

# Moodle On Ubuntu 20.04

## Configuring moodle with php 8.0,nginx and mariadb-server

### Update Ubuntu 20.04

Start with updating the system to install the latest available updates and refresh the APT package index cache.

```
$sudo apt update
```

### Install Nginx and PHP

install the Nginx web server

```
$sudo apt install nginx
```

### Setup PHP 8.x on Ubuntu 20.04

The version of PHP available through the default repository of Ubuntu is 7.4, we are installing PHP 8.0, the latest version of the PHP, we need to add an extra repository.

```
$sudo apt install software-properties-common
```

```
$sudo add-apt-repository ppa:ondrej/php
```

**Install PHP 8.0 and extensions:**

```
$sudo apt install php8.0
```

```
php8.0-{fpm,common,mbstring,xmldrpc,soap,gd,xml,intl,mysql,cli,mcrypt,ldap,zip,curl}
```

To check the version:

```
$php -v
```

**Now, edit the php.ini file to change some values.**

```
$sudo vim /etc/php/8.0/fpm/php.ini
```

**Note:** If you are using the default version of PHP or any other then change **8.0** with that in the above command.

Find the following lines and change their values

```
memory_limit = 256M
```

```
upload_max_filesize = 64M
```

```
max_execution_time = 360
```

```
//also don't forget to remove the semicolon (;) given in front of  
max_input_vars.
```

```
max_input_vars = 5000
```

Replace America/Chicago with your Time zone, if you don't know about it and how to use it, then you can find the **value here**. Now, restart the PHP-FPM process to reflect the changes you have made to your php.ini file:

```
$sudo service php8.0-fpm restart
```

## Create Moodle Database (MariaDB)

We can install MySQL, however, here we are using MariaDB (fork) which is one of the best open-source database servers. It is available to install using the default system repository of Ubuntu 20.04. Hence, just run the given command:

```
$sudo apt install software-properties-common -y
```

```
$curl -LsS -O https://downloads.mariadb.com/MariaDB/mariadb_repo_setup
```

```
$sudo bash mariadb_repo_setup --mariadb-server-version=10.6
```

```
$sudo apt install mariadb-server
```

```
$sudo apt update
```

```
$sudo apt install mariadb-server
```

Secure your Database server by setting up the root DB user password and removing anonymous rights.

```
$sudo mysql_secure_installation
```

### Create a Database and user for Moodle

Once the installation is completed, let's create a dedicated database for Moodle to use and save the data.

**Login to DB server CLI:**

**#Use the password you have set for it in the previous step.**

**\$sudo mysql -u root -p**

**Create Database:**

**Sql > CREATE DATABASE dbname;**

**#Replace dbname name with whatever you want to use:**

**Sql > CREATE USER 'db\_user'@'localhost' IDENTIFIED BY 'db\_password';**

**#Repalce db\_user and db\_password that you want to use.**

**Sql > GRANT ALL ON dbname.\* TO 'db\_user'@'localhost' IDENTIFIED BY 'db\_password' WITH GRANT OPTION;**

**Sql > FLUSH PRIVILEGES;**

**Sql > EXIT;**

## **Install Moodle on Ubuntu 20.04**

Moodle is not a package that we can install using the APT package manager, instead, we have to download it manually.

**\$ cd ~/Downloads**

**Download**

```
$ wget https://download.moodle.org/download.php/stable400/moodle-4.0.4.tgz
```

### **Extract the downloaded file:**

```
$ tar -zxvf moodle-*tgz
```

**OR**

### **Download Moodle**

Change into the /opt directory, clone the Moodle Git repository and change into the resulting moodle subdirectory.

```
#cd /opt  
#sudo git clone git://git.moodle.org/moodle.git  
#cd moodle
```

Determine the latest stable Moodle release from Moodle's official [list of releases](#). At the time of writing, the latest stable release is 3.10.

Identify the branch associated with the latest stable release. For release 3.10, this is MOODLE\_310\_STABLE.

```
#sudo git branch -a
```

Track and then check out the branch you identified.

```
#sudo git branch --track MOODLE_310_STABLE origin/MOODLE_310_STABLE  
#sudo git checkout MOODLE_310_STABLE
```

### **Move the extracted folder to the web root directory:**

```
$ sudo mv moodle /var/www/html/
```

### **Create a data directory to store Moodle-uploaded files and folders.**

```
$ sudo mkdir /var/www/html/moodledata
```

Directory must not be accessible publicly, hence, change its permission as well.

```
$ sudo chown www-data /var/www/html/moodledata
```

We also need to set the permission for the moved Moodle files, so that the web server can access it.

```
$ sudo chown -R www-data:www-data /var/www/html/moodle/
```

```
$ sudo chmod -R 755 /var/www/html/moodle/
```

## Create an Nginx server block for Moodle

```
$ sudo nano /etc/nginx/sites-available/moodle.conf
```

**Paste the following lines:**

**Note: Replace yourdomain.com with the domain you want to use to access Moodle. Also, don't forget to change the PHP version in the below lines if you are using other than php8.0.**

```
server {  
    listen 80;  
    listen [::]:80;  
    root /var/www/html/moodle;  
    index index.php index.html index.htm;  
    server_name yourdomain.com www.yourdomain.com;  
    location / {  
        try_files $uri $uri/ =404;
```

```
}  
location ~ [^/]\.php(/|$) {  
    include snippets/fastcgi-php.conf;  
    fastcgi_pass unix:/var/run/php/php8.0-fpm.sock;  
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;  
    include fastcgi_params;  
}  
}
```

### **Enable the Created Moodle server configuration block for Nginx:**

```
$ sudo ln -s /etc/nginx/sites-available/moodle.conf /etc/nginx/sites-enabled/
```

### **Check the Configuration for errors:**

```
$ sudo nginx -t
```

### **Restart the Nginx server:**

```
$ sudo systemctl restart nginx
```

## **Setting up Moodle on Ubuntu 20.04**

For example – **http://your-domain.com** or **http://your-ip.com**

# Installation

## Language

### Choose a language

Please choose a language for the installation. This language will also be used as the default language for the site, though it may be changed later.

Language

English (en)

Next »



Click on the **Next** button.

A screenshot of a web browser window showing the Moodle installation process. The address bar displays "yourdomain.com/install.php". A light blue informational box contains text about directory permissions. Below this, there are three input fields: "Web address" with "http://yourdomain.com", "Moodle directory" with "/var/www/html/moodle", and "Data directory" with "/var/www/html/moodledata". At the bottom of the form are two buttons: "« Previous" and "Next »". The Moodle logo is visible at the bottom right of the page.

← → ↻ yourdomain.com/install.php ☆

This directory should be both readable and writeable by the web server user (usually 'www-data', 'nobody', or 'apache').

It must not be directly accessible over the web.

If the directory does not currently exist, the installation process will attempt to create it.

Web address

Moodle directory

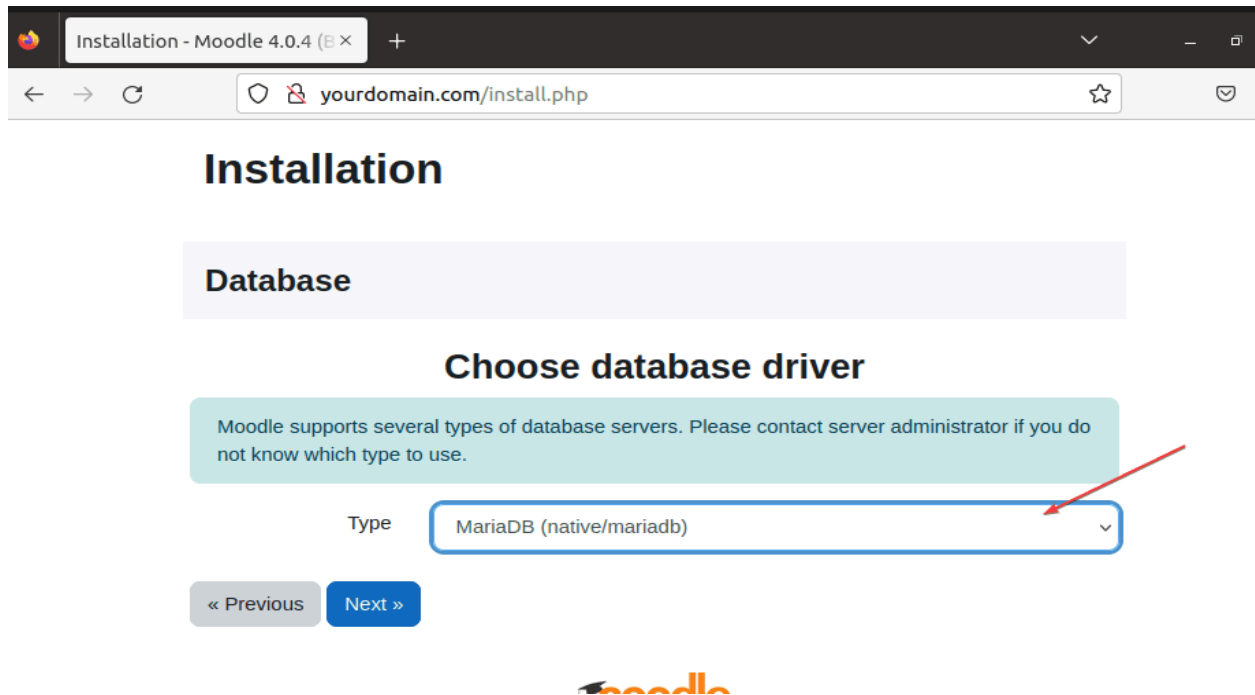
Data directory

« Previous

The Moodle logo, featuring a graduation cap icon and the word "moodle" in a stylized orange font.

Choose the Database driver.





The screenshot shows the Moodle 4.0.4 installation interface in a web browser. The address bar shows 'yourdomain.com/install.php'. The page title is 'Installation - Moodle 4.0.4'. The main heading is 'Installation'. Below it is a section titled 'Database'. The sub-heading is 'Choose database driver'. A light blue box contains the text: 'Moodle supports several types of database servers. Please contact server administrator if you do not know which type to use.' Below this is a dropdown menu labeled 'Type' with 'MariaDB (native/mariadb)' selected. A red arrow points to the dropdown arrow. At the bottom are two buttons: '« Previous' and 'Next »'.

## Installation

### Database

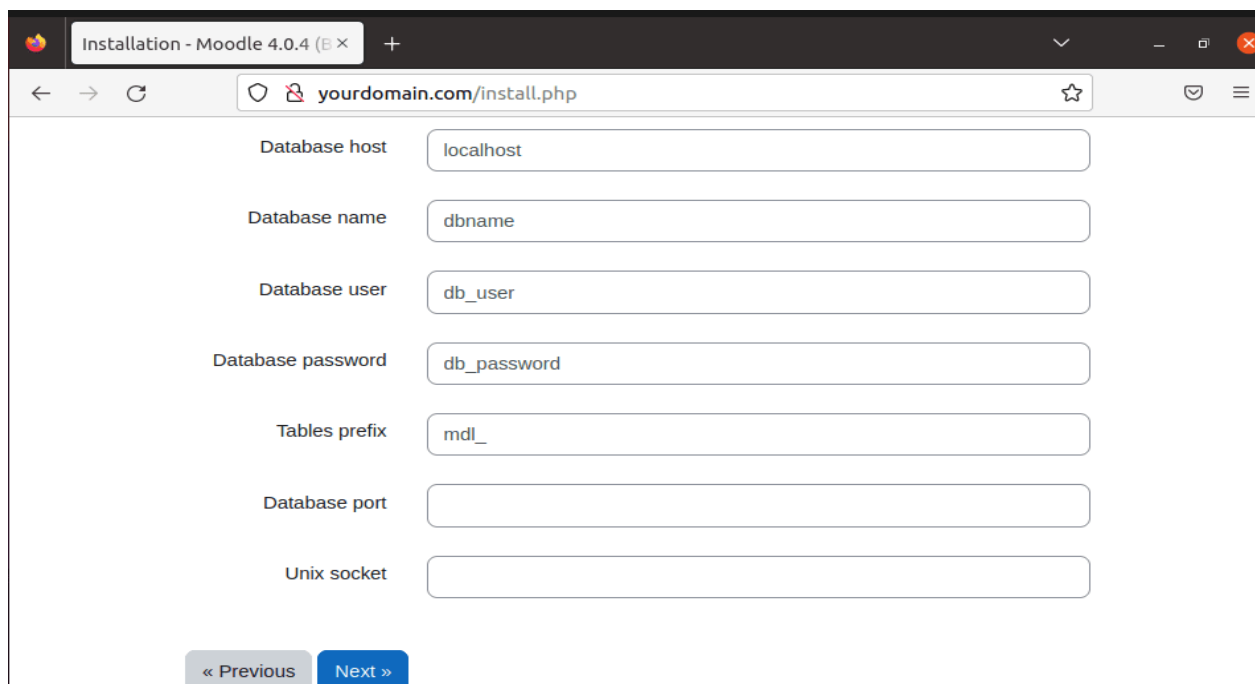
#### Choose database driver

Moodle supports several types of database servers. Please contact server administrator if you do not know which type to use.

Type: MariaDB (native/mariadb)

« Previous Next »

In this step, the setup will ask for the details of the Database such as the Database name, user, and password. Provide them and after that click on **Next**.



The screenshot shows the next step in the Moodle 4.0.4 installation. The address bar shows 'yourdomain.com/install.php'. The page title is 'Installation - Moodle 4.0.4'. The form contains several input fields for database configuration: 'Database host' (localhost), 'Database name' (dbname), 'Database user' (db\_user), 'Database password' (db\_password), 'Tables prefix' (mdl\_), 'Database port' (empty), and 'Unix socket' (empty). At the bottom are two buttons: '« Previous' and 'Next »'.

Database host:

Database name:

Database user:

Database password:

Tables prefix:

Database port:

Unix socket:

« Previous Next »

## Configure the Moodle's System Paths

- ❖ Configuring the system paths for Moodle, while optional, enhances your Moodle site's performance.
  - Navigate again to your Moodle website (example.com/moodle) and select **Site Administration** from the left menu.
  - Click the **Server** tab and select **System paths**.

On the form presented, complete the fields as follows:

- **Path to du**, enter `/usr/bin/du`.
- **Path to aspell**, enter `/usr/bin/aspell`.
- **Path to dot**, enter `/usr/bin/dot`.

### System paths

Path to PHP CLI  
pathtophp

Default: Empty

Path to PHP CLI. Probably something like `/usr/bin/php`. If you enter this, cron scripts can be executed from admin web interface.

Path to du  
pathtodu

`/usr/bin/du`

✓ Default: Empty

Path to du. Probably something like `/usr/bin/du`. If you enter this, pages that display directory contents will run much faster for directories with a lot of files.

Path to aspell  
aspellpath

`/usr/bin/aspell`

✓ Default: Empty

To use spell-checking within the editor, you MUST have **aspell 0.50** or later installed on your server, and you must specify the correct path to access the aspell binary. On Unix/Linux systems, this path is usually `/usr/bin/aspell`, but it might be something else.

Path to dot  
pathtodot

`/usr/bin/dot`

✓ Default: Empty

Path to dot. On Linux it is something like `/usr/bin/dot`. On Windows it is something like `C:\Program Files (x86)\Graphviz2.38\bin\dot.exe`. On Mac it is something like `/opt/local/bin/dot`. To be able to generate graphics from DOT files, you must have installed the dot executable and point to it here.

Path to ghostscript  
pathtogs

`/usr/bin/gs`

✓ Default: `/usr/bin/gs`

On most Linux installs, this can be left as `'/usr/bin/gs'`. On Windows it will be something like `'c:\gs\bin\gswin32c.exe'` (make sure there are no spaces in the path - if necessary copy the files `'gswin32c.exe'` and `'gsdll32.dll'` to a new folder without a space in the path)

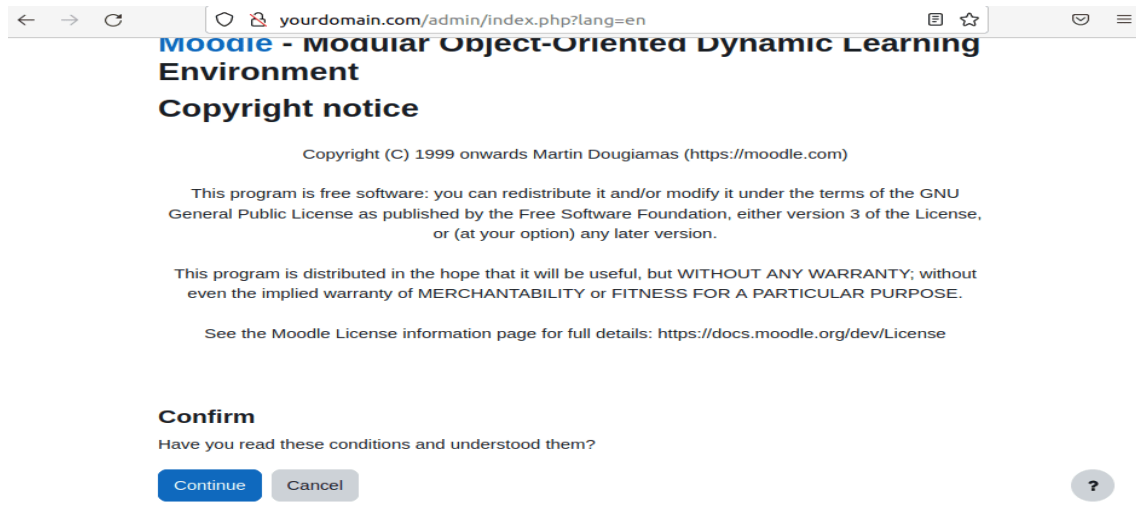
Path to Python  
pathtopython

Default: Empty

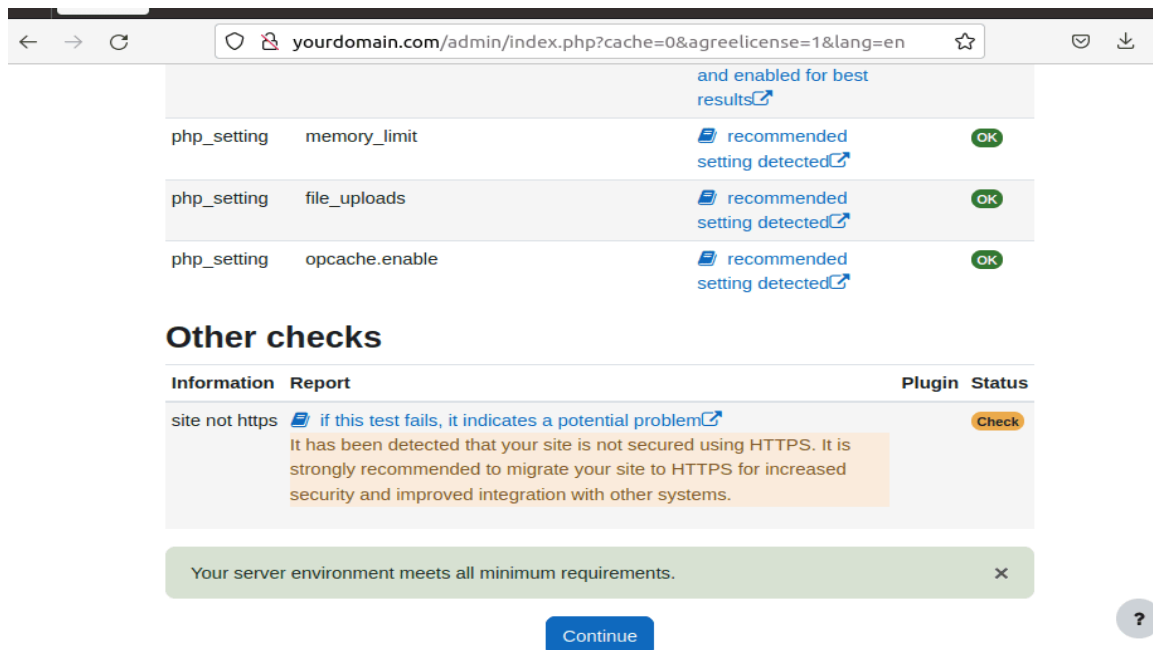
Path to your executable Python binary (both Python 2 and Python 3 are acceptable).

Save changes

Click on the “Continue” button.



The installation wizard of Moodle will check all the requirements on Ubuntu 20.04.





Click on the Continue button.

Create an Admin account.




[Expand all](#)


▼ **General**


Username 

Choose an authentication method  Manual accounts

The password must have at least 8 characters, at least 1 digit(s), at least 1 lower case letter(s), at least 1 upper case letter(s), at least 1 special character(s) such as as \*, -, or #

New password    

☐ Force password change 

First name 

## Set Up an SSL Certificate

- ❖ Although not required, it is recommended that you secure your Moodle site using an SSL certificate. This guide uses [Certbot](#) to create a free certificate from [Let's Encrypt](#).

Remove any existing Certbot installation.

**\$ sudo apt remove certbot**

Install Certbot.

**\$ sudo snap install --classic certbot**

- ★ Link the certbot command from the snap install directory to your path, so you'll be able to run it by just typing certbot

**\$ sudo ln -s /snap/bin/certbot /usr/bin/certbot**

Download a certificate for your site.

**\$ sudo certbot --nginx -d yourdomain.com -d www.yourdomain.com**

- Certbot prompts you to enter your site's domain name. Do so to complete the installation of the SSL certificate.
- Certbot includes a cron job that automatically renews your certificate before it expires. You can test the automatic renewal with the following command

**\$ sudo certbot renew --dry-run**

- ❖ Using your preferred text editor, open the Moodle configuration file, `/var/www/html/moodle/config.php`. Change the `$CFG->wwwroot` value to use https instead of http. Replace `example.com` in the following example with your site's domain name.

**`$CFG->wwwroot = 'https://yourdomain.com/moodle';`**

Restart nginx server

**\$ systemctl restart nginx**

**\*\*\*\*\*DONE\*\*\*\*\***