

GURU DAKSH GOVERNMENT POLYTECHNIC HISAR

APPROVED BY AICTE NEW DELHI, AFFILIATED HSBTE PANCHKULA



DEPARTMENT OF COMPUTER ENGINEERING

APPLICATION DEVELOPMENT USING WEB FRAMEWORK (E-CONTENT)

VI-SEMESTER

January-May 2026

Prepared by:

KAPIL BHORIA

Syllabus

APPLICATION DEVELOPMENT USING WEB FRAMEWORK LABORATORY SEMESTER – 6 (CSE)

Hours / Week

06 Hours Laboratory

Exam Marks 100

1. Practice on HTML, CSS, Java Script, Ajax.PHP & MySql
2. Install WordPress & Create Blogs
3. Manage blogs features e.g. Images, Text Around Images, Comments, Post Formats, Linking, Pages, Categories, Smilies, Feeds, Gravatars, Password Protection
4. Practice various designing features: Colour Scheme, Headers, CSS Horizontal Menus, Dynamic Menu, Highlighting, Navigation Links, and Print
5. Read More, Formatting Date and Time, Finding CSS Styles, Creating Individual Pages, Uploading Files, Using WordPress Themes, Templates, Template Tags, Template Hierarchy, Validating a Website, Know Your Sources, WordPress Site Maintenance
6. Integrate PHP & MySql with WordPress
7. Install Moodle & various plugins
8. Create a Moodle site and Database Schema
9. Design Site appearance, Front page, Front page settings, My Moodle, User profiles, Navigation, Course list, Themes, Theme settings, Header and footer, Language settings, Using web services, Publishing a course, Blogs, RSS feeds
10. Manage Moodle site, Managing authentication, Manual accounts, No login, Email-based self-registration, Account
11. Create Roles and permissions, Assign roles
12. Implement Password salting.
13. Perform Site backup, Course backup, Course restore, automated course backup

LIST OF CONTENTS

Sr. No.	Details	Page No.
1	Introduction	4
2	General Instructions	4-6
	2.1 General Discipline in the Lab	
	2.2 Attendance	
	2.3 Preparation and Performance	
	2.4 Lab Reports	
	2.5 Learning Outcomes	
3	Details of Experiment	
	Experiment No.1	
	Practice on HTML, CSS, Java Script, Ajax.PHP & MySql	7-13
	Experiment No.2	
	Install WordPress & Create Blogs	14-23
	Experiment No.3	
	Manage blogs features e.g. Images, Text Around Images, Comments, Post Formats, Linking, Pages, Categories, Smilies, Feeds, Gravatars, Password Protection	24-40
	Experiment No.4	
	Practice various designing features: Colour Scheme, Headers, CSS Horizontal Menus, Dynamic Menu, Highlighting, Navigation Links, and Print	41-55
	Experiment No.5	
	Read More, Formatting Date and Time, Finding CSS Styles, Creating Individual Pages, Uploading Files, Using WordPress Themes, Templates, Template Tags, Template Hierarchy, Validating a Website, Know Your Sources, WordPress Site Maintenance	56-96
	Experiment No.6	
	Integrate PHP & MySql with WordPress	97-111
	Experiment No.7	
	Install Moodle & various plugins	112-119
	Experiment No.8	

	Create a Moodle site and Database Schema	120-123
	Experiment No.9	
	Design Site appearance, Front page, Front page settings, My Moodle, User profiles, Navigation, Course list, Themes, Theme settings, Header and footer, Language settings, Using web services, Publishing a course, Blogs, RSS feeds	124-147
	Experiment No.10	
	Manage Moodle site, Managing authentication, Manual accounts, No login, Email-based self-registration, Account	148-153
	Experiment No.11	
	Create Roles and permissions, Assign roles	154-160
	Experiment No.12	
	Implement Password salting.	161
	Experiment No.13	
	Perform Site backup, Course backup, Course restore, automated course backup	162-172

INTRODUCTION

Web frameworks have transformed the world of programming and become vitally important in every development process. Even the smallest unit of an application is comprised of coding, and a web framework simply automates it. You might try browsing different sites, books and articles about it, but find only general and ambiguous information – nothing but endless definitions and difficult terms that make your head spin. Well, it's time to handle this issue and get a clear understanding of web frameworks.

GENERAL INSTRUCTIONS

General discipline in the lab

- Students must turn up in time and contact concerned faculty for the experiment they are supposed to perform
- Students will not be allowed after ten minutes from the scheduled time
- Attendance in the laboratory is compulsory. For any absence, students have to fill up the application form in format available in the lab indicating sufficient reasons.
- In case the faculty is not convinced, he may report the matter to the management
- Students will not leave the class till the period is over

- Students must get the connection of the experimental setup verified before switching on the power supply
- After the experiment is over, the experimental kits/modules and any other tools issued for the experiments should be returned to the course instructor
- Students should maintain silence while performing the experiments. Should any necessity arises for discussion amongst them, they should discuss with a very low pitch without disturbing the adjacent groups
- Students should not unnecessarily fiddle with the instruments knob or any other pot. In the instruments which may disturb the calibration accuracy, range or zero etc.
- Violating the above code of conduct may attract disciplinary action.
- Damaging lab equipment or removing any component from the lab may invite penalties and strict disciplinary action.

ATTENDANCE

- Attendance in the lab class is compulsory. As an incentive, certain percentage of lab marks is allotted for attendance in the internal evaluation of lab performance.
- Each group of students must write a satisfactory report for each lab experiments in order to pass the course
- Attendance will be duly recorded by the concerned faculty
- Students should complete the experiments on the day allotted for the same
- Failure to do so or any absence on that day may result in losing marks in the evaluation process of lab records
- However, on the genuine ground, alternate time slot may be given to complete the Experiment
- Students should not attend a different lab group/section other than the one assigned at the beginning of the session
- On account of illness or some family problems, if a student misses his lab classes, he may be assigned a different group to make up the losses in consultation with the concerned faculty member/ lab instructor

PREPARATION AND PERFORMANCE

- Students should come to the lab thoroughly prepared on the experiments they are assigned to perform on that day
- Faculty may check their preparation and understanding of the experiments. If not found satisfactory, students may be debarred from doing the experiments
- Students should record the experimental results and observation in the lab manual
- Any instrument damaged or tools lost during experiments may attract punishment in the form of fine or suspension from the class.

LAB REPORTS

- Each student is required to write a complete report of the experiment he has performed and bring to lab class for evaluation in the next working lab
- Reports should be written very clearly and lab record should be maintained neatly
- The Lab Report must contain the following
 1. Duly completed title page
 2. Each report should include connection diagram, graphs, equations, calculations (where applicable) etc.
 3. Standard symbols should be used to draw the diagrams
 4. Calculations & Comparisons with appropriate equations and comments
 5. Observation can be included explaining your experience in conducting the experiments

LEARNING OUTCOMES

After undergoing the subject, student will be able to:

- Identify basic aspects of web-frameworks.
- Apply the basic concepts, principles and practices of Web-site development using server-side technologies (PHP & MySQL)
- Install Word Press

- Create and manage Blogs, Websites using WordPress.
- Use PHP & MySql with WordPress
- Install Moodle framework
- Create Web Application using Moodle
- Manage features of Moodle sites : Accounts, Enrollment, Roles & Permission
- Take backup of site content

Experiment – 1

Aim: - Practice on HTML, CSS, Java Script, Ajax.PHP & MySql

HTTP and HTML: Berners-Lee's Basics

HTTP is a communication standard governing the requests and responses that take place between the browser running on the end user's computer and the web server. The server's job is to accept a request from the client and attempt to reply to it in a meaningful way, usually by serving up a requested web page—that's why the term server is used. The natural counterpart to a server is a client, so that term is applied both to the web browser and the computer on which it's running. Between the client and the server there can be several other devices, such as routers, proxies, gateways, and so on. They serve different roles in ensuring that the requests and responses are correctly transferred between the client and server. Typically, they use the Internet to send this information. A web server can usually handle multiple simultaneous connections and—when not communicating with a client—spends its time listening for an incoming connection. When one arrives, the server sends back a response to confirm its receipt.

The Request/Response Procedure: At its most basic level, the request/response process consists of a web browser asking the web server to send it a web page and the server sending back the page. The browser then takes care of displaying the page (see Figure 1-1).

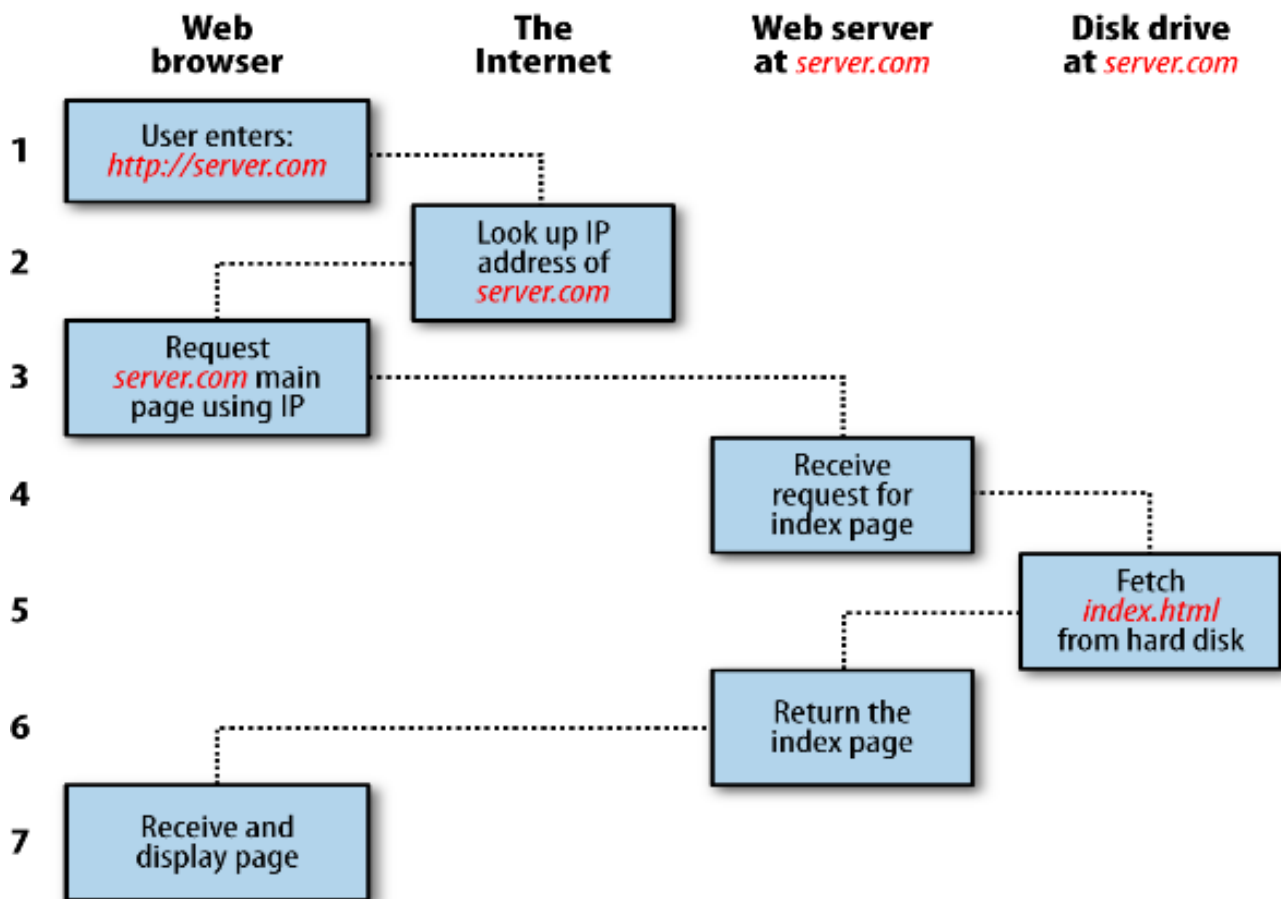


Figure 1-1. The basic client/server request/response sequence

Each step in the request and response sequence is as follows:

1. You enter *http://server.com* into your browser's address bar.
2. Your browser looks up the IP address for *server.com*.
3. Your browser issues a request for the home page at *server.com*.
4. The request crosses the Internet and arrives at the *server.com* web server.
5. The web server, having received the request, looks for the web page on its hard disk.
6. The web page is retrieved by the server and returned to the browser.
7. Your browser displays the web page.

For an average web page, this process takes place once for each object within the page: a graphic, an embedded video or flash file, and even a CSS template.

In step 2, notice that the browser looked up the IP address of *server.com*. Every machine attached to the Internet has an IP address—your computer included. But we generally access web servers by name, such as *google.com*. As you probably know, the browser consults an additional Internet service called

the Domain Name Service (DNS) to find its associated IP address and then uses it to communicate with the computer.

For dynamic web pages, the procedure is a little more involved, because it may bring both PHP and MySQL into the mix (see Figure 1-2).

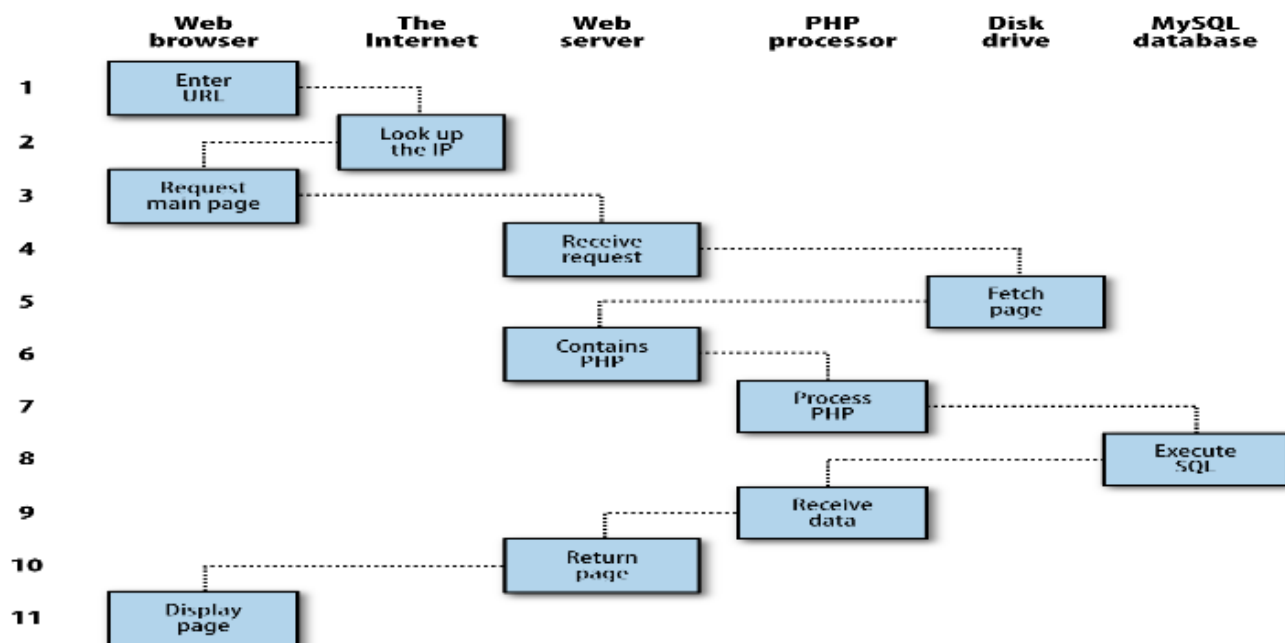


Figure 1-2. A dynamic client/server request/response sequence

Here are the steps for a dynamic client/server request/response sequence:

1. You enter `http://server.com` into your browser's address bar.
2. Your browser looks up the IP address for `server.com`.
3. Your browser issues a request to that address for the web server's home page.
4. The request crosses the Internet and arrives at the `server.com` web server.
5. The web server, having received the request, fetches the home page from its hard disk.
6. With the home page now in memory, the web server notices that it is a file incorporating PHP scripting and passes the page to the PHP interpreter.
7. The PHP interpreter executes the PHP code.
8. Some of the PHP contains MySQL statements, which the PHP interpreter now passes to the MySQL database engine.
9. The MySQL database returns the results of the statements back to the PHP interpreter.
10. The PHP interpreter returns the results of the executed PHP code, along with the results from the MySQL database, to the web server.
11. The web server returns the page to the requesting client, which displays it.

Although it's helpful to be aware of this process so that you know how the three elements work together, in practice you don't really need to concern yourself with these details, because they all happen automatically.

HTML pages returned to the browser in each example may well contain JavaScript, which will be interpreted locally by the client, and which could initiate another request—the same way embedded objects such as images would.

The Benefits of PHP, MySQL, JavaScript, CSS, and HTML5

At the start of this chapter, I introduced the world of Web 1.0, but it wasn't long before the rush was on to create Web 1.1, with the development of such browser enhancements as Java, JavaScript, JScript (Microsoft's slight variant of JavaScript), and ActiveX. On the server side, progress was being made on the Common Gateway Interface (CGI) using scripting languages such as Perl (an alternative to the PHP language) and *server-side scripting* inserting the contents of one file (or the output of a system call) into another one dynamically.

Once the dust had settled, three main technologies stood head and shoulders above the others. Although Perl was still a popular scripting language with a strong following, PHP's simplicity and built-in links to the MySQL database program had earned it more than double the number of users. And JavaScript, which had become an essential part of the equation for dynamically manipulating CSS (Cascading Style Sheets) and HTML, now took on the even more muscular task of handling the client side of the Ajax process. Under Ajax, web pages perform data handling and send requests to web servers in the background—without the web user being aware that this is going on.

No doubt the symbiotic nature of PHP and MySQL helped propel them both forward, but what attracted developers to them in the first place? The simple answer has to be the ease with which you can use them to quickly create dynamic elements on websites. MySQL is a fast and powerful, yet easy-to-use, database system that offers just about anything a website would need in order to find and serve up data to browsers. When PHP allies with MySQL to store and retrieve this data, you have the fundamental parts required for the development of social networking sites and the beginnings of Web 2.0. And when you bring JavaScript and CSS into the mix too, you have a recipe for building highly dynamic and interactive websites.

Using PHP

With PHP, it's a simple matter to embed dynamic activity in web pages. When you give pages the *.php* extension, they have instant access to the scripting language. From a developer's point of view, all you have to do is write code such as the following:

```
<?php
    echo " Today is " . date("l") . ". ";
?>
```

Here's the latest news.

The opening `<?php` tells the web server to allow the PHP program to interpret all the following code up to the `?>` tag. Outside of this construct, everything is sent to the client as direct HTML. So the text Here's the latest news. is simply output to the browser; within the PHP tags, the built-in date function displays the current day of the week according to the server's system time.

The final output of the two parts looks like this:

Today is Wednesday. Here's the latest news.

PHP is a flexible language, and some people prefer to place the PHP construct directly next to PHP code, like this:

Today is <?php echo date("l"); ?>. Here's the latest news.

There are also other ways of formatting and outputting information, which I'll explain in the chapters on PHP. The point is that with PHP, web developers have a scripting language that, although not as fast as compiling your code in C or a similar language, is incredibly speedy and also integrates seamlessly with HTML markup.

Using PHP, you have unlimited control over your web server. Whether you need to modify HTML on the fly, process a credit card, add user details to a database, or fetch information from a third-party website, you can do it all from within the same PHP files in which the HTML itself resides.

Using MySQL

Of course, there's not much point to being able to change HTML output dynamically unless you also have a means to track the changes that users make as they use your website. In the early days of the Web, many sites used "flat" text files to store data such as usernames and passwords. But this approach could cause problems if the file wasn't correctly locked against corruption from multiple simultaneous accesses. Also, a flat file can get only so big before it becomes unwieldy to manage not to mention the difficulty of trying to merge files and perform complex searches in any kind of reasonable time.

That's where relational databases with structured querying become essential. And MySQL, being free to use and installed on vast numbers of Internet web servers, rises superbly to the occasion. It is a robust and exceptionally fast database management system that uses English-like commands.

The highest level of MySQL structure is a database, within which you can have one or more tables that contain your data. For example, let's suppose you are working on a table called users, within which you have created columns for surname, first name, and email, and you now wish to add another user. One command that you might use to do this is:

```
INSERT INTO users VALUES('Smith', 'John', 'jsmith@mysite.com');
```

Of course, as mentioned earlier, you will have issued other commands to create the database and table and to set up all the correct fields, but the INSERT command here shows how simple it can be to add new data to a database. The INSERT command is an example of SQL (Structured Query Language), a language designed in the early 1970s and reminiscent of one of the oldest programming languages, COBOL. It is well suited, however, to database queries, which is why it is still in use after all this time.

It's equally easy to look up data. Let's assume that you have an email address for a user and need to look up that person's name. To do this, you could issue a MySQL query such as:

```
SELECT surname,firstname FROM users WHERE email='jsmith@mysite.com';
```

MySQL will then return Smith, John and any other pairs of names that may be associated with that email address in the database.

As you'd expect, there's quite a bit more that you can do with MySQL than just simple INSERT and SELECT commands. For example, you can join multiple tables according to various criteria, ask for

results in a variety of orders, make partial matches when you know only part of the string that you are searching for, return only the *n*th result, and a lot more.

Using PHP, you can make all these calls directly to MySQL without having to run the MySQL program yourself or use its command-line interface. This means you can save the results in arrays for processing and perform multiple lookups, each dependent on the results returned from earlier ones, to drill right down to the item of data you need.

For even more power, as you'll see later, there are additional functions built right into MySQL that you can call up for common operations and extra speed.

Using JavaScript

The oldest of the three core technologies in this book, JavaScript, was created to enable scripting access to all the elements of an HTML document. In other words, it provides a means for dynamic user interaction such as checking email address validity in input forms, displaying prompts such as “Did you really mean that?”, and so on (note, however, that it cannot be relied upon for security, which should always be performed on the web server).

Combined with CSS (see the following section), JavaScript is the power behind dynamic web pages that change in front of your eyes rather than when a new page is returned by the server.

However, JavaScript can also be tricky to use, due to some major differences in the ways different browser designers have chosen to implement it. This mainly came about when some manufacturers tried to put additional functionality into their browsers at the expense of compatibility with their rivals. Thankfully, the developers have mostly now come to their senses and have realized the need for full compatibility with one another, so they don't have to write multi-exception code. But there remain millions of legacy browsers that will be in use for a good many years to come. Luckily, there are solutions for the incompatibility problems, and later in this book we'll look at libraries and techniques that enable you to safely ignore these differences.

For now, let's take a quick look at how you can use basic JavaScript, accepted by all browsers:

```
<script type="text/javascript">
    document.write("Today is " + Date() );
</script>
```

This code snippet tells the web browser to interpret everything within the script tags as JavaScript, which the browser then does by writing the text Today is to the current document, along with the date, by using the JavaScript function Date. The result will look something like this:

Today is Sun Jan 01 2017 01:23:45

As previously mentioned, JavaScript was originally developed to offer dynamic control over the various elements within an HTML document, and that is still its main use. But more and more, JavaScript is being used for Ajax. This is a term for the process of accessing the web server in the background. (It originally meant “Asynchronous Java-Script and XML,” but that phrase is already a bit outdated.)

Ajax is the main process behind what is now known as Web 2.0 (a term popularized by Tim O'Reilly, the founder and CEO of this book's publishing company), in which web pages have started to resemble standalone programs, because they don't have to be reloaded in their entirety. Instead, a quick Ajax call can pull in and update a single element on a web page, such as changing your photograph on a social networking site or replacing a button that you click with the answer to a question.

Using CSS

With the emergence of the CSS3 standard in recent years, CSS now offers a level of dynamic interactivity previously supported only by JavaScript. For example, not only can you style any HTML element to change its dimensions, colors, borders, spacing, and so on, but now you can also add animated transitions and transformations to your web pages, using only a few lines of CSS.

Using CSS can be as simple as inserting a few rules between `<style>` and `</style>` tags in the head of a web page, like this:

```
<style>
  p {
    text-align:justify;
    font-family:Helvetica;
  }
</style>
```

These rules will change the default text alignment of the `<p>` tag so that paragraphs contained in it will be fully justified and will use the Helvetica font.

There are many different ways you can lay out CSS rules, and you can also include them directly within tags or save a set of rules to an external file to be loaded in separately. This flexibility not only lets you style your HTML precisely, but it can also, for example, provide built-in hover functionality to animate objects as the mouse passes over them. You will also learn how to access all of an element's CSS properties from JavaScript as well as HTML.

And Then There's HTML5

As useful as all these additions to the web standards became, they were not enough forever more ambitious developers. For example, there was still no simple way to manipulate graphics in a web browser without resorting to plug-ins such as Flash. And the same went for inserting audio and video into web pages. Plus, several annoying inconsistencies had crept into HTML during its evolution.

So, to clear all this up and take the Internet beyond Web 2.0 and into its next iteration, a new standard for HTML was created to address all these shortcomings. It was called HTML5 and it began development as long ago as 2004, when the first draft was drawn up by the Mozilla Foundation and Opera Software (developers of two popular web browsers). But it wasn't until the start of 2013 that the final draft was submitted to the World Wide Web Consortium (W3C), the international governing body for web standards.

With nine years for it to develop, you might think that would be the end of the specification, but that's not how things work on the Internet. Although websites come and go at great speed, the underlying software is developed slowly and carefully, and so the stable recommendation for HTML5 is not expected until after this edition of the book has been published—in late 2014. And then guess what? Work will move on to versions 5.1 and higher, beginning in 2015. It's a never ending cycle of development.

However, while HTML5.1 is planned to bring some handy improvements (mainly to the canvas), basic HTML5 is the new standard web developers now need to work to, and it will remain in place for many years to come. So learning everything you can about it now will stand you in very good stead.

There's actually a great deal of new stuff in HTML (and quite a few things that have been changed or removed), but in summary, here's what you get:

Markup

Including new elements such as <nav> and <footer>, and deprecated elements like and <center>.

New APIs

For example, the <canvas> element for writing and drawing on a graphics canvas, <audio> and <video> elements, offline web apps, microdata, and local storage.

Applications

Including two new rendering technologies: MathML (Math Markup Language) for displaying mathematical formulae) and SVG (Scalable Vector Graphics) for creating graphical elements outside of the new <canvas> element. However, MathML and SVG are somewhat specialist, and are so feature-packed they would need a book of their own, so I don't cover them here.

Experiment – 2

Aim: - Install WordPress & Create Blogs

WordPress is an open source **Content Management System (CMS)**, which allows the users to build dynamic websites and blogs. Wordpress is the most popular blogging system on the web and allows updating, customizing and managing the website from its back-end CMS and components.

What is Content Management System (CMS)?

The **Content Management System (CMS)** is software which stores all the data such as text, photos, music, documents, etc. and is made available on your website. It helps in editing, publishing and modifying the content of the website.

WordPress was initially released on 27th May, 2003 by Matt Mullenweg and Mike Little. WordPress was announced as open source in October 2009.

Features

- **User Management:** It allows managing the user information such as changing the role of the users to (subscriber, contributor, author, editor or administrator), create or delete the user, change the password and user information. The main role of the user manager is **Authentication**.
- **Media Management:** It is the tool for managing the media files and folder, in which you can

easily upload, organize and manage the media files on your website.

- **Theme System:** It allows modifying the site view and functionality. It includes images, stylesheet, template files and custom pages.
- **Extend with Plugins:** Several plugins are available which provides custom functions and features according to the users need.
- **Search Engine Optimization:** It provides several search engine optimization (SEO) tools which makes on-site SEO simple.
- **Multilingual:** It allows translating the entire content into the language preferred by the user.
- **Importers:** It allows importing data in the form of posts. It imports custom files, comments, post pages and tags.

Advantages

- It is an open source platform and available for free.
- CSS files can be modified according to the design as per users need.
- There are many plugins and templates available for free. Users can customize the various plugins as per their need.
- It is very easy to edit the content as it uses WYSIWYG editor (What You See Is What You Get is a user interface that allows the user to directly manipulate the layout of document without having a layout command).
- Media files can be uploaded easily and quickly.
- It offers several SEO tools which makes on-site SEO simple.
- Customization is easy according to the user's needs.
- It allows creating different roles for users for website such as admin, author, editor and contributor.

Disadvantages

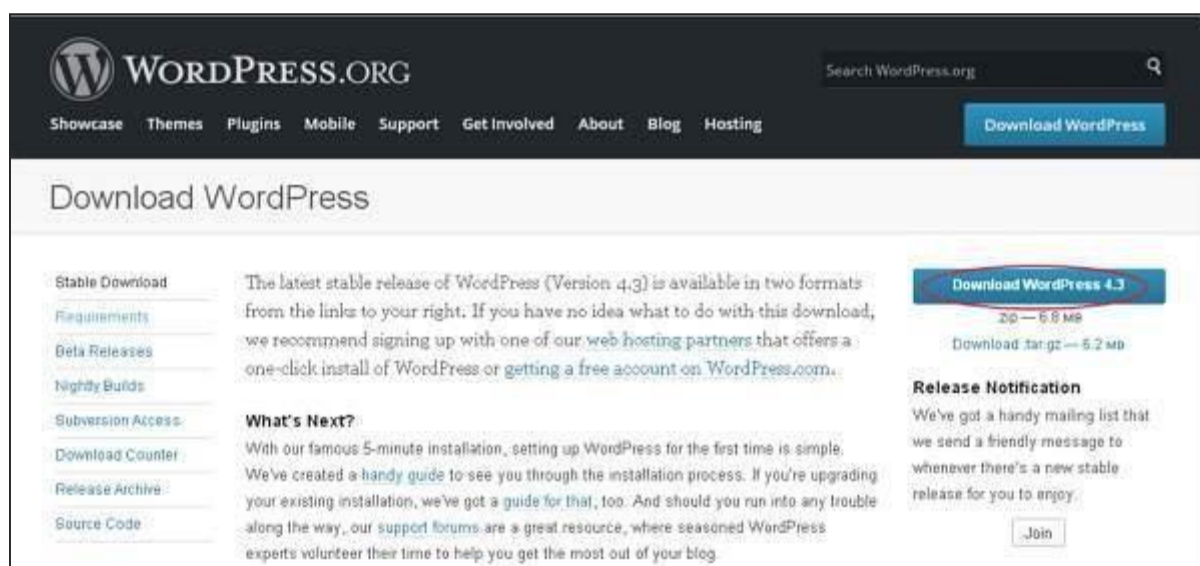
- Using several plugins can make the website heavy to load and run.
- PHP knowledge is required to make modifications or changes in the WordPress website.
- Sometimes software needs to be updated to keep the WordPress up-to-date with the current browsers and mobile devices. Updating WordPress version leads to loss of data, so it a backup copy of the website is required.
- Modifying and formatting the graphic images and tables is difficult.

System Requirements for WordPress

- **Database:** MySQL 5.0+
- **Web Server:**
 - WAMP (Windows)
 - LAMP (Linux)
 - XAMP (Multi-platform)
 - MAMP (Macintosh)
- **Operating System:** Cross-platform
- **Browser Support:** IE (Internet Explorer 8+), Firefox, Google chrome, Safari, Opera
- **PHP Compatibility:** PHP 5.2+

Download WordPress

When you open the link <https://wordpress.org/download/>, you will get to see a screen as the following snapshot:



Download the WordPress zip file from the official site.

Create Store Database

- WordPress requires MySQL database. So create a new empty database with user/password (for example, user as "root" and password as "root" or else you can set as per your convenience).
- Then, you can continue with the installation process as discussed further.

Set up Wizard

It's very easy to set up WordPress into your system. The following steps describe how to set up WordPress locally on your system.

Step (1): Extract the downloaded WordPress folder and upload it into your web server or localhost.

Step (2): Open your browser and navigate to your WordPress file path, then you will get the first screen of the WordPress installer as shown in the following screen. In our case, the path is **localhost/<Your_WordPress_folder >**.



Select your language for the WordPress and click on **Continue**.

Step (3): In this step, you can view the information needed for the database before proceeding with WordPress installation.



Welcome to WordPress. Before getting started, we need some information on the database. You will need to know the following items before proceeding.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

We're going to use this information to create a `wp-config.php` file. If for any reason this automatic file creation doesn't work, don't worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`. Need more help? [We got it.](#)

In all likelihood, these items were supplied to you by your Web Host. If you do not have this information, then you will need to contact them before you can continue. If you're all ready...

[Let's go!](#)

Click on **Let's go!**

Step (4): Here, you have to enter the information about the MySQL database as described in the following screen.

Below you should enter your database connection details. If you're not sure about these, contact your host.

Database Name	wordpress	The name of the database you want to run WP in.
User Name	root	Your MySQL username
Password	root	...and your MySQL password.
Database Host	localhost	You should be able to get this info from your web host, if <code>localhost</code> does not work.
Table Prefix	wp_	If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)

- **Database Name:** Enter the database name which you have created in MySQL database for WordPress.
- **Username:** Enter the user name of your MySQL database.
- **Password:** Enter the password which you had set for MySQL database.
- **Database Host:** Write the host name, by default it will be localhost.

- **Table Prefix:** It is used to add prefix in the database tables which helps to run multiple sites on the same database. It takes the default value.

After filling all information, click on **Submit** button.

Step (5): WordPress checks the database setting and gives you the confirmation screen as shown in the following snapshot.



Click on **Run the install**

Step (6): Enter administrative information.

A screenshot of the WordPress installation "Welcome" screen. At the top center is the WordPress logo. Below it, the heading "Welcome" is followed by a paragraph: "Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world." Below this is the heading "Information needed" and a note: "Please provide the following information. Don't worry, you can always change these settings later." The form contains several fields: "Site Title" with the value "Tutorialspoint"; "Username" with the value "admin" and a note that usernames can have only alphanumeric characters, hyphens, underscores, hyphens, periods, and the @ symbol; "Password, twice" with two masked password fields and a green "Strong" indicator, with a note that the password should be at least 8 characters long; "Your E-mail" with the value "admin@tutorialspoint.com" and a note to double-check the email address; and a "Privacy" section with a checked checkbox for "Allow search engines to index this site." At the bottom left is a button labeled "Install WordPress".

It contains the following fields:

- **Site Title:** Enter the name of the site which you are going to create in WordPress.
- **Username:** Enter the username as per your choice while logging in the WordPress.
- **Password twice:** Enter password two times to protect your site.
- **Your E-mail:** Enter your e-mail address which helps to recover the password or any update.
- **Privacy:** It allows the search engine to index this site after checking the checkbox.

After filling all the information, click on the **Install WordPress** button.

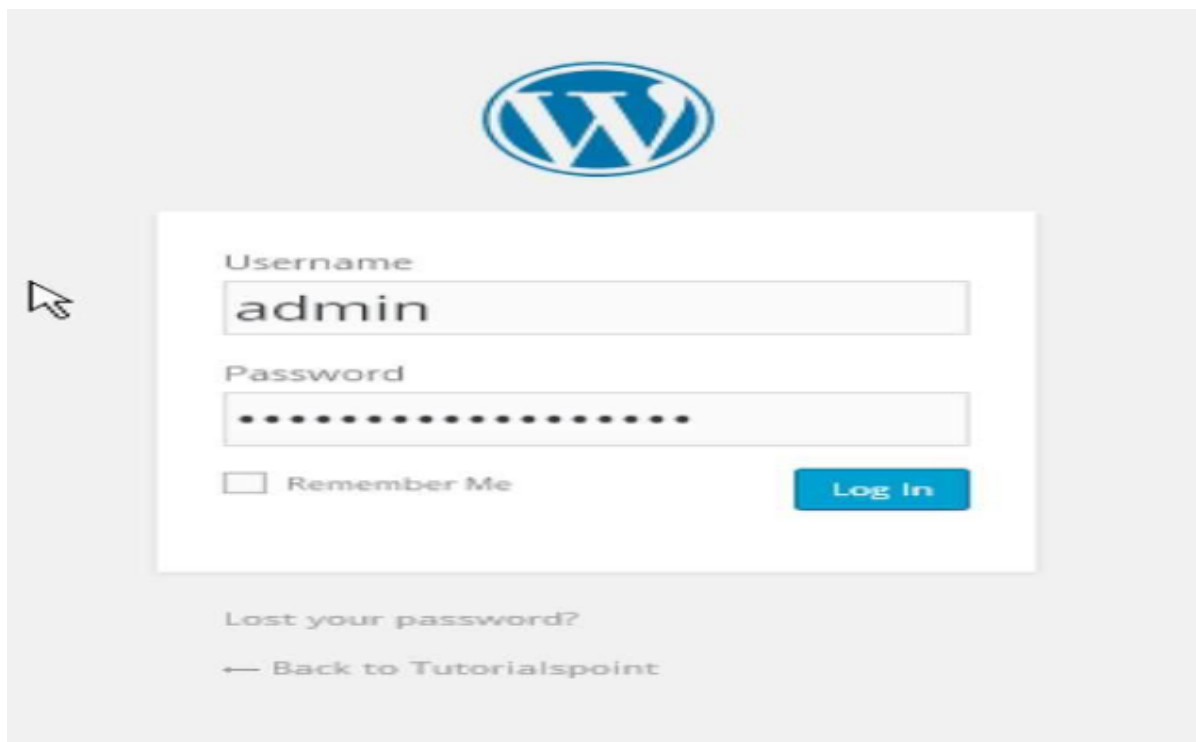
Step (7): After installation being successful, you will get a screen of the stating success as seen in the following screen.



You can view your username and password detail added in WordPress.

Click on **Log In** button.

Step (8): After clicking on login, you will get a WordPress Admin Panel as depicted in the following screen.



Enter the username and password which you had mentioned during installation as shown in step 6 and click on the **Log In** button.

Blogging

A blog is basically a 'weblog', a kind of website, a journal that is updated regularly. It is a discussion platform besides being content achieve. People can discuss a topic on a blog and comment on posts. It is basically run by an individual or a small group.

Every activity and skill set that is required to manage a blog comes under the purview of Blogging, for example, writing posts, publishing them, designing, social marketing, etc. In simple words, blogging is the process of adding new content to a blog and updating it at regular intervals.

- **Blogger Person** – Person involved in blogging is called a 'blogger'.
- **Blogger Service** – It is a service provided by Google that allows us to compose a post, edit, update, and manage it. You can customize your blogs easily with the help of a 'blogger'. It lets you share what is on your mind and earn money. It is a convenient platform for avid as well as novice users. No technical knowledge or coding is required. It runs directly on Google servers and is impossible to hack.
- **WordPress** – It's a free software to create and customize a search engine friendly website or a blog. A number of themes and plugins are available to transform your website as required. Approximately 70 million people are a part of WordPress community. WordPress is available in both free and premium plans.

- **Blogspot** – Blogspot is a free web hosting service gifted to web developers by Google. It helps Google Blogger in delivering its service. We cannot use any other platform to use it.
- **Tumbler** – It is a micro blogging platform that allows us to make a blog and customize it. We may use stories, images, videos, audio, etc. Bloggers can either keep their blogs private or follow other blogs.

Advantages of Blogging

- Better online influence. It assists in making impressions in grabbing a good profile.
- It forms a good source of income. Pat Flynn earns \$100,000 per year from his blog 'Green Exam Academy'.
- Maintaining a blog can help improve your writing skills.
- A blogger can easily become a published author.
- You can get instant feedbacks in the form of comments, shares, etc.
- Bloggers can be sound learners, as they tend to read and engage more in such content.
- Get a long fan following list.
- Blogs can help you build a strong network.
- Maintaining a blog can help you improve your technical skills on Wordpress, SEO, etc.
- Sell Online – Blogs can be a great virtual store.
- Express yourself – Blogging is a great way to share your ideas through inspirational means.
- Blogging is an easy, yet a sure way to build trust online.

Setting a Blog using WordPress

You can create blogs on many platforms. They can be either free or paid. Here is how you can create a blog using WordPress –

- Create your account on WordPress.com. Enter your name and Email address and other required details.
- Select a web address; it can be either free or premium.
- Choose a suitable WordPress plan.
- Set up your profile and build your profile page.
- Configure site settings like Site Title and tagline.

Setting a Blog using Blogger

Blogger.com is a popular platform that is being used widely for creating ready-to-post blogs. Follow the steps given below to create a blog using blogger –

- Sign up and create an account on www.blogger.com.
- Enter display name and click 'create your blog now'.
- Choose your blog title and URL. Verify whether it is available.
- Choose a desirable template from the given options.
- You would come across the '**posting**' tab. Create post, edit posts, and page here.
- Enter the title of your post on the 'title' tab.
- The rest of your post goes in the '**Compose**' text editor. You can customize your post by editing its font, size, text color, etc. '**Edit HTML**' option is available to insert your content in HTML format.
- Select '**post option**' if you want to enable readers' comment.
- Save once you are done with it. You can preview the final page before publishing it. Click '**Publish**' to publish the post directly.

Blog Promotion

You can take the following measures to promote your blog –

- Connect your blog to Google+ to allow auto-sharing.
- Post quality content and post regularly.
- Write search engine friendly content.
- Have an Email button in order to receive feedback from your readers.
- Use your blog link in mail signatures.
- Enable site feed to let more and more people subscribe to your content.
- Add your blog to Blogger's listings. Submit your URL to blog directories.
- Actively comment on other's posts. This way, your profile gets more exposure.
- Participate in online forums.
- Use social media for branding.
- Hold a contest or a carnival.

- Write as a guest blogger. Write multiple pages and link them together.

Blog Commenting

Commenting on a blog is an art. You need to put some effort to post comments that catch users' attention.

- Add a few attractive phrases with exclamation mark to draw attention.
- Comment as early as possible, as the first few comments are the ones read by most.
- Refrain yourself from over-promoting.
- Don't be rude or use offensive language.
- Include questions in your comments. They seek more attention.
- Be influencing by updating an article.

Experiment – 3

Aim: - Manage blogs features e.g. Images, Text around Images, Comments, Post Formats, Linking, Pages, Categories, Smilies, Feeds, Gravatars, Password Protection

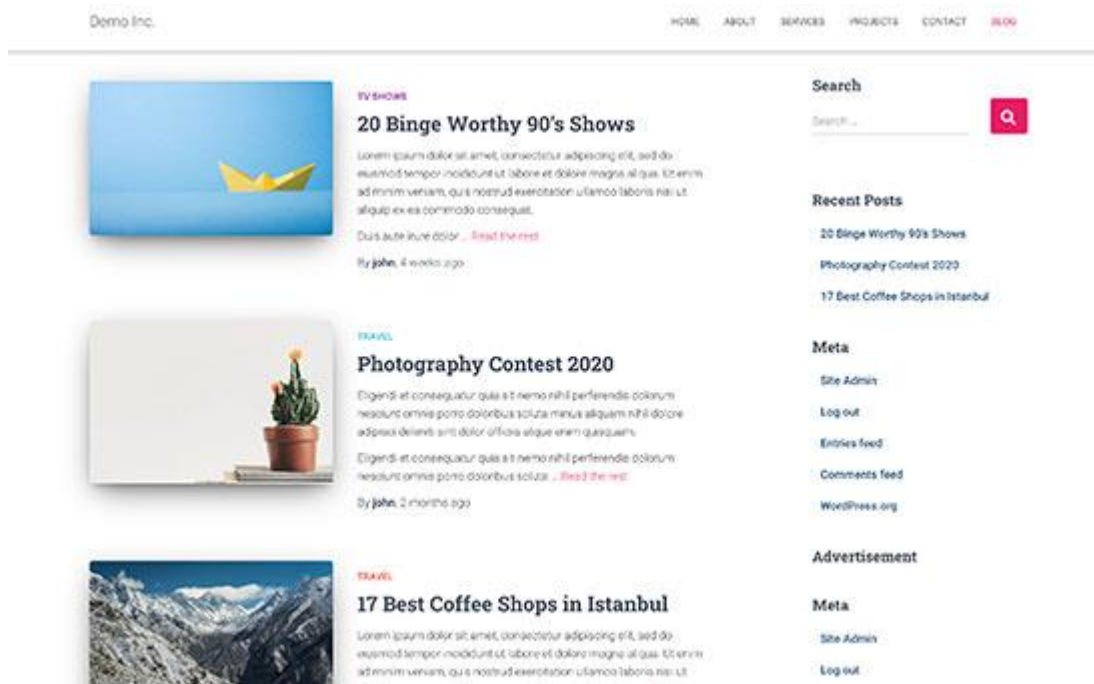
How to add featured images or post thumbnails in Wordpress

Featured images also known as post thumbnails are a well-known WordPress feature supported by most themes.

They are the primary image for your blog posts, and they often appear next to the heading on your home page and social media. All popular WordPress themes come with built-in support for featured images.

Featured Images in WordPress

Featured images are a WordPress theme feature. Almost all WordPress themes come with built-in support for featured images and display them beautifully across different areas of your WordPress website.



As humans, we find visual elements more engaging than plain text. Featured images help you make your blog pages look more appealing.

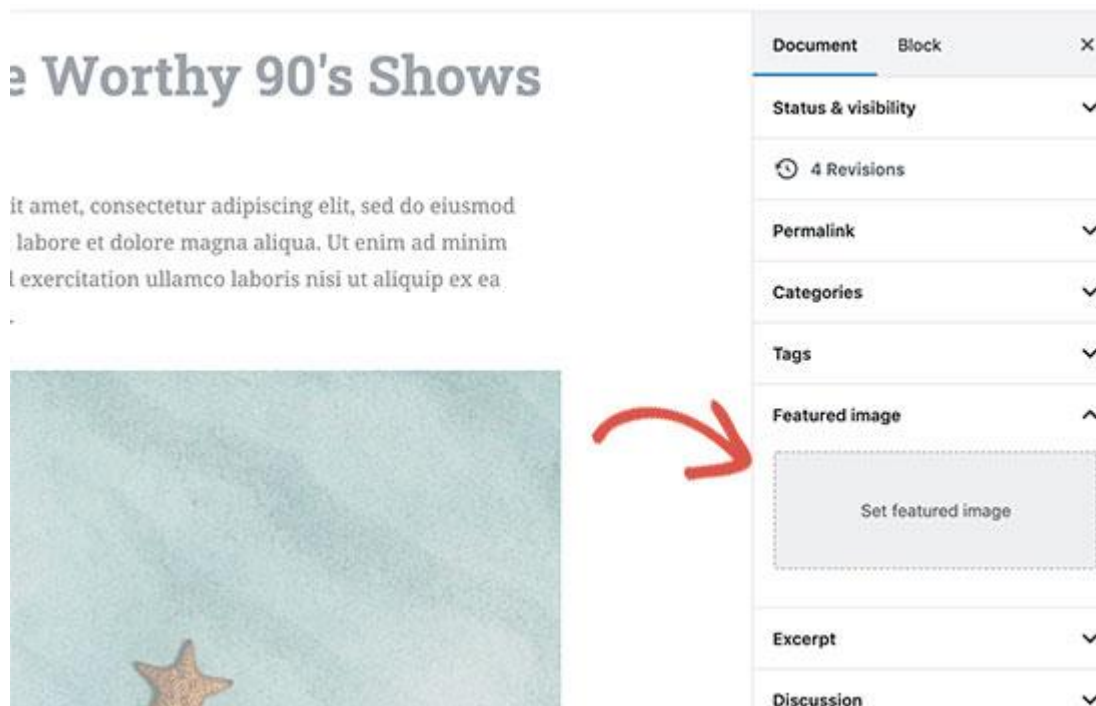
They also help you build user engagement and increase page views. Search engines and social media websites may also use these images and display them in search results and social media news feed.

Mostly featured images are used for blog posts, but you can also set featured images for pages and custom post types.

Adding Post Thumbnail or Featured Image in WordPress

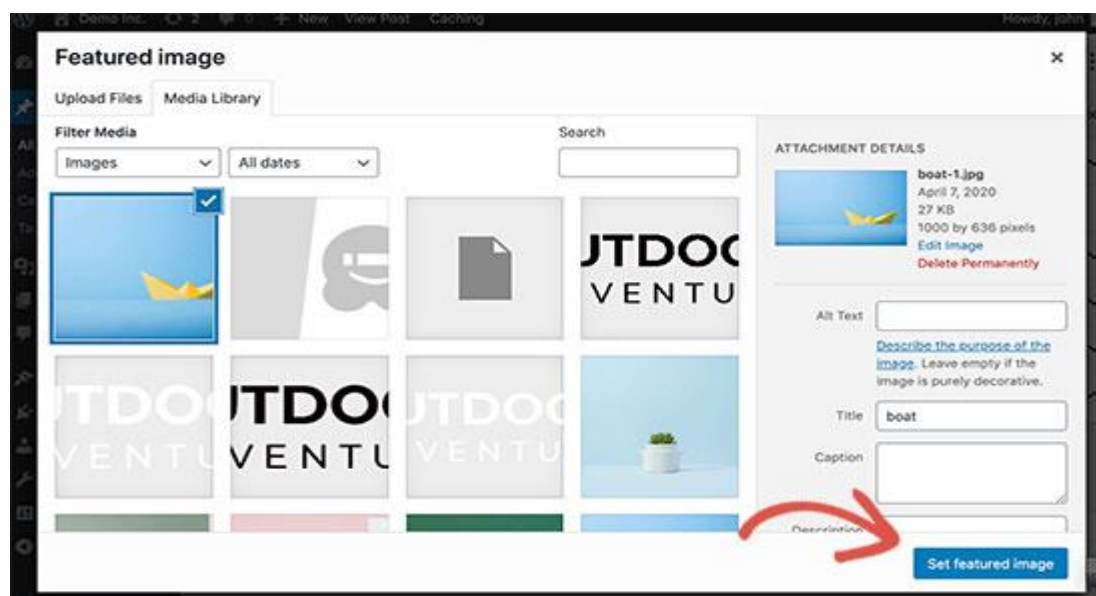
To add a featured image in a WordPress post, simply edit or create a new blog post.

In the [content editor](#), you'll find the featured image tab in the right column.



You need to click on the ‘Set Featured Image’ area, and this will bring up the WordPress media uploader popup.

From here, you can upload an image from your computer or use an existing image from your media library. Once you select the image, simply click on the Set Featured Image button.



WordPress will now show a preview of the featured image in the right column. However, the actual featured image would look different depending on your WordPress theme.

You can now save or publish your post and preview it to see how the featured image will appear on your live website.

Depending on settings defined by your theme developer, your featured image will automatically appear with your posts.

How to Find and Effectively Use Featured Images?

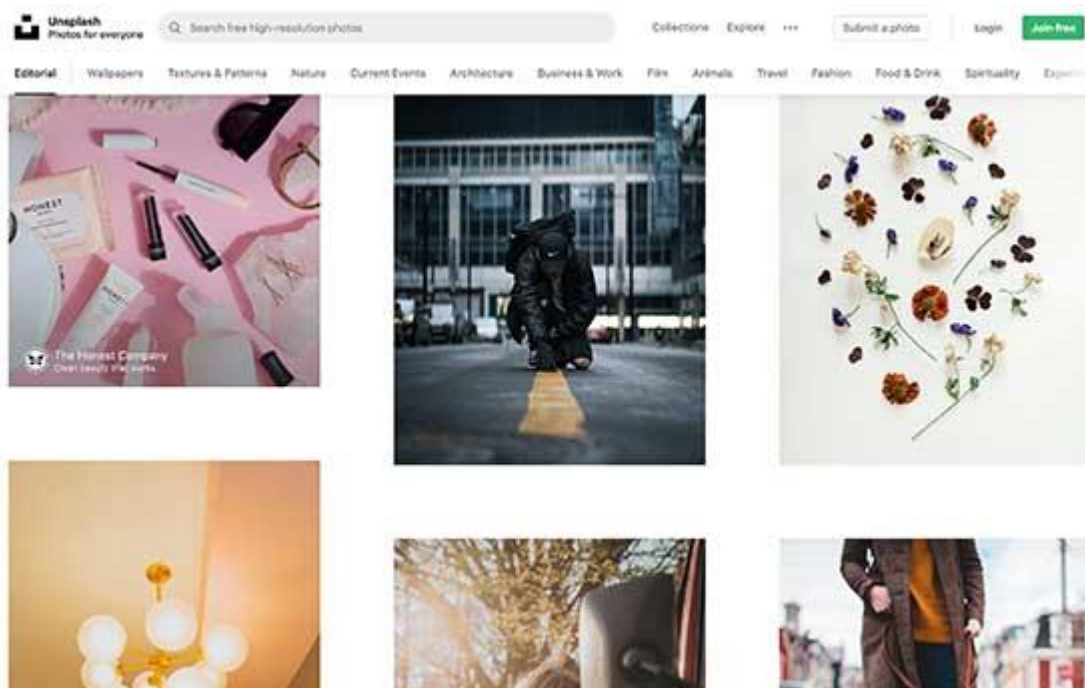
Now, the next big question that most beginners come across is where to find images to use as featured images?

You cannot just use Google image search and use any image from the internet. Those images are protected by copyright laws and using them without proper permission can cause legal trouble.

Luckily, there are several resources available that you can use to find free images for your blog posts.

Our top favorites are:

- Shutterstock – They have a few free images, but the real value comes from their paid plans which give you access to tons of high quality photos, illustrations, drawings, videos, and more. This is what we use for WPBeginner site.
- Unsplash – A popular online resource publishing high-resolution images that you can use on your website and other projects.
- Negative Space – A well-organized collection of copyright-free images.
- New Old Stock – A collection of vintage photographs from public archives.



All these sources provide a high-resolution image, which means they are large in file size and dimensions. You'll need to optimize images for the web before using them as featured images.

Featured images are handled by your WordPress theme. In order to change how your theme displays featured images, you'll need some basic CSS or WordPress coding skills.

What is: Comments

Blog comments are written reactions that readers can add to your blog posts. Leaving a comment on a blog post is an easy way for your visitors to communicate with you and other readers.

Comments usually appear on your site right after the blog post content, along with the commenter's name, the date, and time they left the comment.

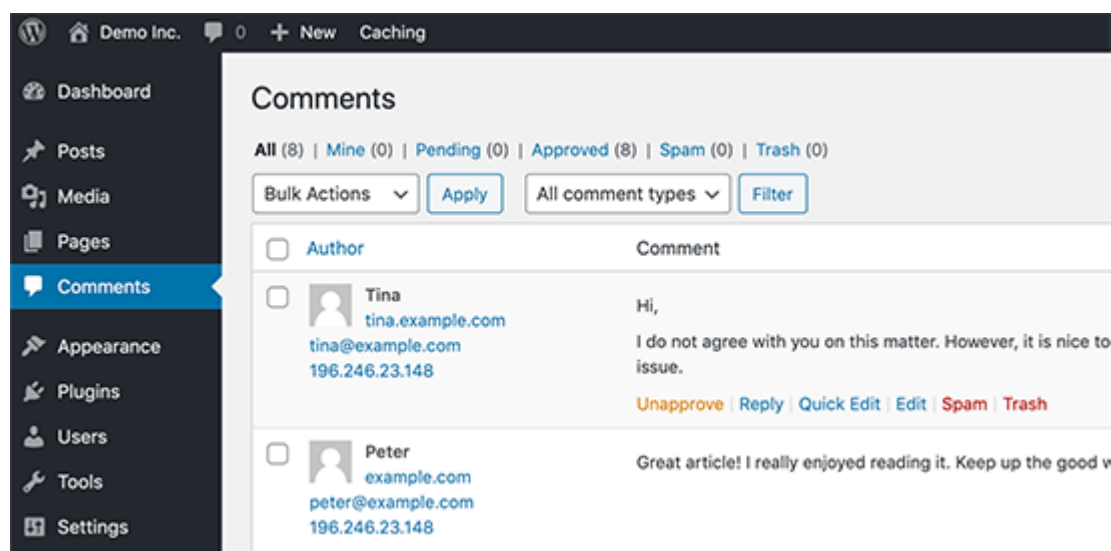
A lively comment area can be a great way for you and your readers to build a community around your blog. Visitors can give feedback, ask questions, offer their own point of view on the topic, and more.

WordPress posts, pages, and other post types can accept comments, if you choose to allow them. However, it's up to you to engage users on your website and encourage them to leave comments.

Let's take a look at how to manage and approve comments first.

The Comments Screen in WordPress

All comments on a WordPress website are displayed on the **Comments** page in your WordPress admin area. By default, it displays all comments.



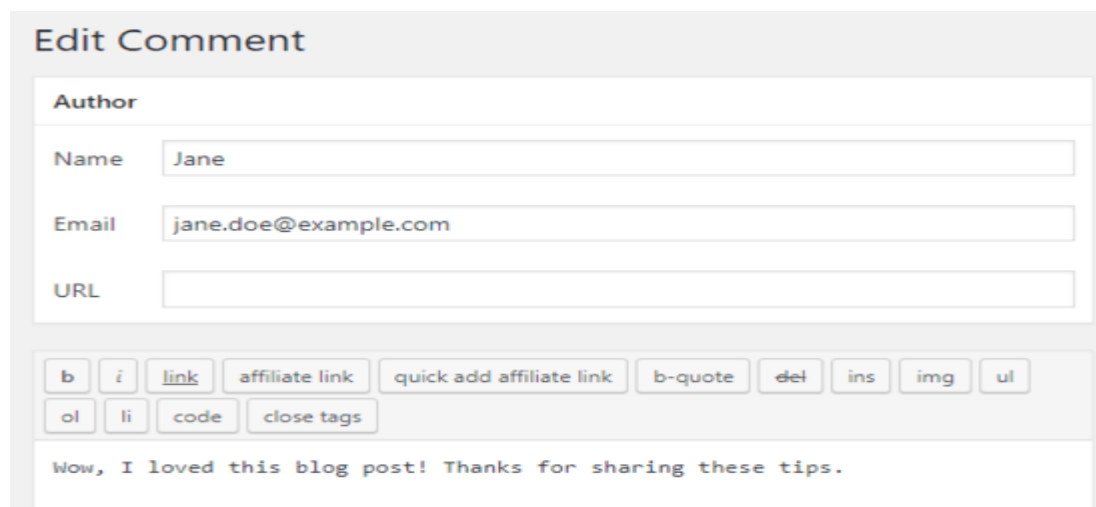
You can switch the view to see Pending, Approved, Spam, and Trash comments:

- **Pending** comments have been submitted by visitors, but they won't be visible on your blog posts unless you approve them.
- **Approved** comments have been published and are publicly visible on your blog posts by all your website visitors.
- **Spam** comments have been flagged as possibly unwanted or irrelevant, and aren't visible to your site visitors.
- **Trash** comments have been marked as unwanted and will be automatically deleted for good after 30 days.

Moving your mouse over to a comment will display the action links for it. You can approve a comment to make it visible to all your site visitors, or unapproved it to put it in pending status again.

You can also reply to comments. By replying to a specific comment, your reply will appear right under that comment. This is called "threaded" or "nested" comments.

You can also choose to edit anyone's comments on your own site. The **Quick Edit** option opens a text area so you can edit the comment without reloading the page. **Edit**, on the other hand, gives you more options to edit the name, user email, and more.



The screenshot shows the 'Edit Comment' form in WordPress. It has a title 'Edit Comment' at the top. Below it is a section for 'Author' with three input fields: 'Name' (containing 'Jane'), 'Email' (containing 'jane.doe@example.com'), and 'URL' (empty). Below the input fields is a row of buttons for text formatting: 'b' (bold), 'i' (italic), 'link', 'affiliate link', 'quick add affiliate link', 'b-quote', 'del' (delete), 'ins' (insert), 'img' (image), and 'ul' (list). Below these buttons are two more buttons: 'ol' (ordered list) and 'li' (list item), followed by 'code' and 'close tags'. At the bottom of the form is a text area containing the comment text: 'Wow, I loved this blog post! Thanks for sharing these tips.'

If you see a comment marked as spam that's actually a real comment, then you can mark it as **Not Spam** by hovering your mouse over it.

Comments in the trash section are comments that have been deleted by you. Any comment in the trash can be restored by clicking on **Restore**. Comments in trash remain there for 30 days after this time WordPress deletes them permanently.

What Is Discussion Settings in WordPress?

The WordPress comment system comes with many options which you can configure from the **Settings » Discussion** page. The first section of these options is for default article settings. These settings apply to all your posts, pages, attachments, and custom post types.



The screenshot shows the 'Discussion Settings' page in WordPress. It has a title 'Discussion Settings' at the top. Below it is a section for 'Default article settings' with three checkboxes: 'Attempt to notify any blogs linked to from the article' (unchecked), 'Allow link notifications from other blogs (pingbacks and trackbacks) on new articles' (checked), and 'Allow people to post comments on new articles' (checked). Below these checkboxes is a note: '(These settings may be overridden for individual articles.)'

Checking "Attempt to notify any blogs linked to from the article" will send trackbacks or pingbacks to the articles you have linked in your post. This feature allows you to notify other blog owners that you have linked to their articles.

If the web page you have linked to is using a pingback-enabled system like WordPress, then the owner of that website will see a notification with a link to your article. Then that website owner can decide to show your trackback in their website's comment section.

Just like the option above, you can also enable your WordPress site to accept pingbacks from other websites by checking "Allow link notifications from other blogs".

You'll see pingbacks from other sites under the Comments screen. You'll also see your pingbacks when you link one of your articles in a new post.

Every WordPress website doesn't need to have comments enabled. You can disable them by unchecking the box next to "*Allow people to post comments on new articles*" option.

Disabling comments will not delete older comments already published on your website. It will just stop accepting new comments. You can re-enable them any time you want too.

What Is Comment Moderation?

Comment Moderation is a WordPress feature that allows website owners to approve or delete comments posted by visitors on their website. Comment moderation settings can be changed from **Settings » Discussion** screen.

To enable moderation, go to **Settings » Discussion** and check the option "Comment must be manually approved".



The image shows a screenshot of the 'Before a comment appears' section in the WordPress Discussion settings. It contains two checkboxes. The first checkbox, labeled 'Comment must be manually approved', is checked with a blue checkmark. The second checkbox, labeled 'Comment author must have a previously approved comment', is unchecked.

Comment moderation allows website owners to prevent potentially harmful, abusive, or spam content from getting published on their website. Using comment moderation with a spam protection plugin such as [Akismet](#) can significantly reduce the chances of any spam comment getting published.

Only Allow Registered Users to Comment

If you want, you can only allow registered users to comment on your website. To do that, first you need to open your WordPress site for registration. Go to **Settings » General** and check the box next to "Anyone can register".

If you don't want new users to write posts on your site then make sure that you have **Subscriber** as the **New User** default role before you save changes.

General Settings

Email Address

This address is used for admin purposes. If you change this we will send you an email at your new address to confirm it. The new address will not become active until confirmed.

Membership

☐ Anyone can register 

New User Default Role

After opening your website for registration go back to **Settings » Discussion** screen. Under ‘Other comment settings’ section check the box next to “Users must be registered and logged in to comment”.

Other comment settings

☒ Comment author must fill out name and email

☒ Users must be registered and logged in to comment 

☐ Automatically close comments on articles older than days

☒ Show comments cookies opt-in checkbox, allowing comment author cookies to be set.

Recieve Email Notifications for WordPress Comments

From the **Settings » Discussion** page, you can also set the options to receive email notifications whenever someone leaves a comment on your website, or comment is waiting to be approved. Checking these options will also send you an email notification when a website sends you trackback or pingback.

However, you will not receive any email notifications for comments or trackbacks marked as spam.

Email me whenever

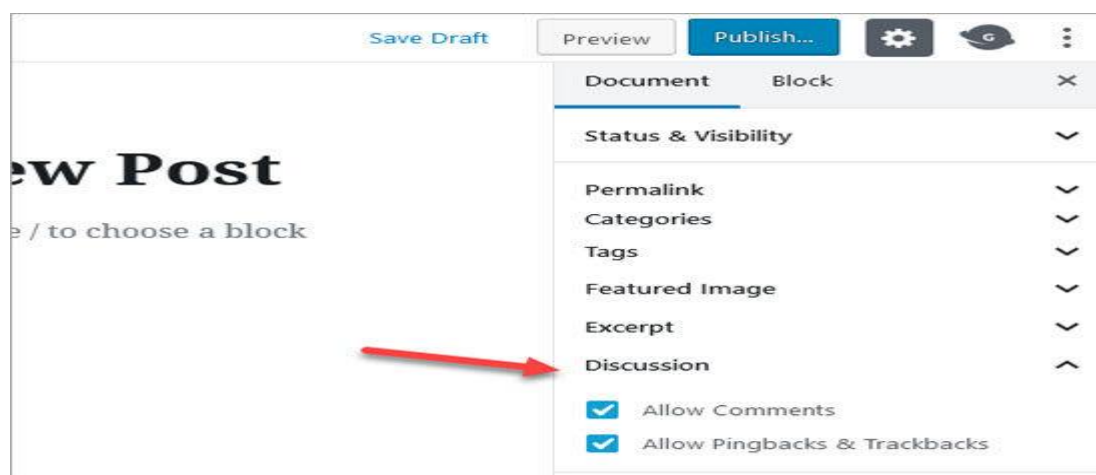
☒ Anyone posts a comment

☒ A comment is held for moderation

If you do not receive email notifications after enabling these checkboxes, then check the email address in **Settings » General** page to make sure it's correct. If the email address is correct then check your email spam folders.

Enable or Disable Comments on a Single Post in WordPress

You can override the default comments settings for articles by enabling or disabling comments on a single post, page, or custom post type. When editing the post, just uncheck the boxes to disable comments and trackbacks, or check them to enable comments and trackbacks.



What Is an Avatar or Gravatar?

In online communities, the word Avatar is used for a user's profile image. Gravatar is a web-based service that allows users to use the same avatar image on thousands of websites that use Gravatars. Gravatars.com is owned by Automattic which is a company founded by WordPress co-founder Matt Mullenweg.

When a user creates an account or leaves a comment, WordPress looks up for a gravatar image matching their email address. If the user has a Gravatar account with an image, then WordPress displays this image as their profile picture with their comments.

If a user's email address doesn't match a Gravatar account, then WordPress displays a default Avatar image.

The default Avatar image in WordPress is called Mystery Person. This can be changed in **Settings » Discussion** under Avatar section.

Avatar Display
☒ Show Avatars

Maximum Rating
☒ G — Suitable for all audiences
☐ PG — Possibly offensive, usually for audiences 13 and above
☐ R — Intended for adult audiences above 17
☐ X — Even more mature than above

Default Avatar
For users without a custom avatar of their own, you can either display a generic logo or a generated one based on their email address.
☒  Mystery Person
☐ Blank
☐  Gravatar Logo
☐  Identicon (Generated)
☐  Wavatar (Generated)
☐  MonsterID (Generated)
☐  Retro (Generated)

You can choose not to use Avatar images on your website by unchecking “**Show Avatars**” option. But since most WordPress themes have templates designed to display Gravatars, it may not look good on the front end of your website.

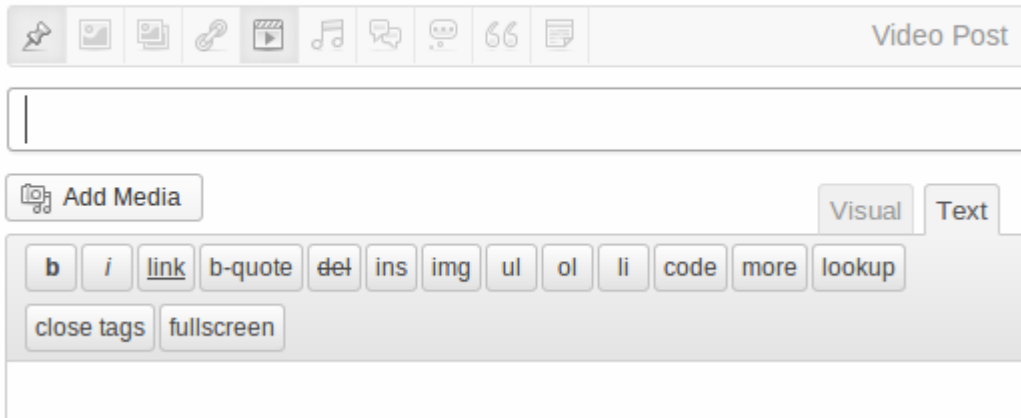
You can choose a maximum rating for your Avatar image. The default option is G which is suitable for all audiences. However, if a user has an Avatar image that could be offensive, then Gravatar automatically displays the default Gravatar image you selected for your site instead.

What is: Post Formats

Post formats is an optional value added to WordPress posts which allows theme developers to define visual representation of a post. Theme developers can create themes with support for post formats. A number of post formats are available, however it is not possible for themes or plugins to introduce custom post formats. It is not necessary for theme developers to support all post formats.

- Standard – The default post format
- Aside – A note like post, usually styled without title.
- Gallery – A gallery of images.
- Link – A link to another site.
- Image – An image or photograph
- Quote – A quotation.
- Status – Twitter like short status update
- Video – A post containing video
- Audio – An audio file.
- Chat – A chat transcript

Add New Post



Standard Post Format

Standard is the default post format in WordPress. It can be an article, a blog post, or anything the user want it to be. A standard post can also be any of the other post formats as well. For example, a standard post can have a gallery or a video. The user can decide whether they want to use their theme's built in support for the formatting and display of a particular post format or they would rather use the standard post format.

Aside

Aside is an extra bit of information that a blogger may want to provide to their readers without writing a standard post about it. It could be an external link, reference to a discussion carried out elsewhere on the web, or an interesting piece of information that does not fit in the regular scope of the blog's posts.

Aside in WordPress is one of the supported post formats. Theme developers can choose to provide support for a post format to define its visual representation.

Gallery

Gallery feature allows you to add multiple images in a WordPress post or page. You can add multiple galleries in a single post. Gallery is also one of the supported post formats so theme developers can add support for it and define the gallery presentation in their theme. An image gallery in WordPress can be inserted using the *Add Media* button and then clicking on "Create Gallery" tab. A gallery is inserted in the post using a shortcode, like this:

The ids are the ids of images in WordPress media, allowing users to easily include images to the gallery.

Link

A link post format contains a link to a web location. Ideally it is used when a user just want to share a link instead of writing a post. They can just add a title of the link and the URL or optionally add their own commentary for the link.

Image

An image post format in WordPress is used to display a single image or photograph.

Quote Post Format

Quote is one of the post formats supported by WordPress. It is used for quotations, especially when a user wants to just share a quote which is not within a standard post or article. A user may decide to add or to wrap a quote around `<blockquote>` HTML tag.

Status Post Format

Status is one of the post formats supported by WordPress. A status is usually a short, twitter-like, status update. However, it is not necessary for a status to be short or twitter-like. A user may choose to write longer status updates.

Video Post Format

Video is one of the post formats supported by WordPress post formats system. A post in video post format usually contains a video either embedded from a third party video hosting service like YouTube or uploaded and played directly from WordPress. Since WordPress version 3.6 there is support for native video upload and playback.

Audio Post Format

Audio is one of the supported post formats in WordPress post formats system. A post with the audio post type usually contains an audio file embedded from a third party hosting service or uploaded directly through WordPress media uploader.

Chat

Chat is one of the post formats supported by WordPress. It is used to display a chat transcript.

How to Create a Separate Page for Blog Posts in WordPress

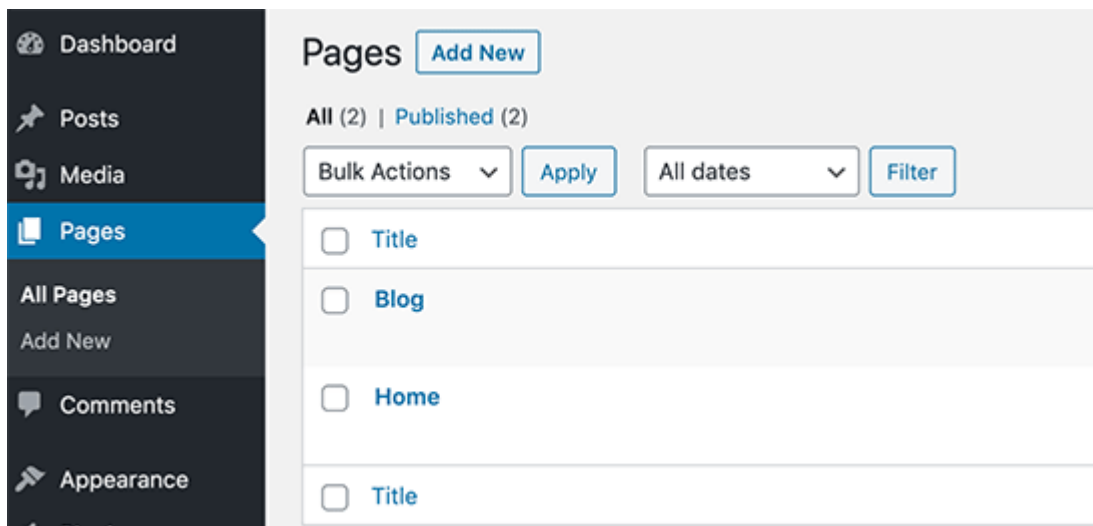
Do you want to create a separate page for blog posts in WordPress? By default, WordPress displays your blog posts on the homepage.

However, if you are making a business website or simply want to customize your homepage, then you may want a separate page for your blog section.

Setup WordPress to Create a Separate Blog Page

WordPress allows you to easily configure your website to have a custom home page. This feature also allows you to choose any other page on your website to be used as your blog page.

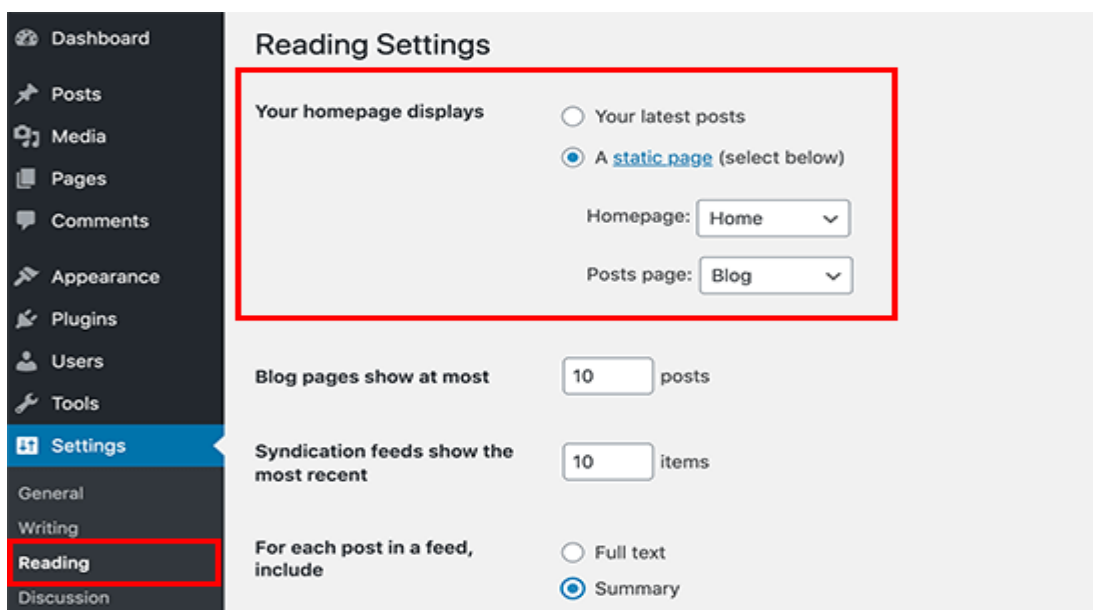
First, you need to create two pages in WordPress that you would like to use as your homepage and blog page.



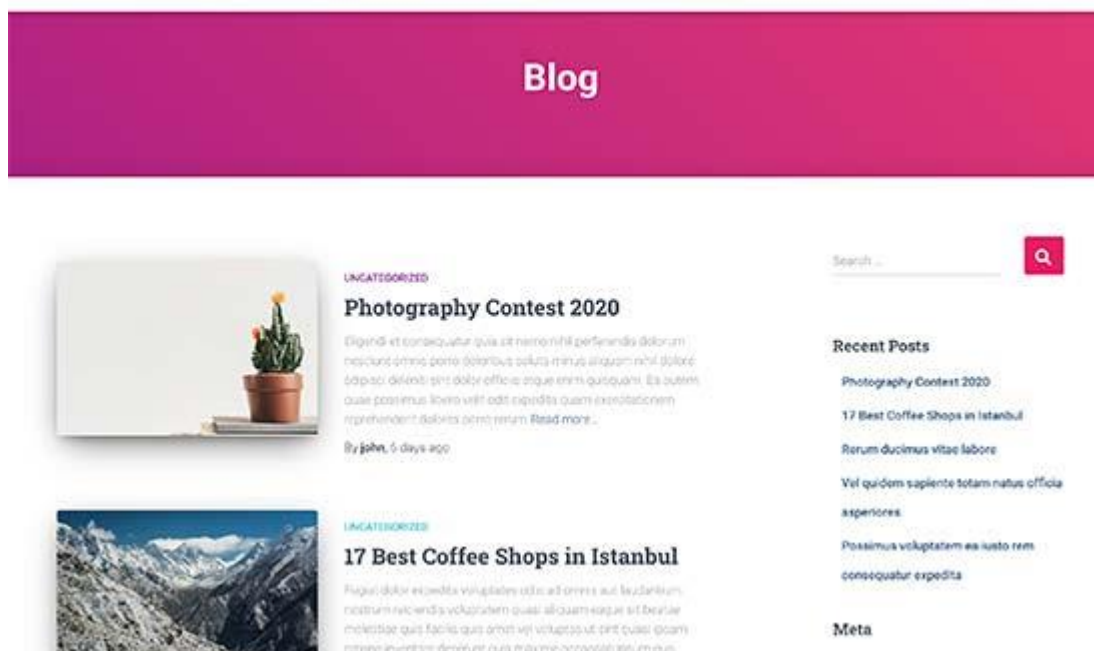
You don't need to add any content to those pages, and you can even name them anything else if you want. For instance, some business websites use News for their blog page.

Once you have created those pages, you can head over to the **Settings » Reading** page in the WordPress admin area.

Under the 'Your homepage displays' section, you need to click on the 'A static page' option. After that, you can select the home and blog pages you created earlier.



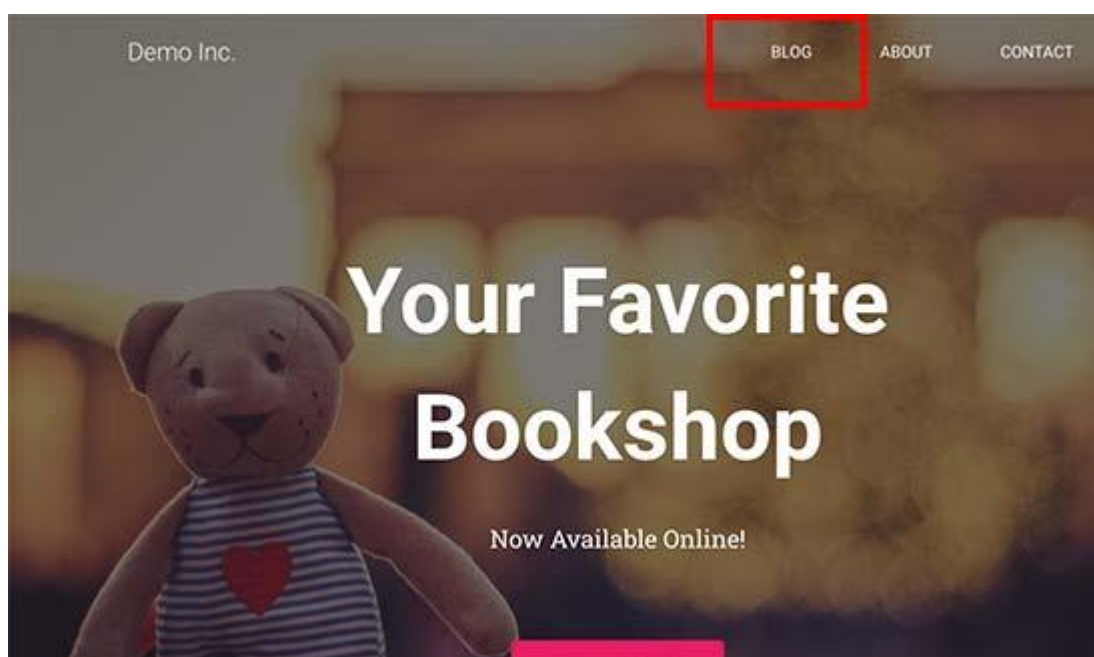
You can now go to **Pages » All Pages** section in admin area. You'll notice your blog and homepage with new labels. Go ahead, and click on the View link under those pages to preview them.



The appearance of these pages depends on your WordPress theme, and you may need to configure your theme settings.

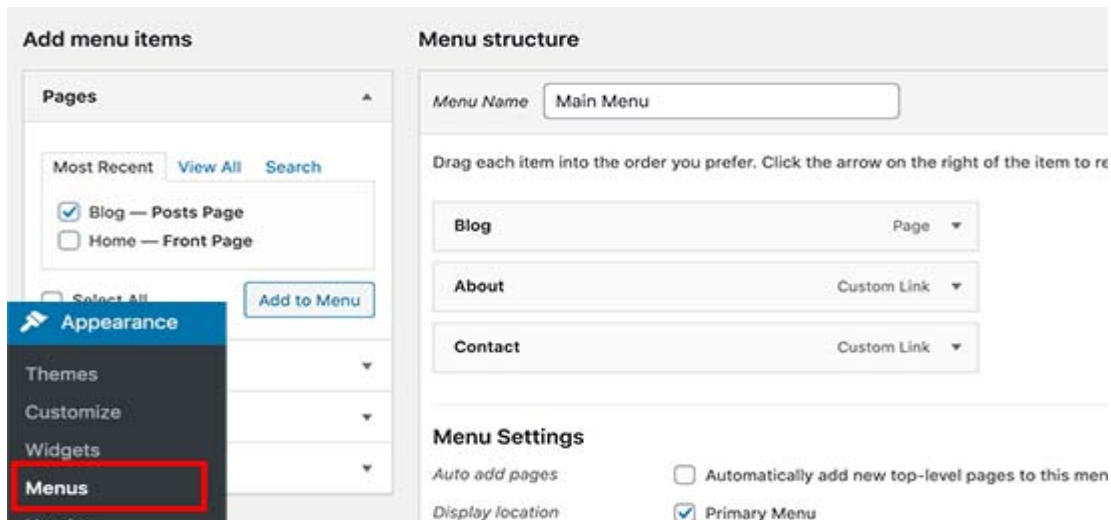
Since each WordPress theme has different options, you'll have to check out your theme's documentation or check the options under **Appearance » Customize** page.

Put the Blog Page Link to Your Website's Navigation Menu



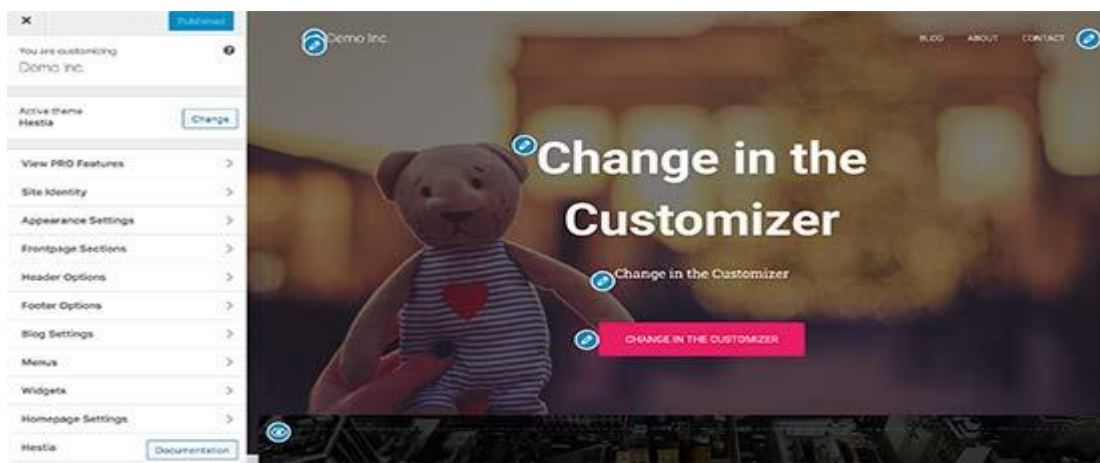
Now that you have created a separate blog page let's help your website visitors find that.

Simply visit the **Appearance » Menus** page and add the Blog page you created earlier to your website's navigation menu.

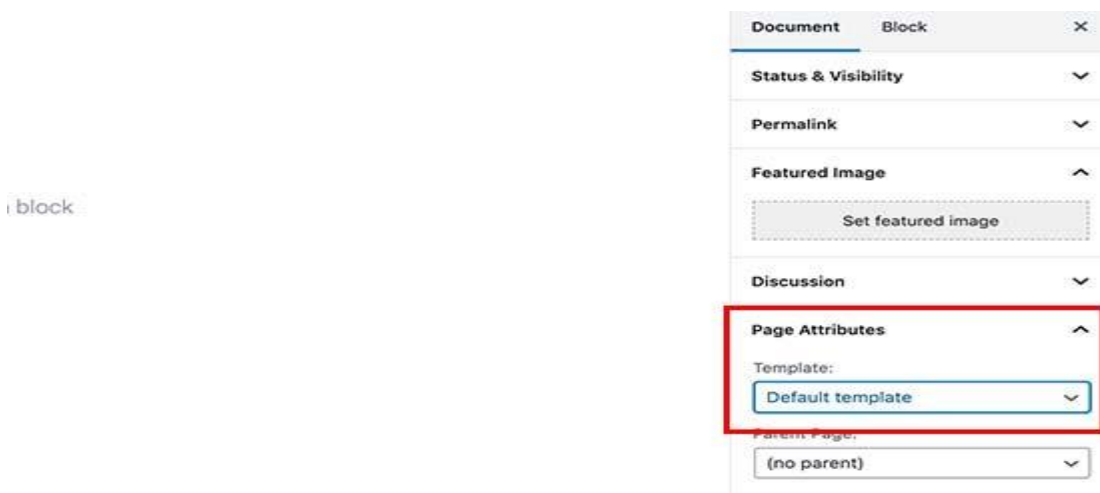


Customizing Your Homepage

By default, many WordPress themes come with a built-in template to display a static page as your website's front page. However, you'll still need to set it up under **Appearance » Customize**.

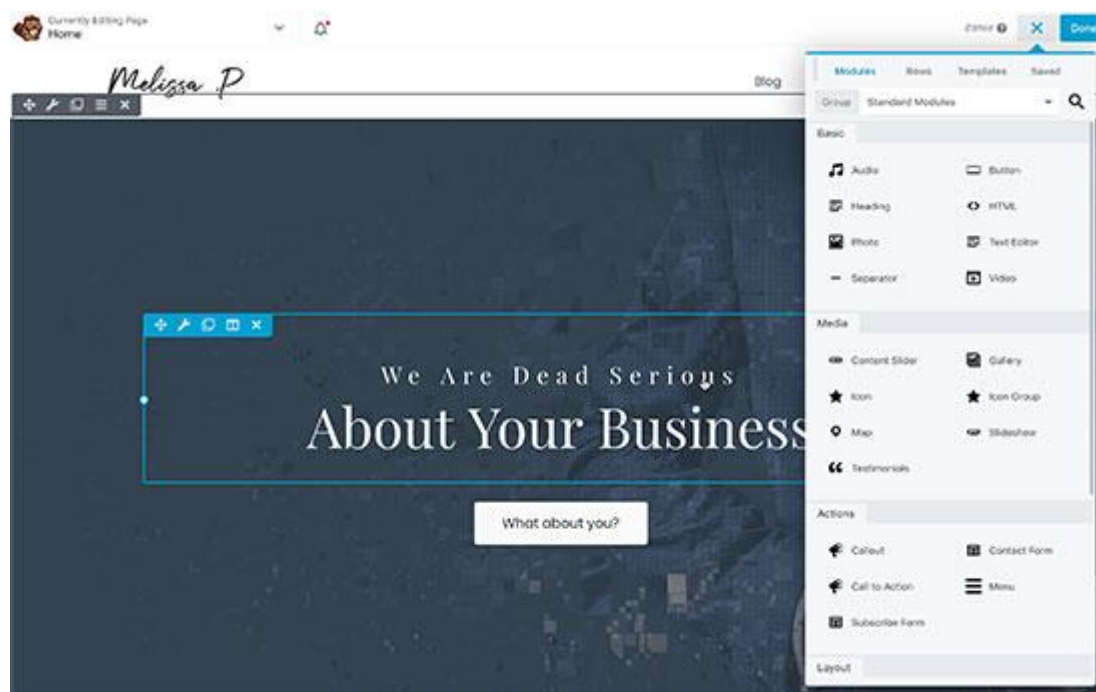


Some WordPress themes also allow you to choose a homepage template for your page. If your theme comes with such a template, then you can select it by editing the Home page.



Finally, you can use a WordPress page builder plugin to use simple drag and tools to create your homepage from scratch.

We recommend using Beaver Builder as it is the best WordPress WordPress page builder plugin on the market. It is super-easy to use, includes ready-made templates, and super-flexible to create any layout you want.



Customize Your Blog Page Design

By default, WordPress automatically looks for a template in your WordPress theme to display posts on a separate blog page.

If your theme does not have one, then it automatically falls back to the archive template. For more details, see our guide on the WordPress template hierarchy.

Now, if you don't like how your theme handles your blog page, then you can change that using some extra tools and without writing any code.

First, you will need to install and activate the Beaver Builder and Beaver Themer plugins. For more details, see our step by step guide on how to install a WordPress plugin.

Beaver Themer is an add-on plugin for Beaver Builder and enables you to not just create pages but entire new WordPress themes from scratch.

First, you need to visit **Beaver Builder » Themer Layouts** page and click on the Add New button.

Add New

Add new builder content using the form below.

Title	Blog
Type	Themer Layout
Layout	Archive

[Add Themer Layout](#)

From here you can provide a name for your layout, select ‘Themer Layout’ as your type, and then select Archive as your layout.

This will create a themer layout for the archive templates. On the next screen, you need to select ‘Post Archive’ for location option and then click on the ‘Launch Beaver Builder’ button.

Blog

Beaver Builder

[Launch Beaver Builder](#) [View Themer Layout](#)

Themer Layout Settings

Type	Archive
Location	Post Archive

[Add Location Rule](#) [Add Exclusion Rule](#)

Rules

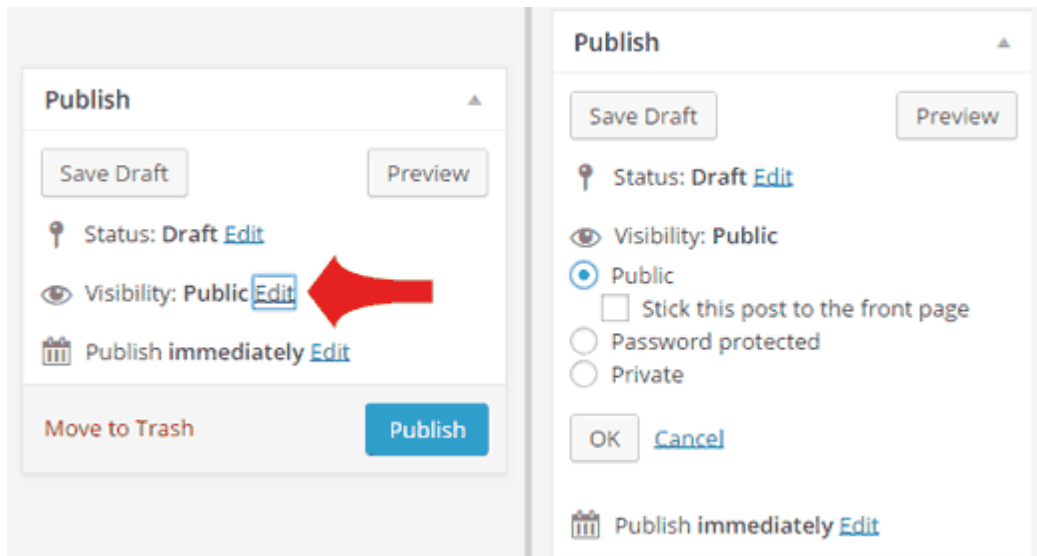
[Add Rule Group](#)

This will open the Beaver Builder with your selected layout. You can simply point and click at any item on the screen and start editing it. You can also add new rows, columns, and web elements to the page.

How to Password Protect a Page or Post in WordPress

Have you ever thought about creating a blog post just for close friends or family members? Or publish a post just for editors on your WordPress site? As a content publishing platform, WordPress comes with some basic content visibility options that most beginners overlook. In this article, we will show you how to create a password protected page or post in WordPress.

When writing posts, there is an option called *Visibility*, which is set to Public by default. If you click Edit, then you will see options such as Password Protected and Private.



Simply click on Password Protected to add a password to your post. Doing so, only the friends, family members, or users you share this password with will be able to see this post.

A private post in WordPress is visible only to logged in users with the Editor or Administrator level user role in WordPress.

The same trick works for password protecting a page in WordPress as well.

Experiment – 4

Aim: - Practice various designing features: Colour Scheme, Headers, CSS Horizontal Menus, Dynamic Menu, Highlighting, Navigation Links, and Print

Style WordPress Navigation Menus

Do you want to style your WordPress navigation menus to change their colors or appearance? While your WordPress theme handles the appearance of your navigation menus, you can easily customize it using CSS to meet your requirements.

We will be showing two different methods. The first method is for beginners because it uses a plugin and does not require any code knowledge. The second method is for intermediate DIY users who prefer to use CSS code instead of a plugin.

Method 1: Styling WordPress Navigation Menus Using a Plugin

Your WordPress theme uses CSS to style navigation menus. Many beginners are not comfortable with editing theme files or writing CSS code on their own.

That's when a WordPress styling plugin comes in handy. It saves you from editing theme files or writing any code.

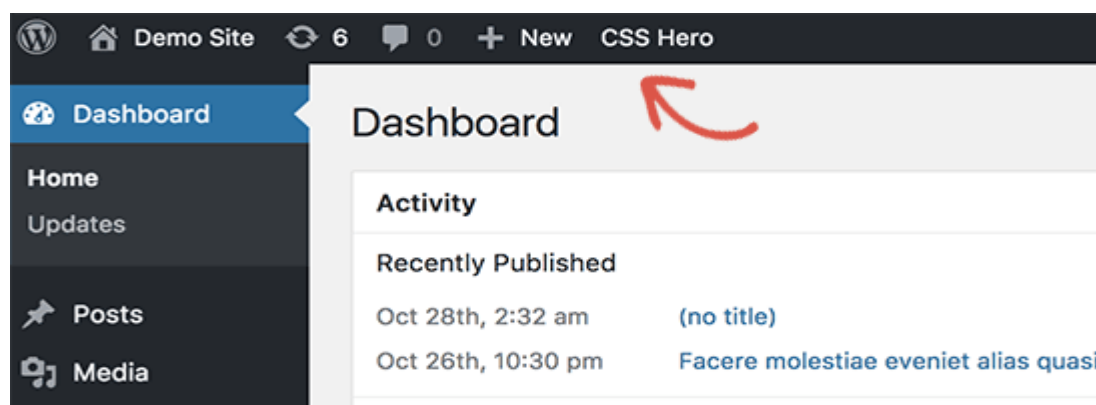
First you need to do is install and activate the CSS Hero plugin. For more details, see our step by step guide on how to install a WordPress plugin.

CSS Hero is a premium WordPress plugin that allows you to design your own WordPress theme without writing a single line of code (No HTML or CSS required). See our CSS Hero review to learn more.

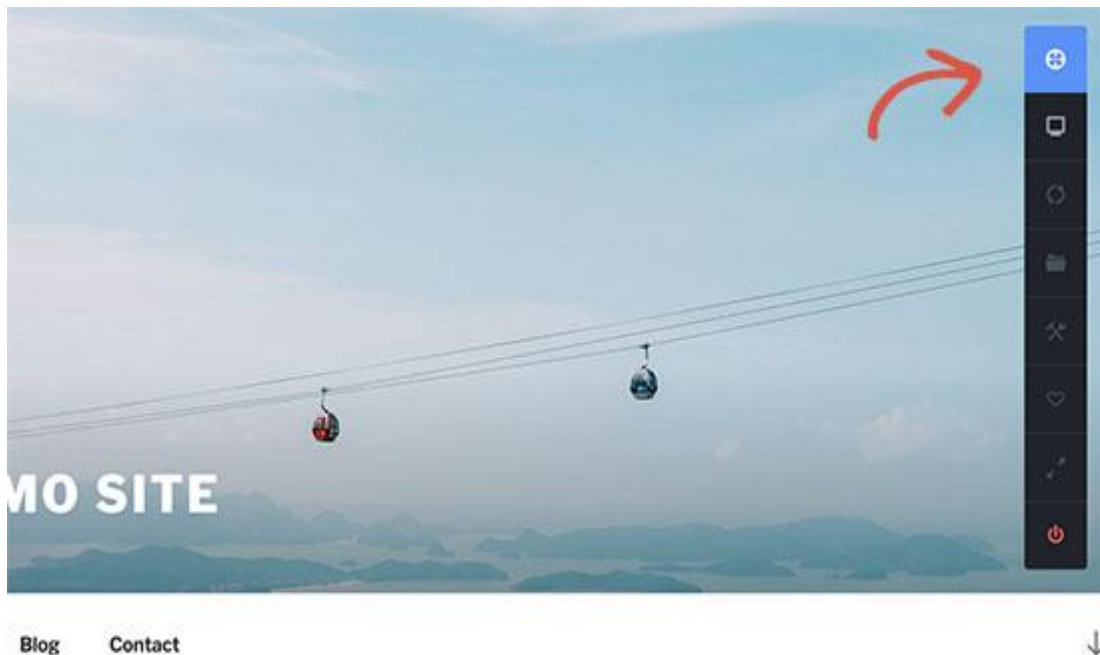
WPBeginner users can use use this CSS Hero Coupon to get 34% discount on their purchase.

Upon activation, you will be redirected to obtain your CSS Hero License key. Simply follow the on-screen instructions, and you will be redirected back to your site in a few clicks.

Next, you need to click on the CSS Hero button in your WordPress admin toolbar.

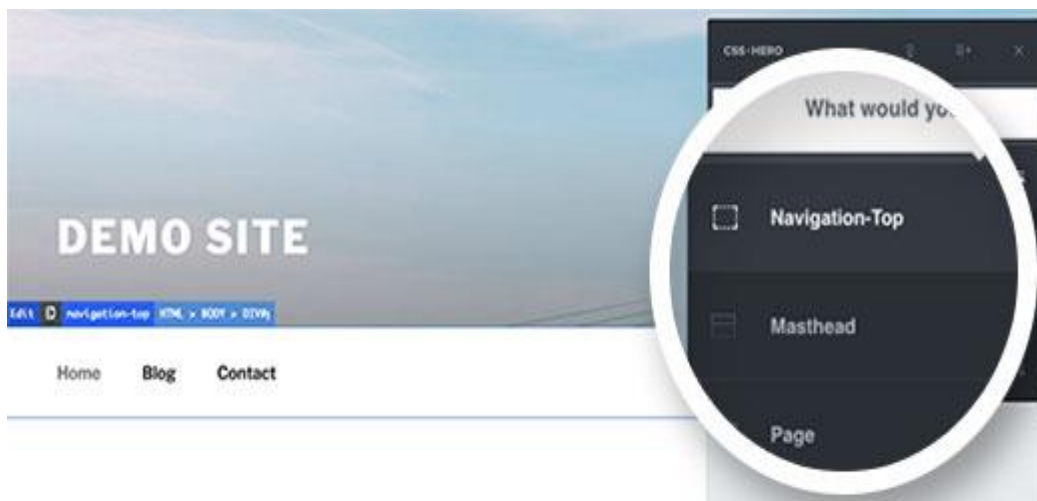


CSS Hero offers a WYSIWYG (What you see is what you get) editor. Clicking on the button will take you to your website with a floating CSS Hero toolbar visible on screen.



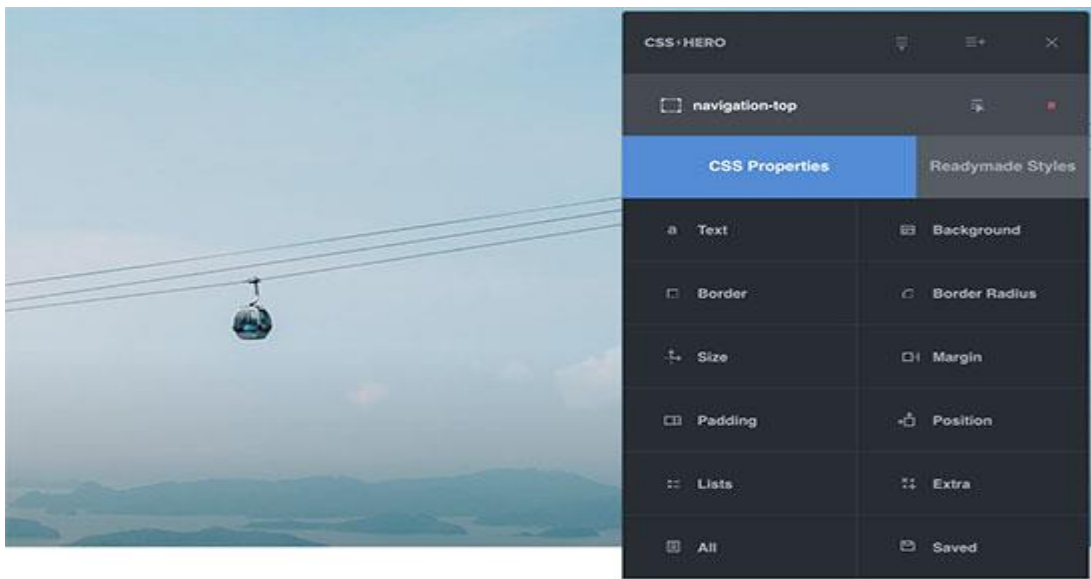
You need to click on the blue icon at the top to start editing.

After you click the blue icon, take your mouse to your navigation menu, and CSS Hero will highlight it by showing the borders around it. When you click on the highlighted navigation menu, it will show you the items that you can edit.

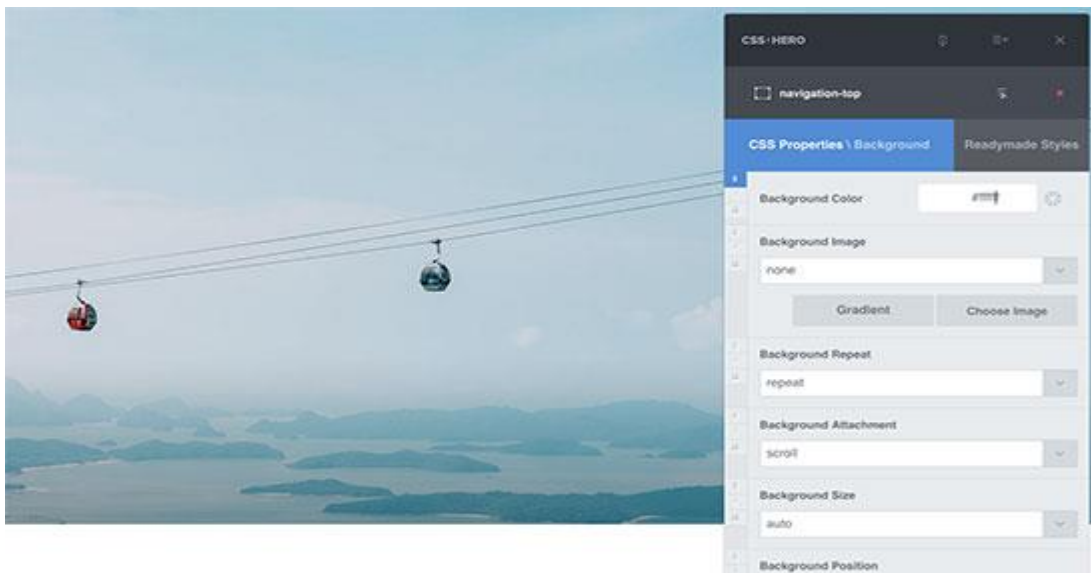


In the screenshot above, it shows us top navigation menu container. Let's assume we want to change the background color of our navigation menu. In that case, we will select top navigation which affects our entire menu.

CSS Hero will now show you different properties that you can edit like text, background, border, margins, padding, etc.

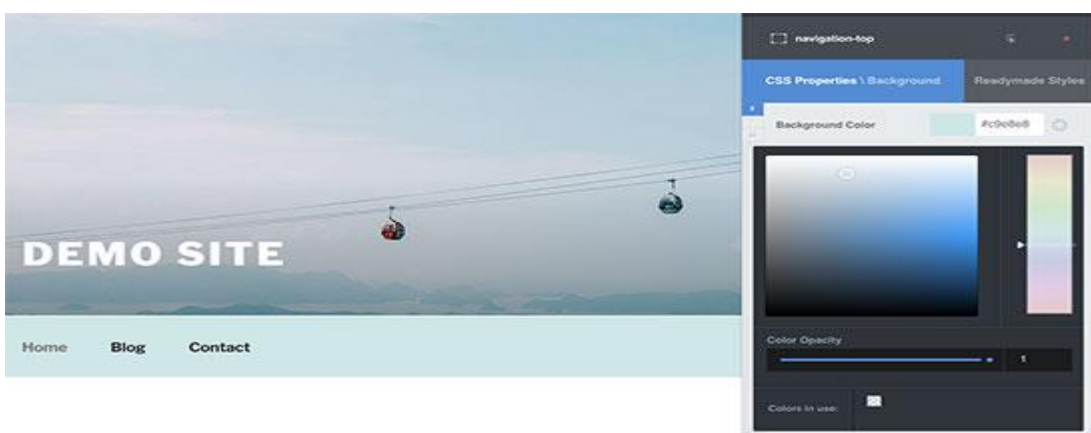


You can click any property that you want to change. CSS Hero will show you a simple interface where you can make your changes.



In the screenshot above, we selected background, and it showed us a nice interface to select background color, gradient, image, and more.

As you make changes, you will be able to see them live in the theme preview.



Once you are satisfied with the changes, click on the save button in CSS Hero toolbar to save your changes.

The best thing about using this method is that you can easily undo any changes that you make. CSS Hero keeps a complete history of all your changes, and you can go back and forth between those changes.

Method 2: Manually Style WordPress Navigation Menus

This method requires you to manually add custom CSS and is meant for intermediate users.

WordPress navigation menus are displayed in an unordered list (bulleted list).

Typically if you use the default WordPress menu tag, then it will display a list with no CSS classes associated with it.

```
1 <?php wp_nav_menu(); ?>
```

Your unordered list would have the class name 'menu' with each list item having its own CSS class.

This might work if you only have one menu location. However, most themes have multiple locations where you can display navigation menus.

Using only the default CSS class may cause conflict with menus on other locations.

This is why you need to define CSS class and menu location as well. Chances are that your WordPress theme is already doing that by adding the navigation menus using a code like this:

```
1 <?php
2     wp_nav_menu( array(
3         'theme_location' => 'primary',
4         'menu_class'     => 'primary-menu',
5     ) );
6 ?>
```

This code tells WordPress that this is where the theme displays primary menu. It also adds a CSS class primary-menu to the navigation menu.

Now you can style your navigation menu using this CSS structure.

```
1 // container class
2 #header .primary-menu{ }
3
4 // container class first unordered list
5 #header .primary-menu ul { }
6
7 //unordered list within an unordered list
8 #header .primary-menu ul ul { }
9
10 // each navigation item
11 #header .primary-menu li { }
12
```

```

13 // each navigation item anchor
14 #header .primary-menu li a {}
15
16 // unordered list if there is drop down items
17 #header .primary-menu li ul {}
18
19 // each drop down navigation item
20 #header .primary-menu li li {}
21
22 // each drop down navigation item anchor
23 #header .primary-menu li li a {}

```

You will need to replace #header with the container CSS class used by your navigation menu.

This structure will help you completely change the appearance of your navigation menu.

However, there are other WordPress generated CSS classes automatically added to each menu and menu item. These classes allow you to customize your navigation menu even further.

```

// Class for Current Page
.current_page_item{}

// Class for Current Category
.current-cat{}

// Class for any other current Menu Item
.current-menu-item{}

// Class for a Category
.menu-item-type-taxonomy{}

// Class for Post types
.menu-item-type-post_type{}

// Class for any custom links
.menu-item-type-custom{}

// Class for the home Link
.menu-item-home{}

```

WordPress also allows you to add your own custom CSS classes to individual menu items.

You can use this feature to style menu items, like adding image icons with your menus or by just changing colors to highlight a menu item.

Head over to **Appearance » Menus** page in your WordPress admin and click on the Screen Options button.

Boxes

☒ Pages
 ☒ Posts
 ☒ Custom Links
 ☒ Categories
 ☐ Tags
 ☒ Format

Show advanced menu properties

☐ Link Target
 ☒ Title Attribute
 ☒ **CSS Classes**
☐ Link Relationship (XFN)
 ☐ Description

Menus [Manage with Live Preview](#)

[Edit Menus](#)
[Manage Locations](#)

Select a menu to edit: ShortMenu (Top Menu) ▾ Select or [create a new menu.](#)

Once you have checked that box, you will see that an additional field is added when you go to edit each individual menu item.

Menu Structure

Drag each item into the order you prefer. Click the arrow on the right of the item to reveal additional configurations.

Home Custom Link ▾

Blog Page ▲

Navigation Label
 Blog

Title Attribute

CSS Classes (optional)
 blog-menu-item

Move [Up one](#) [Down one](#) [Under Home](#) [To the top](#)

Original: [Blog](#)

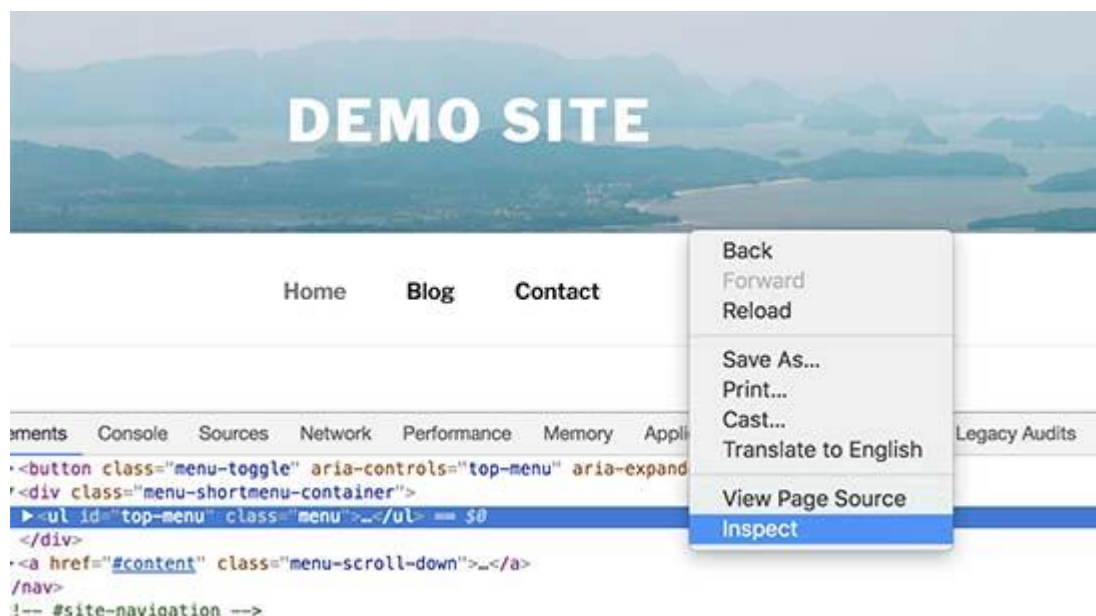
Now you can use this CSS class in your stylesheet to add your custom CSS. It will only affect the menu item with the CSS class you added.

Examples of Styling Navigation Menus in WordPress

Different WordPress themes may use different styling options, CSS classes, and even JavaScript to create navigation menus. This gives you a lot of options to change those styles and customize your navigation menus to meet your own requirements.

The inspect tool in your web browser will be your best friend when it comes to figuring out which CSS classes to change. If you haven't used it before, then take a look at our guide on how to use inspect tool to customize WordPress themes.

Basically, you just need to point the cursor to the element you want to modify, right click and then select Inspect tool from browser's menu.



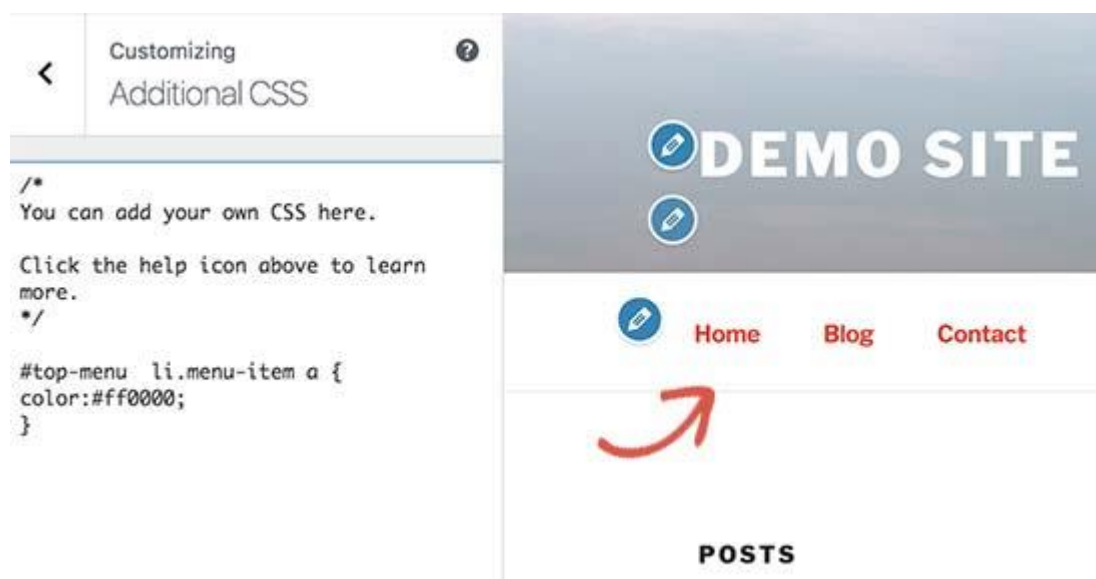
That being said, let's take a look at some real life examples of styling navigation menus in WordPress.

1. How to Change Font Color in WordPress Navigation Menus

Here is the sample custom CSS that you can add to your theme to change font color of navigation menus.

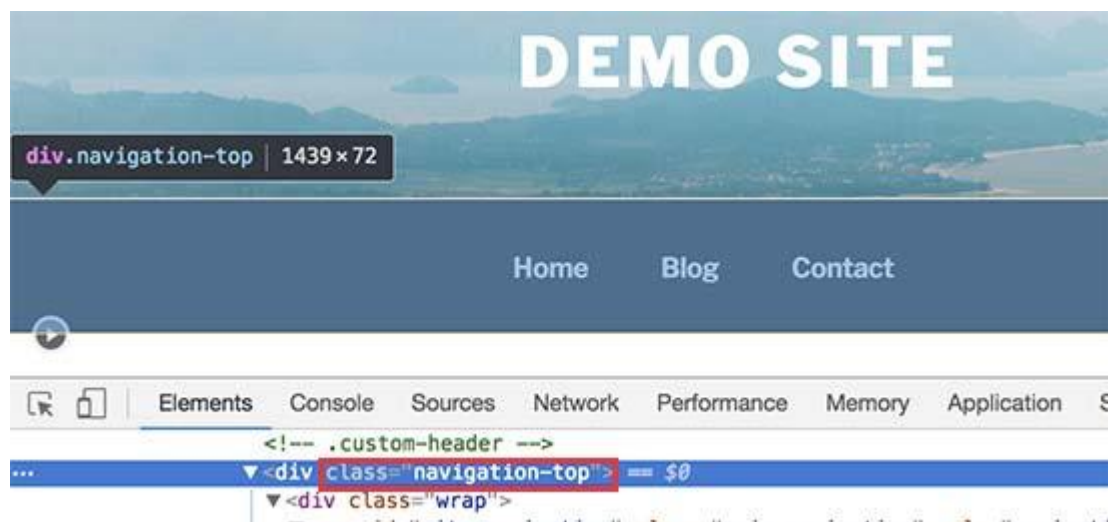
```
1 #top-menu li.menu-item a {  
2   color:#ff0000;  
3 }
```

In this example, the #top-menu is the ID assigned to the unordered list that displays our navigation menu. You will need to use the inspect tool to find out the ID used by your theme.



2. How to Change Background Color of Navigation Menu Bar

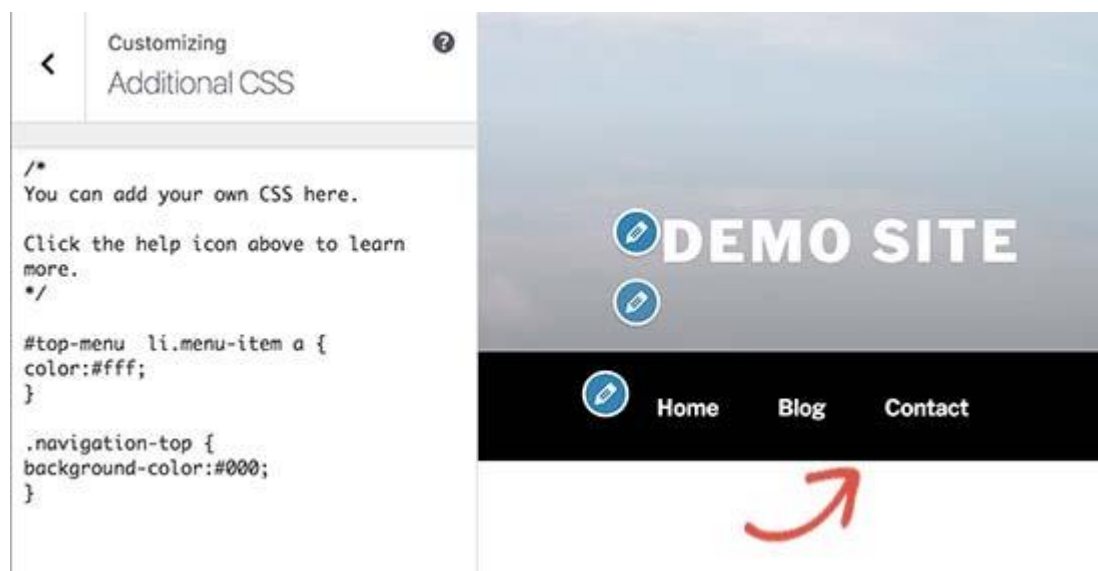
First you'll need to find out the CSS ID or class used by your theme for the container surrounding navigation menu.



After that you can use the following custom CSS to change background color of navigation menu bar.

```
1 .navigation-top {  
2   background-color:#000;  
3 }
```

Here is how it looked on our demo website.

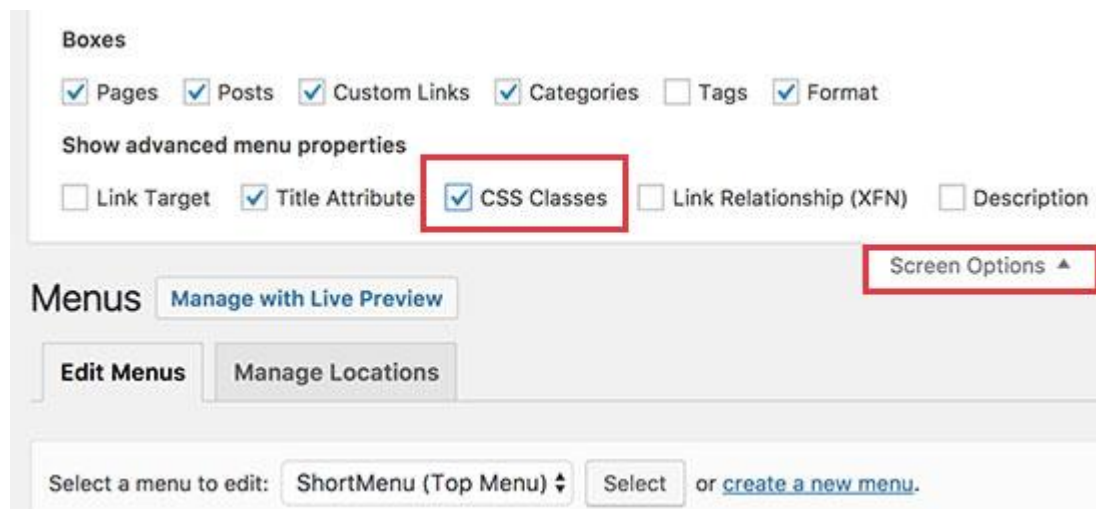


3. How to Change Background Color of a Single Menu Item

You may have noticed that many websites use a different background color for the most important links in their navigation menu. This link could be a login, sign up, contact, or buy button. By giving it a different color, it makes the link more noticeable.

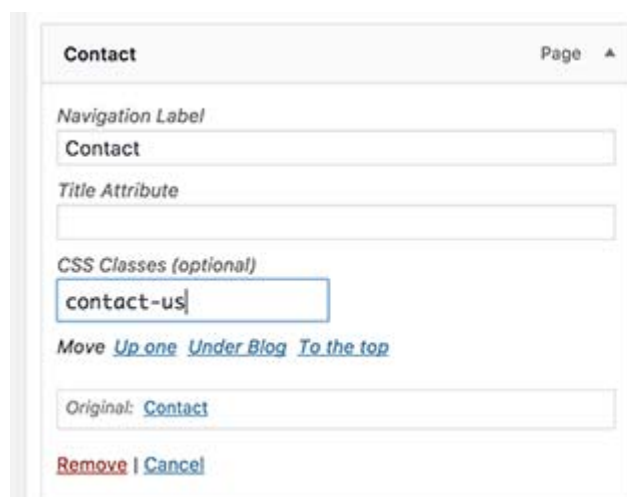
To achieve this, we will add a custom CSS class to the menu item that we want highlight with a different background color.

Head over to **Appearance » Menus** and click on the Screen Options button at the top right corner of the screen. This will bring up a fly down menu where you need to check the box next to ‘CSS classes’ option.



The screenshot shows the WordPress 'Menus' screen. At the top, under 'Boxes', there are checkboxes for 'Pages', 'Posts', 'Custom Links', 'Categories', 'Tags', and 'Format'. Below this, under 'Show advanced menu properties', there are checkboxes for 'Link Target', 'Title Attribute', 'CSS Classes', 'Link Relationship (XFN)', and 'Description'. The 'CSS Classes' checkbox is checked and highlighted with a red box. To the right of the 'Show advanced menu properties' section, there is a 'Screen Options' button with an upward arrow, also highlighted with a red box. Below the 'Boxes' section, there are buttons for 'Edit Menu' and 'Manage Locations'. At the bottom, there is a dropdown menu for 'Select a menu to edit:' with 'ShortMenu (Top Menu)' selected, and a 'Select' button next to it. There is also a link to 'or create a new menu.'

After that you need to scroll down to the menu item that you want to modify and click to expand it. You will notice a new option to add your custom CSS class.



The screenshot shows a single menu item titled 'Contact' expanded. It has a 'Page' dropdown set to 'Page'. Below the title, there is a 'Navigation Label' field with the value 'Contact'. Below that is a 'Title Attribute' field. Below the 'Title Attribute' field is a 'CSS Classes (optional)' field with the value 'contact-us'. Below the 'CSS Classes' field are links for 'Move Up one', 'Under Blog', and 'To the top'. Below these links is a field for 'Original:' with the value 'Contact'. At the bottom are links for 'Remove' and 'Cancel'.

Now you can use this CSS class to style that particular menu item differently.

```
1 .contact-us {  
2   background-color: #ff0099;  
3   border-radius:5px;  
4 }
```

Here is how it looked on our test site.

```
.navigation-top {  
background-color:#666;  
}  
  
.contact-us {  
background-color: #ff0099;  
border-radius:5px;  
}
```



4. Adding Hover Effects to WordPress Navigation Menus

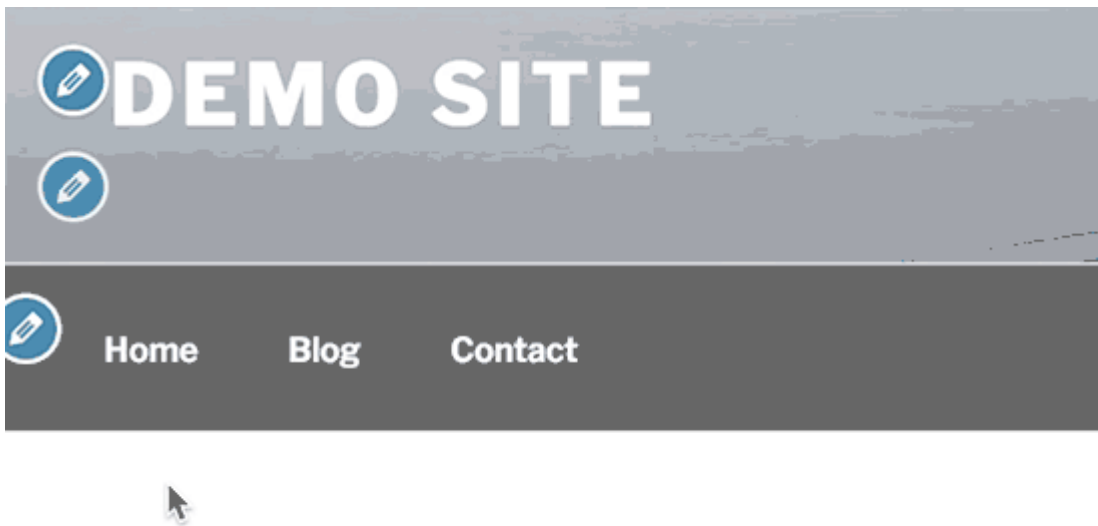
Do you want your menu items to change colors on mouse-over? This particular CSS trick makes your navigation menus look more interactive.

Simply add the following custom CSS to your theme.

```
1 #top-menu li.menu-item a:hover {  
2 background-color:#fff;  
3 color:#666;  
4 border-radius:5px;  
5 }
```

In this example, #top-menu is the CSS ID used by your theme for the unordered navigation menu list.

Here is how this looked on our test site.



5. Create Sticky Floating Navigation Menus in Wordpress

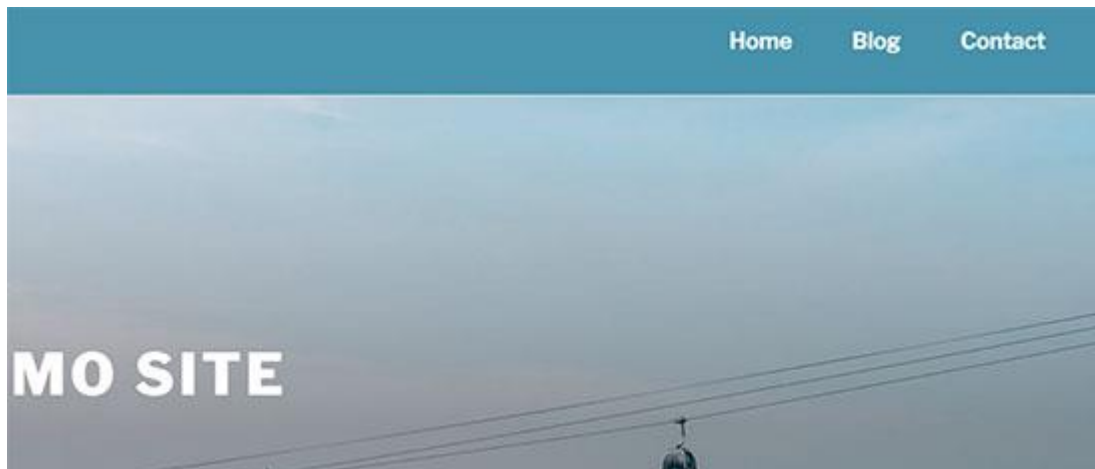
Normally navigation menus appear on top and disappear as a user scrolls down. Sticky floating navigation menus stay on top as a user scrolls down.

You can add the following CSS code to your theme to make your navigation menus sticky.

```
1  #top-menu {  
2    background:#2194af;  
3    height:60px;  
4    z-index:170;  
5    margin:0 auto;  
6    border-bottom:1px solid #dadada;  
7    width:100%;  
8    position:fixed;  
9    top:0;  
10   left:0;  
11   right:0;  
12   text-align: right;  
13   padding-right:30px  
14 }
```

Simply replace #top-menu with the CSS ID of your navigation menu.

Here is how it looked in our demo:



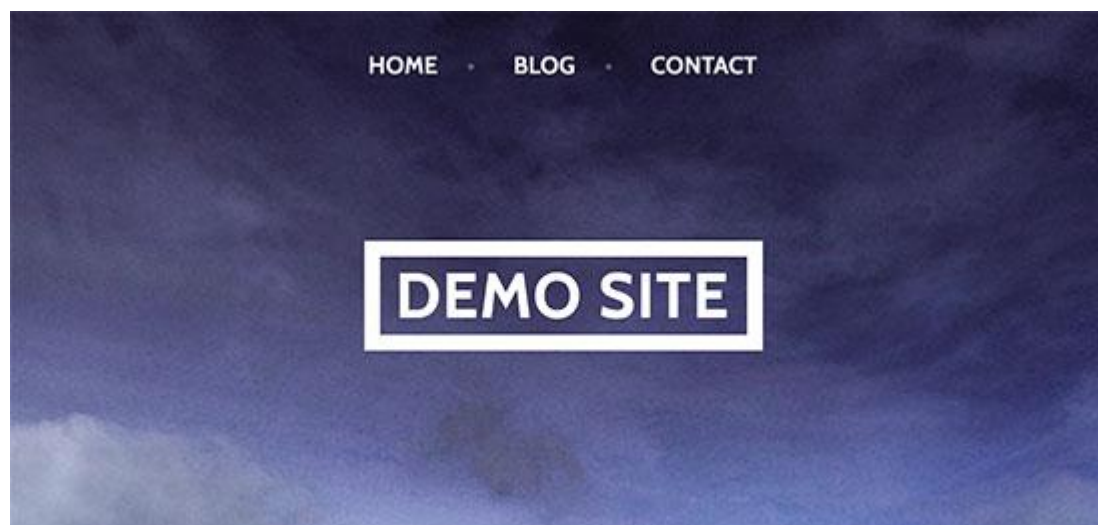
6. Create Transparent Navigation Menus in WordPress

Many websites use large or fullscreen background images with their call to action buttons. Using transparent menus makes your navigation blend in with the image. This makes users more likely to focus on your call to action.

Simply add the following sample CSS to your theme to make your navigation menus transparent.

```
1  #site-navigation {  
2    background-color:transparent;  
3  }
```

This is how it looked on our demo site.



Depending on your theme, you may need to adjust the position of the header image so that it covers the area behind your transparent menus.

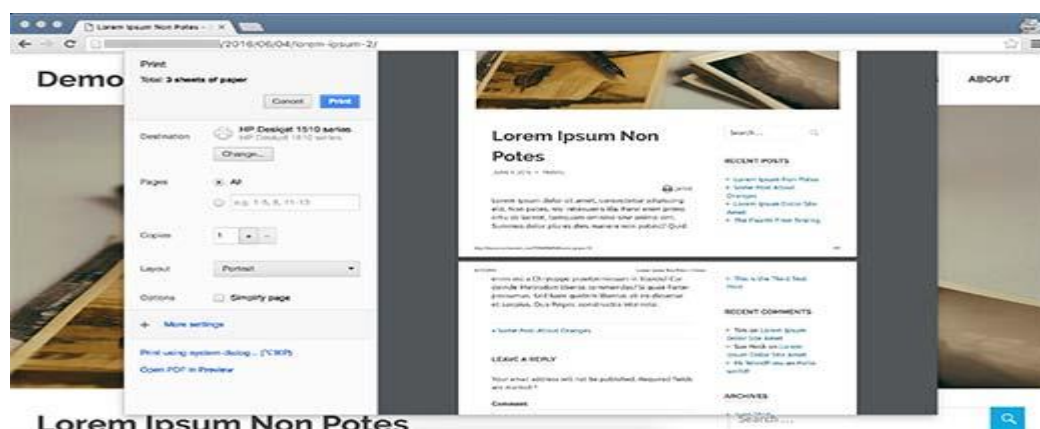
How to Add a Printer Friendly Option to Your WordPress Posts

Often users want to print articles they find interesting. By default, WordPress itself and most WordPress themes would print your page as it is, with all the graphics, colors, and text in sidebars.

Why and When You Need a Printer Friendly Option for WordPress Posts?

Some WordPress themes come with a separate CSS stylesheet for printing. This stylesheet is used when a user prints a page. Other WordPress themes, may not change your site's appearance when a user prints it.

Usually this results into users printing all the images, sidebars, headers, menus, etc. Most of the time it is not very pleasant to read or look at.



You can see how your WordPress theme handles print by pressing CTRL+P when looking at a page from your site. Your browser will show print preview of your site.

If your theme shows your site as it appears in the browser, then it is not handling print any differently.

In that case, you can use a printer friendly option on your WordPress site. It will not only make your content printer friendly, it will also encourage users to print with an on-screen print icon.

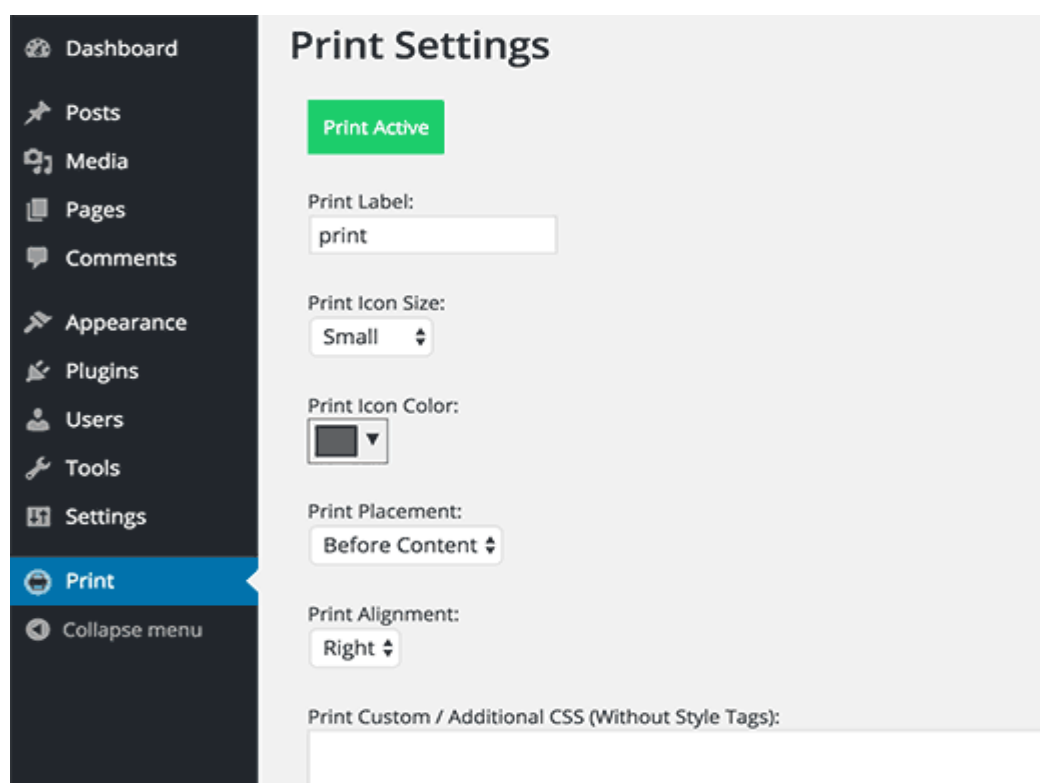
We will be showing you two plugins that can add a print option to WordPress, and you can choose the one that best suits your need.

Method 1: Add Print Button in WordPress with Print Post and Page Plugin

This method is recommended for users who do not want to add code to their WordPress theme files.

First thing you need to do is install and activate the [Print Post and Page](#) plugin.

Upon activation the plugin will add a new menu item labeled ‘Print’ to your WordPress admin bar. Clicking on it will take you to the plugin’s settings page.



Here you can choose the print icon size and color, text to accompany the icon, and print alignment. Optionally, you can add custom CSS that you want to include into your print template.

Don’t forget to click on the ‘Save and Activate’ button to store your plugin settings.

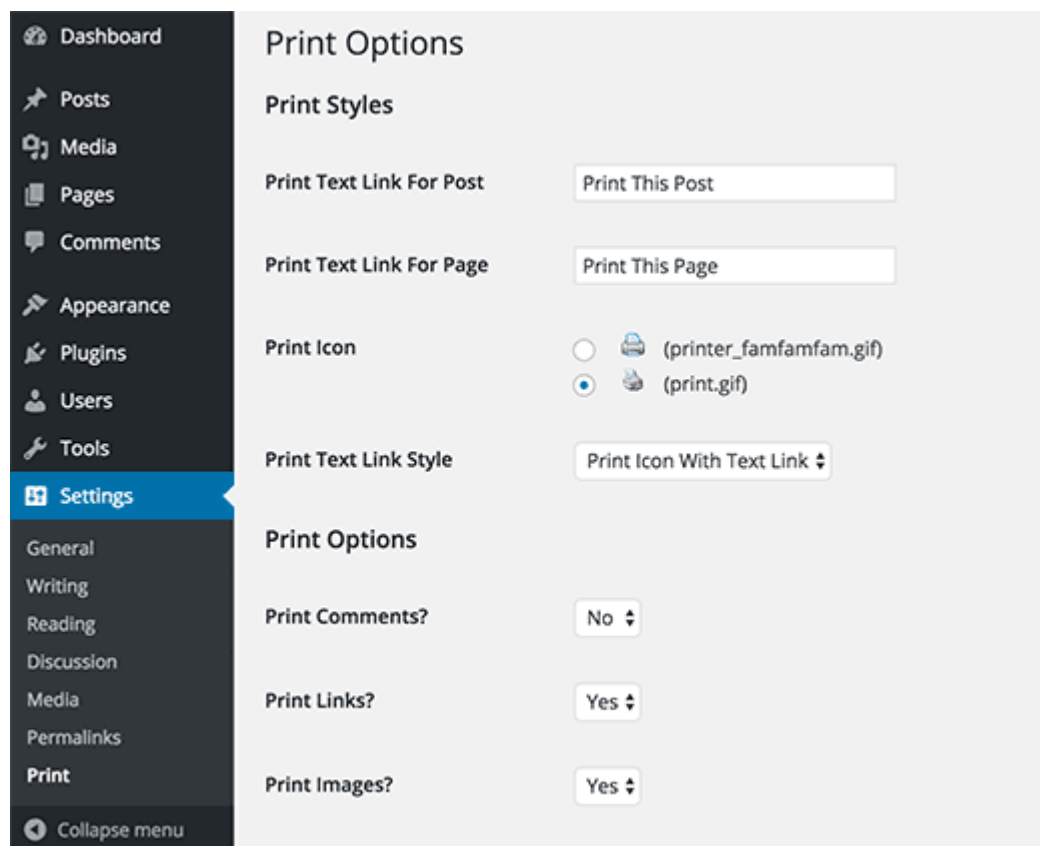
You can now visit your WordPress site and you will see a print button on your single posts.

However, the plugin does not automatically adds print button to your WordPress pages. You will need to add `[printicon align="left"]` shortcode to your WordPress pages to display the print icon on pages.

Method 2: Adding Printer Friendly Option Using WP-Print

First thing you need to do is install and activate the WP-Print plugin. For more details, see our step by step guide on how to install a WordPress plugin.

Upon activation, you need to visit **Settings » Print** to configure plugin settings.



You can display print icon with text, just the icon, or just the text. You can choose between two icons, and you also change the text that appears with the icon.

Under print options, you will be asked if you want to print comments, links, images, or videos. By default, the plugin does not print comments and videos. You can change that if you want the plugin to print comments and videos as well.

At the bottom of each printed post, the plugin will add a disclaimer/copyright text. By default, it will show your site's name. You can change that to anything you want, and you can also use HTML tags inside the disclaimer field.

Don't forget to click on the save changes button to store your settings.

Now comes the tricky part. The plugin will not actually add the print button into your posts. You will have to manually edit your WordPress theme files to add the button.

Depending on your theme, you may need to add the code into single.php, page.php, archive.php, content.php, etc.

Locate the line in your theme template that looks like this:

```
1 <?php while (have_posts()) : the_post(); ?>
```

Below this line, you need to add the following code where you want to display the print button:

```
1 <?php if(function_exists('wp_print')) { print_link(); } ?>
```

That's all, you can now visit your website and you will be able to see the print button.

Sometimes you may have content that you do not want to be included into the print version. Simply wrap that content between [donotprint] and [/donotprint] shortcode, like this:

```
1 [donotprint]This text will not be displayed when printing[/donotprint]
```

Experiment – 5

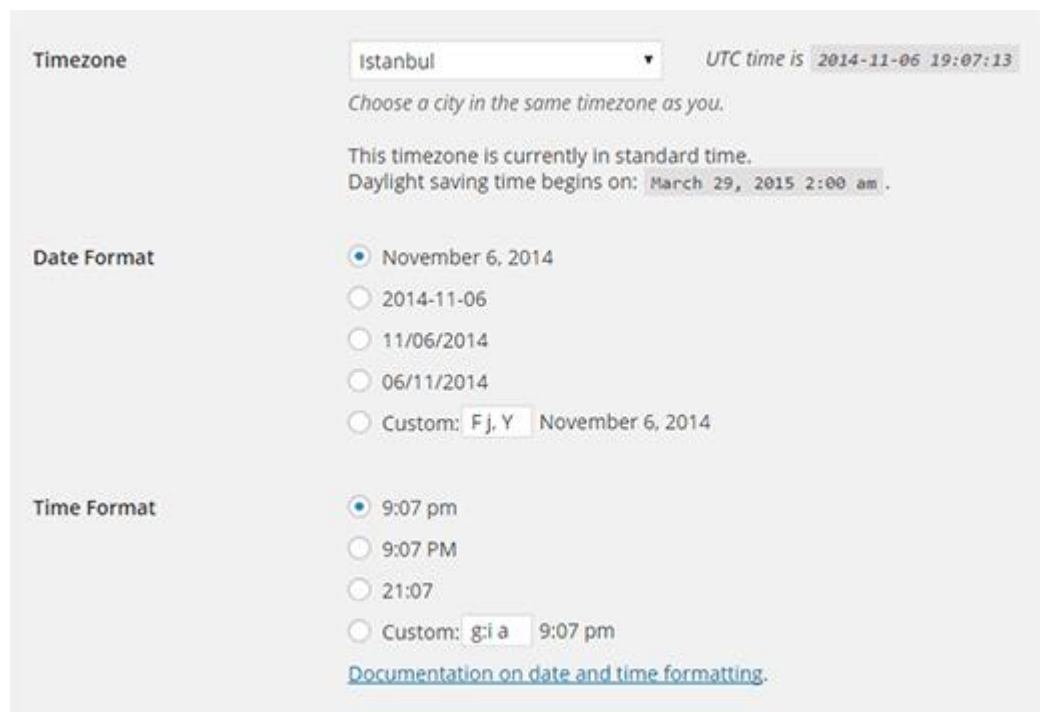
Aim: - Read More, Formatting Date and Time, Finding CSS Styles, Creating Individual Pages, Uploading Files, Using WordPress Themes, Templates, Template Tags, Template Hierarchy, Validating a Website, Know Your Sources, WordPress Site Maintenance

How to Change Date and Time Format in WordPress

Have you ever wanted to change the date format in WordPress? Maybe you wanted to abbreviate the month or hide the year or display time?

Date and Time Settings in WordPress

WordPress comes with built-in functions which allow users as well as developers to format date and time. You can modify your format by going to **Settings » General** page in WordPress admin area and scroll down to timezone section.



The screenshot shows the 'Settings - General' page in WordPress. The 'Timezone' section has a dropdown menu set to 'Istanbul' and a text box showing 'UTC time is 2014-11-06 19:07:13'. Below it, a note says 'Choose a city in the same timezone as you.' and another note says 'This timezone is currently in standard time. Daylight saving time begins on: March 29, 2015 2:00 am.' The 'Date Format' section has four radio button options: 'November 6, 2014' (selected), '2014-11-06', '11/06/2014', and '06/11/2014'. There is also a 'Custom:' option with a text box containing 'F j, Y' and the preview 'November 6, 2014'. The 'Time Format' section has four radio button options: '9:07 pm' (selected), '9:07 PM', '21:07', and 'Custom:'. The 'Custom:' option has a text box containing 'g:i a' and the preview '9:07 pm'. At the bottom, there is a link: '[Documentation on date and time formatting.](#)'

This is where you can choose a timezone for your website along with the date and time formats.

There are couple of built-in options that you can choose from by clicking on the radio button, or you can enter your own custom format.

What are Format Characters for Date and Time in WordPress?

The date and time format are based alphabetic characters that represent a time format. For example, entering Y will output the year in numeric four digits, like 2014.

Entering smaller case y will output the year in numeric two digits, like 14. You can see a full list of [format characters here](#).

Some examples of commonly used date and time format characters:

M d, Y – will output – Nov 06, 2014
d M, Y – will output – 06 Nov, 2014
F jS, Y – will output – November 6th, 2014
l, F jS, Y – will output – Thursday, November 6th, 2014
H:i:s – will output – 21:26:12

You can try different format in the custom box for both date and time settings. After you enter the format, WordPress will show you a preview of what your date or time will look like.

Once you save your settings, it will use this format through out your website unless your theme has pre-defined the format.

Another commonly asked question about date and time formats is how to display time next to dates on your WordPress posts.

The difficult but better solution would be to edit your WordPress theme by creating a child theme and add this line where you want to display the time.

```
1 <?php get_the_time(); ?>
```

This code will only fetch time for the post or page that is currently being viewed.

However, most beginners don't want to create child themes and go through the code. For those users, there is an easier way to add the time next to date.

Technically, when WordPress outputs date it actually outputs the complete date and time. However, since most people only use format characters for date, it has no choice but to display only the date. If you entered time related characters in the date format it will display time as well. For example,

d/m/Y g:i a will output 06/11/2014 12:15 pm

Similarly, you can also use date related format characters in the time field if you need.

Depending on your needs, there are many other things you can do with date and time in WordPress. For example, you can show last updated time for posts instead of publish date.

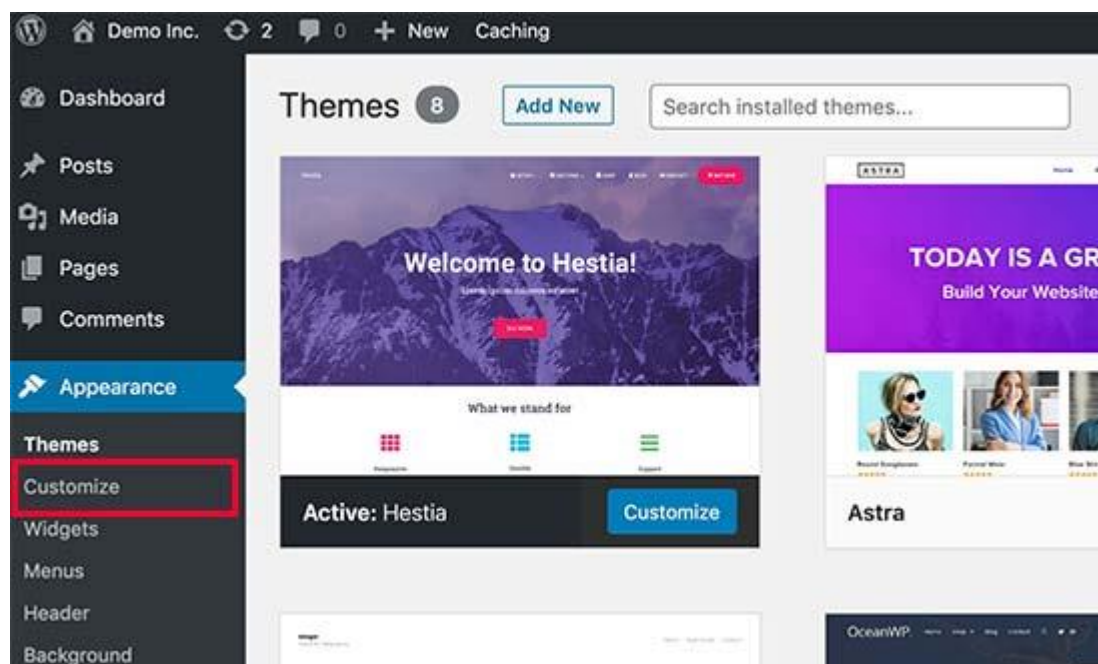
How to Easily Add Custom CSS to Your WordPress Site

Sometimes you may find yourself needing to add Custom CSS in your WordPress site.

Method 1: Adding Custom CSS Using Theme Customizer

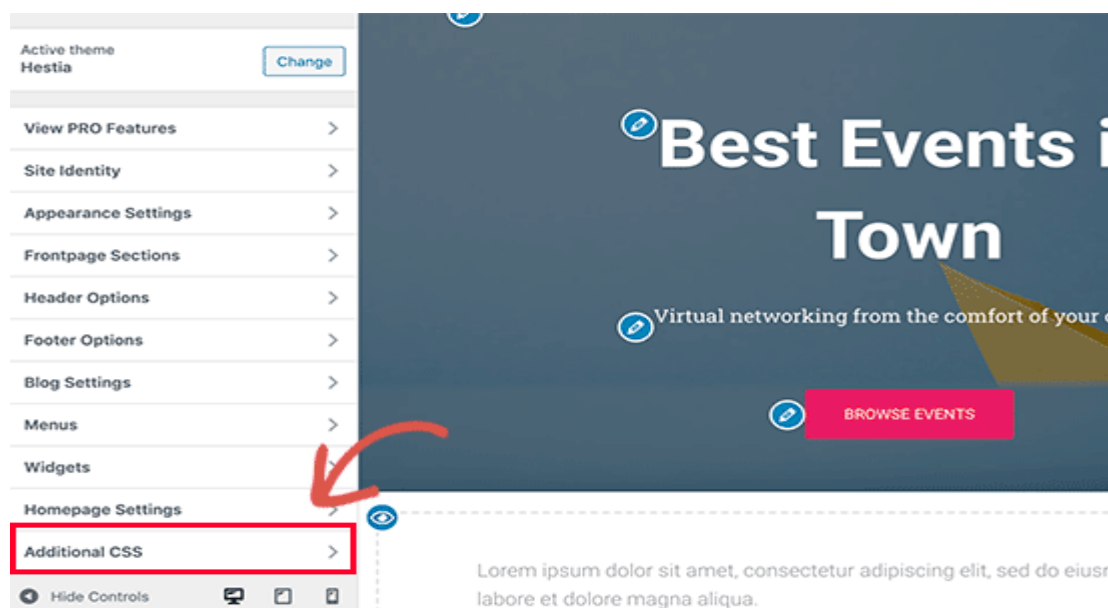
Since WordPress 4.7, users can now add custom CSS directly from the WordPress admin area. This is super-easy, and you would be able to see your changes with a live preview instantly.

First, you need to head over to the **Themes » Customize** page.

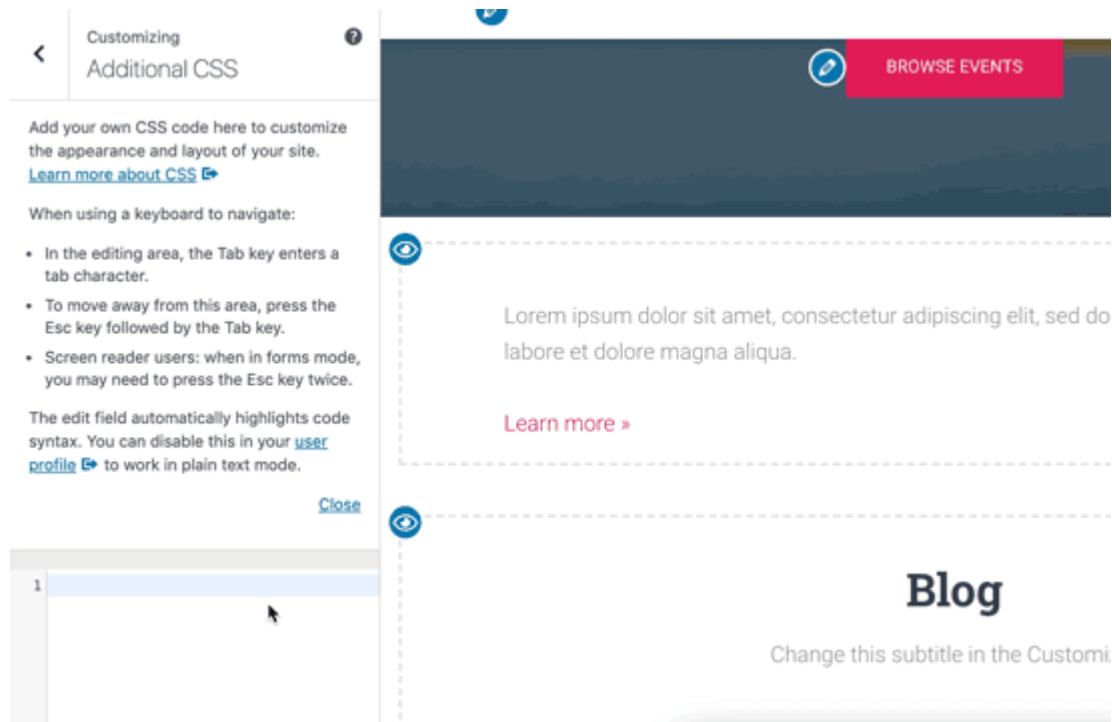


This will launch the WordPress theme customizer interface.

You will see your site's live preview with a bunch of options on the left pane. Click on the Additional CSS tab from the left pane.

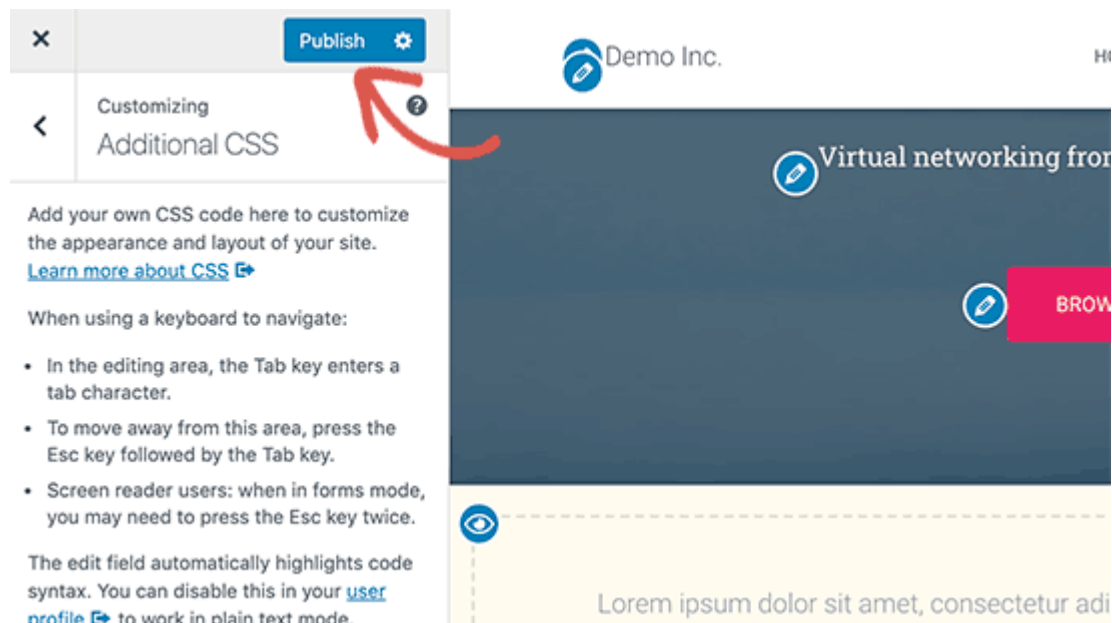


The tab will slide to show you a simple box where you can add your custom CSS. As soon as you add a valid CSS rule, you will be able to see it applied on your website's live preview pane.



You can continue adding custom CSS code, until you are satisfied with how it looks on your site.

Don't forget to click on the 'Save & Publish' button on the top when you are finished.



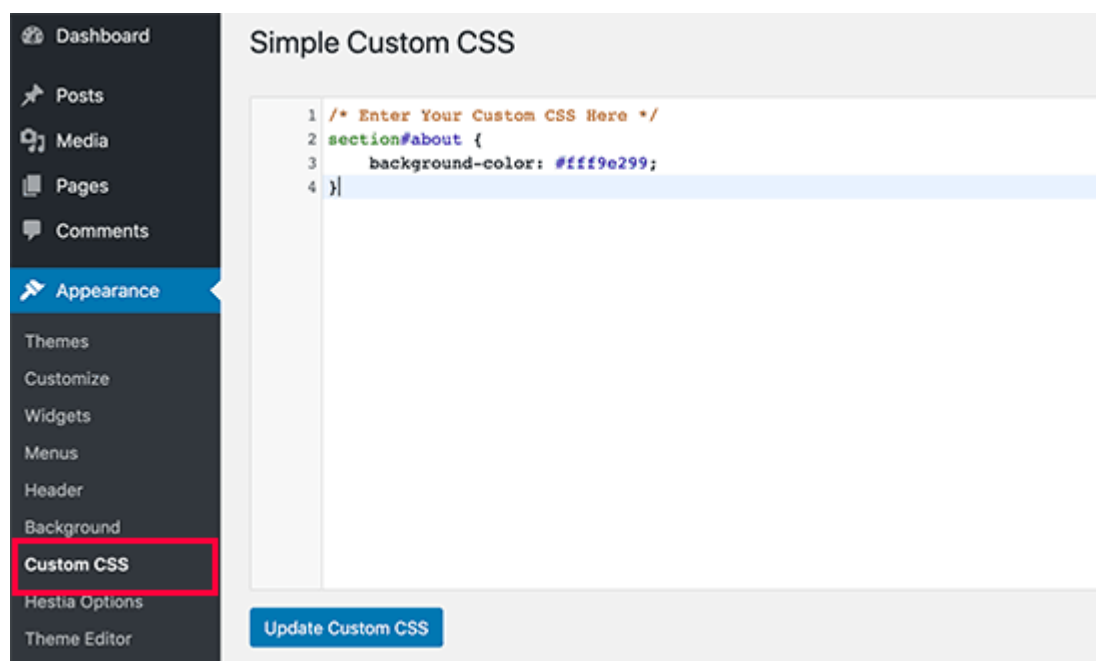
Any custom CSS that you add using theme customizer is only available with that particular theme. If you would like to use it with other themes, then you will need to copy and paste it to your new theme using the same method.

Method 2: Adding Custom CSS Using a Plugin

The first method only allows you to save custom CSS for the currently active theme. If you change the theme, then you may need to copy and paste your custom CSS to the new theme.

If you want your custom CSS to be applied regardless of which WordPress theme you are using, then this method is for you.

First thing you need to do is install and activate the Simple Custom CSS plugin. Upon activation simply go to **Appearance » Custom CSS** and write down or paste your custom CSS.



Don't forget to press the 'Update Custom CSS' button to save your changes.

You can now view your WordPress website to see the custom CSS in action.

Using a Custom CSS plugin vs Adding CSS in Theme

Both methods described above are recommended for beginners. Advanced users can also add custom CSS directly to their themes.

However, adding custom CSS snippets into your parent theme is not recommended. Your CSS changes will be lost if you accidentally update the theme without saving your custom changes.

The best approach is to use a child theme instead. However, many beginners don't want to create a child theme. Apart from adding custom CSS, often beginners really don't know how they will be using that child theme.

Using a custom CSS plugin allows you to store your custom CSS independently from your theme. This way you can easily switch themes and your custom CSS will still be there.

Another great way to add custom CSS to your WordPress site is by using the CSS Hero plugin. This wonderful plugin allows you to edit almost every CSS style on your WordPress site without writing a single line of code.

How to Create a Separate Page for Blog Posts in WordPress

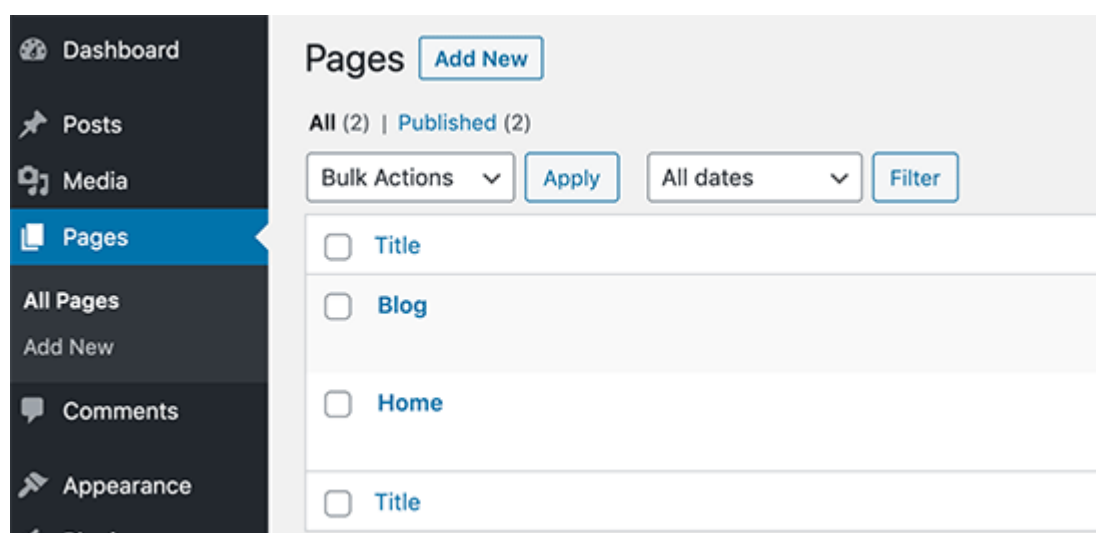
Do you want to create a separate page for blog posts in WordPress? By default, WordPress displays your blog posts on the homepage.

However, if you are making a business website or simply want to customize your homepage, then you may want a separate page for your blog section.

Setup WordPress to Create a Separate Blog Page

WordPress allows you to easily configure your website to have a custom home page. This feature also allows you to choose any other page on your website to be used as your blog page.

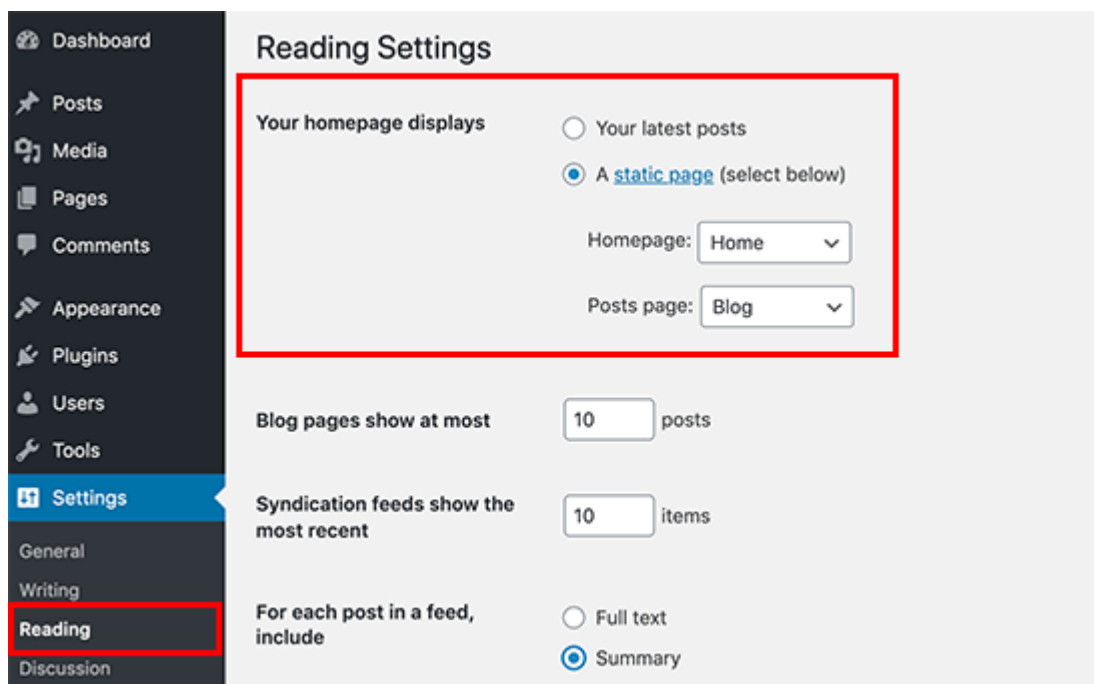
First, you need to create two pages in WordPress that you would like to use as your homepage and blog page.



You don't need to add any content to those pages, and you can even name them anything else if you want. For instance, some business websites use News for their blog page.

Once you have created those pages, you can head over to the **Settings » Reading** page in the WordPress admin area.

Under the 'Your homepage displays' section, you need to click on the 'A static page' option. After that, you can select the home and blog pages you created earlier.



You can now go to **Pages » All Pages** section in admin area. You'll notice your blog and homepage with new labels. Go ahead, and click on the View link under those pages to preview them.

The appearance of these pages depends on your WordPress theme, and you may need to configure your theme settings.

Since each WordPress theme has different options, you'll have to check out your theme's documentation or check the options under **Appearance » Customize** page.

How to Create a File Upload Form in WordPress

Do you need to have your website visitors upload files on your contact form?

Maybe you're hiring employees and want to collect resumes, or perhaps you're running another application form or photography contest where you need the applicant to attach files.

Why Create a File Upload Form in WordPress?

If you want to collect any kind of file from your users, then a file upload form is a great idea. You could let users upload images, PDF files, Word documents, or other types of files.

Using a file upload form makes life easy for you, and your users. Instead of the email back-and-forth, you can create a form that includes all the fields you need, so you can collect all information at once.

Also, your form should automatically save the form data in your WordPress database. That way, you can easily find the submissions, even if you miss or delete an email.

How to Create a File Upload Form in WordPress

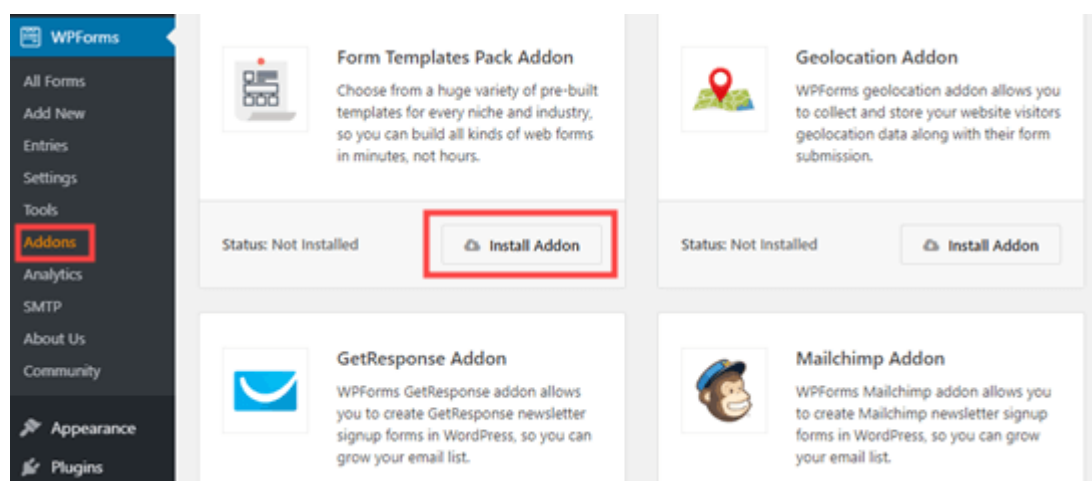
For this tutorial, we'll be using WPForms, which is the best contact form plugin for WordPress.

If you're on a tight budget, you can get the file upload feature on WPForms' Basic plan. However, the fully-featured Pro plan gives you access to bonus form templates, survey reports, and tons of other powerful features.

First, you'll need to install and activate the WPForms plugin. If you're not sure how, just check out our step by step guide on how to install a WordPress plugin.

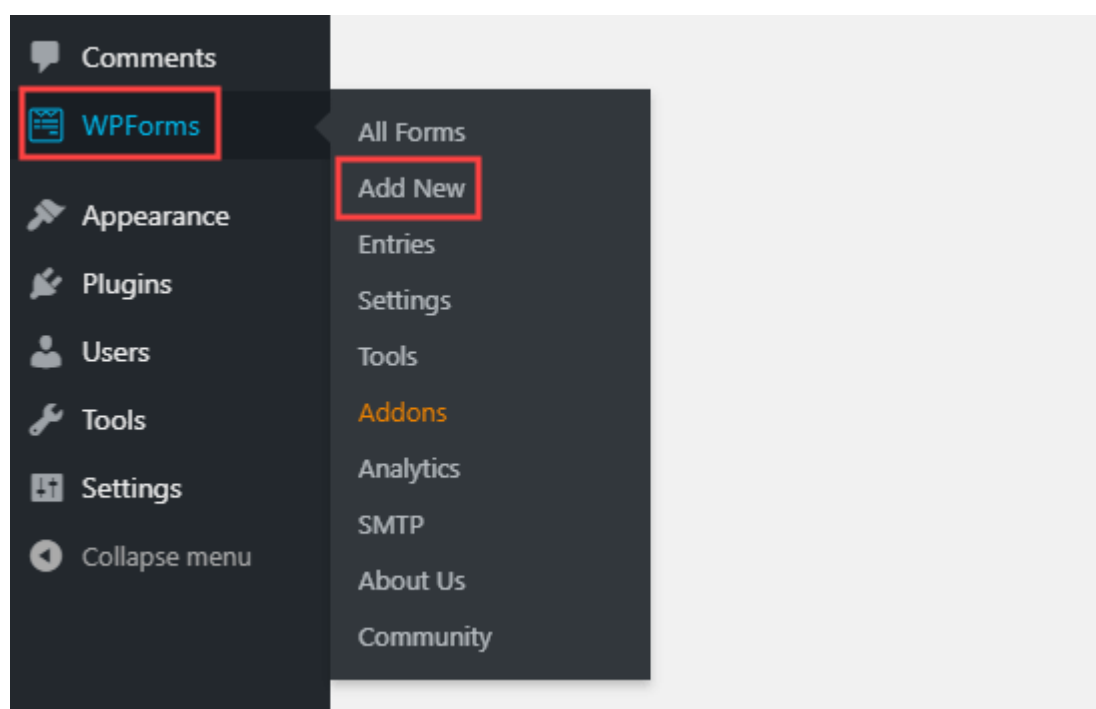
Once you've activated the plugin, you'll see a new WPForms tab in your dashboard.

Go to the **WPForms » Addons** page and scroll down to the Form Templates Pack Addon. Click on the 'Install Addon' button to install and activate it.

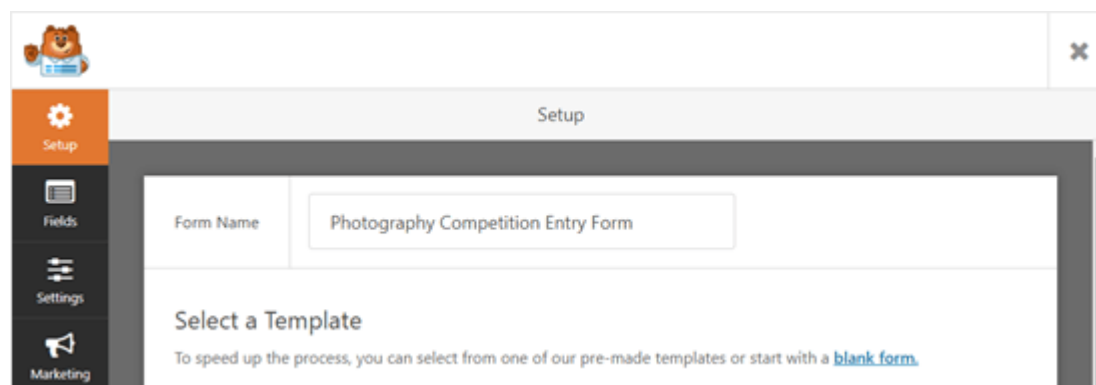


You are now ready to create your first file upload form.

Head over to the **WPForms » Add New** page.

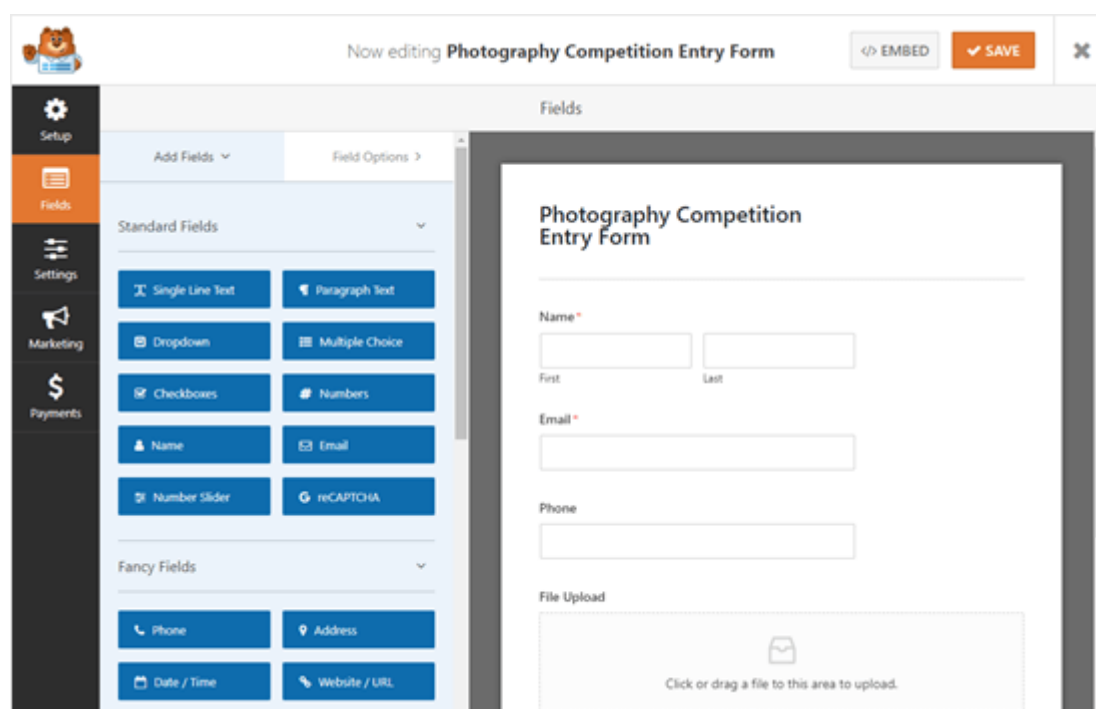


You'll be taken straight to the WPForms builder. You can give your form a name at the top of the screen:



You'll also need to choose a template. We recommend using the File Upload Template, which you can find in the Additional Templates section. Click on the template to use it.

The File Upload Form will automatically be created for you.



You can use the file upload form without making any changes at all, if you want.

By default, it'll allow users to upload the following file types:

- Images files: .png, .gif, .jpg
- Documents: .doc, .xls, .ppt, .pdf
- Audio: .wav, .mp3, .mp4
- Video: .mpg, .mov, .wmv

If you'd like to restrict the file types further, click on the File Upload field to edit it. You'll see the Field Options on the left hand side of your screen:

The screenshot shows the Gravity Forms configuration interface. On the left, a sidebar contains icons for Setup, Fields, Settings, Marketing, and Payments. The 'Fields' section is highlighted. The main area is titled 'Fields' and has two tabs: 'Add Fields >' and 'Field Options >'. Under 'Field Options', the 'File Upload (ID #4)' field is selected. Below the field name, there are input boxes for 'Label' (containing 'File Upload'), 'Description', 'Allowed File Extensions', 'Max File Size', and 'Max File Number' (set to '1'). There is also a 'Required' checkbox. On the right, a preview of the form shows the 'File Upload' field as a dashed box with a file icon and the text 'Click or drag a file to this area to upload.' Below the preview is an 'Additional Comments' text area and a 'Submit' button.

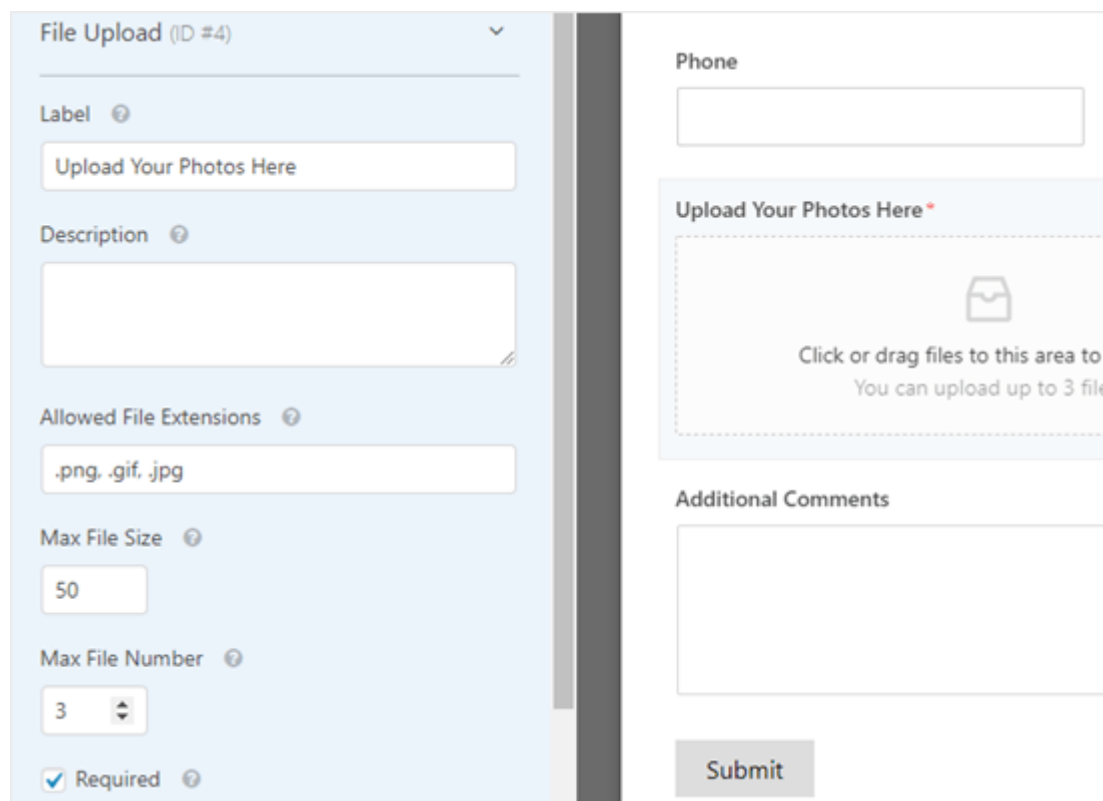
In the Allowed File Extensions box, you can type in whatever file formats you want to accept. We're going to allow the image formats .png, .gif, and .jpg. Make sure you separate them with a comma.

This is a close-up of the 'Allowed File Extensions' field in the Gravity Forms configuration. The field is highlighted with a red rectangle. It contains the text '.png, .gif, .jpg'. The field is labeled 'Allowed File Extensions' with a help icon. Below it is the 'Max File Size' field, which is currently empty.

Note: WordPress limits the types of files that can be uploaded to the site for WordPress security reasons. You can add additional file types to WordPress if you want to allow users to upload file types that aren't normally allowed.

You can also set a maximum file size and a maximum number of files. For our competition, we're going to allow people to submit up to 3 photos of up to 50MB each.

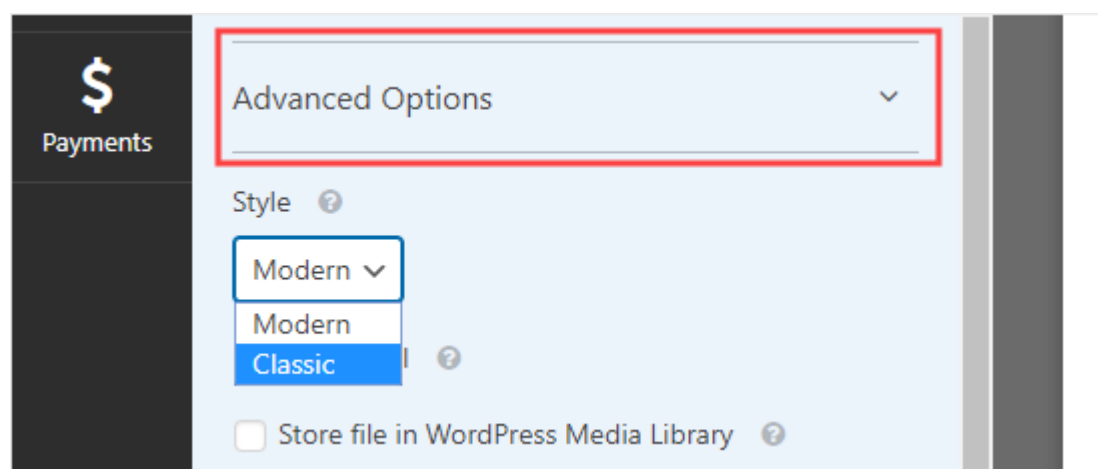
If you want to prevent people from submitting the form without attaching their file(s), you can check the Required box. This means the form can't be submitted unless at least 1 file has been uploaded.



The image shows the configuration for a 'File Upload' field (ID #4) and a preview of the form. The configuration panel on the left includes a 'Label' field with the text 'Upload Your Photos Here', a 'Description' field, 'Allowed File Extensions' set to '.png, .gif, .jpg', 'Max File Size' set to 50, 'Max File Number' set to 3, and a checked 'Required' checkbox. The form preview on the right shows a 'Phone' input field, the 'Upload Your Photos Here' field with a dashed border and a file icon, and an 'Additional Comments' text area. A 'Submit' button is at the bottom right.

By default, the File Upload Field uses a modern format that lets users drag and drop their files. However, if you have limited space for your form, or you prefer a classic upload field, you can change this.

Simply click on 'Advanced Options' to open up more options for the field. You can then switch the style from Modern to Classic using the dropdown.



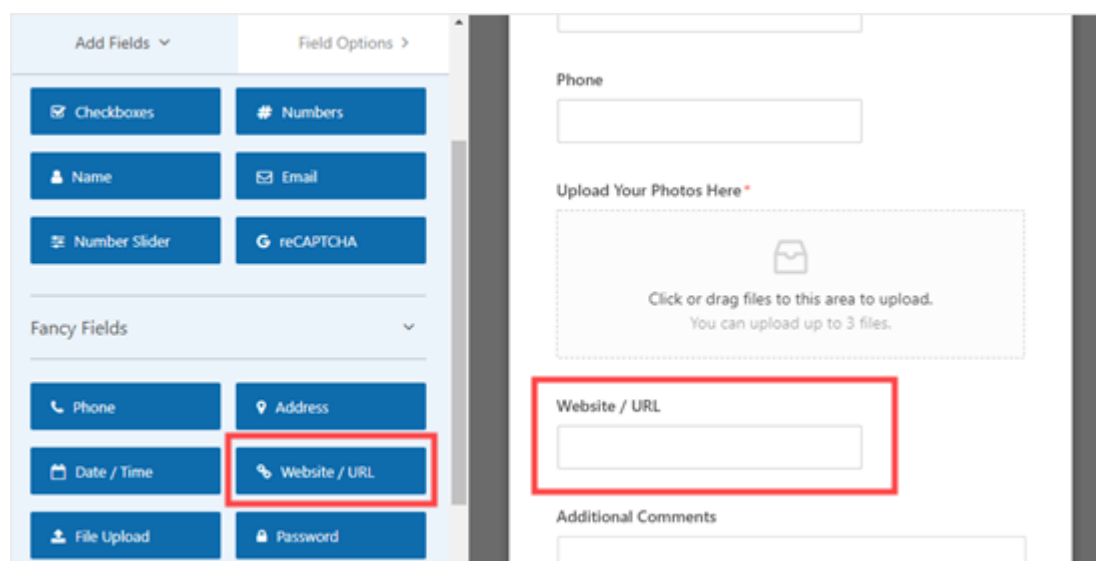
The image shows the 'Advanced Options' configuration panel for the File Upload field. The 'Advanced Options' header is highlighted with a red box. Below it, the 'Style' dropdown menu is open, showing 'Modern' and 'Classic' options, with 'Classic' selected. There is also a checkbox for 'Store file in WordPress Media Library'.

You can also choose to store the submitted files in the WordPress Media Library, if you want. This makes it easy for you to add them to a post or page. For instance, with our photography competition, we might include the best photos when we announce the winner.

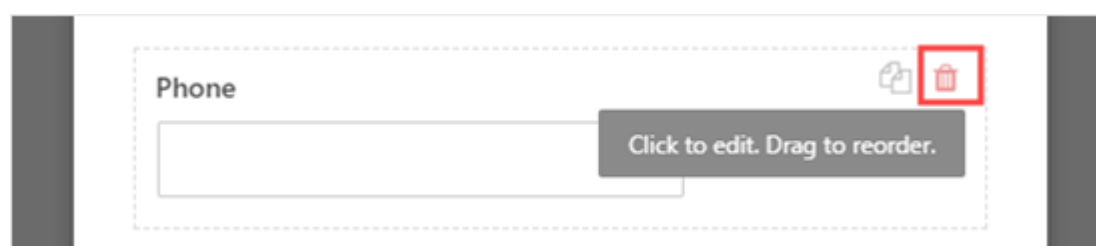
If you want to edit any of the other fields on your form, you can click to edit them on the left hand side in the same way.

You can also add new fields using the Add Fields tab. Simply drag and drop any field that you want to add to your form.

Here, we've added a Website/URL field to the form, so that photographers can give us the link to their website.



If you want to remove a field, hover your mouse cursor over it and click the red trash can icon. We're going to remove the Phone field as we don't need the photographers' phone numbers.



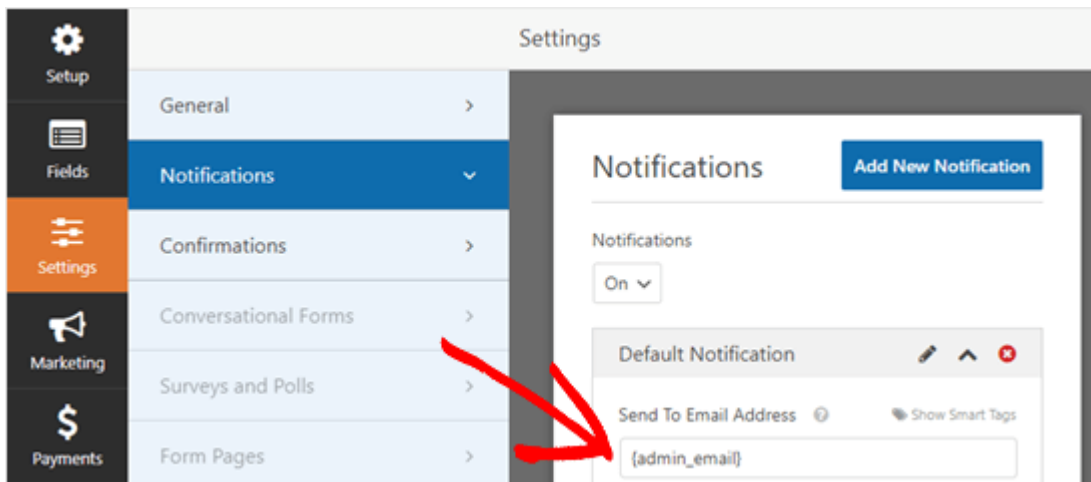
Once you're happy with your form, click the Save button at the top of the screen.

Setting Up Your File Upload Form's Notifications

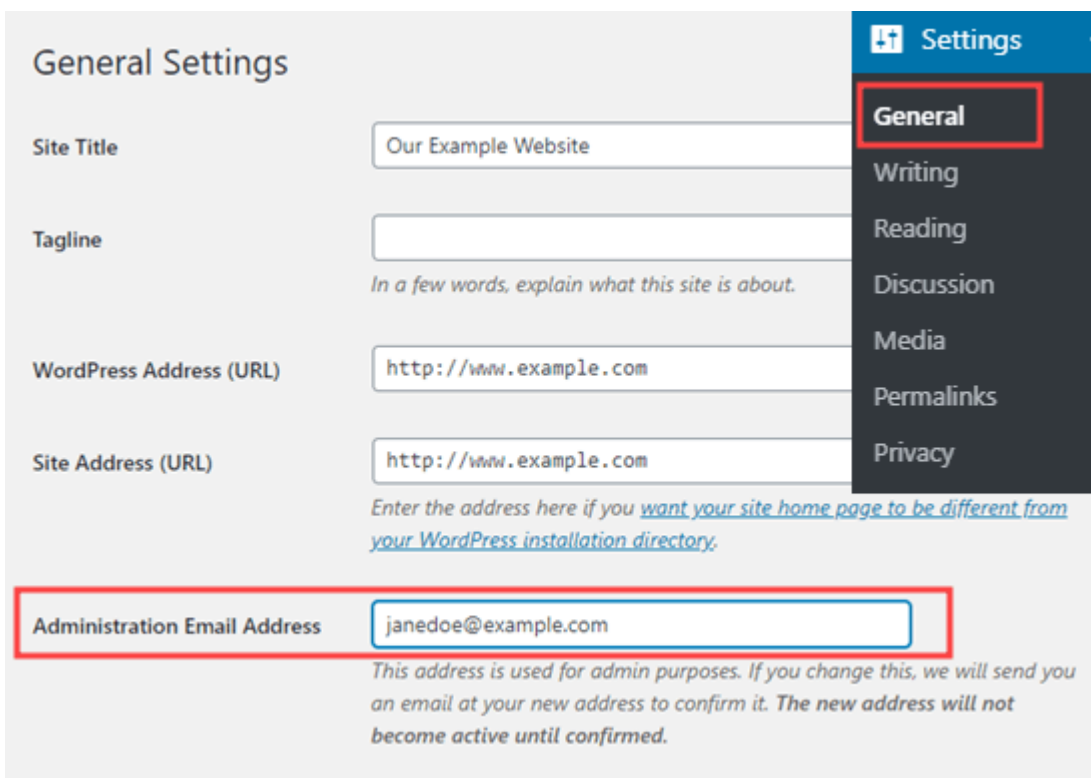
Under the Settings tab, you can click Notifications to set up your form's email notifications. You may want to change the default settings.

By default, the form sends an email notification to {admin_email}.

If you created your WordPress site by yourself, then this will be your email address. If not, then you can delete {admin_email} and type your own email address here instead.



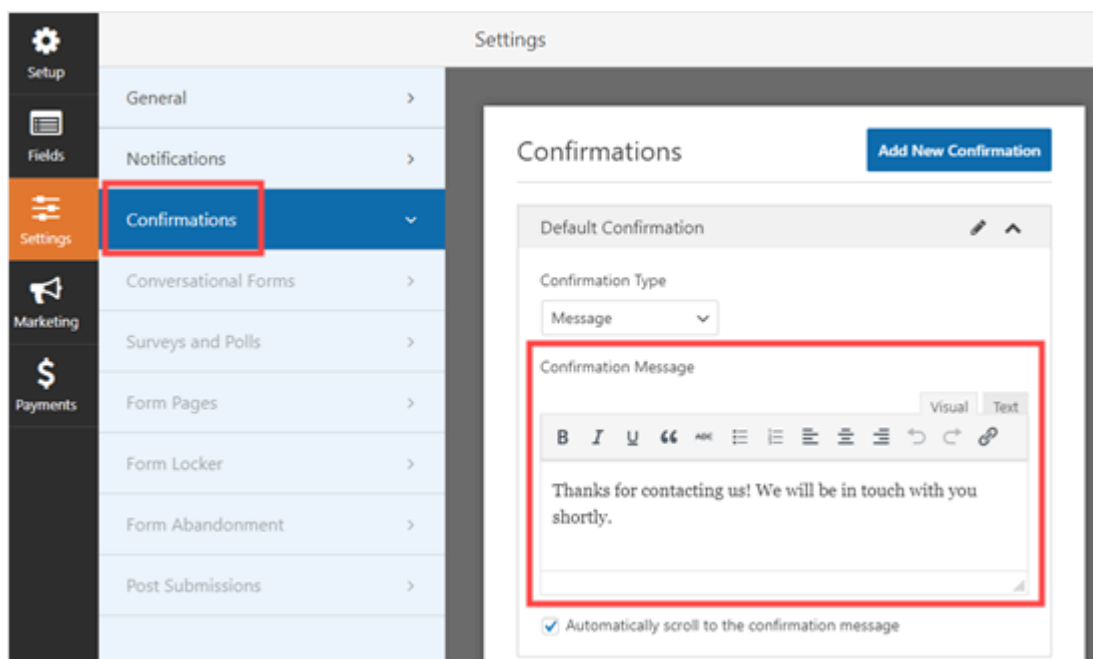
Tip: Not sure if {admin_email} is your email address or not? Go to **Settings » General** in your WordPress dashboard. Look for the Administration Email Address. You can change it here. Just click Save Changes at the bottom after doing so.



If you want the submitted files to be sent to more than one person, you can also set up multiple notifications here.

You may also want to change the confirmation message that users sees after the form is submitted. You can do this under **Settings » Confirmations**.

Simply type the message you want into the box. You can add formatting like bold and italic if you want to. This is how the default message looks:

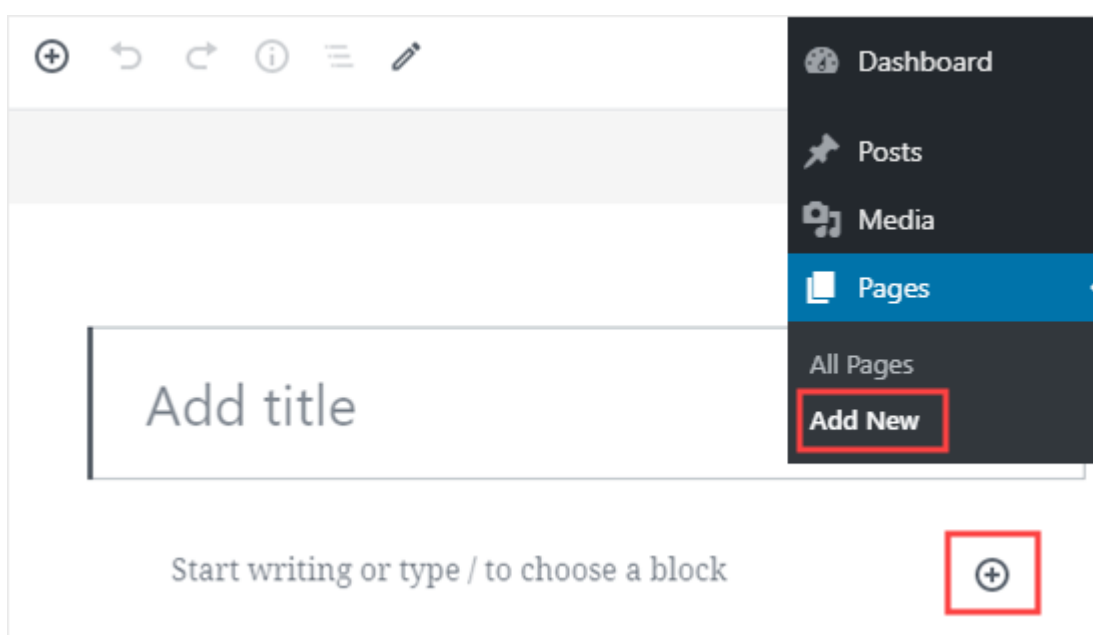


Don't forget to save your form once you've made changes. You can exit the form builder by clicking the X in the top right.

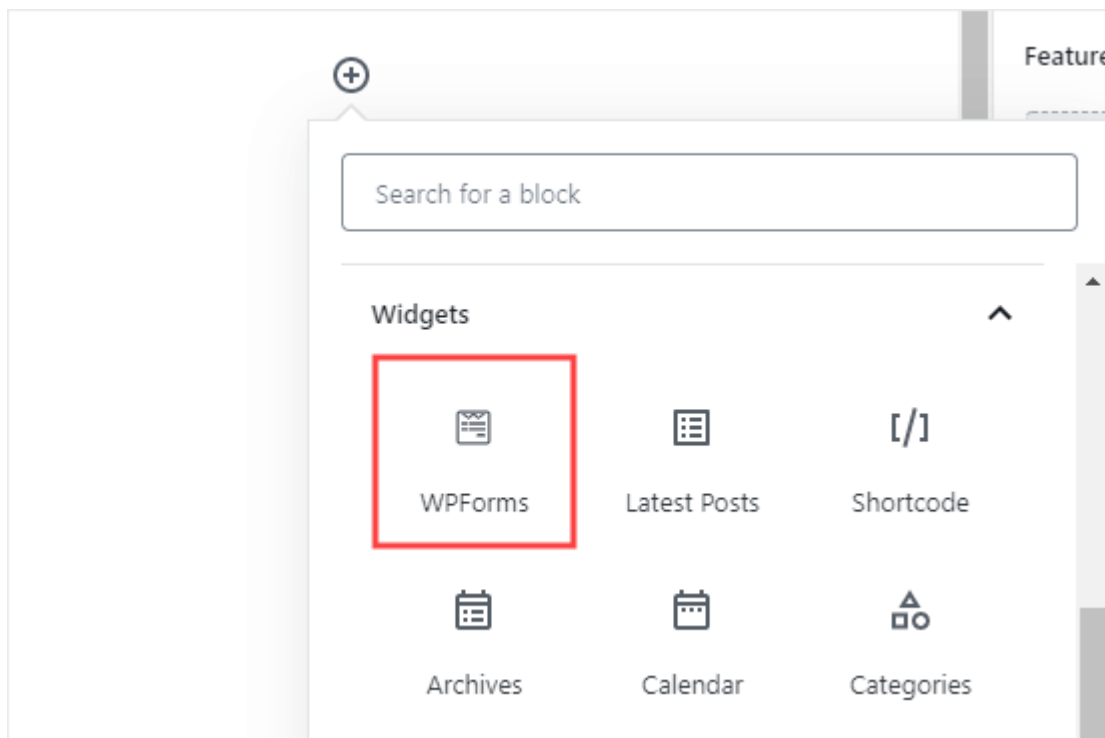
Adding Your File Upload Form to Your Website

Once you've created your form, you'll need to add it to your website. You can add it on an existing post or page, or you can create a new page.

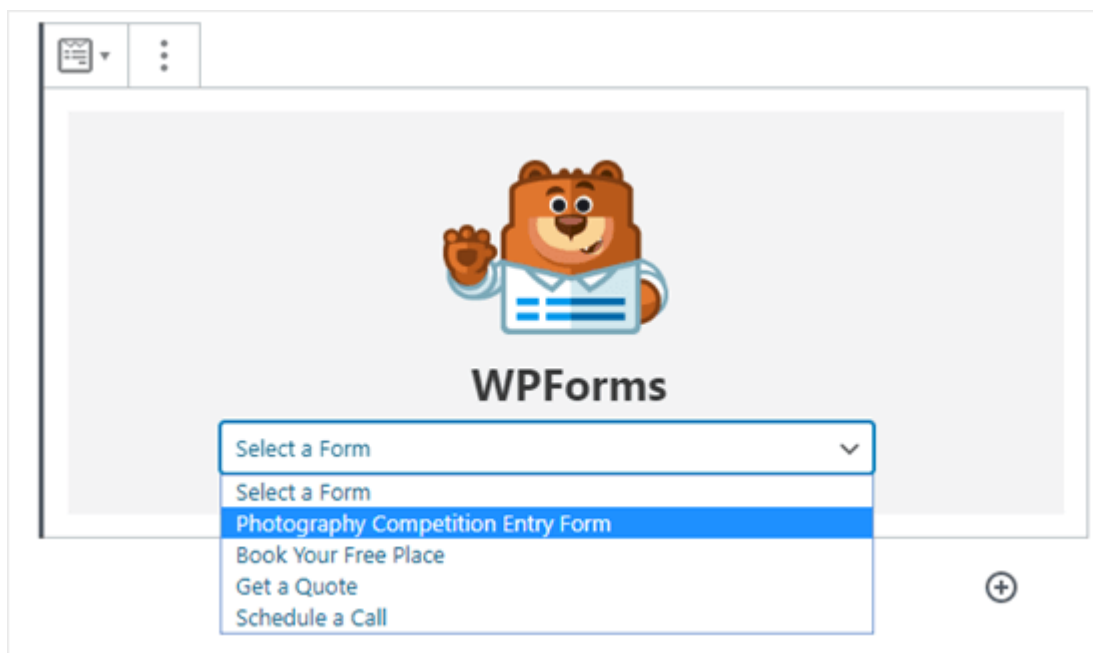
We're going to add our form to a new page. To create a new page, go to **Pages » Add New**. Then, add a new block to your page by clicking the (+) icon.



Next, you'll need to select the WPForms block from the Widgets section. You can use the search bar to find it if you prefer.



After you've added the block to your page, select the form that you created from the WPForms dropdown.



Now, preview or publish your page and take a look at your form live on your site. Here's how our form looks.

Photography Competition Entries

Name *

First Last

Email *

Upload Your Photos Here *



Click or drag files to this area to upload.
You can upload up to 3 files.

The URL of Your Website

Tell Us About Your Photos (Optional)

You can make changes to your form by going to **WPForms » All Forms**. Just click on the name of the form, or run your mouse over it and click the ‘Edit’ link, to edit it.

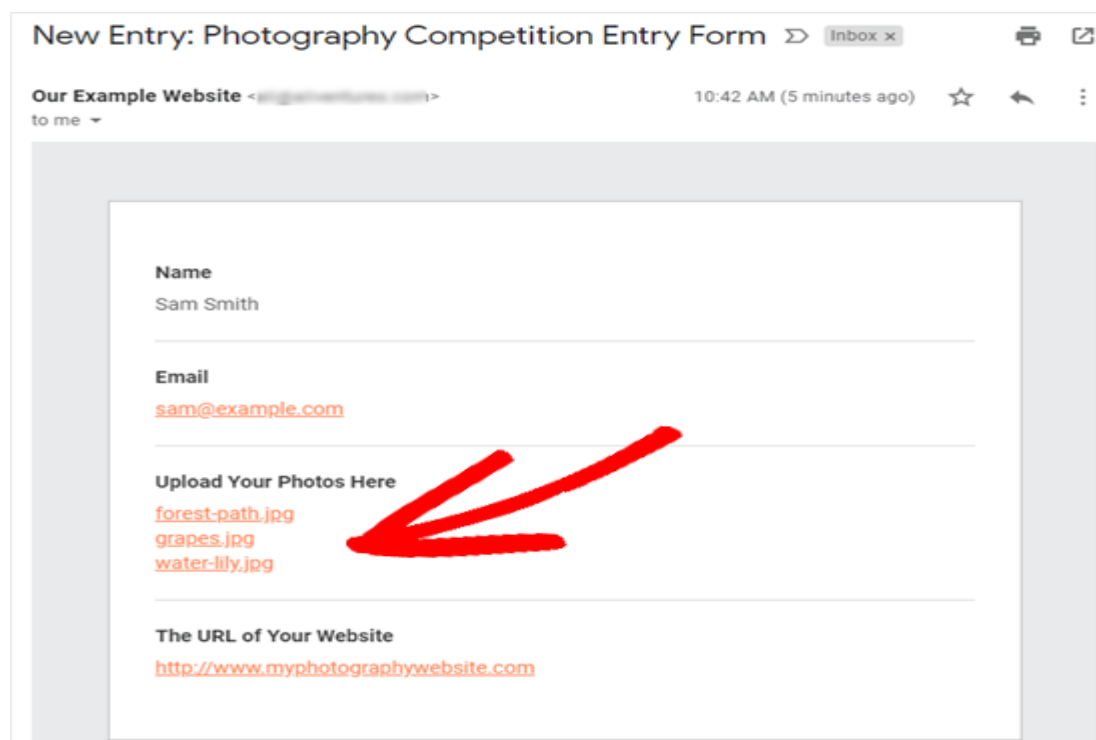
Forms Overview Add New	
Bulk Actions ▾ Apply	
☐	Name Shortcode
☐	Photography Competition Entry Form [wpforms id="754"] Edit Entries Preview Duplicate Delete
☐	Book Your Free Place [wpforms id="743"]

When you edit your form, it’s automatically updated on your website. You don’t need to add it to your page again.

Viewing or Downloading the Uploaded Files

When someone sends a file through your file upload form, you can get it through your email inbox or through your WordPress dashboard.

With each form entry, you'll get an email like this:



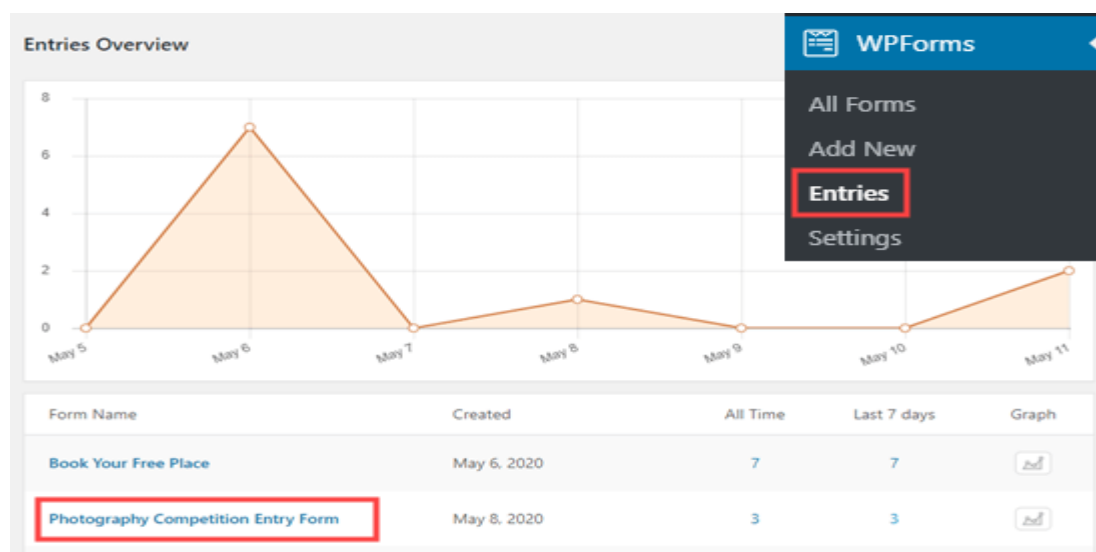
The screenshot shows an email interface with the subject 'New Entry: Photography Competition Entry Form'. The email body contains the following information:

- Name:** Sam Smith
- Email:** sam@example.com
- Upload Your Photos Here:** [forest-path.jpg](#), [grapes.jpg](#), [water-lily.jpg](#)
- The URL of Your Website:** <http://www.myphotographywebsite.com>

A large red arrow points to the photo links in the 'Upload Your Photos Here' section.

Just click the links to view or download the files themselves.

You can also easily find the files through WordPress itself. Go to **WPForms » Entries** then click on the name of your form.



We hope this tutorial helped you learn how to create a file upload form in WordPress.

What is: Template

In WordPress theme development, a template defines part of a web page generated by a WordPress theme.

Example: header.php is a default template used in most WordPress themes. It defines the header area of web pages generated by WordPress. The header file will typically be loaded on every page of your WordPress site, allowing changes to be made to a single file, that will apply across the entire website.

Most WordPress themes have some default templates with code to generate HTML output for particular sections of a website.

- Main – index.php. To display the main page of a website.
- Header – header.php. Displays the header section.
- Sidebar – sidebar.php. Generates HTML output for the sidebar section.
- Footer – footer.php. Displays the footer section.
- Theme Functions – functions.php. Contains code and functions to be used in a theme.
- Single Post – single.php. Displays the single post page.
- Comments – comments.php. Displays comments and comment form.

Templates can be included into each other. For example, single.php can have header, footer, sidebar, and content template included into it. WordPress allows users to add as many as they would like. Some themes will have them for custom loops and sidebars as well.

WordPress also has a templates hierarchy. If a theme doesn't have single.php for some reason, then WordPress will automatically fall back to a more general templates like index.php.

Apart from these templates, each WordPress theme must have a style definition file named style.css.

What is: Template Tag

A template tag is a PHP function used to generate and display information dynamically. WordPress Themes contain different templates and theme developers use template tags to fetch and display dynamic data. WordPress has many built-in template tags that can be used in WordPress themes. WordPress plugins and themes can also define their own template tags and use them in different templates.

Example:

```
1 <?php the_author(); ?>
```

The author template tag displays the name of the post author in WordPress.

Usage example:

```
1 <p>This post is written by <?php the_author(); ?></p>
```

Template tags can also return a data set and users can choose what to display using parameters.

Example:

```
1 <a href="<?php bloginfo('url'); ?>" title="<?php bloginfo('name'); ?>"><?php  
  bloginfo('name'); ?></a>
```

Template tags are basically PHP functions, so any PHP function defined by a WordPress plugin or theme can be used as a template tag. To use a theme function as a template tag, the function should be defined in the theme's `functions.php` file.

Template tags are PHP functions, so they can also be used inside other PHP functions and template tags. In the example below, we have defined a function that displays some text.

Example:

```
1 function donation_request() {  
2     $this_article = wp_title("",true);  
3     echo '<p>Hi, if you enjoyed reading '.$this_article.' please consider <a  
4     href="http://www.example.com/donate/">donating</a>.';  
    }  
}
```

To use this function in a template, add this line of code:

```
1 <?php donation_request(); ?>
```

Multiple template tags can also be combined to accomplish a goal.

WordPress Template Hierarchy (Cheat Sheet)

Why Learn About WordPress Template Hierarchy?

WordPress has a standard template hierarchy. This means that templates with certain filenames affect specific areas on your website. It also tells you what template name to use for a specific item.

Most modern WordPress themes come with templates to display category, date, archives, single posts, custom pages, and more. As a user, you can create more templates by creating a child theme.

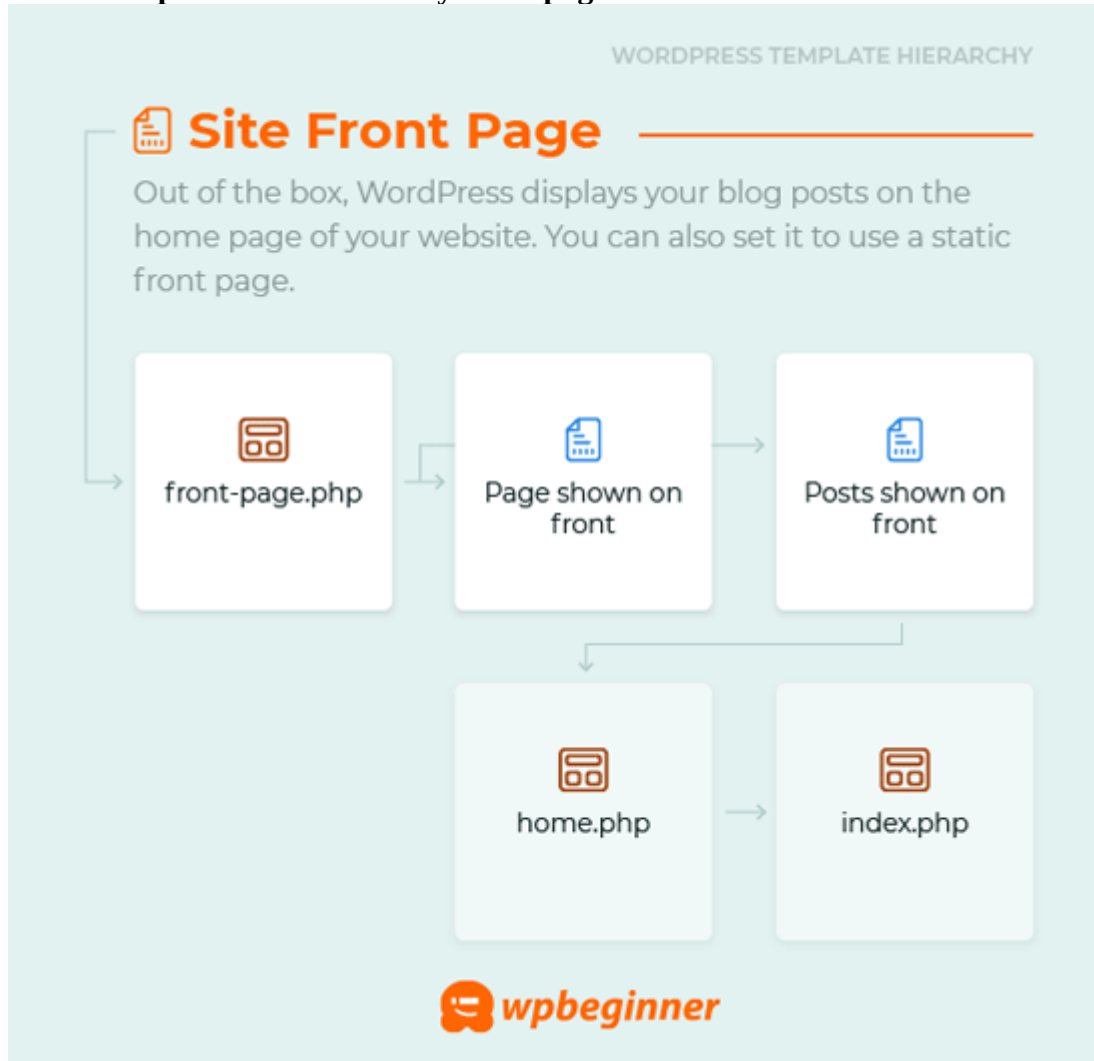
Having said that, let's take a look at how this WordPress template hierarchy works behind the scenes.

Visualizing WordPress Template Hierarchy (Cheat Sheet)

WordPress uses an easy to understand and meaningful pattern for template names. The visual cheat sheet below explains which template files are used to display different pages on a WordPress site.

Here is a breakdown of which WordPress template files to edit for different pages in a typical WordPress site.

Which template files are used by home page?



Out of the box, WordPress displays your blog posts on the home page of your website. You can also set it to use a custom home page (also known as front page) by visiting **Settings » Reading** page in WordPress admin area.

1. front-page.php – If you are using a static front page, then WordPress will first look for this template file and use it to display front page.

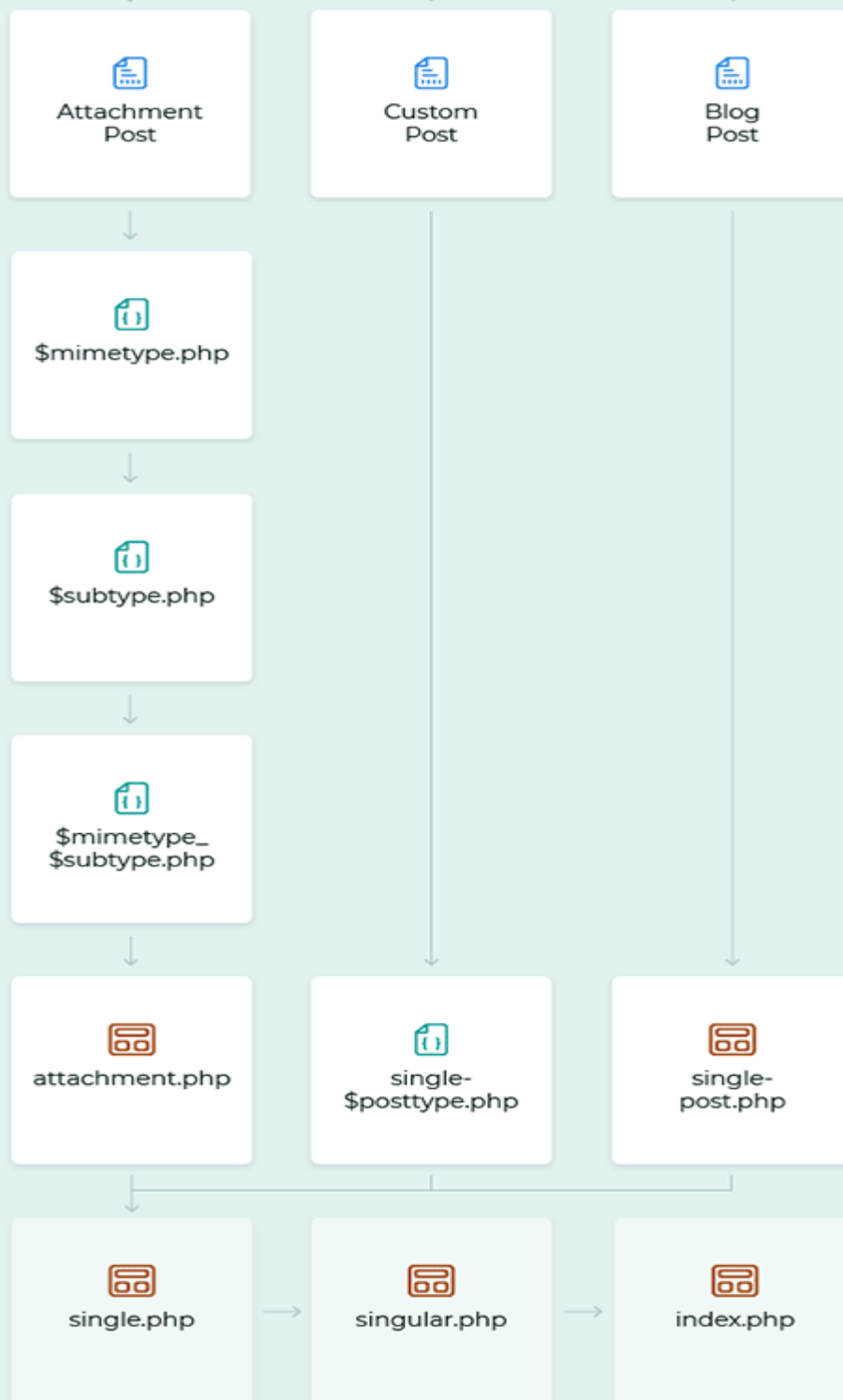
2. home.php – If you are using static front page, and you don't have front-page.php template in your theme, then WordPress will look for home.php template. It is also used to display default blog posts on homepage.

3. index.php – If front-page.php or home.php do not exist, then WordPress falls back to index.php template to display homepage. This template is the default fallback template in WordPress to display any page.

Which template files are used by single post?

Single Post Page

WordPress looks for these files to display a single post.



WordPress looks for these files to display a single post.

1. single-{post-type}-{slug}.php – Use this template to modify the display of a specific individual post in any post type. For example, if post type is ‘review’ and the post slug is acme-phone, then WordPress would look for single-review-acme-phone.php.

2. single-{post-type}.php – WordPress will then check if there is a template to display this specific post type. For example, if the post type is review, then WordPress would look for single-review.php.

3. single.php – WordPress will then fall back to single.php.

4. singular.php – This template adds another fallback to display a single item from any post type.

5. index.php – Finally, as mentioned above, WordPress ultimately falls back to index.php.

Which template files are used by single page?



WordPress pages are one of the default post types. They allow you to create static pages in your website instead of posts. See our guide on the difference between posts vs pages.

1. Custom Page Template – The page template assigned to the page. See how to create a custom page template in WordPress.

2. page-{slug}.php – If the page slug is contact-us, WordPress will look to use page-contact-us.php.

3. page-{id}.php – If the page ID is 17, then WordPress will look for a template file named page-17.php.

4. page.php – The template to display all static pages.

5. singular.php – This template is a default fallback to all single post type items.

6. index.php – The default fallback template.

Which template files are used by category archives?



WordPress uses these files to display category related pages in WordPress.

1. category-{slug}.php – This template is used to display category archive page for a specific category. For example, if category slug is reviews, then WordPress will look for category-reviews.php template.

2. category-{id}.php – WordPress then looks for a template with category ID. For example, if category ID is 17, then WordPress will look for category-17.php.

3. category.php – This is the default template to display all category archive pages in WordPress.

4. archive.php – This is the default template used by WordPress to display any archive pages.

5. index.php – The default fallback template.

Which template files are used by tag archives?



WordPress uses these files to display tag archive pages.

1. **tag-{slug}.php** – If the tag's slug is fruits, WordPress will look for tag-fruits.php.
2. **tag-{id}.php** – If the tag's ID is 17, WordPress will look for tag-17.php template.
3. **tag.php** – The default template for tag archives.
4. **archive.php** – The default template for any archive page.
5. **index.php** – The default fallback template.

Which template files are used by custom taxonomy archives?



Categories and tags are two default WordPress taxonomies. Users can also create their own custom taxonomies as well. Here is how WordPress looks for templates to display custom taxonomy pages.

1. taxonomy-{taxonomy}-{term}.php – If you have a custom taxonomy called genre, and there is a term ‘thriller’, then WordPress will look for taxonomy-genre-thriller.php.

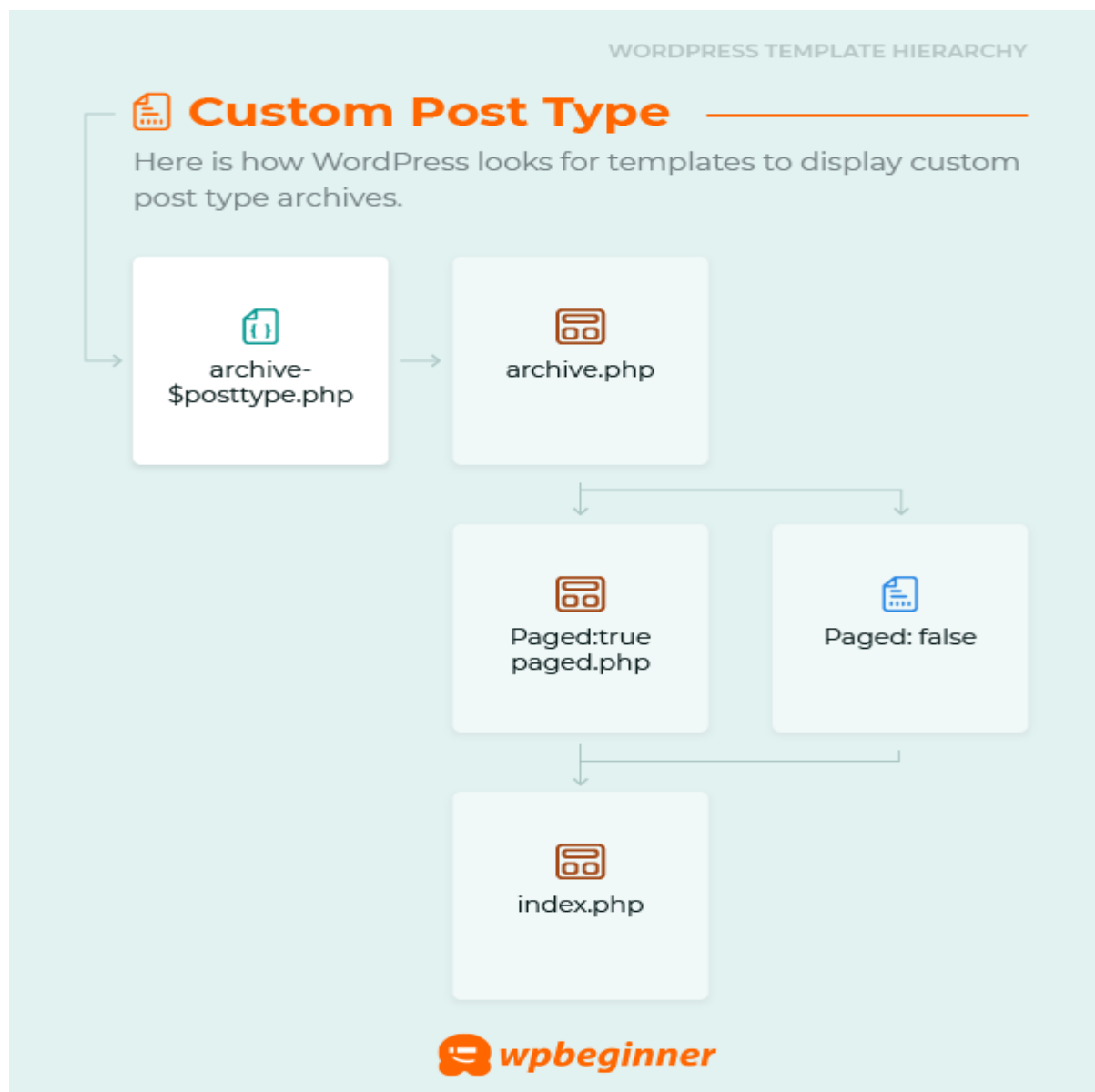
2. taxonomy-{taxonomy}.php – If the taxonomy were genre, WordPress would look for taxonomy-genre.php.

3. taxonomy.php – The default template to display any custom taxonomy archives.

4. archive.php – The default fallback for all archive pages in WordPress.

5. index.php – The default fallback template in WordPress.

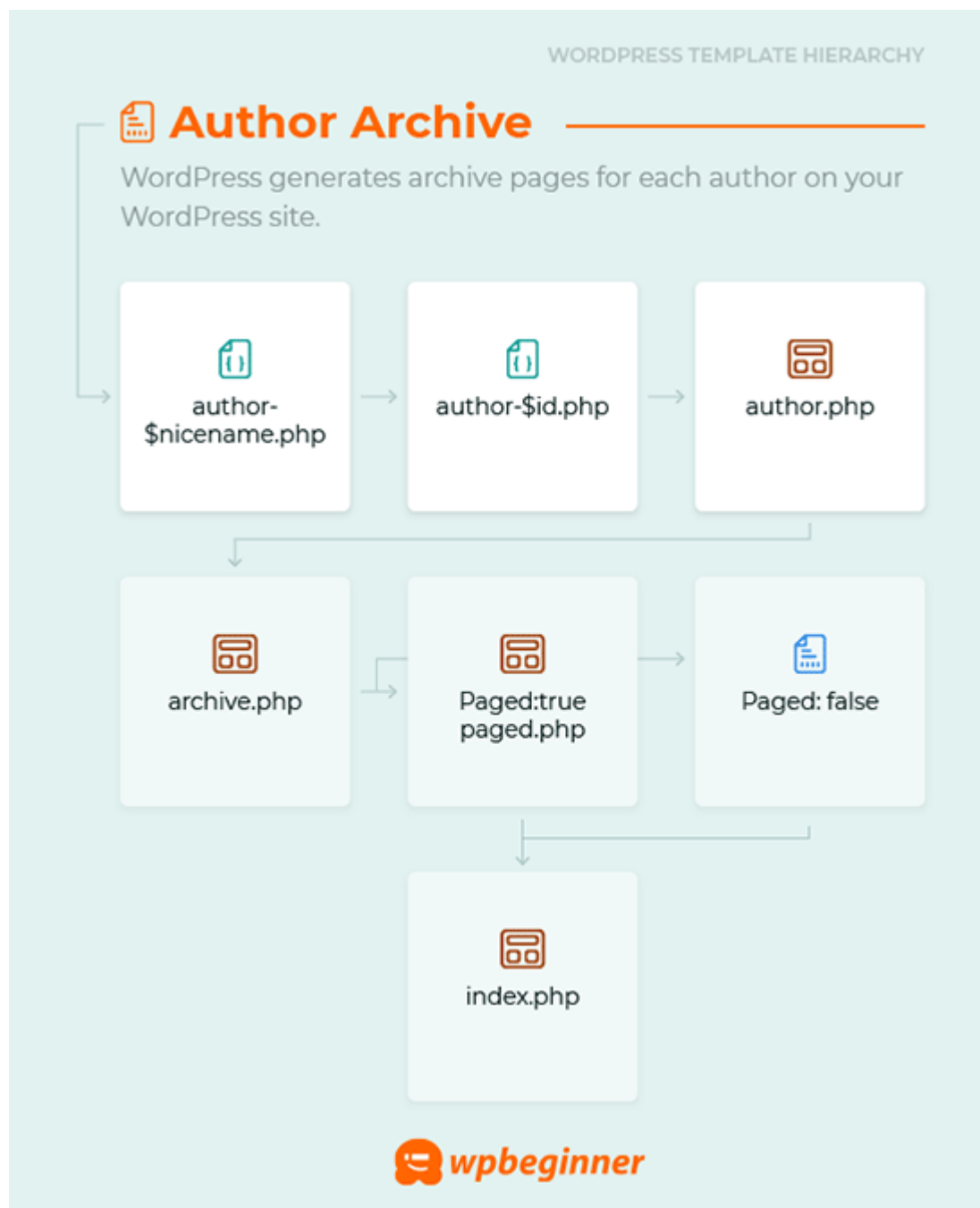
Which template files are used by custom post types?



Here is how WordPress looks for templates to display custom post type archives.

1. **archive-{post_type}.php** – If you have a post type is review, WordPress will look for archive-review.php.
2. **archive.php** – The default template to display all archive pages in WordPress.
3. **index.php** – The default fallback template in WordPress.

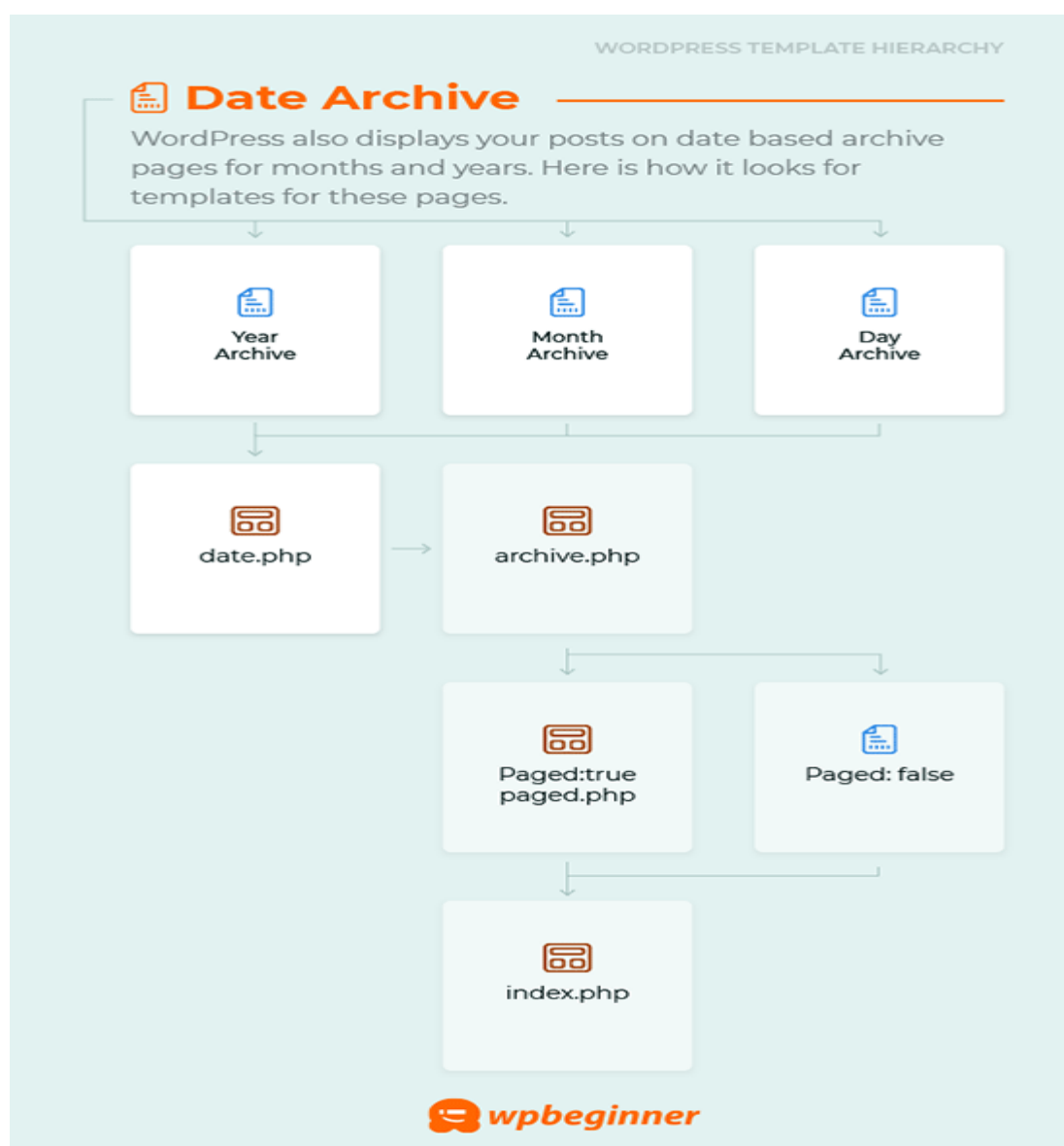
Which template files are used to display author archives?



WordPress generates archive pages for each author on your WordPress site. Here is how it looks for author archive template.

1. **author-{nickname}.php** – If the author's nice name is matt, WordPress will look for author-matt.php.
2. **author-{id}.php** – If the author's user ID is 6, then WordPress will look for author-6.php.
3. **author.php** – The default template used to display author archive pages in WordPress.
4. **archive.php** – The default template to display all archive pages in WordPress.
5. **index.php** – The default fallback template in WordPress.

Which template files are used to display date based archives?



WordPress also displays your posts on date based archive pages for months and years. Here is how it looks for templates for these pages.

1. **date.php** – The default template for date based archives.
2. **archive.php** – The default template used to display author archive pages in WordPress.
3. **index.php** – The default fallback template in WordPress.

Which template files are used to display search pages?



1. **search.php** – The default page to display search results in WordPress.
2. **searchform.php** – The template to display a search form in WordPress.
3. **index.php** – The default fallback template in WordPress.

Which template files are used to display 404 error pages?

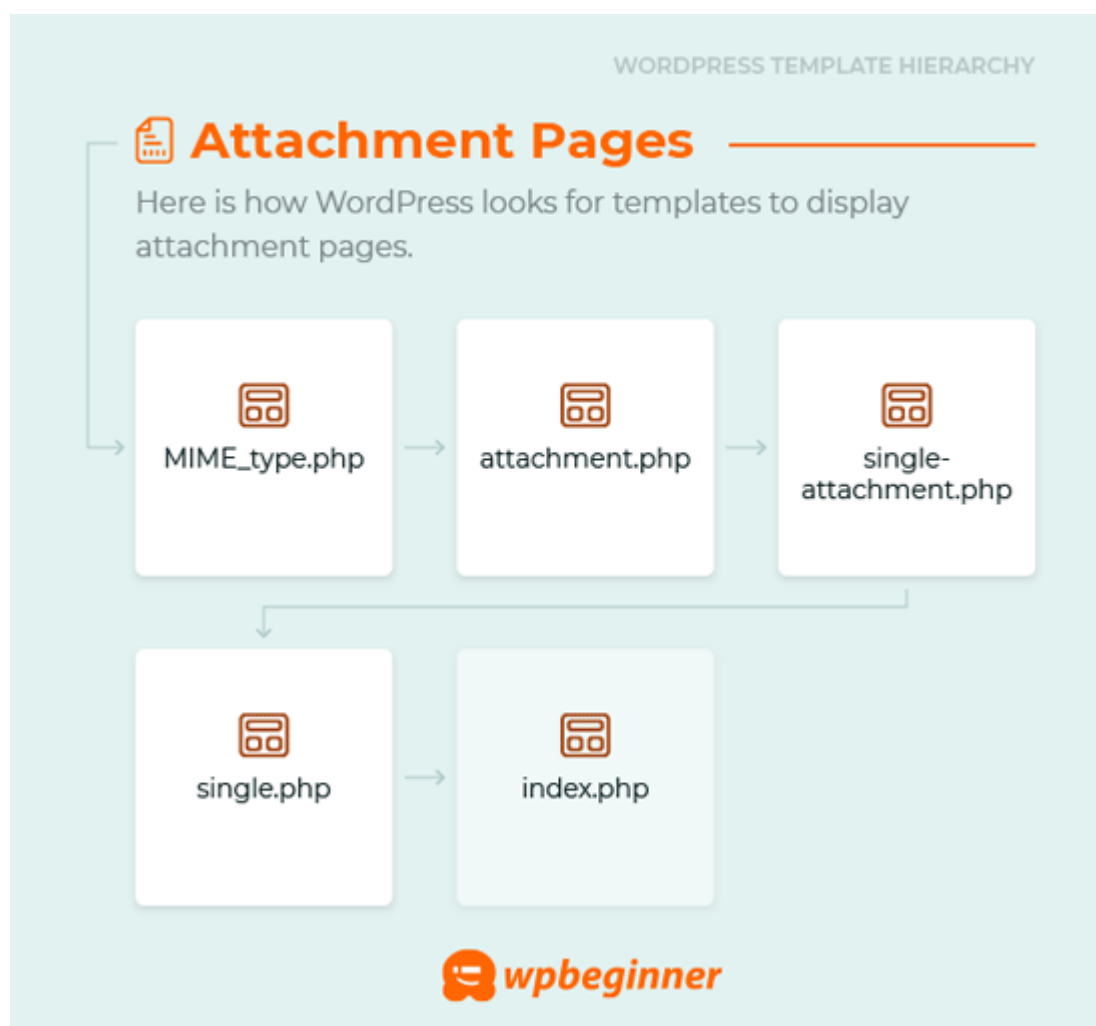


The 404 error page is displayed when WordPress is unable to find the requested content. See our guide on how to improve your 404 page template.

1. **404.php** – The default template to display 404 error page in WordPress.

2. **index.php** – The default fallback template in WordPress.

Which template files are used to display attachment pages?



1. **MIME_type.php** – Mime_type stands for file type. For example, image.php, video.php, application.php.

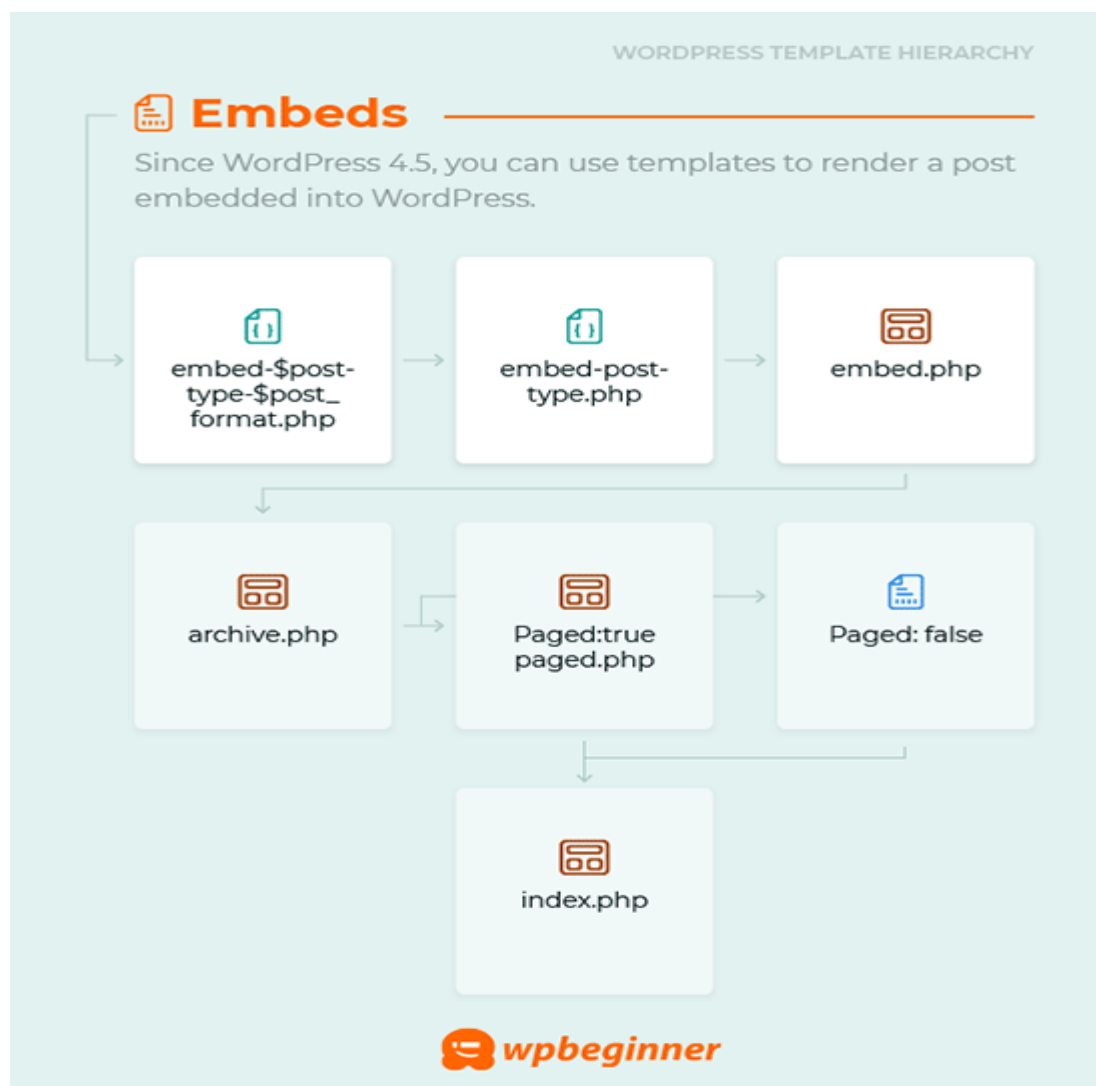
2. **attachment.php** – The default template to display attachment pages.

3. **single-attachment.php** – To display a single attachment.

4. **single.php** – The default template to display single post type items.

5. **index.php** – The default fallback template in WordPress.

Which template files are used to display embeds?



Since WordPress 4.5, you can use templates to render a post embedded into WordPress.

1. **embed-{post-type}-{post_format}.php** – WordPress will look for a post type and post format template first. For example, if you have a review with video, then WordPress will look for `embed-review-video.php`.
2. **embed-{post-type}.php** – If the post type is review, WordPress would look for `embed-review.php`.
3. **embed.php** – The default fallback for all embeds.

We hope this tutorial helped you learn about the WordPress template hierarchy.

How to Effectively Validate a WordPress Website

Validation is undoubtedly one of the most important aspects of good web design. And in a content management system like WordPress, you need to be careful about your HTML & CSS both in your theme and your content. In this tutorial, we're going to talk about validating a WordPress website.

Why Validate a Website?

Or, perhaps the better question is what is the meaning of validation?

Simply put, validation of a web page means making the page's code compliant with the World Wide Web Consortium's (W3C) standards. This is usually done by checking the HTML & CSS output of your web pages with a validation tool. Just like there are grammar rules in different languages, there are rules in computer programming, too. Validation checks your web page to see if the page follows those rules.

In addition to that, we also need to keep our theme's code compliant with WordPress Coding Standards. This could be done manually (by having common sense while coding a WordPress theme) or using tools like the famous Theme Check plugin.

And why do we need to validate, you ask? There's more than one answer. Let's have a look:

Validation Allows You to Debug Your Pages

When I first started to toy around with HTML in 2005, I coded those awful web pages, full of errors like improper nesting of elements or forgetting to close the tags I opened. CSS was even worse: Just a simple semicolon error or an undetectable typo meant that I needed more and more time for my experiments and projects.

But even after nine years of hard work, I can make mistakes despite the fact that I'm *almost* an expert in HTML and CSS. Luckily, tools like W3C's HTML validator can help us debug almost every error there is in your HTML code. Whether it be by pasting the code directly, uploading the HTML file or submitting the URI of the web page, you can see what did you do wrong in your code, where the problem is, and suggestions on how to fix it. Fixing those errors might even prevent greater problems.

Validation Makes Your Website More Accessible & Compatible

Have you ever thought about how a browser *reads* a web page? They have "engines" to parse the code and convert it into a visual experience for human beings. Unfortunately, different web browsers have different engines and that can result in handling of your pages differently.

In a valid web page, most differences are really tiny and acceptable for almost everyone, but an invalid web page might be read so differently that people might not even be able to see your pages properly in their iPhones, Kindles or whatever browser they may be using. Validation fixes almost every major difference and make your web page readable by nearly all web browsers.

Crucial WordPress Maintenance Tasks to Perform Regularly

Ever wondered which important maintenance tasks you should perform regularly on your WordPress site?

Routine WordPress maintenance tasks take very little time, but they ensure that your site is secure, safe, and running at peak performance.

Why and When to Perform WordPress Maintenance Tasks

Your WordPress site is a powerful system made of several parts. This includes your WordPress hosting, the WordPress software itself, plugins, and themes.

On top of that, you add your own content with text and images. Together, all of these make a website that is loved by your visitors and customers.

After starting a blog or website, many website owners do not perform maintenance checks unless something breaks.

However, if you want optimal performance, then you need to perform simple maintenance tasks on a regular basis. These maintenance tasks ensure that your website is always in the best shape.

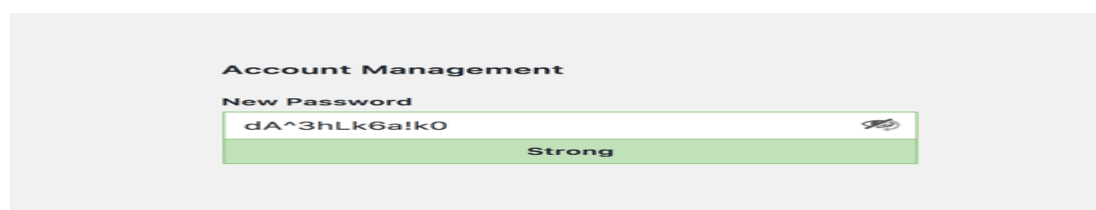
How often should you perform WordPress maintenance tasks?

If you run a busy website with a lot of traffic, then you should go through this maintenance checklist every three months. For smaller websites with low traffic and content, you need to do these maintenance tasks every six months.

Let's take a look at the essential WordPress maintenance tasks you need to perform and how to do them. You can use these quick links to jump straight to each task.

1. Change All Your WordPress Passwords
2. Create a Complete Backup of Your Website
3. Check and Update All WordPress Files
4. Check and Delete Spam Comments
5. Test All Your WordPress Forms
6. Optimize Your WordPress Database
7. Run Performance Tests
8. Find and Fix 404 Errors
9. Find and Fix Broken Links
10. Perform a Thorough Content and SEO Audit
11. Optimize Images on Your WordPress Site
12. Review WordPress Security Logs
13. Troubleshoot Maintenance Tasks

1. Change All Your WordPress Passwords



Passwords are your first defense against unauthorized access to your website. You should always use strong unique passwords for all your online accounts. These include your WordPress website, FTP accounts, and database.

However, even if you are using strong passwords, they could be compromised without you even knowing.

That's why WordPress security experts recommend changing your WordPress passwords regularly. This includes passwords for your WordPress admin area, FTP or SSH accounts, and your WordPress database password.

The problem with strong passwords is that they are harder to remember. This is why we recommend using password manager apps like LastPass to securely store and easily fill passwords without having to type them.

2. Create a Complete Backup of Your Website



Backups are one of the most important WordPress plugins in your arsenal.

There are plenty of great WordPress backup plugins like UpdraftPlus or BackupBuddy. These can help you completely automate the WordPress backup process.

However, sometimes your backup solution may suddenly stop working without you even noticing.

Once in a while, you need to manually run your backup plugin to create a complete backup of your website. After running the backup, check that your backup files are properly stored at the remote location of your choice (Dropbox, Google Drive, etc).

3. Check and Update All WordPress Files



WordPress comes with a built-in system to manage updates for WordPress core, plugins, and themes. You should always update WordPress to use the latest version as well as keeping all your plugins and themes updated.

With that said, there are some situations when you may miss an update. For example, when a premium plugin or theme's license expires, it may fail to check for an update.

Go to the WordPress Updates page to manually check for updates. Review all your installed plugins and themes to make sure that they are running the latest version. If they're not, then make sure to document a reason why you're choosing not to update.

4. Check and Delete Spam Comments



Many website owners use Akismet to combat comment spam in WordPress. It automatically keeps spam away from your comment moderation queue.

However, sometimes Akismet may end up marking a legitimate comment as spam. Once in a while, you need to take a quick look at the spam comments to ensure that there are no real comments incorrectly marked as spam.

Once you are done, you can safely delete all spam comments from your website. If you have thousands of spam comments, then you should batch delete all spam comments in WordPress. It will not necessarily improve performance, but it will ensure that you don't miss genuine comments.

5. Test All Your WordPress Forms



WordPress form builder plugins like WPForms make it super easy to create beautiful forms on your website.

However due to misconfiguration on your WordPress hosting server or your email service provider, sometimes these forms may suddenly stop sending emails.

You need to check all forms on your website to make sure that they are working properly. If a form is not working, then see our guide on fixing the WordPress not sending email issue.

As a best practice, we recommend using WP Mail SMTP plugin because it has email logging, and it will alert you when an email fails to send.

6. Optimize Your WordPress Database

WordPress stores most data in your WordPress database. It contains all your content, comments, users, and settings.

Over time, your database may gather a lot of unnecessary data. This increases your WordPress backup sizes which may affect uploading, downloading, and restoring backups.

Optimizing your WordPress database allows you to clean up clutter, defragment tables, and improve database performance.

7. Run Performance Tests



Many users optimize their WordPress performance when they first start their website, and then forget about it.

As time goes by, you add new content, install new plugins, or may even change the theme. All of these items may affect the performance of your WordPress site.

Faster websites are not just good for user experience, but they also improve your SEO rankings. This is why you need to regularly do a thorough performance review of your website.

When reviewing your site's performance, don't just limit it to improving your homepage. Also test your most popular content and all your important pages.

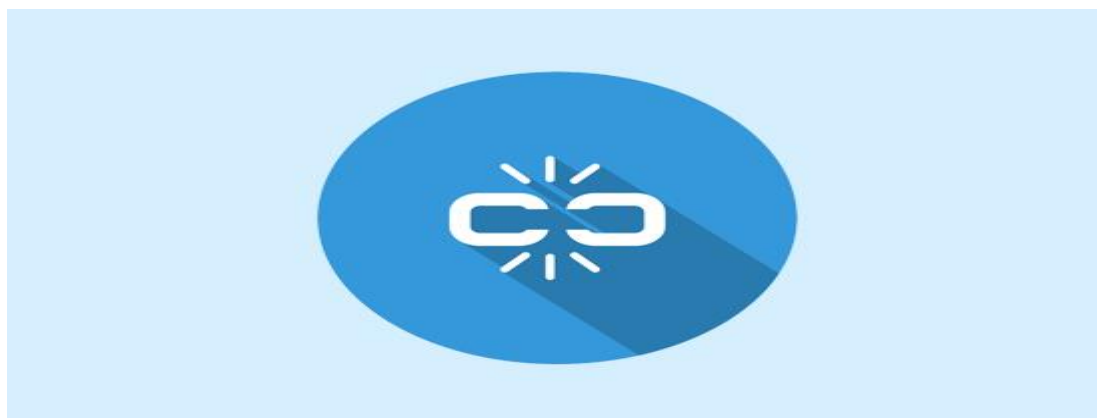
8. Find and Fix 404 Errors



When a user requests a page that doesn't exist on your website, then WordPress will show them a 404 error page.

404 errors that occur because a user mistyped an address are normal and nothing to be worried about. However, 404 errors that occur because a page is no longer available are frustrating for users and create a bad user experience.

9. Find and Fix Broken Links



As your website grows, you will realize that some external websites that you linked to in your older articles do not exist anymore. Some may have moved to new locations, while others may just disappear.

The broken links issue is not just limited to external links. You might accidentally add broken images, poorly formatted links, or misspell your own links. This can be frustrating for your visitors and harms your site's user engagement.

You need to check your website for broken links as part of your WordPress maintenance routine.

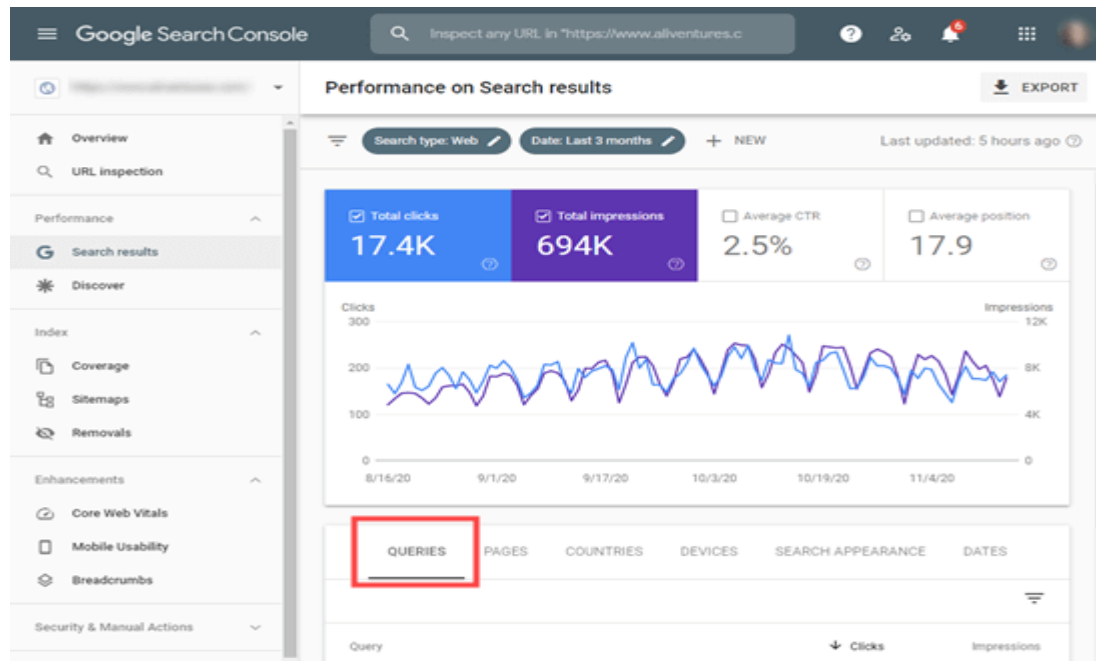
10. Perform a Thorough Content and SEO Audit



The next thing you need to include in your regular maintenance tasks is a thorough in-depth review of your content. This is where the data from Google Search Console and Google Analytics comes in.

Google Analytics shows you where your visitors are coming from and what they are doing on your website. This data allows you to discover content on your website where you have lots of traffic but your conversion rate is low.

Google Search Console's Search Performance page can help you find search keywords where your site appears in the results. You can sort it to show you keywords where your site can easily rank higher by updating those articles.



If you are using All in One SEO, then you can set a particular keyword as your focus keyphrase. All in One SEO will give you a True SEO Score plus specific optimization tips.

Focus Keyphrase

blog post ideas 62/100

- Focus Keyphrase in URL ▼
Focus Keyphrase not found in the URL.
- Focus keyphrase in introduction
- Focus Keyphrase in Subheadings ▼
Use your focus keyphrase more in your H2 and H3 subheadings
- Focus keyphrase in image alt attributes
- Focus keyphrase length
- Focus keyphrase density ▼
Focus keyphrase Density is 0.07 which is low, the Keyphrase appears 1 times.

Additional Keyphrases

Add Additional Keyphrases

Even with an excellent SEO score, you can still further improve your content. Try adding new information, adding images, and linking to it from other pages on your website.

11. Optimize Images on Your WordPress Site



Images take longer to load than text. This means they decrease your page load speed. You will discover some overly large images during the performance checkup of your site.

However, you will likely miss those in your less popular articles. If you run a multi-author WordPress site, then some of your authors may not be as careful about image sizes as you are.

Reviewing your images and media library allows you to stay on top of the issue. You can perform this check to find images that can be reduced in size or images that are just too large.

12. Review WordPress Security Logs



Some WordPress users don't realize that their site is under attack until it slows down or their search rankings drop.

We have already mentioned some security precautions like changing passwords, and creating manual backups as proactive measures. You also need to review your site's access and error logs to see if there is any unusual activity on your site.

Another good option is to add a security audit plugin to your site.

We also recommend using Sucuri. It is a website security company that offers a website firewall to protect your website against common threats.

13. Troubleshoot Maintenance Tasks



Most WordPress website maintenance tasks are quite harmless and will not affect your website's normal functioning. However, some may slow down your site, like checking for broken links or running an image optimizer plugin.

If you run a staging site, then you can perform your maintenance tasks on your staging site and then push them live.

However, most WordPress users don't run a staging site. In that case, you will have to expect a temporary slow site and some unexpected errors.

One way to deal with this is by putting your WordPress site in maintenance mode. Alternatively, you can perform these tasks during your low traffic hours.

If you run across an issue, then see our guide on how to fix common WordPress errors. If the error you are seeing is not listed there, then follow the steps in our WordPress troubleshooting guide. It will help you locate the problem and find a solution.

We hope this article helped you learn crucial WordPress maintenance tasks to perform regularly on your website. You may also want to see our tips on how to increase your blog traffic and our comparison of best WordPress page builders to create custom design layouts without any code.

Experiment – 6

Aim: - Integrate PHP & MySql with WordPress

WordPress is written using PHP as its scripting language and MySQL as its database management system. In order to use WordPress, you don't really need to learn either of them.

However, a very basic understanding of the two could help you troubleshoot problems as well as increase your understanding of how WordPress works

We will also show you how to manage a WordPress database using phpMyAdmin which will include creating database backups, optimizing the WordPress database, and more.

What is a Database and How WordPress Uses it?

A database is a system of storing and fetching data in an organized way. Database allows software to manage data in a programmable way.

For example: WordPress uses PHP (programming language) to store and retrieve data from the database.

The information stored in a WordPress database includes posts, pages, comments, categories, tags, custom fields, users, and other WordPress settings. We will elaborate on this later in the tutorial when we cover all WordPress database tables.

When you first install WordPress, it asks you to provide a database name, host, username, and password. This information is stored in the configuration file called wp-config.php.

The image shows the WordPress installation screen for database configuration. At the top is the WordPress logo. Below it, a message says: "Below you should enter your database connection details. If you're not sure about these, contact your host." There are five input fields with labels and descriptions to their right: "Database Name" with value "wordpress" (description: "The name of the database you want to use with WordPress."), "Username" with value "username" (description: "Your database username."), "Password" with value "password" (description: "Your database password."), "Database Host" with value "localhost" (description: "You should be able to get this info from your web host, if localhost doesn't work."), and "Table Prefix" with value "wp_" (description: "If you want to run multiple WordPress installations in a single database, change this."). A "Submit" button is at the bottom left of the form area.

During the installation, WordPress uses the information you provide about the database to create tables and store default installation data inside those tables.

After the installation, WordPress runs queries to this database to dynamically generate HTML pages for your website or blog.

This is what makes WordPress extremely powerful because you don't have to create a new .html file for each page that you want to create. WordPress handles everything dynamically.

Where is my WordPress database stored?

WordPress uses MySQL as its database management system, which is a software installed on your WordPress hosting server. Your WordPress database is also stored on the same server.

This location however is not accessible on most shared hosting environments. However, if you are on a VPS hosting plan or a dedicated server, then you can use command line tools to locate the database.

Most often it is located at the following path:

```
/var/lib/mysql/your_database_name
```

However, this could differ from one hosting provider to another.

It is important to keep in mind that you don't really need to access the database file itself. You can use other tools like phpMyAdmin to manage your database.

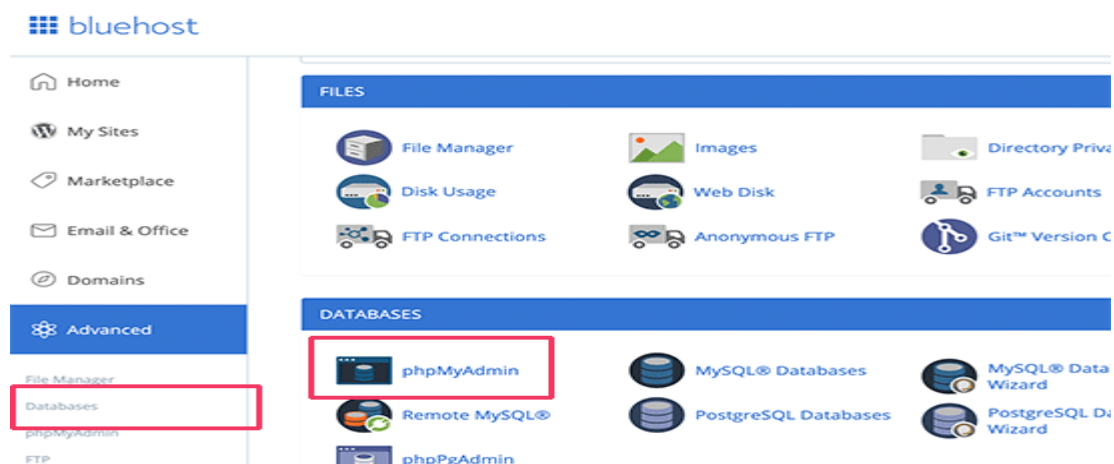
What is phpMyAdmin?

PhpMyAdmin is a web-based software that allows you to manage MySQL databases using your web browser. It offers an easy to use interface that allows you to run MySQL commands and database operations.

You can also use it to browse and edit database tables, rows, and fields. It also allows you to import, export, or delete all data inside a database.

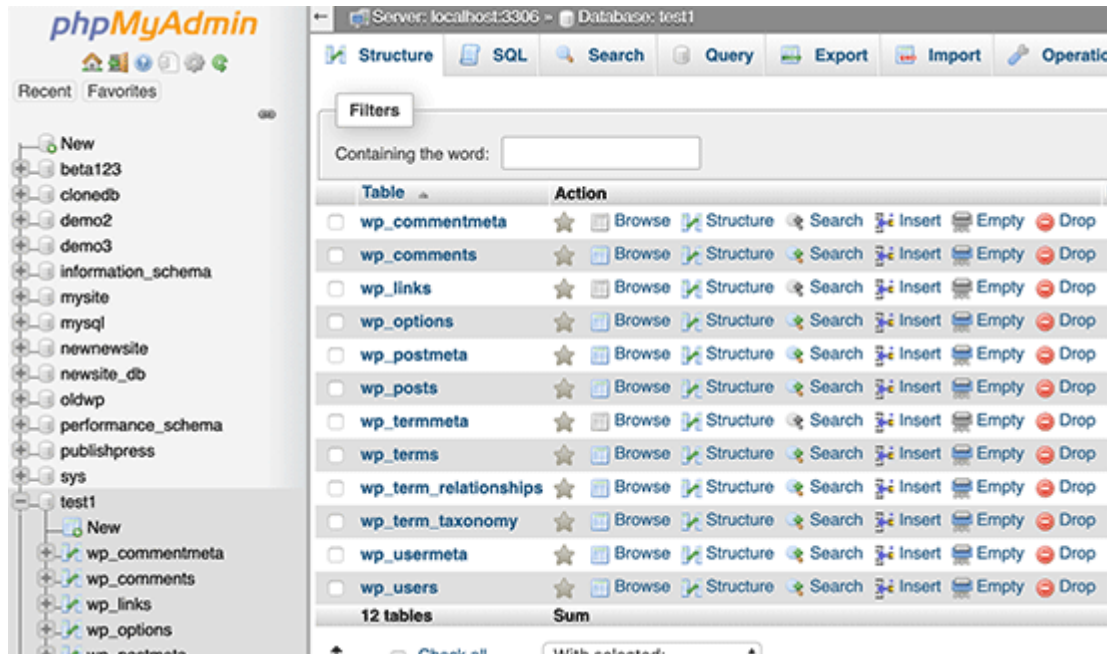
How to Access PhpMyAdmin?

PhpMyAdmin comes pre-installed with all top WordPress hosting companies. You can find it under the Databases section of your hosting account's cPanel dashboard. Below is an example screenshot from Bluehost control panel:



Depending on your hosting provider, your cPanel interface may look different than the above screenshot. You would still be able to find phpMyAdmin icon under the databases section.

Clicking on it will open the phpMyAdmin interface where you can select your WordPress database from the left column. After, that phpMyAdmin will display all tables in your WordPress database.



Let's take a look at the WordPress database tables.

Understanding WordPress Database Tables

Each WordPress installation has 12 default tables in the database. Each database table contains data for different sections, features, and functionality of WordPress.

Looking at the structure of these tables, you can easily understand where different parts of your website are stored. Currently, a default WordPress installation creates the following tables:

Note: wp_ before each table name is the database prefix you choose during the installation. It can be different if you changed it at the time of installation.

wp_commentmeta : This table contains meta information about comments posted on a WordPress website. The table has four fields meta_id, comment_id, meta_key, and meta_value. Each meta_id is related to a comment_id. One example of comment meta information stored is the status of comment (approved, pending, trash, etc).

wp_comments : As the name suggests this table contains your WordPress comments. It contains comment author name, url, email, comment, etc.

wp_links : To manage blogrolls create by earlier versions of WordPress or the Link Manager plugin.

wp_options : This table contains most of your WordPress site-wide settings such as: site url, admin email, default category, posts per page, time format, and much much more. The options table is also used by numerous WordPress plugins to store plugin settings.

wp_postmeta : This table contains meta information about your WordPress posts, pages, and custom post types. Example of post meta information would be which template to use to display a page, custom fields, etc. Some plugins would also use this table to store plugin data such as WordPress SEO information.

wp_posts : The name says posts but actually this table contains all post types or should we say content types. This table contains all your posts, pages, revisions, and custom post types.

wp_termmeta : This table allows developers to store custom metadata for terms under their custom taxonomies. For example, WooCommerce uses it to store metadata for product attributes and categories.

wp_terms : WordPress has a powerful taxonomy system that allows you to organize your content. Individual taxonomy items are called terms, and they are stored in this table. For example, your WordPress categories and tags are taxonomies, and each category / tag inside them is a term.

wp_term_relationships : This table manages relationship of WordPress post types with terms in wp_terms table. For example this is the table that helps WordPress determine post X is in Y category.

wp_term_taxonomy : This table defines taxonomies for terms defined in wp_terms table. For example, if you have a term “WordPress Tutorials“, then this table contains the data that says it is associated with a taxonomy called category. In short, this table has the data that helps WordPress differentiate between which term is a category, which is a tag, etc.

wp_usermeta : Contains meta information about registered users on your website.

wp_users : Contains user information like username, password, user email, etc.

Managing WordPress Database using phpMyAdmin

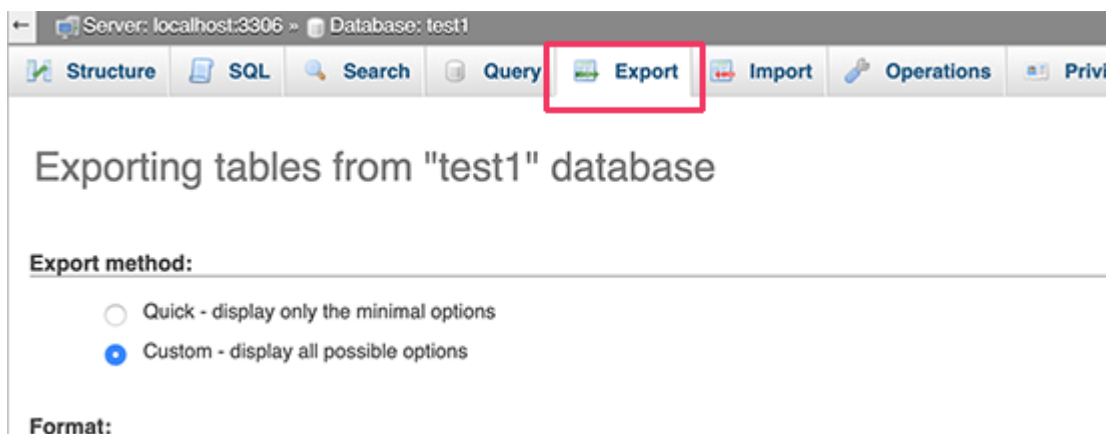
Your WordPress database contains important WordPress settings, all your blog posts, pages, comments, and more. You need to be extremely careful when using phpMyAdmin, or you may end up accidentally deleting important data.

As a precaution, you should always create a complete database backup. This would allow you to restore your database the way it was before.

Let’s take a look at how to easily create a WordPress database backup.

Creating a WordPress Database Backup using phpMyAdmin

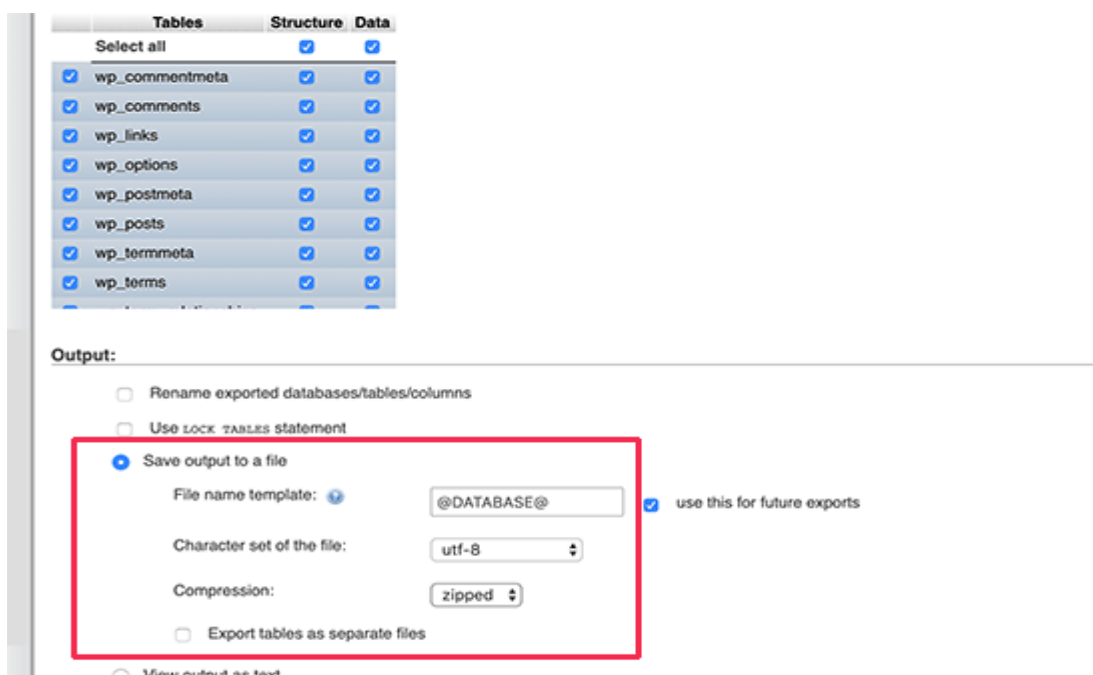
To create a backup of your WordPress database from phpMyAdmin, click on your WordPress Database. On the top menu, click on the **Export** tab.



In newer versions of phpMyAdmin, it will ask you for an export method. The quick method will export your database in a .sql file. In the custom method, it will provide you with more options and ability to download backup in compressed zip or gzip archive.

We recommend using the custom method and choosing zip as the compression method. The custom method also allows you to exclude tables from the database.

Let's say if you used a plugin that created a database table, and you are no longer using that plugin, then you can choose to exclude that table from the backup if you want.



Your exported database file can be imported back into a different or the same database using phpMyAdmin's import tab.

Creating a WordPress Backup using a Plugin

Keeping regular backups of your WordPress site is the best thing you can do for your WordPress security. While the WordPress database contains the majority of your site information, it still lacks several fairly important elements such as template files, images, uploads, etc.

All your images are stored in the uploads folder in your /wp-content/ directory. Even though the database has the information which image is attached to a post, it is useless if the image folder doesn't have those files.

Often beginners think that the database backup is all that they need. It is NOT true. You need to have a full site backup that includes your themes, plugins, and images.

While most hosting companies claim they have daily backups, **we recommend that you keep backups yourself just in case.**

Note: If you are on a managed WordPress hosting solution like WPEngine, then they do create daily backups.

For the majority of us who are not on WPEngine, you should use a WordPress backup plugin to set up automated WordPress backups on your site.

Importing a WordPress Database Backup via phpMyAdmin

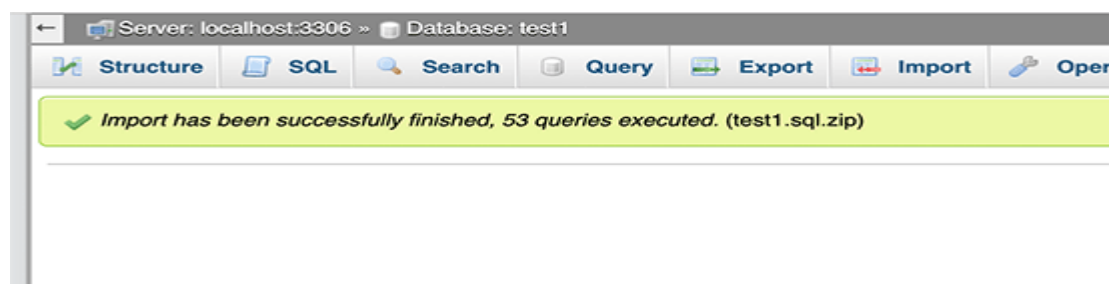
PhpMyAdmin also allows you to easily import your WordPress database. Simply launch phpMyAdmin and then select your WordPress database.

Next, you need to click on the 'Import' link from the top menu.



On the next screen, click on the Choose file button and then select your database backup file you downloaded earlier.

PhpMyAdmin will now process your backup file upload and import it into your WordPress database. Once finished, you will see a success message.

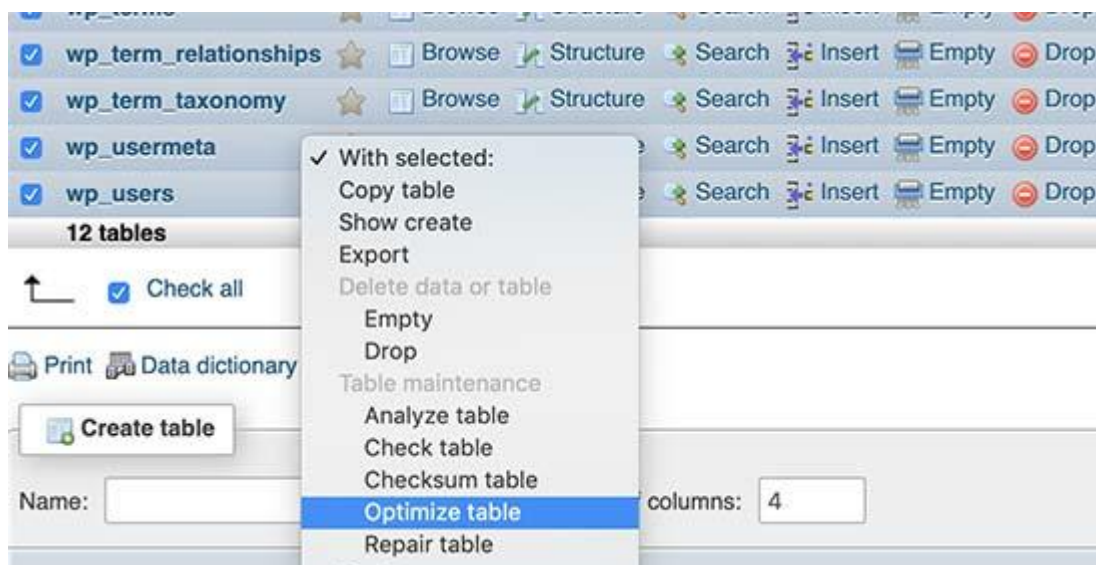


Optimizing your WordPress Database in phpMyAdmin

After using WordPress for a while, your database becomes fragmented. There are memory overheads which increase your overall database size and query execution time.

MySQL comes with a simple command that allows you to optimize your database. Simply go to phpMyAdmin and click on your WordPress database. This will show you a list of your WordPress tables.

Click on the check All link below the tables. Next to it, there is a “With Selected” drop down, you need to click on it and choose Optimize table.



This will optimize your WordPress database by defragmenting selected tables. It will make your WordPress queries run a little faster and slightly reduce the size of your database.

Fixing WordPress Issues using PhpMyAdmin

As we mentioned earlier, phpMyAdmin is a handy tool for troubleshooting and fixing some common WordPress errors and issues.

Let’s check out some common WordPress issues that can be easily fixed using phpMyAdmin.

Reset WordPress Password Using PhpMyAdmin

If you forgot your WordPress admin password and cannot recover it via lost password email, then this method allows you to quickly reset WordPress password.

First, launch the phpMyAdmin and select your WordPress database. This will display your WordPress database tables where you need to browse next to **wp_users** table.

☐	wp_terms	★	Browse	Structure	Search	Insert	
☐	wp_term_relationships	★	Browse	Structure	Search	Insert	
☐	wp_term_taxonomy	★	Browse	Structure	Search	Insert	
☐	wp_usermeta	★	Browse	Structure	Search	Insert	
☐	wp_users	★	Browse	Structure	Search	Insert	
12 tables		Sum					

Note: Table names in your WordPress database may have a different table prefix than the one we are showing in our screenshot.

You will now see the rows in your WordPress users table. Go ahead and click on the edit button next to the username where you want to change the password.

☐ Show all
 | Number of rows: 25
 | Filter rows: Search this table

+ Options

		ID	user_login	user_pass
☐	Edit	1	peter	\$P\$BfrBEpitP3A4Q941m7Rr
☐	Edit	2	elle	\$P\$B2lxKFmzv/52fPmhr9jgT

☐ Check all
 | With selected: Edit Copy Delete Export

PhpMyAdmin will show you a form with all the user information fields.

You will need to delete the value in the **user_pass** field and replace it with your new password. Under the function column, select MD5 from the drop-down menu and click on the **Go** button.

Column	Type	Function	Null	Value
ID	bigint(20) unsigned			1
user_login	varchar(60)			peter
user_pass	varchar(255)	MD5		pQ\$3trI^ds
user_nicename	varchar(50)			peter
user_email	varchar(100)			

Your password will be encrypted using the MD5 hash and then it will be stored in the database.

Congratulations! You have successfully changed your WordPress password using phpMyAdmin.

Now some of you may be wondering why did we select the MD5 hash to encrypt the password.

In the older version, WordPress used MD5 hash to encrypt passwords. Since WordPress 2.5, it started using stronger encryption technologies. However, WordPress still recognizes MD5 to provide backward compatibility.

As soon as you log in using a password string stored as an MD5 hash, WordPress recognizes it and changes it using the newer encryption algorithms.

Adding a New Admin User to WordPress using PhpMyAdmin

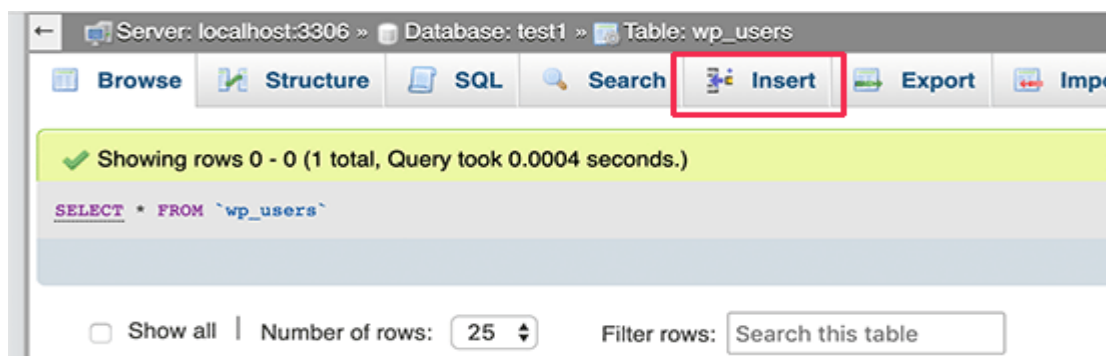
Let's suppose you have access to the WordPress database but not the WordPress admin area. While you can change the admin user password, it will prevent the other admin user from using their account.

An easier solution would be to add a new admin user via phpMyAdmin.

First, you need to launch the phpMyAdmin and then select your WordPress database. This will show your WordPress database tables where you need to click on the 'Browse' link next to wp_users table.



phpMyAdmin will now show you the rows inside the wp_users table. Go ahead and click on the 'Insert' link from the menu on the top.



This will bring up a form that you need to fill up to add a new user to your WordPress site.

Column	Type	Function	Null	Value
ID	bigint(20) unsigned			
user_login	varchar(60)			jsmith
user_pass	varchar(255)	MD5		Y3SMaj#8qjA9FZGus5M
user_nicename	varchar(50)			jsmith
user_email	varchar(100)			jsmith@example.com
user_url	varchar(100)			
user_registered	datetime	CURRENT_TIME		0000-00-00 00:00:00 
user_activation_key	varchar(255)			
user_status	int(11)			0
display_name	varchar(250)			

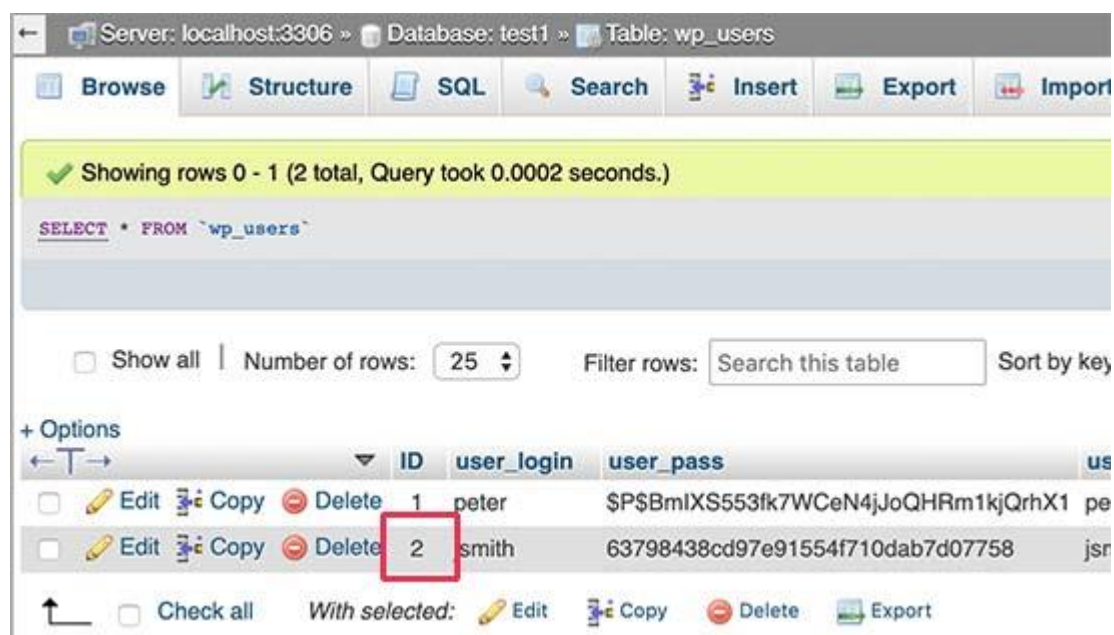
Here is how you need to fill each field in this form.

- **ID** – You can ignore this one as this is automatically generated.
- **user_login** – This will be your WordPress username that you’ll use to login.
- **user_pass** – This is your WordPress password you need to enter the password and select MD5 in the function column.
- **user_nicename** – This is the URL friendly username you can use the as your login.
- **user_email** – Enter a valid email address as you may need it to recieve password reset and WordPress notification emails.
- **user_url** – Add your website URL or you can leave it blank.
- **user_registered** – You need to select the CURRENT_TIME in the function column to automatically insert current time here.
- **user_activation_key** – You can leave this field blank as well it is used to approve user registration.
- **user_status** – You can leave this field blank as well.
- **display_name** – You can enter the user’s full name as you want to be displayed on the articles. You can also leave it blank.

After filling in the form, click on the Go button to insert it into your WordPress database. PhpMyAdmin will now run the MySQL query to insert the data.

We have added the user, but that user does not have an administrator user role on your WordPress site. This value is saved in another table called **wp_usermeta**.

Before we can make this user an administrator, we will need to find the user ID. Simply click on the 'Browse' link next to **wp_users** table and you will see a row containing your newly added user with their user ID.



Server: localhost:3306 » Database: test1 » Table: wp_users

Showing rows 0 - 1 (2 total, Query took 0.0002 seconds.)

```
SELECT * FROM `wp_users`
```

Number of rows: 25 Filter rows: Search this table Sort by key

	ID	user_login	user_pass
☐ Edit ☐ Copy ☐ Delete	1	peter	\$P\$BmIXS553fk7WCeN4jJoQHRm1kjQrhX1 pe
☐ Edit ☐ Copy ☐ Delete	2	smith	63798438cd97e91554f710dab7d07758 jsr

Check all With selected: Edit Copy Delete Export

Note down the user ID as you'll need it in the next step.

Now, let's open the wp_usermeta table by clicking on the Browse link next to it.



☐ wp_termmeta	★	Browse	Structure	Search	Insert	Empty	Drop
☐ wp_terms	★	Browse	Structure	Search	Insert	Empty	Drop
☐ wp_term_relationships	★	Browse	Structure	Search	Insert	Empty	Drop
☐ wp_term_taxonomy	★	Browse	Structure	Search	Insert	Empty	Drop
☐ wp_usermeta	★	Browse	Structure	Search	Insert	Empty	Drop
☐ wp_users	★	Browse	Structure	Search	Insert	Empty	Drop
12 tables	Sum						

Next, you need to click on the Insert link at the top to add a new row to the table.



Server: localhost:3306 » Database: test1 » Table: wp_usermeta

Browse Structure SQL Search **Insert** Export Import

Showing rows 0 - 14 (15 total, Query took 0.0003 seconds.)

```
SELECT * FROM `wp_usermeta`
```

You'll now see a form to enter a new row. This is where you'll tell WordPress that the user you created earlier has the administrator user role.

Column	Type	Function	Null	Value
umeta_id	bigint(20) unsigned		☐	
user_id	bigint(20) unsigned		☐	2
meta_key	varchar(255)		☐	wp_capabilities
meta_value	longtext		☐	a:1:{s:13:"administrator";b:1;}

Here is how you'll fill in this form.

- **umeta_id** – You need to leave it blank as it is automatically filled in.
- **user_id** – Enter the user ID you copied earlier.
- **meta_key** – You need to enter **wp_capabilities** in this field. However, you may need to replace **wp_** if your WordPress table names use a different prefix.
- **meta_value** – You need to enter the following serialized value:
a:1:{s:13:"administrator";s:1:"1";}

Finally, click on the Go button to save your changes.

Next, we need to add another row to define the user level. Click on the Insert link on the top menu to add another row to the usermeta table.

Column	Type	Function	Null	Value
umeta_id	bigint(20) unsigned		☐	
user_id	bigint(20) unsigned		☐	2
meta_key	varchar(255)		☐	wp_user_level
meta_value	longtext		☐	10

Here is how you'll fill in this form.

- **umeta_id** – You need to leave it blank as it is automatically filled in.
- **user_id** – Enter the user ID for your newly added user.
- **meta_key** – You need to enter **wp_user_level** in this field. However, you may need to replace **wp_** if your WordPress table names use a different prefix.
- **meta_value** – Here you need to enter **10** as the user level value.

Don't forget to click on the **Go** button to insert the data.

That's all, you can now visit the WordPress admin area and log in with your newly added admin user.

Change a WordPress Username via PhpMyAdmin

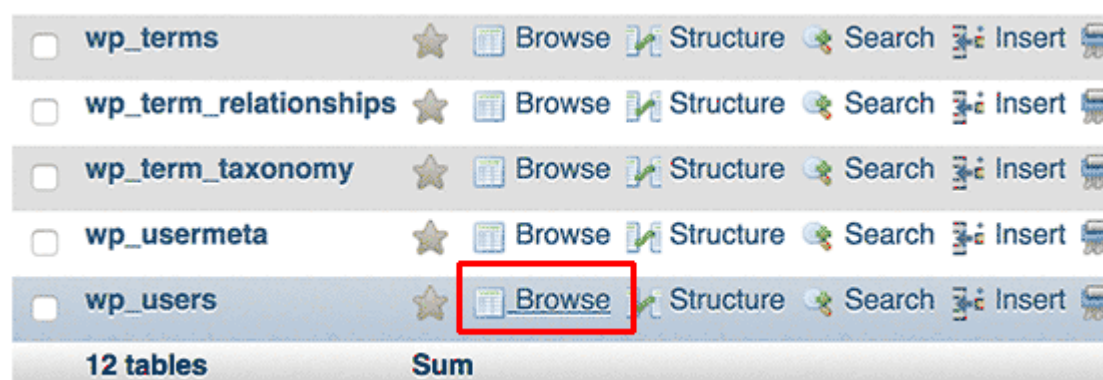
You may have noticed that while WordPress allow you to change user's full name or nickname, it does not allow you to change the username.

Now a lot of users end up choosing usernames during the installation that they may later want to change. Here is how you can do this via phpMyAdmin.

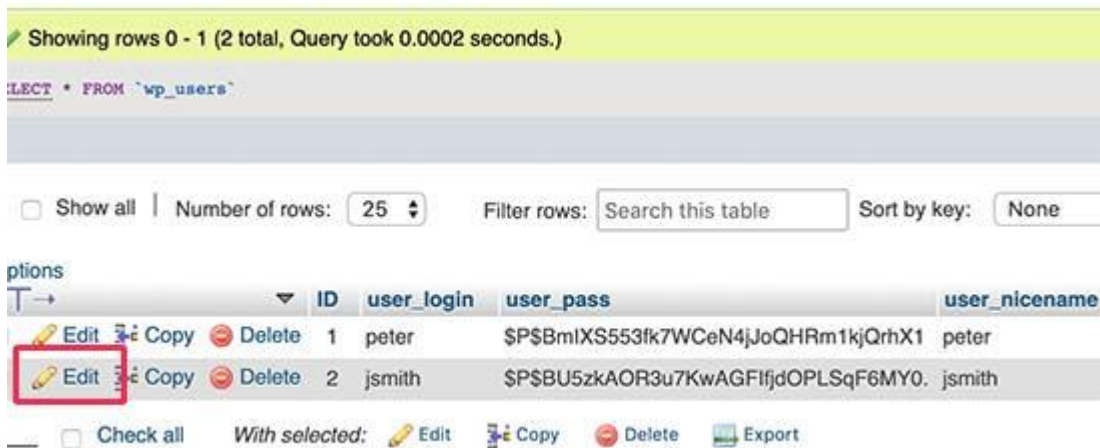
Note: there is an easier way to [change WordPress username using a plugin](#).

First, you need to launch the phpMyAdmin from your hosting account's cPanel dashboard. After that, you need to select your WordPress database.

PhpMyAdmin will show your WordPress database tables. You need to click on the 'Browse' link next to **wp_users** table.



This table will now list all the registered users on your website. Click on the edit link next to the username that you want to change.



PhpMyAdmin will now show you the user data. You need to locate the user_login field and change the value to the username you want to use.

Column	Type	Function	Null	Value
ID	bigint(20) unsigned			2
user_login	varchar(60)			jsmith
user_pass	varchar(255)			\$P\$BU5zkAOR3u7KwAGFIIfjdOPLSqF6MY
user_nicename	varchar(50)			jsmith
user_email	varchar(100)			jsmith@example.com
user_url	varchar(100)			
user_registered	datetime			2010-05-23 00:40:24

Don't forget to click on the Go button to save your changes. You can now visit your website and login with your new username.

Other Useful PhpMyAdmin Tricks

Your WordPress database is the engine behind your website. With phpMyAdmin you can tweak settings to improve performance, fix issues, or simply change things that cannot be changed from inside WordPress admin area.

Following are just a few more tricks you can use with phpMyAdmin.

- Deactivate all WordPress plugins
- Change WordPress theme via phpMyAdmin
- Duplicate WordPress database via phpMyAdmin
- Find and replace text in WordPress database

Securing your WordPress Database

Before we get into this, we want to emphasize that every site can be hacked. However, there are certain measures you can take to make it a little harder.

First thing that you can do to is to change WordPress database prefix. This can significantly reduce the chances of SQL injection attacks on your WordPress database because often hackers target sites in masses where they are targeting the default wp_ table prefix.

You should always choose a strong username and password for your MySQL user. This will make it difficult for someone to get access to your WordPress database.

For WordPress security, we strongly recommend that you use Sucuri. It is the best WordPress security plugin that would catch any attempts at MySQL injection even before it reaches your website.

Experiment – 7

Aim: - Install Moodle & various plugins

Installing Moodle

If you just want to try Moodle on a standalone machine there are 'one-click' installers for Windows (see [Complete install packages for Windows](#)) and for OSX (see [Complete Install Packages for Mac OS X](#)) or [install on OS X](#). These are unsuitable for production servers.

If you want to avoid installing Moodle yourself completely, consider <https://moodle.com/moodlecloud/>

Requirements

Moodle is primarily developed in Linux using [Apache](#), [PostgreSQL/MySQL/MariaDB](#) and [PHP](#) (sometimes known as the LAMP platform). Typically this is also how Moodle is run, although there are other options as long as the software requirements of the [release](#) are met.

If you are installing Moodle in a Windows server, note that from php5.5 onwards, you will also need to have the Visual C++ Redistributable for Visual Studio 2012 installed from: <http://www.microsoft.com/en-us/download/details.aspx?id=30679> Visual C++] (x86 or x64)

The basic requirements for Moodle are as follows:

Hardware

- Disk space: 200MB for the Moodle code, plus as much as you need to store content. 5GB is probably a realistic minimum.
- Processor: 1GHz (min), 2GHz dual core or more recommended.
- Memory: 512MB (min), 1GB or more is recommended. 8GB plus is likely on a large production server
- Consider separate servers for the web "front ends" and the database. It is much easier to "tune"

All the above requirements will vary depending on specific hardware and software combinations as well as the type of use and load; busy sites may well require additional resources. Further guidance can be found under [performance recommendations](#). Moodle scales easily by increasing hardware.

For very large sites, you are much better starting with a small pilot and gaining some experience and insight. A "what hardware do I need for 50,000 user?" style post in the forums is highly unlikely to get a useful answer.

Software

See the [release notes](#) in the dev docs for software requirements.

Set up your server

Depending on the use case a Moodle server may be anything from a Desktop PC (e.g. for testing and evaluating) to a rackmounted or clustered solution to cloud VMs or other hosted solutions. As mentioned above there are lots of possibilities for installing the basic server software, some links and pointers are at [Installing AMP, IIS, Nginx](#).

It will help hugely, regardless of your deployment choices, if time is taken to understand how to configure the different parts of your software stack (HTTP daemon, database, PHP etc). Do not expect the standard server configuration to be optimal for Moodle. For example, the web server and database servers will almost certainly require tuning to get the best out of Moodle.

If a hosting provider is being used ensure that all Moodle requirements (such as PHP version) are met by the hosting platform before attempting the installation. It will help to become familiar with changing settings within the hosting provider's platform (e.g. PHP file upload maximums) as the options and tools provided vary.

Download and copy files into place

IMPORTANT: While there are now a number of places you can get the Moodle code (including host provided Moodle installers), you are strongly advised to only obtain Moodle from moodle.org. If you run into problems it will be a great deal easier to support you.

You have two options:

- Download your required version from <http://moodle.org/downloads> and unzip/unpack...
- **OR** Pull the code from the Git repository (recommended for developers and also makes upgrading very simple):

```
$ git clone -b MOODLE_{Version3}_STABLE git://git.moodle.org/moodle.git
```

For a fuller discussion see [Git for Administrators](#).

Either of the above should result in a directory called **moodle**, containing a number of files and folders.

You can typically place the whole folder in your web server documents directory, in which case the site will be located at **<http://yourwebserver.com/moodle>**, or you can copy all the contents straight into the main web server documents directory, in which case the site will be simply **<http://yourwebserver.com>**. See the documentation for your system and/or web server if you are unsure.

Tip: If you are downloading Moodle to your local computer and then uploading it to your hosted web site, it is usually better to upload the compressed Moodle file and then decompress on your hosted web site. If you decompress Moodle on your local computer, because Moodle is comprised of over 25,000 files, trying to upload over 25,000 files using an FTP client or your host's "file manager" can sometimes miss a file and cause errors.

- **Secure the Moodle files:** It is vital that the files are not writeable by the web server user. For example, on Unix/Linux (as root):

```
chown -R root /path/to/moodle
```

```
chmod -R 0755 /path/to/moodle
```

(files are owned by the administrator/superuser and are only writeable by them - readable by everyone else)

On test/dev sites you *may* want to make the files writeable in order to use the built-in plugin installer. This is discouraged for live sites (at least, revert to more secure settings if you do).

Create an empty database

Next create a new, empty database for your installation. You need to find and make a note of following information for use during the final installation stage:

- **dbhost** - the database server hostname. Probably *localhost* if the database and web server are the same machine, otherwise the name of the database server
- **dbname** - the database name. Whatever you called it, e.g. *moodle*
- **dbuser** - the username for the database. Whatever you assigned, e.g. *moodleuser* - do not use the root/superuser account. Create a proper account with the minimum permissions needed.
- **dbpass** - the password for the above user

If your site is hosted you should find a web-based administration page for databases as part of the control panel (or ask your administrator). For everyone else or for detailed instructions, see the page for your chosen database server:

- [PostgreSQL](#)
- [MariaDB](#)
- [MySQL](#)
- [MSSQL](#)
- [Oracle](#) (not recommended)

Create the (*moodledata*) data directory

Moodle requires a directory to store all of its files (all your site's uploaded files, temporary data, cache, session data etc.). The web server needs to be able to write to this directory. On larger systems consider how much free space you are going to use when allocating this directory.

Due to the default way Moodle caches data you may have serious performance issues if you use relatively slow storage (e.g. NFS) for this directory. Read the [Performance recommendations](#) carefully and consider using (e.g.) redis or memcached for [Caching](#).

IMPORTANT: This directory must **NOT** be accessible directly via the web. This would be a serious security hole. Do not try to place it inside your web root or inside your Moodle program files directory. Moodle will not install. It can go anywhere else convenient.

Here is an example (Unix/Linux) of creating the directory and setting the permissions for **anyone** on the server to write here. This is only appropriate for Moodle servers that are not shared. Discuss this with your server administrator for better permissions that just allow the web server user to access these files.

```
# mkdir /path/to/moodldata
# chmod 0777 /path/to/moodldata
```

Securing moodldata in a web directory

If you are using a hosted site and you have no option but to place 'moodldata' in a web accessible directory. You may be able to secure it by creating an .htaccess file in the 'moodldata' directory. This does not work on all systems - see your host/administrator. Create a file called .htaccess containing only the following lines:

```
order deny,allow
deny from all
```

Start Moodle install

It's now time to run the installer to create the database tables and configure your new site. The recommended method is to use the command line installer. If you cannot do this for any reason (e.g. on a Windows server) the web-based installer is still available.

Command line installer

It's best to run the command line as your system's web user. You need to know what that is - see your system's documentation (e.g. Ubuntu/Debian is 'www-data', Centos is 'apache')

- Example of using the command-line (as root - substitute 'www-data' for your web user):

```
# chown www-data /path/to/moodle
# cd /path/to/moodle/admin/cli
# sudo -u www-data /usr/bin/php install.php
# chown -R root /path/to/moodle
```

The chowns allow the script to write a new config.php file. More information about the options can be found using

```
# php install.php --help
```

You will be asked for other settings that have not been discussed on this page - if unsure just accept the defaults. For a full discussion see [Administration via command line](#)

Web based installer

For ease of use you can install Moodle via the web. We recommend configuring your web server so that the page is not publicly accessible until the installation is complete.

To run the web installer script, just go to your Moodle's main URL using a web browser.

The installation process will take you through a number of pages. You should be asked to confirm the copyright, see the database tables being created, supply administrator account details and supply the site details. The database creation can take some time - please be patient. You should eventually end up at the Moodle front page with an invitation to create a new course.

It is very likely that you will be asked to download the new config.php file and upload it to your Moodle installation - just follow the on-screen instructions.

Final configuration

Settings within Moodle

There are a number of options within the Moodle Site Administration screens (accessible from the 'Site administration' tab in the 'Administration' block (Classic theme) or the Site administration button in the navigation bar (Boost). Here are a few of the more important ones that you will probably want to check:

- Administration > Site administration > Server > Email > Outgoing mail configuration: Set your smtp server and authentication if required (so your Moodle site can send emails). You can also set a noreply email on this page.
- Administration > Site administration > Server > Server > Support contact. Set your support contact email.
- Administration > Site administration > Server > System paths: Set the paths to du, dot and aspell binaries.
- Administration > Site administration > Server > HTTP: If you are behind a firewall you may need to set your proxy credentials in the 'Web proxy' section.
- Administration > Site administration > Location > Update timezones: Run this to make sure your timezone information is up to date. (more info Location)
 - Set _____ server's _____ local _____ timezone inside php.ini (should _____ probably _____ be inside /etc/php.ini or /etc/php.d/date.ini, depending on the underlying OS):

[Date]

; Defines the default timezone used by the date functions
date.timezone = "YOUR LOCAL TIMEZONE"

Remaining tasks

- **Configure Cron:** Moodle's background tasks (e.g. sending out forum emails and performing course backups) are performed by a script which you can set to execute at specific times of the day. This is known as a cron script. Please refer to the Cron instructions.
- **Set up backups:** See Site backup and Automated course backup.
- **Secure your Moodle site:** Read the Security recommendations.
- **Increasing the maximum upload size** See Installation FAQ Maximum upload file size - how to change it?
- **Check mail works :** From Site administration > Server > Test outgoing mail configuration, use the link to send yourself a test email. Don't be tempted to skip this step.

Installation is complete :)

- Create a new course: You can now access Moodle through your web browser (using the same URL as you set during the install process), log in as your admin user and create a new course. See [create a new course](#).

If something goes wrong...

Here are some things you should try...

- Check the [Installation FAQ](#)
- Check your file permissions carefully. Can your web server read (but not write) the Moodle program files? Can your web server read and write your Moodle data directory? If you don't fully understand how file ownership and permissions work on your operating system it would be time very well spent to find out.
- Check your database permissions. Have you set up your database user with the correct rights and permissions for your configuration (especially if the web server and database server are different machines)?
- Create your [Configuration file](#) (config.php) by hand. Copy config-dist.php (in the root of the Moodle program directory) to config.php, edit it and set your database/site options there. Installation will continue from the right place.
- Once you have a config.php (see previous tip) you can edit it to turn on debugging (in section 8). This may give you extra information to help track down a problem. If you have access, check your web server error log(s).
- Re-check your php.ini / .htaccess settings. Are they appropriate (e.g. memory_limit), did you edit the correct php.ini / .htaccess file and (if required) did you re-start the web server after making changes?
- Did you include any non-core (optional) plugins, themes or other code before starting the installation script? If so, remove it and try again (it may be broken or incompatible).
- Explain your problem in the [Installation problems forum](#). **PLEASE** list your software versions; explain what you did, what happened and what error messages you saw (if any); explain what you tried. There is no such thing as 'nothing', even a blank page is something!

Platform specific instructions

Note: Much of this information is provided by the community. It may not have been checked and may be out of date. Please read in conjunction with the above installation instructions.

- [Windows installation](#)
 - [Installing Moodle on SmarterASP.NET](#)
- [Unix or Linux Installation](#)
- [Mac Installation](#)
- [Amazon EC2 Cloud Services Installation](#)

Installing plugins

There are optional plugins or add-ons available that can extend Moodle in all sorts of ways - new activities, new quiz question types, new reports, integrations with other systems and many more. This page gives some important advice you should consider before adding them and general advice on how to install them.

What is meant by the word 'plugin'?

Moodle = core code + plugins

Plugins may be either standard plugins (included in the Moodle download package) or add-ons.

Before you do anything else...

VERY IMPORTANT Warning: Please be aware that some add-ons have not been reviewed, and the quality and/or suitability for your Moodle site has not been checked. Please be careful when considering using one. It may not do what you expect, it may have serious security issues or it may even not work at all.

- Do you **really** need the add-on? More functionality means more things to support, more things to (potentially) go wrong and more things to worry about at upgrade time.
- Is the add-on supported and maintained? If something goes wrong can you get support? Will bugs be fixed?
- If the add-on does not work in a future version of Moodle, what will you do about it?
- Beware of *patches*! If an add-on modifies or replaces core files then be very careful. It can only be guaranteed to work with the exact build (version) of Moodle it was created for and is highly unlikely to survive a Moodle upgrade.

Moodle requirements

- You must have access to your Moodle code files to upload or install the add-on. There is no way to install add-ons through the web interface.
- Your add-on may impose additional requirements on the server. In particular, add-ons to access other systems or services may need specific PHP plugins (e.g. LDAP or other database drivers). Check the documentation.

You should always look for a 'readme' file or other documentation supplied with the add-on and read that carefully.

Downloads

Add-ons may be downloaded from the [Moodle plugins directory](#).

Note: Make sure you select the correct version. Most add-ons for Moodle 2.0+ will work in all 2.x versions of Moodle. Question Types changed in Moodle 2.1 and require specific plugins for Moodle 2.1+. Plugins (and themes) for Moodle 1.9 and earlier will NOT work. There is no point trying!

Installation

First, establish the correct place in the moodle code 'tree' for the plugin. Some plugins need elements installed in multiple locations. To complicate matters not everybody packages up plugins in the same way. It's best to unzip/unpack the plugin in an empty directory first just to see what it looks like and to check for 'readme' files. Common locations are (there are others):

- /path/to/moodle/mod/ - modules (i.e. activities and resources)
- /path/to/moodle/blocks/ - blocks (in the sidebars)
- /path/to/moodle/question/type/ - questiontypes

- /path/to/moodle/course/format/ - course formats
- /path/to/moodle/admin/report/ - admin reports
- /path/to/moodle/theme/ - themes

Then:

1. Download the zip file
2. Upload or copy it to your Moodle server
3. Unzip it in the right place for the plugin type (or follow add-on instructions).
4. In your Moodle site (as admin) go to *Settings > Site administration > Notifications* (you should, for most plugin types, get a message saying the add-on is installed)
5. Test the add-on

Note: The add-on may contain language files. They'll be found by your Moodle automatically. These language strings can be customized using the standard *Settings > Site administration > Language* editing interface.

If you have problems...

- Check the file permissions. The web server needs to be able to read the plugin files. If the the rest of Moodle works then try to make the plugin permissions and ownership match.
- Did you **definitely** unzip or install the add-on in the correct place?
- Because Moodle scans plugin folders for new plugins you cannot have any other files or folders there. Make sure you deleted the zip file and don't try to rename (for example) an old version of the plugin to some other name - it will break.
- Make sure the directory name for the plugin is correct. All the names **have** to match. If you change the name then it won't work.
- Turn on Debugging - any error messages?
- Ask in the appropriate forum in Using Moodle. Make sure you describe your system (including versions of MySQL, PHP etc.), what you tried and what happened. Copy and paste error messages exactly. Provide the link to the version of the add-on you downloaded (some have very similar names).

Experiment – 8

Aim: - Create a Moodle site and Database Schema

Create Moodle site database

Moodle supports [MySQL](#), [PostgreSQL](#) and MS SQL 2005 and Oracle databases. Other databases may also work with Moodle.

If your Moodle installer, does not create or add to an existing MySQL database, these instructions may help you.

MySQL

phpMyAdmin

If you have access to phpMyAdmin you can use this program to create an empty database which the Moodle Installer will then populate with its tables.

Tip: Be sure that it includes one of the UTF standard formats for Moodle 2.x.

See the [create the MySQL database](#) section in the XAMPP installation page.

MySQL client

In Linux, here are some command line prompts using a MySQL Client program (commands which you type-in are shown in bold):

- Start the MySQL Client program:

```
#mysql -u root -p
```

Enter password: **MartinMoodle**

```
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 2 to server version: 5.0.22-log  
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.  
mysql>
```

Note: The prompt changes to "mysql>" to indicate that you are now working in the MySQL Client program. When working in MySQL, all commands which you type-in must end in a semi-colon. (If you hit the Enter key without the final semi-colon, you'll get the line continuation symbol '->'; this is your second chance to type the semi-colon and hit Enter.)

- Begin by checking for any existing databases called "moodle" - if there are any you should change the name in all the commands which follow:

```
mysql> SHOW DATABASES;
```

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| test               |
+-----+
3 rows in set (0.03 sec)
```

- Create a database to store the Moodle tables. We'll call this "moodle", as there are none with that name already in the above list, but change it if you need to.

```
mysql> CREATE DATABASE moodle;
```

Query OK, 1 row affected (0.00 sec)

- Change the default character set and collation of the "moodle" database to UTF8. Leave this out if you are installing Moodle 1.5 or earlier):

```
mysql> ALTER DATABASE moodle DEFAULT CHARACTER SET utf8 COLLATE
utf8_unicode_ci;
```

Query OK, 1 row affected (0.00 sec)

- Create a username and password to access the database "moodle" and grant database access permissions. We'll call the user "moodleuser" and set the password as "yourpassword". It's a good idea to change these for your installation however most people keep the username as "moodleuser". Remember the username and password you have set, as you'll need it in the configuration screens later. This is a long command so has been split over several lines by pressing the Return key.

```
mysql> GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,CREATE TEMPORARY
TABLES,DROP,INDEX,ALTER ON moodle.* TO moodleuser@localhost IDENTIFIED BY
'yourpassword';
Query OK, 0 rows affected (0.01 sec)
```

Security Warnings: Never leave the password as the one shown here. Make sure you have a strong password (a mixture of letters and numbers, upper and lower case). Avoid granting "ALL" permissions on the database.

- Exit the MySQL Client program:

```
mysql> QUIT
```

Alternative

If you want to run all this at once without so many commands then copy and paste the lines below into the mysql command line.

```
create database moodle default character set utf8;  
grant all privileges on moodle.* to 'moodleuser'@'localhost' identified by 'yourpassword';  
flush privileges;  
quit
```

Note: 'Grant all' gives a lot of privilege to the moodleuser account.

PostgreSQL

And some example command lines for those who wish to use a PostgreSQL database:

```
# su - postgres  
> psql -c "create user moodleuser createdb;" template1  
> psql -c "alter user moodleuser with encrypted password 'yourpassword';" template1  
> psql -c "create database moodle with encoding 'unicode';" -U moodleuser template1  
> psql -c "alter user moodleuser ncreatedb;" template1  
> su - root  
# /etc/init.d/postgresql reload
```

If the Postgres create user command above (>psql -c "create user moodleuser...") gives an error message you may want to try initdb as postgres, then restarting postgresql as root, then start steps again from the beginning:

```
initdb -D data  
su  
# /etc/init.d/postgresql restart
```

If the Postgres create database command above (>psql -c "create database moodle...") gives an error message you may want to try:

```
psql -c "create database moodle with template=template1 encoding = 'unicode' owner = moodleuser  
location = '/var/mydata';"
```

If the create database command asks you for a password, run the line containing 'encrypted password' first before proceeding.

Using a SQLite database

SQLite is a software library that implements a self-contained, serverless, zero-configuration, transactional SQL database engine.

Moodle 2.0 offers experimental support for SQLite3 database installations. In this case, no database setup is required. The database file will be created by the installation script. By default, the database file will be stored in Moodle's data directory (see [Creating the data directory](#)). During install, the web server must have write access on the directory where the database file will be stored. After installation, the web server must have read-write access to the database file.

Continue with [Creating the data directory](#)

Using a hosted server

If you are using a webhost, they will probably have a control panel web interface for you to create your database.

The **cPanel** system is one of the most popular of these. To create a database using cPanel:

1. Click on the **MySQL Databases** icon.
2. Type **moodle** in the New Database field and click **Create Database**.
3. Type a *username* and *password* (not one you use elsewhere) in the respective fields and click **Create** **User**.
Note that the *username* and *database* names may be prefixed by your cPanel account name and an underscore, and truncated to 16 characters. When entering this information into the Moodle installer - use the full names.
4. Now use the **Add Users to Databases** button and give this new user account **ALL** rights to the new database.

Continue with [Creating the data directory](#)

Caution- be aware of Moodle version requirements

See the [\[download page\]](#) at Moodle.org or [PHP settings by Moodle version](#) for supported database requirements for each version of Moodle.

For example Moodle 2.0 requires:

- * MySQL 5.0.25 or later (InnoDB storage engine highly recommended)
- * PostgreSQL 8.3 or later
- * Oracle 10.2 or later
- * MS SQL 2005 or later

Experiment – 9

Aim: - Design Site appearance, Front page, Front page settings, My Moodle, User profiles, Navigation, Course list, Themes, Theme settings, Header and footer, Language settings, Using web services, Publishing a course, Blogs, RSS feeds

Site appearance

There are many ways to customise the appearance of your Moodle site so that it blends in with, for example, your college site or your company's corporate brand. The links below provide more information on personalising the appearance of Moodle:

- [Front page](#) - how best to display the entry page to your Moodle. Here you can decide how to display announcements and courses (if you want to) and how to add attractive images and descriptive text to promote your organisation.
- [Theme settings](#) - how to add custom dropdown menus and (if you wish) personalise the [user menu](#) top right.
- [Header and footer](#) - how to display useful information at the top and bottom of your course and link to Google Analytics.
- [Themes](#) - general information on changing the "skin" of your Moodle for the whole site or just sections. Look at [Standard themes](#) for examples of how to easily customise Moodle's 'More' theme.
- [Dashboard](#) - a personalised page for each user. Find out how to set the default dashboard and how to allow users to select their own home page.
- [User profiles](#)
- [Navigation](#) - control how users find their way around Moodle
- [Course list](#) - control who appears in the list of courses

Front page

The front page is the initial page seen by someone reaching a Moodle site after or before a login. Typically a student will see [courses](#), some [blocks](#) of information, displayed in a [theme](#). In the [Navigation bar](#) and [Navigation block](#) it is called "Home".

A combination of [site policies](#), [user authentication](#) and [front page settings](#) determine who can get to the front page. And once they get there [what they can see](#) and [what they can do](#).

In many ways, a Front Page has similar features and functions to those in a [Course](#). For example, [Activities](#), [Resources](#) and [Blocks](#) can be added to the Front Page to give it different looks.

Front page settings

The site's Front page is similar to a course page. The front page settings can be found in 'Front page settings' in the Site administration.

Full site name

This name appears at the top of every page above the navigation bar.

Short name for site

The short name appears at the beginning of the navigation bar as a link back to your site front page.

Front page summary

This summary can be displayed on the left or right of the front page using the course/site summary block.

The summary is also used as the HTML metadata description in some themes, for the front page of the site. This is not generally seen by users, but can be useful for search engines that index the page.

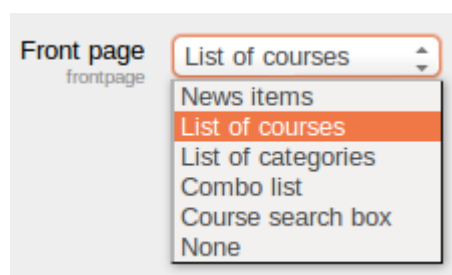
Front page and Front page items when logged in

The centre of the front page can display any combination of the following: news items, a list of courses, a list of enrolled courses, a list of course categories, a list of categories and courses, a course search box or none. The order is determined by a combination box.

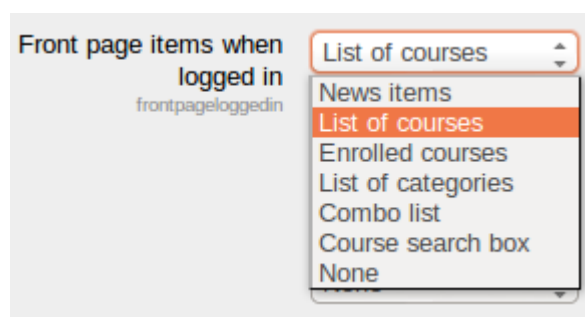
It is possible to create one setting for anyone who gets to the front page without logging in to the Moodle site, and another look for only those who have logged into the Moodle site. There are two combination boxes.

Combination Boxes

The image below shows the first Front Page settings dialog - what visitors will see before they log in.



The image below shows a second front Page settings dialog - what Authenticated Users will see when logged in.



An alternative option for logged-in users is their dashboard (can be set in Site administration > Appearance > Navigation)

Maximum category depth

This specifies the maximum depth of child categories expanded when displaying categories or combo list. Deeper level categories will appear as links and user can expand them with AJAX request.

Maximum number of courses

Maximum number of courses to be displayed on the site's front page in course listings.

Include a topic section

This adds a topic section to the centre-top of the front page. When editing is turned on, resources and/or activities can be added to the topic section using the dropdown menus, in the same way as on a course page.

Note: The label resource can be used to add text and/or an image to the centre-top of the front page.

News items to show

This setting only applies if the front page is set to display news items or if you are using the [Latest announcements block](#).

Comments per page

Each course may include a [Comments block](#) which allows the display of user comments. The number of comments displayed at any time is set here.

Default front page role

The default front page role enables logged-in users to participate in front page activities. See 'How do I enable logged-in users to participate in front page activities?' in [Front page FAQ](#) for more details. It is recommended that it is set to 'Authenticated user on frontpage'. It can also be set to Student. It should not be set to Guest.

Front Page roles

As in a course, you can assign users [roles](#) just in the [context](#) of this page. Note this is different from the default Authenticated User on Front Page role found in front page settings. As a security issue, usually only admins can modify the front page, other users do not.

It is possible to upload users as students in the front page with a csv file using the site short name as 'course1' field, student in 'role1' field (and, if needed, chosen group name in 'group1' field.)

Front Page backup

You can backup the front page, similar to a [Course backup](#).

Front Page restore

As in a course, you can [Restore](#) a backed up version of the front page.

Front Page questions

The Question bank is accessible from the Front Page. For example, if "topic" is checked in the front page settings, you can add a Quiz module activity.

Tips and Tricks

A Moodle site's front page can be reconfigured from the default standard to give it a different look and or change how it functions. Depending upon the changes, this can be a simple or a more complex process. See Front page FAQ for some ideas on customising your front page.

Visual examples

3 column Front Page, topic checked



2 column Front Page, blocks right



2 column Front Page, blocks left



3 column Front Page, Categories and courses



2 column Front Page, Categories only



Clean look, topic checked, no blocks



Central area, Courses or Categories?

This is a major decision, what is it you want in the centre of your front page? You can have a list of categories, a list of categories and courses, or neither, a label with a table full of images works just as well. The issue is always the same, actually, what looks good.

Clean look, topic checked, no blocks

The standard Moodle page format of Blocks on the left and blocks on the right is not always appropriate or considered pretty. The site administrator can change this basic format by simply deleting or hiding blocks that can be seen by users. Navigation can be placed on the dock.

The two column format

The Administrator decides which blocks should appear on the front page and moves all of them to the left or right side. This format allows you to decide what you want in the centre, and where you want the blocks, left or right.

Tip: It is possible to force a block column to assume a specific width. For example, placing an image of 400 px in a block will force that column to 400 px.

The clean look example

Some sites want an uncluttered look. A site administrator or designer wants to start with the most basic look and then add features.

1. The site administrator will turn on editing and hide or delete all blocks that can be seen by teachers or students on the front page.
2. Then go to the Front Page settings and:
 1. Set the "Front page" list to None, None, None, None
 2. Set "Front page items when logged in" list to the same
 3. Check the box for "Include Topic section"
 4. Save

All the above was done with 1 theme, and a simple front page.

If you're using a theme such as Clean or More you can then do some cool things with Bootstrap layout elements, as described in the thread [LOOK & LEARN: How to add marketing spots as a front page topic](#).

Block settings

Each block has a number of configuration settings that you can change. See [Managing blocks](#) for more details.



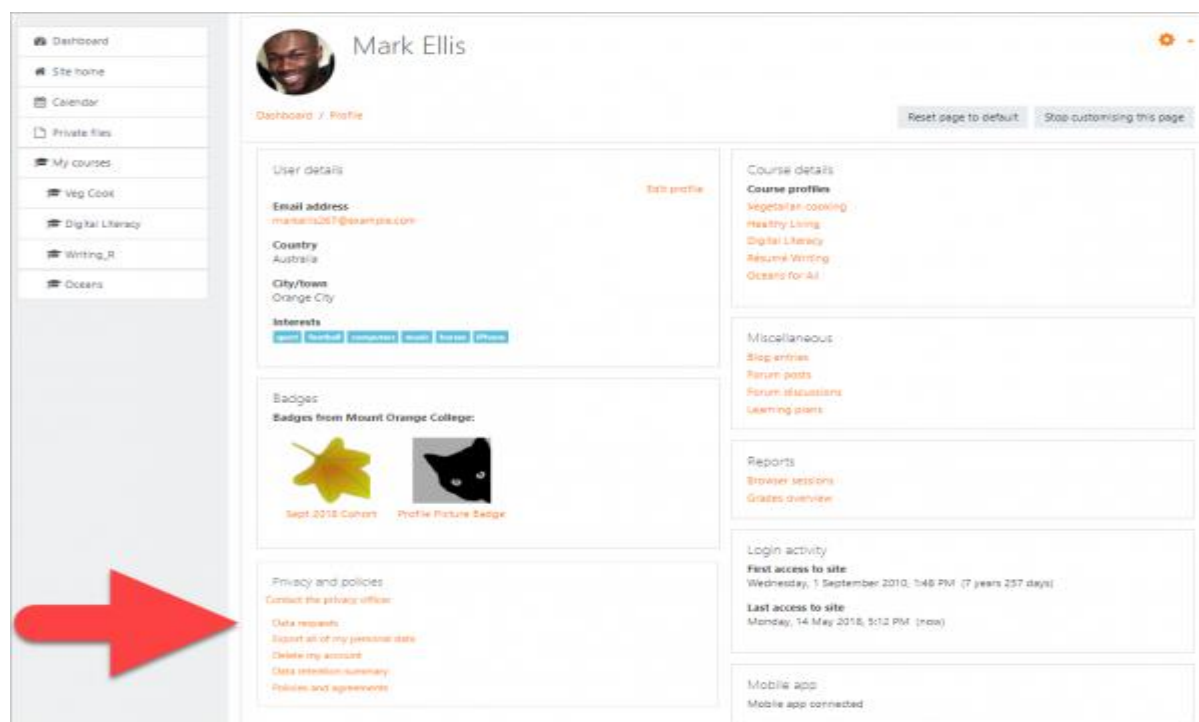
Other settings that change the look of the Main Page

Some components of the Main Page, such as the logo, heading and Navigation bar may be changed by an administrator in Themes in the Site administration and clicking on the theme name. Different themes may have fewer or more options to set. There are some other settings available in 'Theme settings'.

User profiles

Every user has a profile page which may be reached from the user menu top right and then clicking Profile. This page contains links to further pages allowing the user to [edit their profile information](#) and preferences, view their forum/blog posts, and check any reports they have access to.

The Privacy and policies section provides links for contacting the Privacy officer, requesting data export and account deletion (new in 3.6), along with a data retention summary page (new in 3.6) (unless disabled by the administrator).



If mobile access is enabled, a link to download the [Moodle Mobile app](#) is visible on the profile. (See screenshot above.)

Viewing others's profiles

Users with permission to view the profiles of other users can view them by clicking on their name. If they click on the name of a user within a course, the course profile will be displayed and the full profile may be viewed (if allowed) by clicking the link "Full profile" in the Miscellaneous section:

Note: All users are allowed to view the full profile of users listed as course contacts in the course description.

See [View profile](#) for more information on how the profile information is displayed and [Edit profile](#) for information on updating profiles.

Site administration settings

Site policies

An administrator can force users to login for profiles and select which roles are visible in user profiles (by default teacher, non-editing teacher and student) in *Administration > Site administration > Security > Site policies*.

Default profile page

An administrator or manager (or any other user with the capability [moodle/user:managesyspages](#)) can add additional blocks onto the default profile page for all users from *Administration > Site administration > Appearance > Default profile page*. Blocks can be added to the left, right or middle of the page.

Note that this does NOT allow you to remove the following blocks, or to change the layout/order of these blocks:

- User details
- Privacy and policies
- Course Details
- Miscellaneous
- Reports
- Login activity
- Mobile app (only displays if [Mobile web services](#) are configured)

These are all 'fake' blocks and cannot be removed through the normal blocks administration interface.

Clicking the button 'Reset profile for all users' will then apply these settings to the profile pages of everyone on the site.

Preventing users from customizing their profile page

By default, users can customize their public profile page and add blocks. To prevent this

1. Go to Administration > Site administration > Users > Permissions > Define roles
2. Edit the authenticated user role and untick the capability [moodle/user:manageownblocks](#)

User profile capabilities

System:

- [Edit own user profile](#)
- [Manage blocks on own public user profile](#)
- [Configure default page layout for public user profiles](#)
- [Update user profiles](#)

Users:

- [Edit user profile](#)
- [Manage blocks on user profile of other users](#)
- [View user full information](#)

Course:

- [View user profiles](#)

Navigation

Navigation bar

The [navigation bar](#) is the row of links you will find at the top left of your Moodle site. The navigation bar shows a user their current context path, with links to the higher contexts.

Navigation when using different themes

When using the Clean or More theme, or a custom theme based on Clean, the navigation and administration blocks provide access to other pages on the site.

When using the Boost theme, or a custom theme based on it, access is instead provided via the navigation drawer.

Navigation block

The [Navigation block](#) provide easy access to **view** various sections of the Moodle site and includes

- Dashboard- a personalised home page displaying links to the courses a user is associated with and activity information (such as unread forum posts and upcoming assignments)
- Site pages - links to site pages and resources from the front page of Moodle
- My courses - expands to show courses the user is enrolled in.

Keyboard support for navigation

Keyboard/Screenreader and Voice input users can access navigation by using the Tab, Space, Enter and right/left arrow keys.

Administration block

The Administration block provides you with easy access to change various settings of a Moodle site. This block shows contextual settings (the settings for a forum when you are viewing it as an editing user) as well as settings for anything else you have permissions for.

Site administration settings

An administrator can change navigation settings, such as the default home page for users, and whether to show course categories in the navigation, in *Administration > Site administration > Appearance > Navigation*.

If *Show course full names* is enabled, courses in the Navigation block will be shown with their full name rather than their short name.

If *navshowcategories* is enabled, but there is only one category on the site, it will not display in the navigation bar.

The setting *Always link course sections* can be enabled to link the course sections. When a section name is clicked either in the navigation block or the central course content area, it will go straight to the items in that section. When using the Boost theme, 'Always link course sections' must be enabled for course section links to display in the navigation drawer.

Note for administrators

If you are not *directly* associated with a course (that is, if you don't have a teacher, student or other role actually defined in the course) it will not appear in your Navigation menu. Also, if you are not directly associated with at least one course in a category, the category will not appear in your Navigation menu.

Course list

Order of courses listed

The order in which courses are listed on the front page can be set by an administrator in *Administration > Site administration > Courses > Manage courses and categories*

Course contacts

An administrator can change who is displayed as a course contact in *Administration > Site administration > Appearance > Courses*.

Users with the selected role(s) e.g. teachers are listed in the course description.

Available courses

Moodle Features Demo

Teacher: Terri Teacher

This course outlines Moodle's features by providing examples of activities and resources.



Course Fullname 101

Teacher: Terri Teacher

This course is in weekly format.



By default, users with the role of teacher in a course are listed in the course description for that course. If a course has several teachers, the course description can become rather long. In this case, you can create a role e.g. head of subject with no capabilities set and assign it to selected users in addition to their teacher role. If you then select only the head of subject role in the course managers setting, the course description will be shorter. Note: Only users enrolled in the course are displayed as course contacts.

If Self enrolment is used in a course, the welcome message email may be sent from a user with the role of course contact. If there is more than one user with the role, then the welcome message is sent from the first user to be assigned the role of course contact.

Display extended course names

By default, only course full names are displayed in the list of courses. If you check this box, then the course short name will also be displayed. Enabling this will also allow you to edit the wording using Language customization. See the forum discussion Course Extended Name Display Language Modification for further details.

Courses per page

Here you can select how many courses you wish to be displayed per page in the course listing

Courses with summaries limit

When courses are listed, they show the full course summary. Displaying the full summary is very useful, but can make pages very long. If you have a great number of courses, it might be better to display the clickable icon taking you to more information (ie, the full summary) Here you can decide the number of courses that will display the full summary before Moodle reverts to the "more information" icon.

Course summary files limit and Course summary files extensions

Allows course managers to add files to the course. They will be accessible by anyone from outside of the course just like course name and/or summary. Only users with moodle/course:changesummary capability are able to upload/change course overview files. By default it is configured that course managers can add up to one image file to the course. The theme determines how those files are displayed in course lists.

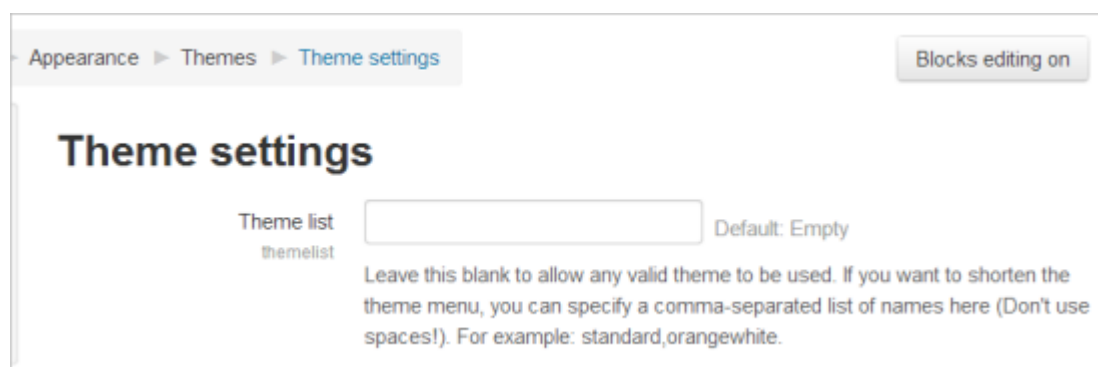
Themes

Themes provide a "skin" to completely change the look and feel of your site (or even an individual course).

- [Theme settings](#)
- [Standard themes](#) including information on changing your theme
- [Installing a new theme](#)

Theme settings

An administrator can change theme settings in Administration > Site administration > Appearance > Themes > Theme settings.



Theme list

This lists the themes available for course and user themes. Leave this blank to allow any valid theme to be used. If you want to shorten the theme menu, you may specify a comma-separated list of names, though don't use spaces (e.g. standard,orangewhite).

You can preview the available themes in *Administration > Site Administration > Appearance > Themes > Themes selector*.

Theme designer mode

Turn this on if you are designing and testing themes as it will stop the themes being cached and enable you to see theme changes quickly. (You can also do this with the Clear theme cache button on the theme selector page.)

Allow user themes

If the option *allowuserthemes* is enabled, each user may select their preferred theme on the edit profile page. All Moodle pages will be displayed in the user's theme, apart from courses where a course theme has been set.

NOTE: the user's theme will not be available in mobile and tablet devices unless the option *enableddevicedetection* is unchecked.

Allow course themes

If you enable this, then teachers will be able to set their own course themes. Course themes override all other theme choices (site, user, or session themes).

- If the option *allowcoursethemes* is enabled, each editing teacher may select their course theme via the Force theme option on the [Course settings](#) page. The course will always be displayed in the theme specified in the course setting, with user and the site themes being overwritten.

NOTE: the course theme will not be available in mobile and tablet devices unless the option *enableddevicedetection* is unchecked.

Allow category themes

When enabled, themes can be set at the category level. This will affect all child categories and courses unless they have specifically set their own theme. **WARNING:** Enabling category themes may affect performance, as it will result in a few extra DB queries on each page, so only turn this on if you need it!

NOTE: the category theme will not be available in mobile and tablet devices unless the option *enableddevicedetection* is unchecked.

Allow cohort themes

When enabled, cohort themes may be specified when editing or creating a cohort. See [Cohort sync](#).

Allow theme changes in the URL

- If this is checked then the theme may be changed by adding

```
?theme=theme_name
```

(or

```
&theme=theme_name
```

if there are other URL parameters) to the URL in the browser.

Theme names used as the URL parameter should be the shortname of the theme, which you can see in *Site administration > Plugins > Plugins overview* and scroll down to the Themes section near the bottom. Remove the "theme_" prefix to find the proper shortname. Theme shortnames are usually the same as the main theme name, eg. Clean is 'clean', so your URL parameter for it will be:

```
?theme=clean
```

Shortnames are always lower case. Themes with multiple words are usually the same but spaces are replaced by underscores, for example the theme "Formal white" should be entered as

```
?theme=formal_white
```

Clean theme_clean	Standard	2013110500
Elegance theme_elegance	Add-on	2014060200
Essential theme_essential	Add-on	2013120200
Formal white theme_formal_white	Standard	2013110500



Tip: You can turn this on to aid in troubleshooting problems that may be theme related. This will allow you to switch to a standard or different theme for your own session without affecting other users or the site theme.

Allow users to hide blocks

Allows users to display and hide blocks

Allow blocks to use the docks

- If the theme allows it, then checking this will allow the user to move blocks to the side dock.

Custom menu items

Note: The Boost theme does not support submenus in the custom menu setting.

The custommenuitems setting allows you to create a drop down menu that can be displayed by themes that support it.

You are able to create the custom menu by entering custom menu items one per line into the setting. Each item is preceded by a number of hyphens (-), the number of hyphens determines the depth of the item. So items that are **NOT** preceded by a hyphen appear on the top level of the menu (always visible), items with a single hyphen appear on a drop down menu below the previous top level item, and items with two hyphens appear on a drop down menu below the previous first level item and so on - *if the theme allows submenus*.

The content of each item is constructed of up to three bits, each separated by a | (Shift + \) character. The bits are **label** | **url** | **tooltip**.

label

This is the text that will be shown within the menu item. You must specify a label for every item on the menu.

url

This is the URL that the user will be taken to if they click the menu item. This is optional, if not provided then the item will not link anywhere.

tooltip

If you provide a URL you can also choose to provide a tooltip for the link that is created with the URL. This is optional and if not set the label is used as the tooltip for the menu item. The following is an example of how you would create a custom menu:

```
Moodle community|http://moodle.org
-Moodle free support|http://moodle.org/support
-Moodle development|http://moodle.org/development
--Moodle Tracker|http://tracker.moodle.org
--Moodle Docs|https://docs.moodle.org
-Moodle News|http://moodle.org/news
Moodle company
-Moodle commercial hosting|http://moodle.com/hosting
-Moodle commercial support|http://moodle.com/support
```

Note: The custom menu does not escape characters within the label. If you want to use a special HTML character such as an ampersand you must escape it yourself within the label. e.g. use `&` instead of `&`.

Multilanguage support

You can add a language code (or a comma separated list of codes) as the 4th item of the line. The line will be then printed if and only if the user has currently selected the listed language. For example:

```
English only|http://moodle.com|English only item|en
German only|http://moodle.de|Deutsch|de,de_du,de_kids
```

Adding other attributes to the HTML

Other attributes, such as **target** can be added:

```
Moodle Homepage|http://moodle.org" target="_blank
```

The first quote closes the href attribute, allowing other attributes to be added. Do not add the closing quotation mark on the final attribute, it is added automatically by Moodle.

User menu items

The `customusermenuitems` setting allows you to add or remove items from the user menu. Three items, Dashboard, Profile and Log out are automatically created and can not be changed. Other user menu items can be set in this area by removing them, reordering them, or adding custom ones. There are three default items:

```
grades,grades/grade/report/mygrades.php|grades
messages,message/message/index.php|message
preferences,moodle/user/preferences.php|preferences
```

The format for each is the | "pipe" character delineating three parts:

1. a string in "langstringname, componentname" form or as plain text
2. a path to a file to run
3. an icon either as a pix icon or as a path to a file URL

Notes:

1) You can refer to a language string in the users language pack by its string name. For example, in the third line above for the default settings, "preferences" is the name of the language string, and "moodle" is the language file it is located in. See Language customisation for more information. If you put a text here instead of a reference to a language string, that text will always show up and not be language specific.

2) The examples are to php files inside the Moodle site, so `"/message/index.php"` is the path on the current Moodle instance. You could also put a path or URL here to files outside of Moodle as well.

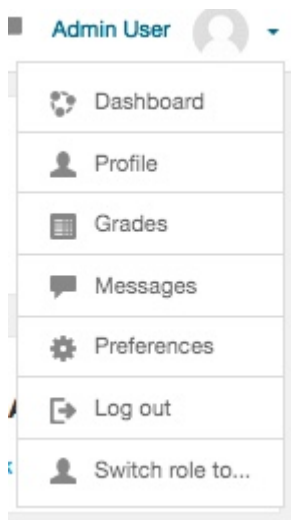
3) The example defaults of icons point to files in the `/pix/t/` directory on the Moodle server. You can use any of the pre-existing icon names there (without a file extension). You can add your images to that directory to expand the icon list, or you can prefix the icon name with `../` if you want to use icons from other pix folders. Alternatively, you can use a URL to point to a file elsewhere. "Icon" files are in both png and svg format.

Tip: If you do not specify an icon, the Moodle will supply the gear icon that is used by "Preferences" next to the entry.

Custom item example:

```
Help Center/course/view.php?id=2|check
```

This will add item called "Help Center" which links to a course on the same site with `id=2` which has been set up as a help center for all users, and it will show using the standard check mark icon.



Enable device detection

If a tablet or mobile theme is set, AND the user is using a tablet or mobile, theme associated with the device is used, always overriding the course, category or user theme.

If no tablet or mobile theme is set, course, category or user themes will be used.

Device detection regular expressions

This will allow you to customize the theme selector options. For example, you can add a custom theme for IE6 and another for Windows CE by entering the identifying expressions the browser sends and the "Return value" you want to display as the theme selectors "Device type".

Special themes

Session themes

Moodle offers an additional way to set a theme - the **session theme**. This is set with the URL and lasts until you log out. When you next login, the site/course/user themes are active again. This option is great for theme testing and works perfectly when you want to enable different themes for different situations.

For example you can offer a special link for PDA users and integrate the session theme *orangewhitepda* in that link. Nobody needs to change any settings, you just click on that link. The session theme is called by the URL parameter

```
&theme=orangewhitepda
```

The whole URL without the session theme could look like

```
http://mymoodlesite.org/course/view.php?id=18
```

and with the parameter for the PDA theme like

```
"http://mymoodlesite.org/course/view.php?id=18&theme=orangewhitepda"
```

In a standard Moodle installation, session themes are not active. To activate them the administrator must add the parameter

```
$CFG->allowthemechangeonurl = true;
```

to the Moodle *config.php* file in the Moodle base directory.

Page theme

A page theme is for special page-only themes set by code, use

```
$PAGE->force_theme()
```

Theme hierarchy

Here is the usual order that themes are considered by the Moodle interface.

Theme type	Overwrites	Display	Setting type
Site	-	all pages*	saved in theme profile
User	Site theme	all pages*	saved in user profile
Course	Site, user and session themes	one course	saved in course profile
Session	Site and user themes	all pages*	temporary until logout

(* except courses with the course theme set)

Change default hierarchy

The priority of themes can be set via the */moodle/config.php* file. The order defines which theme wins when there are several set at different levels. You can set a variable called *\$CFG->themeorder* (see *config-dist.php* for more details). By default is set in the file */lib/pagelib.php* to:

```
$CFG->themeorder = array('course', 'category', 'session', 'user', 'cohort', 'site');
```

Particular theme settings

Each theme in Moodle will have its individual menu and setting area in *Site administration > Appearance > Themes > Name of theme*. Themes can vary greatly in how they organize their own settings, so menus here may look very different from theme to theme.

Each theme should have all its settings in this location. Note, however, that some themes, especially child themes of Boost, may also use some common settings like the site Logos, set in *Site administration > Appearance > Logos* instead of having their own logo settings.

Header and footer

You can easily add links and other information to the top and bottom, and header and footer of your Moodle site.

Customising the header

- The administrator can add links, metatags, CSS or Java Script to the header section of your page via *Site administration > Appearance > Additional HTML*
- You can specify HTML or CSS or JavaScript that you want added to every page without altering the Moodle code files.
- Add your HTML/CSS to the 'Within HEAD' box

Within HEAD additionalhtmlhead	<pre><p> For support please email the Helpdesk </p></pre>
-----------------------------------	---



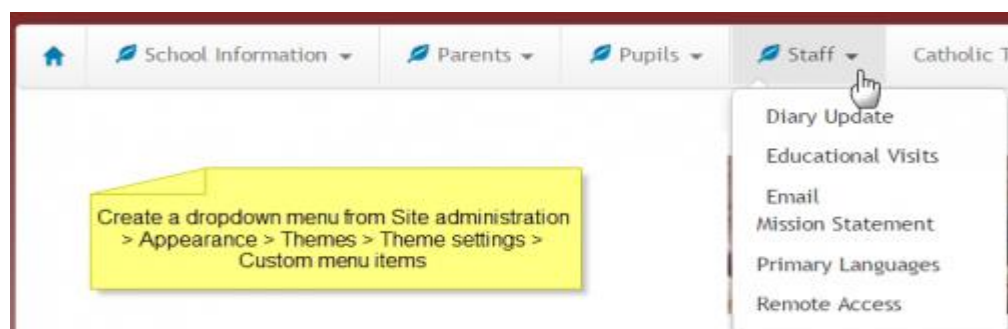
For support please [email the Helpdesk](#)



Charlemont High School

Dropdown menus

Although not strictly in the 'header' section, custom dropdown menus may be added to the top of your site via *Site administration > Appearance > Themes > Theme settings* and adding your links to the Custom menu box, using the examples given to help you.

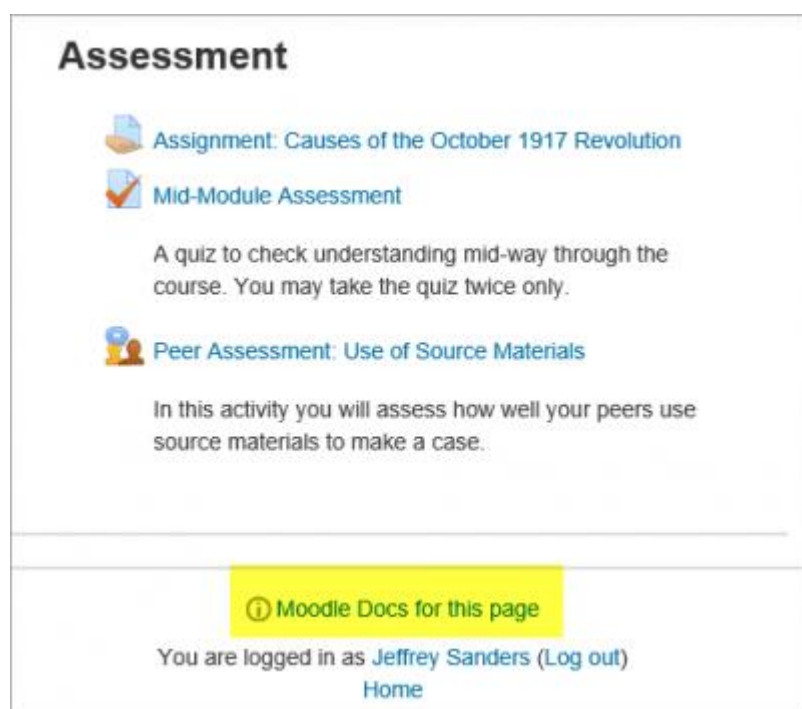


Customising the footer

- As with header options in the section above, the administrator can add links, CSS or Java Script to the footer section of your page via Site administration > Appearance > Additional HTML
- This might be useful for adding Google Analytics, for example.
- Add your HTML/CSS to the 'Before BODY is closed' box.

Footer link to Moodle docs

- Moodle Docs is an abbreviation for Moodle documentation.
- By default, managers, teachers and non-editing teachers (and any other users with the capability `moodle/site:doclinks`) have "Moodle Docs for this page" links at the bottom of each page in Moodle for accessing context-specific documentation.



- An administrator can change Moodle Docs settings in Administration > Site administration > Appearance > Moodle Docs. The link can be made to open in a new window; the docs language may be forced, and the docs link may be removed by removing the link from the Moodle docs document root field.
- If you have your own custom documentation online or wish to link to other documentation online, you can change the link in this Moodle docs document root field.
- You can download the Moodle Docs files for several branches and languages of Moodle Docs from <http://wimski.org/docs/>. These files can be hosted in a local intranet and this may be useful for places that have strict internet policies that prevent Moodle users from reaching The Moodle Docs site.

'Get the mobile app' link

In Moodle 3.4 onwards, the link 'Get the mobile app' in the footer encourages users to make use of the Moodle Mobile app.

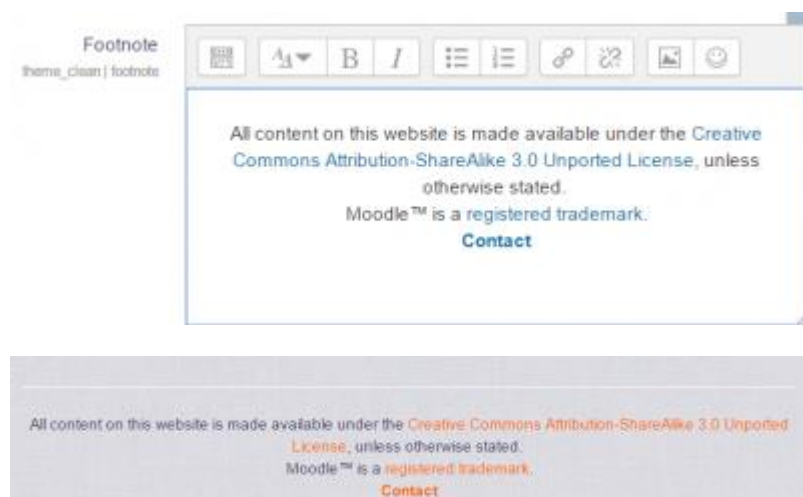
By default, the link is <https://download.moodle.org/mobile> however the setting 'App download page' (in 'Mobile appearance' in the Site administration) can be changed e.g. for sites using a branded mobile app.

To prevent the 'Get the mobile app' link being displayed in the footer, simply remove the URL from the 'App download page' setting so that the field is blank and then save changes.

If mobile services are not enabled for the site, then the 'Get the mobile app' link is not displayed.

Theme settings options

Some themes allow you to add custom CSS and footer text via their settings pages. The standard 'Clean' and 'More' themes offer this option, via Site administration > Appearance > Themes > Clean/More



Customising the 'main body' section

- As with header and footer options, you can add code and script to the main body of the page.
- This might be useful for adding a banner image, a text heading or an alert message for example.
- Add your content to the 'When BODY is opened' box.

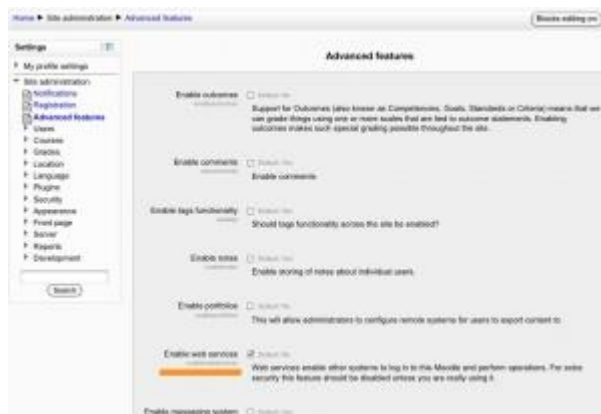
Using web services

This document explains how an administrator can set up a web service for users to access a service. Each user will have a specific and unique security key (also known as a "token") to access the service.

Enabling Web Services

1. Access Administration > Site administration > Advanced features
2. Check 'Enable web services' then click 'Save Changes'

Note: For security reasons, web services should only be enabled if you intend to make use of it.



Enabling protocols

Usually external applications that users wish to use dictate which protocols should be enabled.

1. Access *Administration > Site administration > Plugins > Web services > Manage protocols*
2. Enable the protocols (SOAP, REST, XMLRPC, AMF, ...) as required



Enabling web service function documentation

Enabling web service function documentation (also on the Manage protocols page) results in user-specific web service documentation being available for each user on their Security keys page. This option is mainly useful to web service client developers. If nobody is creating a web service client, there is no need to enable this feature.



Creating a custom external service

If none of the pre-build web services match your needs, you can create a custom service i.e. select which of the standard web service functions are available via that service.

You can enable only the specific functions that you need to expose, so not compromising on security.



1. Access Administration > Site administration > Plugins > Web services > External services
2. Click Add new custom service
 - o 'Authorised users only' - If enabled, you will need to select the authorised users manually. Otherwise all users with appropriate permissions are allowed
 - o 'Required capability' - If enabled, any user accessing the web service will be checked against this selected capability. (This is just an additional optional security layer.)
3. Enter a name and check Enabled
4. Click the button 'Add service'

Adding functions to the service

Your service is currently empty and doesn't do anything. Web service functions need to be added. Your choice will be dictated by what you allow the external application to do. For this example, select 'Create group'.

1. Click 'Add functions' link
2. Select 'create group' function and click the 'Add functions' button

Note that deprecated functions can not be added to services although the ones that are already part of a service can remain there until they are removed from Moodle codebase.

You should be back to the service functions list. 'Required capabilities' are indicated for each function. Users need the required capabilities to run a function. The function descriptions in the API Documentation can also give you more information about the required capabilities (Administration > Site administration > Plugins > Web services > API Documentation).



Enabling capabilities

The final step is to grant appropriate permissions. The following capabilities should be allowed:

- moodle/webservice:createtoken - for allowing users to generate a security key

- [webservice/rest:use](#), [webservice/soap:use](#), [webservice/xmlrpc:use](#), [webservice/amf:use](#) which match the enabled protocols.
- The service *Required capability* if set (*Administration > Site administration > Plugins > Web services > Manage services > **Edit** link*).
- The required capabilities for the web service functions. These required capabilities are listed when you add a function to the service. For more information about roles and capabilities, read the [Manage roles](#) documentation.

Once done, the web service should be set up. Users should be able to [obtain a personal security key](#).

Alternative settings

Authorise only specific users

1. Administration > Site Administration > Plugins > Web services > External Services
2. Select **Authorised users** link (the service must have been set as **Authorised users only** in the **Edit** link)
3. Select some users and click **Add**

Moodle indicates if some capabilities need to be assigned to an authorised user. Moreover, if you click on the authorised user fullname, you can set up some specific options: IP restriction and Valid until.

Create a token

This feature allows you to create a token for specific user. It can be useful if a user doesn't have the `moodle/create:token` capability. This is also the only way to create a token for an administrator. For security reason, tokens are not automatically generated in the administrator security keys page.

Administration > Site Administration > Plugins > Web services > Manage tokens

1. Click on **Add**
2. Select the created user and service
3. Click on **Saves changes**

As you created a token for this user, you do not need to assign "`moodle/webservice:createtoken`" to him/her. Finally, note that, as for authorised users, you can also set *IP restriction* and *Valid until* on a token.

Blogs

The word 'blog' is a contraction of 'web log'. Blogs are a form of online journal used by millions of people around the world for self-expression and communicating with family and friends. Blogs are usually organized as a chronological series of postings created by the author of the blog. Blogs usually are written by one person, although some blogs can be authored by groups of people. Blogs in Moodle are user based - each user has their own blog. Users can also register their external blogs, such as Blogger or WordPress, so that entries are automatically included in their Moodle blog.

- [Blog settings](#)
- [Using Blogs](#)

Using RSS feeds

Activities and other areas that produce RSS feeds

- [Blogs](#)
- [Database](#)
- [Forum](#)
- [Glossary](#)

Subscribing to RSS feeds

To subscribe to an RSS feed from Moodle, click on the activity and look in the Administration block for a link "RSS feeds for (this activity)" and copy the address from your browser bar to your RSS reader software.

Restricting access to RSS feeds

To ensure security and privacy, RSS feed URLs contain a special token that identifies the user they are for. This prevents users from accessing areas of Moodle they shouldn't have access to via RSS feeds.

Users have this token automatically created for them the first time they access an area of Moodle that produces an RSS feed. If a user feels that their RSS feed token has been compromised in some way they can request a new one by going to the User menu > *Preferences* > *Security keys* and clicking the Reset link next to the RSS key.

To provide a public RSS feed for use on another site, login to your Moodle site as a guest, and then copy the RSS feed address. Make sure guests do not have to agree to a Site policy document.

Experiment – 10

Aim: - Manage Moodle site, Managing authentication, Manual accounts, No login, Email-based self-registration, Account

Managing a Moodle site

A user with the role of [Administrator](#) is typically in charge of a Moodle site once it has been installed, although some tasks may be delegated to others by assigning them a role such as [Manager](#). The links below provide more information about how to manage your Moodle site.

New to the role of site admin? See our [Guide for new administrators](#).

Users and Courses

- [Authentication](#) - different methods of adding new users to your Moodle
- [Managing accounts](#) - how to search for, edit, delete or perform bulk actions on users
- [Roles and permissions](#) - how to add or remove permissions from students, teachers and other users on your Moodle
- [Enrolments](#) - different methods of adding users to courses

Server and Security

- [Admin tools](#) - a list of useful tools, such as [Database search and replace](#) and [database transfer](#)
- [Backup](#) - how to backup your site and courses
- [Developer tools](#) - how to debug your site, purge caches and tools for test sites only
- [Performance](#) - ways to check the efficiency and smooth running of your Moodle
- [Security](#) - how to keep your Moodle safe
- [Web services](#) - how to connect other systems and applications that communicate using web services, including [Moodle Mobile](#), the official mobile app for Moodle

Site-wide settings

- [Language](#) - how to add new languages and alter the default terms used
- [Location](#) - how to set time zones for the site and users, and default city and country
- [Logging](#) - Manage log stores
- [Server settings](#) - registration, maintenance and default settings
- [Site-wide reports](#) - a list of useful reports for administrators
- [Site appearance](#) - ways to change the display and navigation of your site

More

- [Site registration](#)
- [More _____ features](#) - [Badges](#), [Blocks](#), [Blogs](#), [Calendar](#), [Comments](#), [Filters](#), [Messaging](#), [Notes](#), [RSS feeds](#), [Tags](#), [Competencies](#)

Authentication

Authentication is the process of allowing a user to log in to a Moodle site with a username and password.

Authentication plugins

Moodle provides a number of ways of managing authentication, called *authentication plugins*. Standard authentication plugins are:

- Manual accounts - accounts created manually by an administrator
- No login - suspend particular user account
- Email-based self-registration - for enabling users to create their own accounts
- CAS server (SSO) - account details are located on an external CAS server
- External database - account details are located on an external database
- LDAP server - account details are located on an external LDAP server
- LTI - works with the Publish as LTI tool enrolment method to connect courses and activities
- Moodle Network authentication - how different Moodle sites can connect and authenticate users
- No authentication - for testing purposes or if the Moodle site is not available on the Internet. Do NOT use on public servers!
- Shibboleth - account details are located on an external Shibboleth server
- Web services authentication

There are also many additional authentication plugins in the Moodle plugins directory.

Manual accounts

- Location: Administration>SiteAdministration>Plugins>Authentication>Manage authentication.
- Manual authentication is enabled by default in Moodle and requires the administrator to manually create all the user accounts.

Manage authentication						
Available authentication plugins						
Name	Users	Enable	Up/Down	Settings	Test settings	Uninstall
Manual accounts	21			Settings		
No login	0			Settings		
Email-based self-registration	0			Settings		Uninstall

Password expiry settings

An administrator can specify an expiry time for passwords and when to notify users of the imminent expiry by enabling the setting in Site administration>Plugins>Authentication>Manual accounts.

Password expiry settings

Enable password expiry	No	Allow passwords to expire after a specified time.
Password duration	30 days	Length of time for which a password is valid.
Notification threshold	Never	Number of days before password expiry that a notification is issued.

Locking user fields

- The settings page in Administration>Site Administration>Plugins>Authentication>Manual accounts allows for certain fields to be "locked". Users are not allowed to change the locked fields.

Lock user fields

First name	Unlocked	You can lock user data fields. This is useful for sites where the user data is maintained by the administrators manually by editing user records or uploading using the 'Upload users' facility. If you are locking fields that are required by Moodle, make sure that you provide that data when creating user accounts or the accounts will be unusable.
Surname	Unlocked	
Email address	Unlocked	
City/town	Unlocked	Consider setting the lock mode to 'Unlocked if empty' to avoid this problem.
Country	Unlocked	
Language	Unlocked	
Description	Unlocked	

Adding a new user

This may be done on an individual basis in Administration>Site Administration>Users>Accounts> Add a new user

Adding users in bulk

This may be done via a text file upload in Administration>Site Administration>Users>Accounts> Upload users

No login

An administrator may "turn off" or suspend a user's account by selecting "No Login" in the "Choose an authentication method setting" in the user's profile, or by ticking the 'Suspended account' checkbox.

Different users have to have unique email addresses in their profile, even when email is not enabled. When the "No Login" setting is used, the account email may not be re-used to create another account. This essentially is a way to bar a particular user account from using Moodle for the duration that the account is assigned to this authentication type.

Email-based self-registration

The email-based self-registration authentication method enables users to create their own accounts via the 'Create new account' button on the login page. They then receive an email at the address they specified in their account profile to confirm their account.

Enabling email-based self-registration

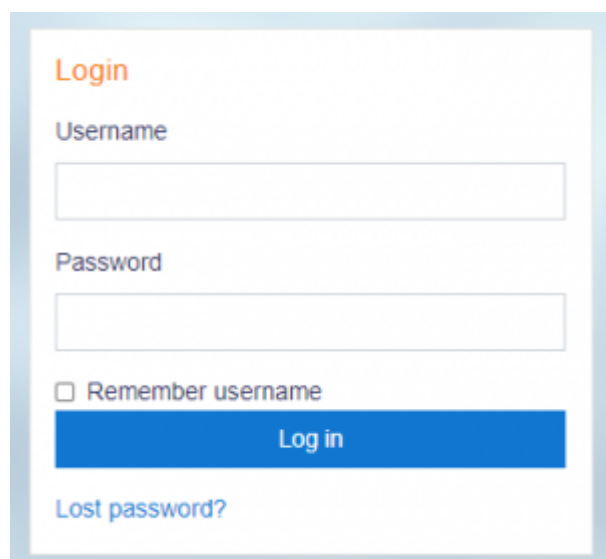
1. Go to Administration > Site administration > Plugins > Authentication > Manage authentication and click the crossed-out-eye icon opposite email-based self-registration
2. Select email-based self-registration from the self registration drop-down menu in the common settings (further down the same page, underneath the authentication plugins)
3. Click the 'Save changes' button

Warning: Enabling self registration results in the possibility of spammers creating accounts in order to use forum posts, blog entries etc. for spam. This risk can be minimized by limiting self registration to particular email domains with the allowed email domains setting in Administration > Site administration > Plugins > Authentication > Manage authentication. Alternatively, self registration may be enabled for a short period of time to allow users to create accounts, and then later disabled.

Note: The Email-based self-registration authentication plugin must be enabled to allow users who previously self-registered to login. Selecting Email-based self-registration as the self registration method allows potential users to self register.

You may change the text that appears under "Is this your first time here?" in two ways:

1. Add text in the Instructions box found in Site administration ... Plugins ... Authentication ... Manage authentication or,
2. Use the Language Customization tool to edit the language string 'loginsteps' found in moodle.php. Note, this string is different from 'loginstepsnone', which is the string used for No Authentication registration.

A screenshot of the Moodle login form. It features a light blue border. At the top left, the word "Login" is written in orange. Below it, the label "Username" is in blue, followed by a white text input field. Underneath, the label "Password" is in blue, followed by another white text input field. Below the password field is a checkbox with the label "Remember username" in blue. A prominent blue button with the white text "Log in" is positioned below the checkbox. At the bottom left, the text "Lost password?" is displayed in blue.

Enable reCAPTCHA element

A CAPTCHA is a program that can tell whether its user is a human or a computer. CAPTCHAs are used by many websites to prevent abuse from bots, or automated programs usually written to generate spam. No computer program can read distorted text as well as humans can, so bots cannot navigate sites protected by CAPTCHAs.

Spam protection may be added to the email-based self-registration new account form with a CAPTCHA element - a challenge-response test used to determine whether the user is human.

In addition to enabling the reCAPTCHA element, email-based self-registration should be set as the self registration authentication plugin and reCAPTCHA keys should be set in the manage authentication common settings.

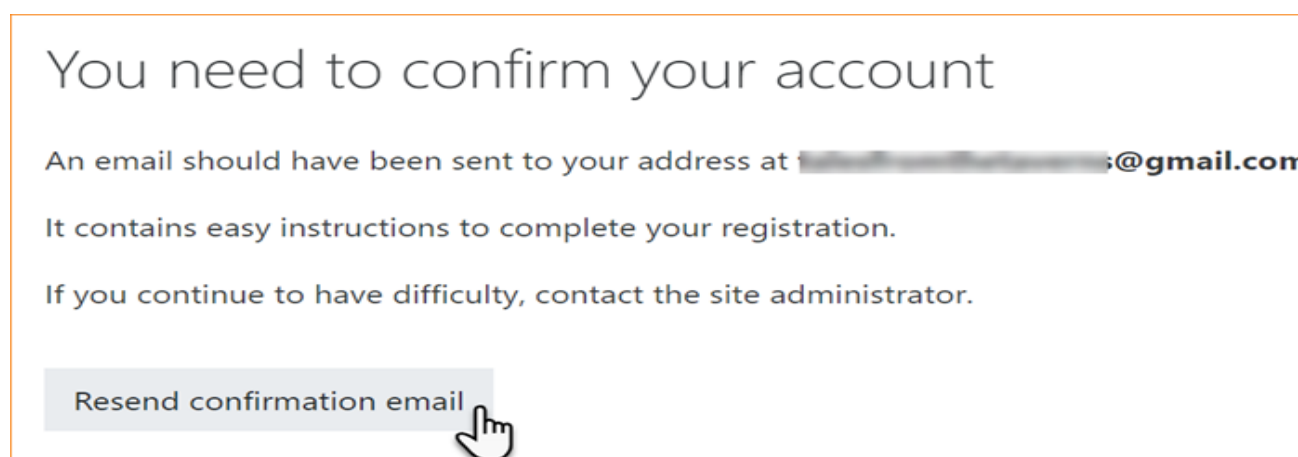
Email confirmation message

An automated email confirmation message is sent to the user using the support contact email address. You can change this text in Administration > Site administration > Language > Language customization by choosing the appropriate language pack. selecting 'moodle.php' from 'core' and searching for the string identifier name 'emailconfirmation' and editing it with a local customization.

Component	String	Standard text
core	emailconfirmation	Hi {\$a->firstname}, A new account has been requested at '{\$a->sitename}' using your email address. To confirm your new account, please go to this web address: {\$a->link} In most mail programs, this should appear as a blue link which you can just click on. If that doesn't work, then cut and paste the address into the address line at the top of your web browser window. If you need help, please contact the site administrator, {\$a->admin}  string contains a placeholder

Resending the confirmation email

A button is available for new users to have their confirmation link resent if they cannot find the email, or if they access it beyond the deadline.



Support contact

An administrator can specify a support name, email and/or support page in 'Support contact' in the Site administration for including in the confirmation email. If left blank, the name of the primary administrator is used.

Tips

- Check your user list regularly for spammy/suspect names/emails and/or users in the system but not enrolled in the course
- Disable Blogs unless actually using them; some spambots know how to post there
- Enable some sort of login failure notification in Administration > Site administration > Security > Notifications so you can see who is having login issues
- Check Administration > Site administration > Reports > Spam cleaner from time to time
- Potential users may not receive the account confirmation email due to it ending up in the spam folder, being refused by the remote server, an invalid email address entered etc. Such accounts may be confirmed manually by an admin.

Experiment – 11

Aim: - Create Roles and permissions, Assign roles

Roles and permissions

A role is a collection of permissions defined for the whole system that you can assign to specific users in specific contexts. The combination of roles and context define a specific user's ability to do something on any page. The most common examples are the roles of student and teacher in the context of a course.

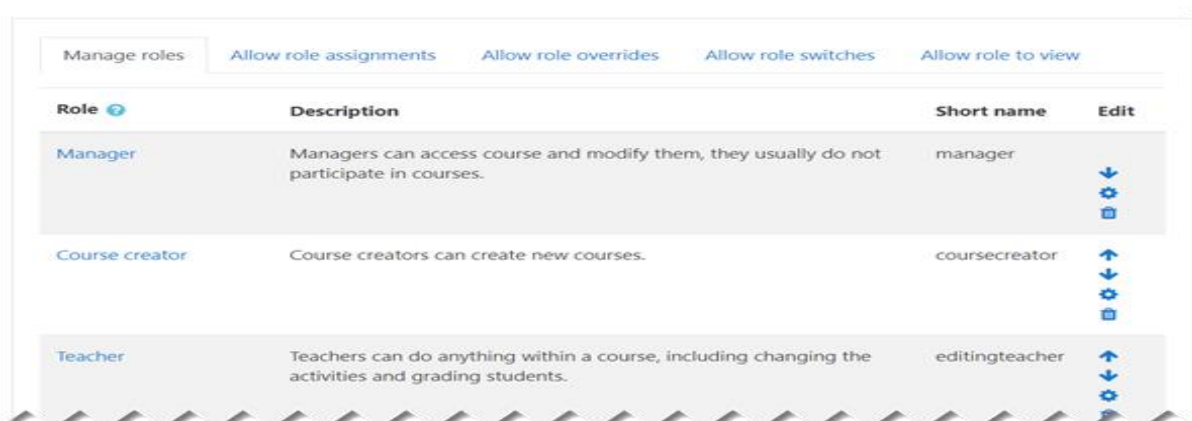
- [Managing roles](#)
- [Assign roles](#)
- [User policies](#)
- [Using roles](#)
- [Standard roles](#)
- [Creating custom roles](#)
- [Role export and import](#)
- [Permissions](#)
- [Override permissions](#)

Managing roles

Managing overall role capabilities can be done by an administrator using Administration > Site administration > Users > Permissions > Define roles. This is the place to add custom roles or modify existing roles. The "Manage roles" tab, allows the system administrator to edit any one of over 350 different capabilities associated with any role. The "Allow role assignments", "Allow role overrides" and "Allow role switches" contain a matrix which give the ability for a specific role to work with other specific roles.

Manage roles

The 'Manage roles' tab contains a list of roles on your site. The edit column contains icons for editing, deleting roles and copying roles, and for moving them up or down in the list (affecting the way that roles are listed around Moodle).



The screenshot shows the 'Manage roles' tab in Moodle's administration interface. It features a table with columns for 'Role', 'Description', 'Short name', and 'Edit'. Three roles are visible: 'Manager', 'Course creator', and 'Teacher'. Each role has a set of icons in the 'Edit' column for moving, editing, deleting, and copying.

Role	Description	Short name	Edit
Manager	Managers can access course and modify them, they usually do not participate in courses.	manager	Move, Edit, Delete, Copy
Course creator	Course creators can create new courses.	coursecreator	Move, Edit, Delete, Copy
Teacher	Teachers can do anything within a course, including changing the activities and grading students.	editingteacher	Move, Edit, Delete, Copy

To edit a role:

1. Go to *Administration > Site administration > Users > Permissions > Define roles*.
2. Click the edit icon opposite the role you want to edit. For example "student".
3. On the editing role page, change permissions as required for each capability.
4. Scroll to the bottom of the page and click the "Save changes" button.

See [Creating custom roles](#) for information about adding a new role and creating a duplicate role.

Role name localization

If a standard role name or description is empty Moodle uses a default string from the current language pack. Custom roles can be customised using multilang syntax.

You may also override the role names separately in each course.

Resetting a role

To reset a role back to the default permissions:

1. Go to *Administration > Site administration > Users > Permissions > Define roles*.
2. Click on the name of the role, for example "student".
3. Click the 'Reset' button.

Note that if you have students who have been given extra permissions at course and/or activity levels (such as forum ratings), then they will no longer be able to do this once the role has been reset to its default. A teacher would need to go back and set up these extra permissions in the course/activity levels again.

Allow role assignments

The "Allow role assignments" tab allows (or does not allow) a specific role to be able to assign specific roles to a user.

Enabling teachers to assign other teachers

By default, teachers can only assign other users the roles of non-editing teachers, students and guests. If you want teachers to be able to assign other teachers in their course, you can allow the role assignment:

1. Click on *Administration > Site administration > Users > Permissions > Define roles*.
2. Click the Allow role assignments tab.
3. Click the checkbox where the teacher row and column intersect.
4. Click the "Save changes" button.

Allow role overrides

The "Allow role overrides" tab allows (or does not allow) a specific role to be able to override specific roles for a user. For example, it might allow a teacher role to override a student's role to a non-editing teacher's role.

Note that the settings only apply to roles that have the capabilities moodle/role:override or moodle/role:safeoverride allowed.

Allow role switches

The "Allow role switches" tab allows (or does not allow) a specific role to be able to temporarily change their role to another specific role. For example, this might allow a users assigned to a custom role in a course to see "Student" in the Settings > Switch role list.

Note: the selected role must also have the moodle/role:switchroles capability to be able to switch.

Allow role to view

This setting allows the administrator to decide which roles users can see, search and filter by, according to their existing role.

Roles capabilities

- Create and manage roles
- Assign roles to users
- Switch to other roles

Assign roles

Introduction to context and roles

- Apart from the manager and course creator, users do not normally have a site-wide (or system) role. Even though you may be a teacher offline, when you are in Moodle you could have a teacher role in the course you teach in but a student role in another course where you are studying for a diploma.
- Because of the way Moodle works, assigning roles is done for a particular context. A site and course are examples of two different contexts. When you create a new role or tweak a pre-existing role via *Administration > Site Administration > Users > Permissions > Define roles*, you are asked in which context(s) you want the role to be assigned:

Context types ☐ System
where this role ☐ User
may be ☒ Category
assigned ☒ Course
☒ Activity module
☐ Block

- Here are some examples of contexts; how to get to the assign roles screen, and when/why you would assign roles here:

System context

To assign a role in the system context, go to Site administration > Users > Permissions > 'Assign system roles'.

Any roles assigned here apply across the whole site. It makes sense therefore that only roles that need this functionality can be assigned here. The Manager role and Course creator role are examples of two such roles. Assigning a teacher or student here would result in their being able to teach/study in every single course on the site, which is not usually desirable.

Assigning system roles by CSV

Where certain custom roles are applied in the system context, it is possible to upload users to that role in bulk by adding the field sysrole1 (etc) to a CSV file.

F13						
	A	B	C	D	E	F
1	username	password	firstname	lastname	email	sysrole1
2	jdep	1234	Johnny	Depp	jdepp@example.com	manager
3	bbardot	1234	Brigitte	Bardot	bbardot@example.com	coursecreator

When previewed, there is a column indicating their system role:

Upload users preview						
CSV						
line	username	password	firstname	lastname	email	sysrole1
2	jdep	1234	Johnny	Depp	jdepp@example.com	manager
3	bbardot	1234	Brigitte	Bardot	bbardot@example.com	coursecreator

Once uploaded, the users are present on the 'Assign system roles' screen:

Assign roles in System ⓘ

WARNING! Any roles you assign from this page will apply to the assigned users throughout the entire system, including the front page and all the courses.

Please choose a role to assign

Role	Description	Users with role
Manager		1 Johnny Depp
Course creator		1 Brigitte Bardot

See Upload users for information on adding users to roles via CSV.

Front page context

- Administration > Site Administration > Front Page>Users>Permissions>Assigned roles
- Those with a role in the system context do not need to be assigned a role here as well.
- However you might want to allow a teacher to manage items on the front page; in this instance, you would assign them the role on the Front page.

Course Category context

- See [Category enrolments](#) - users may be enrolled in the category to save enrolling them in each individual course in that category.
- See also [Assign Manager role at category level](#).

Course context

- Go to the Participants page
- Click the "Enrol users" button and click those users you wish to enrol

The dropdown menu at the top shows roles for which you are allowed to enrol; typically those users with lower roles than you. See [Participants](#) for more details.

Block context

- (Within the block) Administration > Assign roles
- You may wish to assign roles to a block if, for instance you want specific people to see the block but for it to be hidden from others

Activity Module context

- (Within the activity settings) Administration > Locally assigned roles
- An example of this is assigning a student the teacher role locally in an individual activity like a forum so they can moderate their classmates' posts while still retaining the student role in the rest of the course.

User context

The user context is used for roles such as mentor, team leader or the [Parent role](#). The role to be assigned must have 'User' ticked as the context type where it is to be assigned.

To assign a user the role of mentor in the context of their mentee, click the mentee's profile, then Preferences then 'Assign roles relative to this user'.

If a mentor has lots of mentees, the role of mentor can be assigned to them all in one go as follows:

1. Put all mentees in a cohort
2. Go to Site administration > Users > Permissions > Assign user roles to cohort

Hierarchy

By assigning a role to a user in a certain context, you grant them the permissions contained in that role for the current context and all lower contexts.

The list of contexts in hierarchical order is as follows:

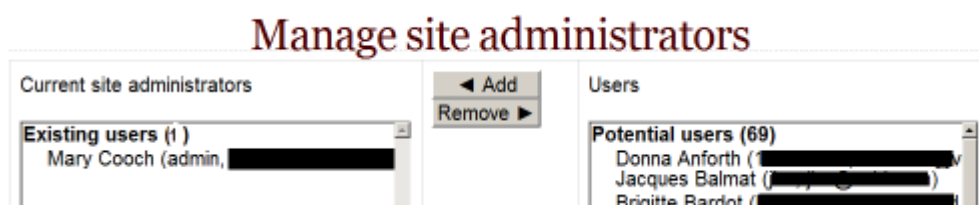
- System (no parent)
- Front page (parent = system) -
- Course category (parent = parent category or system)
- Course (parent = category or system)
- Module (parent = course or system)
- Block (parent = course or system)
- User (parent = system)

Roles can be inherited. For example if a user is assigned a Teacher role in a specific course category then the user will have this role in ALL courses within the category. Tip: use the override feature in a specific context for exceptions.

Roles will only work if the role assignment is made in the correct context. Some examples: a Teacher role should be assigned to a user in the course or course category context, a Forum moderator for a particular forum should be assigned in that specific forum.

Assigning someone the role of Site administrator

- Site administrators are assigned via a special page: Administration > Site Administration > Users > Permissions > Site Administrators. Select the name from the right and move it over to the left:



- The original (primary) administrator cannot be deleted.

Checking a user's role assignments

To check a user's role assignments go to their profile page, then under the Administration heading, click on 'Preferences'. On the Preferences page, under the 'Roles' heading click on 'This user's role assignments'.

Hidden roles

If you want to provide users with access to the course, but don't want them to be visible in the participants list, use the Other users link in the course administration menu (Administration > Course Administration > Users > Other Users). Assigning roles here provides course access, and editing rights according to the permissions set for the role assigned without actually enrolling the user in the course. This is similar to the functionality of the "hidden user" check box in previous versions of Moodle.

Note: By default, the only role which can be assigned to other users is the manager role. To enable other roles, such as teacher to be assigned, the capability moodle/course:view should be allowed for the role (see below).

Enabling teachers to assign the role of teacher

By default, teachers are only allowed to assign the roles of non-editing teacher, student and guest. To enable teachers to assign the role of teacher:

1. Access Site Administration > Users > Permissions > Define roles.
2. Click the tab "Allow role assignments".
3. Click the checkbox where the teacher row and column intersect.
4. Click the "Save changes" button.

Beware of assignments that don't make sense

There are many role assignments that do not make sense as the underlying functionality does not exist. Just because you give someone the "right" to do something does not guarantee that the interface or facility actually exists within the context that you have assigned that right. For example, you can assign a user the right to create new categories in the category context. However there is no interface within Moodle to do that (category creation is only available at the system level).

Multiple assignments

A significant part of the roles infrastructure is the ability to assign a user into multiple roles (at the same time). The capabilities of each role are merged to produce the effective set of capabilities. For example, a user could be both a Teacher and Student in the same course. You should be careful to ensure that if you change a user's role that you remove them from any other roles as required as this will no longer be done automatically.

Experiment – 12

Aim: - Implement Password salting.

Password salting

What is password salting?

Password salting is a way of making password hashing more secure by adding a random string of characters to passwords before their hash is calculated, which makes them harder to reverse.

How does Moodle use password salting?

Prior to Moodle 2.5 there was a single site-wide salt which was used when hashing every user's password. From Moodle 2.5 onwards Moodle automatically generates and adds a different salt for each individual user. This is more secure and means that a site-wide configuration variable for the salt is no longer required for **new** installations of 2.5 or greater.

Backwards compatibility for site upgrades

Important! If you are upgrading a site from 2.4 or below and you are already using a site-wide salt in your configuration file, **you need to keep using it to ensure your existing users can still log in.**

Each time a user logs in their password hash will be converted to the new scheme, but it may take a long time before all your users have logged in. Alternatively, if you would like to force all your users to use the new scheme you could force reset all passwords using [Bulk user actions](#).

For more details about the old site-wide salt configuration, see the [Moodle 2.4 Password Salt documentation](#).

Sites running PHP version below 5.3.7

The new password hashing mechanism relies on bcrypt support from PHP which is only normally available in PHP version 5.3.7 or greater (see note below). If you are using a version of PHP which doesn't properly support bcrypt, Moodle will fall back to the old password hashing scheme, so we recommend that you continue to use a site-wide salt until you are able to upgrade PHP.

Note: While an important fix to PHP's hashing algorithm was added in 5.3.7, some distributions of Linux have backported the fix to bcrypt to earlier versions of PHP. This means that some earlier versions of PHP may still work. To confirm if your PHP supports the new hashing scheme you can use [this test](#).

Experiment – 13

Aim: - Perform Site backup, Course backup, Course restore, automated course backup

Site backup

A site backup allows a site administrator to save everything associated with a moodle site. These backups can be restored to bring a site back to the point in time when the backup was made.

Performing regular backups are highly recommended to reduce the amount of lost information in the event of a problem on the site and to speed the overall recovery process.

What needs to be backed up?

A Moodle system comprises three parts:

1. The data stored in the database (For example, a MySQL database)
2. The uploaded files (For example, site and course files uploaded via Moodle located in moodledata)
3. The **Moodle code** (For example, everything in server/htdocs/moodle)

You can confirm where all these things are located in a Moodle installation by checking the **config.php** file.

1. **\$CFG->dbname** shows the database name
2. **\$CFG->prefix** shows the the database table name prefix
3. **\$CFG->dataroot** controls where the *uploaded files* are stored; and
4. **\$CFG->wwwroot** points to where the *code* is stored.

Tip

Generally speaking, the database ("dbname and prefix") and the uploaded files (dataroot) are the two most important to copy on a regular basis. These contain information that will change most often.

The Moodle code (wwwroot) is less important as a frequent backup, since it will only change when the the actual code is changed through upgrades, addins and code tweaks. You can always get a copy of the standard Moodle code from <http://download.moodle.org> so you only have to backup the parts you added or changed yourself.

Creating a backup of your Moodle site

Database

The right way to back up your database depends on which database system you are using. The instructions below are one way to back up a MySQL database. Another option would be to use a tool like phpMyAdmin to manually make a backup. The documentation for your database will give more options.

There are many ways to do such backups. Here is an outline of a little script you can run on Unix to backup the database (it works well to have such a script run daily via a cron task):

```
cd /my/backup/directory
mv moodle-database.sql.gz moodle-database-old.sql.gz
mysqldump -h example.com -u myusername --password=mypassword -C -Q -e --create-options
mydatabasename > moodle-database.sql
gzip moodle-database.sql
```

Also consider using an option to help ensure that the backup is consistent (e.g. avoid the data changing half-way through the backup), e.g. for MySQL/MariaDB and mysqldump try [--single-transaction](#).

Character encoding

Make sure that a database backup uses the correct character encoding. In most databases, use UTF-8. When dumping the entire Moodle database, check for possible character encoding issues. In some instances, backups created with mysqldump or phpMyAdmin may not properly encode all of the data. This will result in non-readable characters when the database is restored.

Tip: One solution is to use MySQL Administrator 1.1 or another tool that will force a UTF-8 dump of the data.

Tools for database backups

phpMyAdmin

phpMyAdmin is the tool of choice with most web hosting providers.

MySQLDumper

MySQLDumper is a backup script for MySQL databases, written in PHP and Perl. MySQLDumper uses a proprietary technique to avoid execution interruption when running PHP scripts (the max. execution time is usually set to 30 seconds). MySQLDumper also cares for the encoding problems mentioned above. It also works with compressed files and allows setting up regular cron jobs for updating and updating to a remote FTP site.

AutoMySQLBackup

AutoMySQLBackup is a backup script for MySQL databases, written in PHP. AutoMySQLBackup will create Daily, Weekly and Monthly backups of one or more of your MySQL databases from one or more of your MySQL servers. Features include email notification, compression and encryption, configurable backup rotation, incremental database backups, and more.

Uploaded files (moodledata)

Through the Moodle interface, users can upload or create files and folders. These are located in a directory, often called "moodledata". Since they are just files and folders, there are many different ways to backup or copy moodledata.

- For example, using a file transfer program, copy the entire moodledata directory to a different area, drive or computer. Example of file transfer programs include: FTP, WinSP, wget, rsync.
- You might use a compression program to create compact files (tar, zip, 7z, XZ, BZIP2, GZIP, and WIM are a few file formats) of the entire directory. This can be done before or after file transfers.

Tips

- Typically not all moodledata files change between regular/periodic backups. A new Administrator might want to look into [incremental or other efficient backups](#) procedures.
- Depending upon the operating environment there are many tools for backing up server files and ways of backing up moodledata. See [Tools for backing up server files](#) for tips on using rsync, FTP or Wget.

Moodle code

Backing up the Moodle code, will be similar to backing up moodledata. See [Tools for backing up server files](#).

Tip

It is always a good idea to have several backup copies of your Moodle code files. While you can always download a fresh base copy of the Moodle code from <http://download.moodle.org>, you might have customized that code. It is a good idea to create a separate backup of your Moodle code before you customize the code. This includes installing [Contributed code](#), [Themes](#) and upgrading.

Course backup

A course can be saved with some or all of its parts by using the course backup. Typically, the site administrator will set a schedule of [automated course backups](#) for the whole site. A teacher can create a backup or download an existing backup for safe keeping, or for use on another Moodle site.

Backing up a course

To backup a course

1. Go into the course.
2. Click the Backup link either in the gear menu or the Administration block (depending upon the theme).
3. Initial settings - Select activities, blocks, filters and other items as required then click the Next button. Users with appropriate permissions, such as administrators and managers, can choose whether to [include users](#), anonymize user information, or include user role assignments, groups, groupings, user files, comments, user completion details, course logs and grade history in the backup.
4. Schema settings - Select/deselect specific items to include in backup, then click the Next button.
5. If desired, select specific types of activity to be backed up by clicking the link 'Show type options'
6. Confirmation and review - Check that everything is as required, using the Previous button if necessary, otherwise click the 'Perform backup' button
7. Complete - Click the Continue button

Backup settings

IMS Common Cartridge 1.1

Include enrolled users

Anonymize user information

Include user role assignments

Include activities and resources

Include blocks

Include filters

Include comments

Include badges

Include calendar events

Include user completion details

Include course logs

Include grade history

Include question bank

Include groups and groupings

A backup file (with distinctive .mbz extension to avoid confusion with .zip files) is then saved in the course backup area. Backup file names are of the form backup-moodle2-course-coursename-date-hour.mbz, ending in -nu.mbz when backed up with no users and -an.mbz with anonymized names.

Tip: If you are satisfied with the default settings and don't wish to go through all the backup screens, you can simply click 'Jump to final step' to perform the backup.

Anonymizing user information

Anonymize user information is a backup feature which "protects user identities" by making each user anonymous. If this box is checked in the backup initial settings, Moodle will substitute aliases for real names, substituting @doesntexist.com email addresses and so forth. For example "Max Manager" might become "anonfirstname4 anonlastname4".

Asynchronous course backups

Note: With large courses it is helpful to be able to continue working while a course backup is being made. To achieve this, you need to enable asynchronous backups in Site Administration / Courses / Backups / Asynchronous backup/restore.

Backup via CLI for administrators

Site administrators can backup selected courses using a CLI script.

Options

```
--courseid=INTEGER (Course ID for backup.)  
--courseshortname=STRING (Course shortname for backup.)  
--destination=STRING (Path where to store backup file. If not set the backup will be stored within the  
course backup file area.)  
-h, --help (Print out this help.)
```

Example

```
\$sudo -u www-data /usr/bin/php admin/cli/backup.php --courseid=2 --destination=/moodle/backup/
```

On Windows you will use cmd.exe to run the /admin/cli/backup.php script. If you have your Moodle data folder setup on a separate data server you can specify a UNC path to your backup folder on that data server. From the command line CD to the Moodle \admin\cli folder and run this command:

```
php backup.php --courseid=25 --destination=\\moodledata\backup
```

where '--courseid' is the id of the course that you want to backup.

The .mbz backup file for courseid=25 will be stored in the backup subfolder in the Moodle data (\$CFG->dataroot) folder.

Tips and tricks

Lose content after a restore in Moodle 2.0 ? Do you see topic headings that say "Orphaned activities"? Solution: Go to course settings and increase the number of topic sections and things will return to normal.

Creative uses

The backup and restore processes can offer the teacher and administrators many creative solutions.

- Duplicating courses or specific activities in one course to another course (similar to Import)
- Updating a production Moodle site course, with material from a localhost site course
- Transferring a course to a new Moodle site.
- In earlier versions of Moodle, a way of rolling a course forward without past student activity
- Creating a blank activity, save just that activity and then restore it to the course or another course one or more times.

General backup defaults

Default settings for course backups can be set by a site administrator in 'General backup defaults' in the Site administration.

Selected settings may be locked, so that they cannot be changed when creating a course backup.

By selecting a time in the "Keep logs for.." dropdown, it is possible to specify how long backup logs information is kept before being deleted. As this information may be very large, it is recommended the length of time chosen be quite short.

Course backup stops above 90%, not showing any errors

This has been reported to be caused by:

- a non-standard course format. Try replacing the course format.
- not enough server RAM. Adding more RAM to your server is usually the first performance recommendation.

Course backup capabilities

- Anonymise user data on backup
- Backup courses
- Backup sections
- Backup for import
- Configure backup options
- Download files from backup areas
- Backup user data

Error: Trying to restore user 'admin' from backup file will cause conflict

- Before proceeding, it would be advisable to have a second admin account just in case something goes wrong.
- Check that the admin accounts on each server have **different** unique usernames and email addresses and then try again to backup up the course with user data and restore it.
 - If you can not make a new backup and you must use an existing 'conflicting' backup file, you will have to change the new site's main admin account with extreme care to remember the new main admin's password and email.
- Note that the admin account from the imported file may lose the password and become unusable;
- You are advised to update/reset the 'imported' main 'admin's account's password before logging out, as this will most likely be the main 'admin' account for the site, and you would get into serious problems if this account stops working.
- If something goes terribly wrong, you can use the second admin account to fix the main admin account.

Course restore

Restoring a course backup

A course backup file (.mbz) may be restored from within any existing course for which you have permission. During the restore process, you will be given the option to restore as a new course or into an existing course.

Overview of course restore. (Instructions are for MoodleCloud but also apply to other sites.)

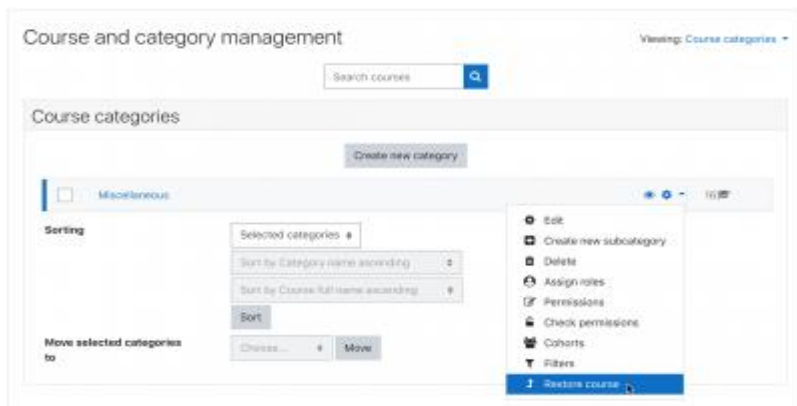
1. With the Boost theme click the gear menu and then 'Restore' Otherwise, go to *Site administration > Front page settings > Restore* (if you have front page permissions) or *Administration > Course administration > Restore* (if you have an empty course to restore into.)
2. Upload the backup file or choose a file in the course backup area or user private backup area and click Restore
3. Confirm - Check that everything is as required then click the Continue button
4. Destination - Choose whether the course should be restored as a new course or into an existing course then click the Continue button
5. Settings - Select activities, blocks, filters and possibly other items as required then click the Next button
6. Schema - Select/deselect specific items and amend the course name, short name and start date if necessary then click the Next button
7. Review - Check that everything is as required, using the Previous button if necessary, then click the 'Perform restore' button
8. Complete - Click the continue button

Notes:

1. If you are restoring a course with user data (forum posts for example) then the user data will keep the original dates, whereas the activities and sections will change according to the new course start time you set.
2. If your new course has fewer sections than the course you are restoring, the extra sections will appear as "orphaned activities". Increase the number of sections in the new course to make them visible.
3. If you are restoring a course from a different Moodle site and you obtain the error message "Trying to restore user (admin) from backup file will cause conflict", you can enable the setting 'Allow admin conflict resolution' in *Site admin > Courses > Backups > General import defaults*
4. When restoring a course containing calendar events from activities e.g. assignment due dates, these events will appear in the calendar a short while after restoring the course. The events are added to the calendar via a cron task. Restoring multiple courses will result in queued tasks with a possible delay before events appear in the calendar.

Restoring from course and category management screens

Users with permission to access *Site administration > Courses > Manage courses and categories* or *Site administration > Courses > Manage courses and categories > (Category name)* can restore courses from here:



General restore defaults

An administrator can set defaults and lock selected restore settings in 'General restore defaults' in the Site administration. If a setting is locked, then it can not be changed when restoring a course.

Different defaults may be set for restoring and merging into another course and for restoring and deleting the course contents.

Course restore capabilities

- [Create users on restore](#)
- [Configure restore options](#)
- [Restore activities](#)
- [Restore courses](#)
- [Restore sections](#)
- [Restore from files targeted as import](#)
- [Roll activity configuration dates on restore](#)
- [Upload files to backup areas](#)
- [Restore user data](#)
- [Restore courses from automated backups](#)

Restore via CLI for administrators

In Moodle 3.10 onwards, site administrators can restore large backup files using a CLI script. Options:

```
-f, --file=STRING      Path to the backup file.
-c, --categoryid=INT   ID of the category to restore too.
-s, --showdebugging    Show developer level debugging information
-h, --help             Print out this help.
```

Example:

```
$sudo -u www-data /usr/bin/php admin/cli/restore_backup.php --file=/path/to/backup/file.mbz --categoryid=1
```

Automated course backup

Automating course backups

Automated course backups runs the same functions as individual [course backups](#). It's a good idea to schedule backups for when your server isn't usually busy. Running the backup tool over all the courses can be processor-intensive, so you shouldn't run it when there are a lot of students trying to access the server. You should always alert users with an announcement on the first page that if they log on during the backup hours they may notice a decrease in performance.

In order to make scheduled backups, you have to set up CRON to run periodically. Please refer to the [cron instructions](#).

Setting an automated course backup schedule

To set the backup schedule:

1. Go to *Administration > Site administration > Courses > Backups > Automated backup setup*.
2. Set backup_auto_active to enabled.
3. Select the days of the week to run the backup.
4. Set the execution time for the backup process. For most servers, early morning will be the best time. Keep in mind the time zone your server is using.
5. Set the "Save to..." path. If you can, choose a backup path on another machine or on a different drive than the one Moodle is on. You don't want to lose your backups at the same time you lose your Moodle site if the drive fails. If you leave the field blank, then backup zip files will be saved in the backup folder of each course files folder. On a Linux server the path might look like: **/home/(your user or domain)/public_html/archive/** or if it's below the publicly accessible folders: **/home/(your user or domain)/archive/**. On Windows the path might look like: **D:\Archive**.

If the path is not recognized a red X is shown after saving, otherwise a green check.

1. Set all other options appropriately.
2. Click the "Save changes" button.

Tip: To save disk space, you can choose to delete backups older than a specified number of days. If so, you should also specify a minimum number of backups to be kept. (The settings 'Delete backups older than' and 'Minimum number of backups kept' are new in Moodle 3.0 onwards.)

Once you've set up your backup schedule, Moodle will automatically create archives of all the courses on the server at the time you specified. Once the backup is complete, Moodle will send you an email describing the status of the backup.

Backups report

A last execution log of scheduled course backups can be found in *Administration > Site administration > Reports > Backups*. It displays a list of all course backups made, the time taken to create them, their status (see below), and the next scheduled automated backup execution time.

Clicking the 'View logs' icon next to the course name (1 in screenshot below) will display a detailed backup log for a particular course, including date, hour and minute when each step of the backup was performed (2). This log is useful for troubleshooting backup errors.

The screenshot shows two parts of the Moodle backup interface. The top part, titled 'Last execution log', contains a table with the following data:

Course	Time taken	Status	Next backup
Teaching with Moodle 	29 Apr, 01:51 - 29 Apr, 01:53	OK	6 May, 01:00

Below the table is a detailed view of the backup logs for 'Teaching with Moodle', indicated by a circled '2'. The title is 'Backup logs of Teaching with Moodle'. Below the title is a pagination bar: 'Page: 1 2 3 4 5 6 7 8 9 10 11 (Next)'. The main heading is 'Logs of the backup executed on Wednesday, 29 April 2015, 1:51 AM'. Below this is a box containing 'Status - OK'. A message states 'There were no logs found for this backup.' At the bottom is another pagination bar: 'Page: 1 2 3 4 5 6 7 8 9 10 11 (Next)'.

Backup status

- **OK** - The course backup was successfully completed.
- **Skipped** - The course was skipped (not backed up) because the course was unavailable to students and had not been changed in the last 31 days. This is not an error, but a feature designed to save process time.
- **Unfinished** - The backup was not finished. This might have happened because the job to execute backups ran out of time or resources (e.g., the cron running the backups ran out of time before finishing all the courses waiting). The unfinished jobs should be completed when the next scheduled backups are repeated.
- **Error** - There were errors in the backup process for the course, so a backup was not completed. For additional details about the error (1) check the backup log for the affected course (i.e. click

the 'View logs' icon next to the course name) or (2) back up the course manually with debugging turned on to see what error message is displayed.

Course versus site backups

- Automated course backups are more expensive in terms of time and CPU usage. The recovery time to have your site running again is longer.
- Course backups are useful for obtaining "fresh" copies of courses to be re-used or distributed individually, however they should never be used as a primary backup system (unless your hosting doesn't allow the preferred site backups).

Capabilities

Both the download course backup files capability and the restore user data from a course backup capability are required to be able to download automated course backups. There is also a restore courses from the automated backup storage capability.