Phase 1 (Digital Ocean + Lamp):
Get a remote Linux server running LAMP stack on Digital Ocean

Notes for LAMP stack creation using Digital Ocean – Instructions to recreate the color app

You will need a digitalocean.com account for this. I suggest being logged onto the account before you begin. You will also need to have purchased a domain. I am using COP4331-5.com since I already own it and am not using it. .xyz domains are inexpensive, so you might want to consider those.

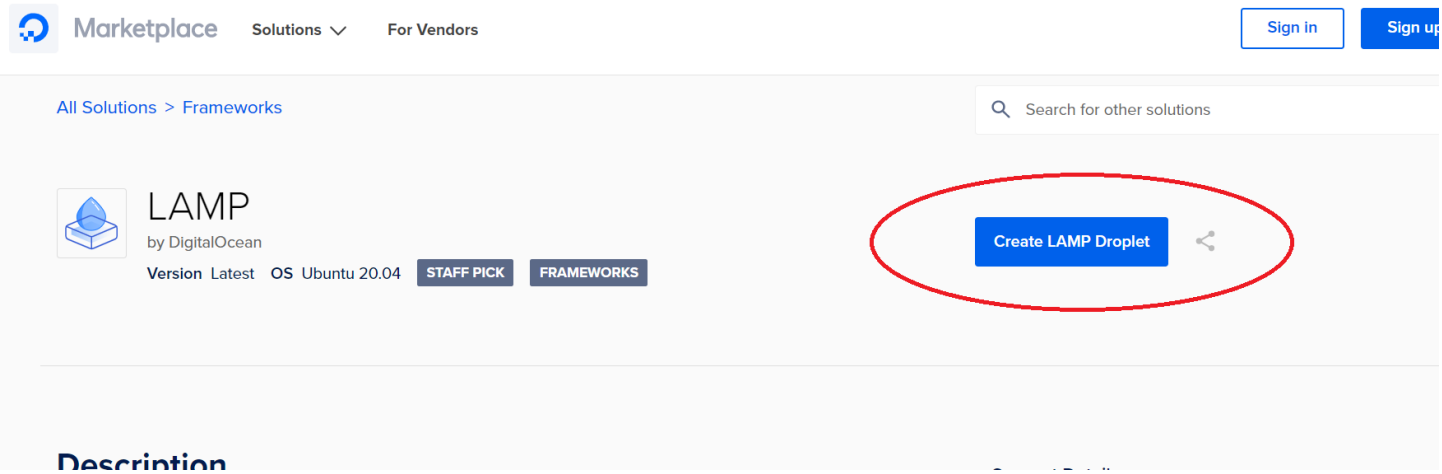
The digital ocean hosting will cost $7 per month, and you will need it for two months. The domain will cost something, too.

Hosting

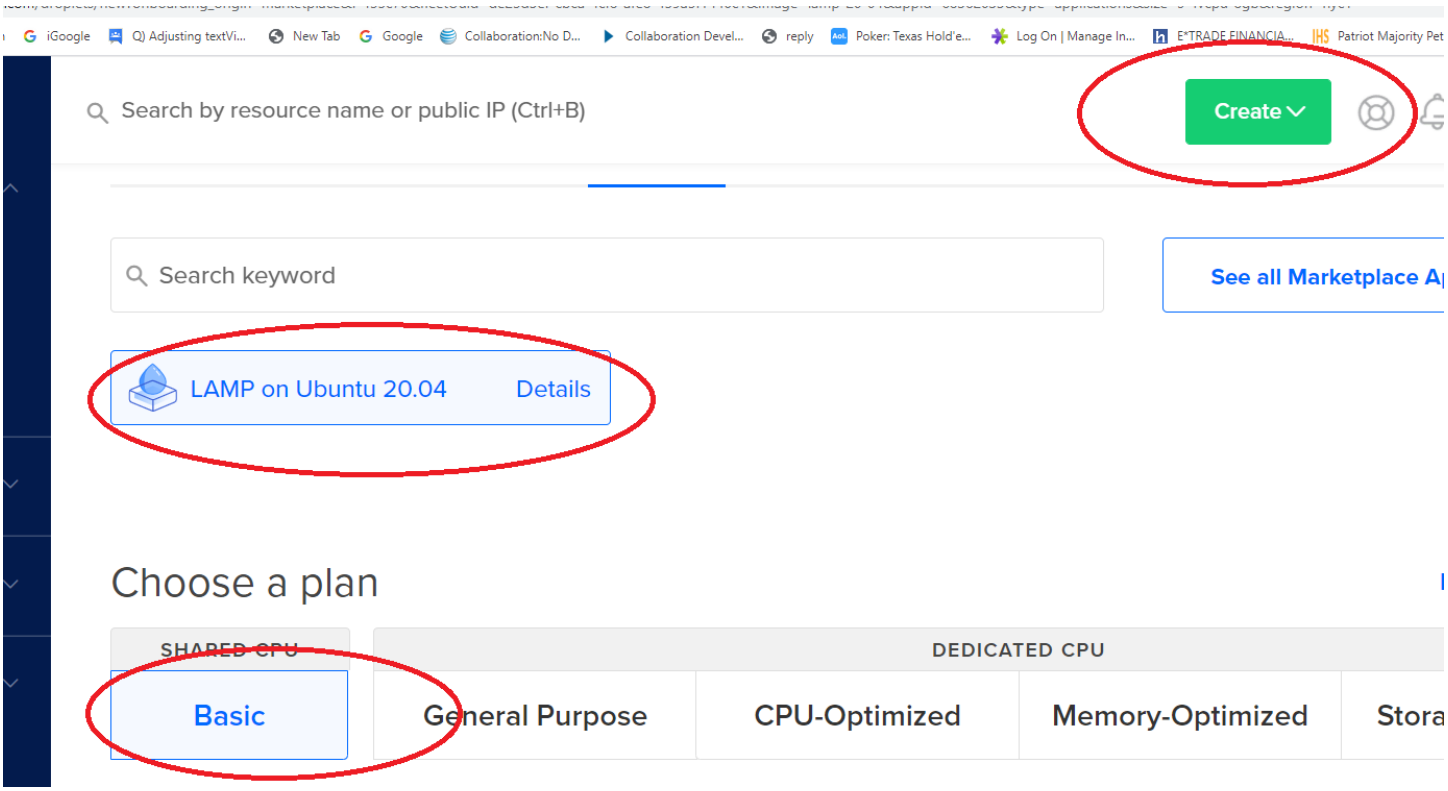
(Note to self, sign in to Digital Ocean before going to the following link. Also sign in to GoDaddy. Run Postman.)

Make sure you are logged in to your Digital Ocean account. Go to
https://marketplace.digitalocean.com/apps/lamp

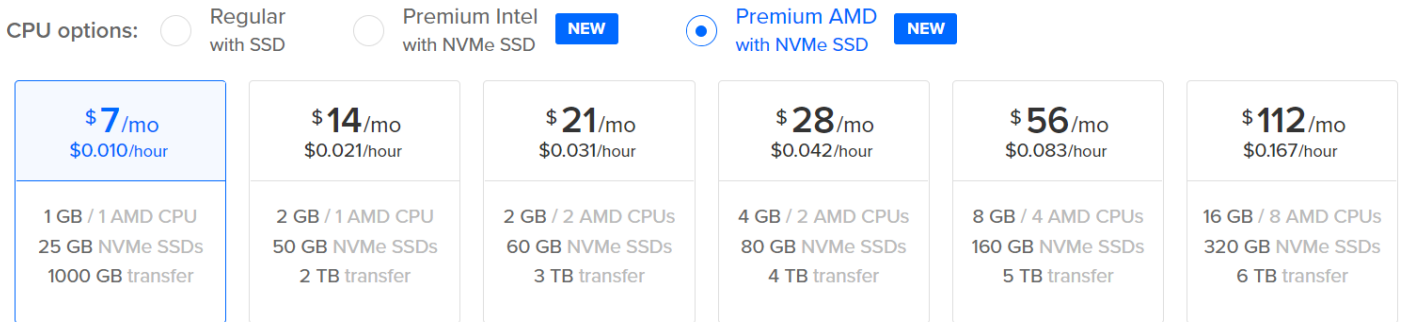
Create a LAMP Droplet



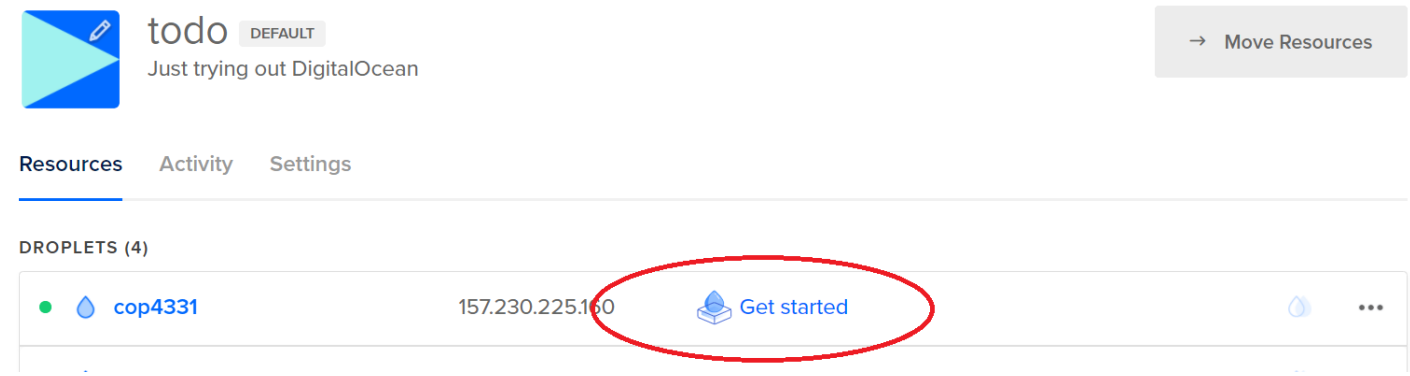
Select Ubuntu, Basic Plan, and the *Create*



You will need to select the $5 (now $7) / mo plan. Also, create a root password (suggest writing it down somewhere). The droplet will then be provisioned while you wait several minutes. (You can optionally edit the hostname.)



Once done, click **Get started**



Open command prompt and run SSH

ssh root@MyDomainOrIPAddress

Install Putty in order to easily SSH in to your droplet

Then enter your password.

Please note that everything you essentially need is already installed in the droplet. This includes MySQL Apache, and PHP.

Navigate to the root – **cd /root**

The web root is in /var/www/html – Go to that directory now with **cd /var/www/html**

View the contents of the directory with **ls**

View the contents of index.html with **cat index.html** – it's a lot

Now we will edit the contents of index.html – open for editing with **vi index.html**

(Note to self, change the SSH font. Right Click, Properties, Size, OK Button)

You can highlight and delete a block by positioning the cursor at the top of the block and pressing Shift-V, cursoring down to the end of the block and pressing d

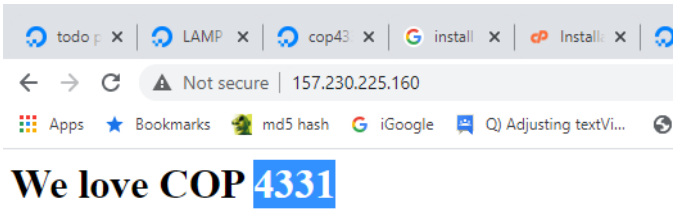
You need to add the <body> and be in insert mode, so press the insert key

Your index.html file should look like the following:

```
<html>
  <body>
    <h1>We love COP 4331</h1>
  </body>
</html>
```

To save and quit hit the **escape key** (to get out of insert mode), type **:wq** – now verify the edit with **cat index.html**

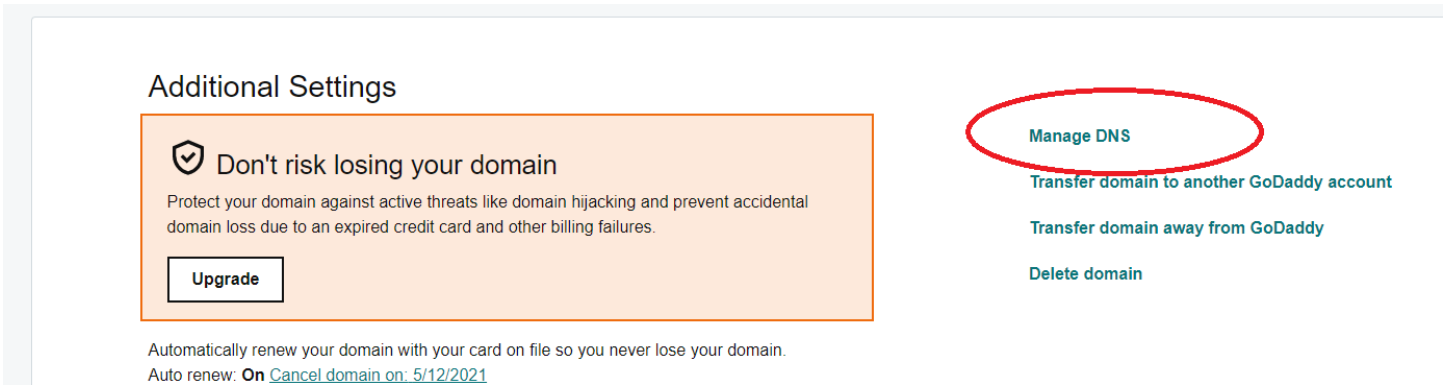
You can access this via a web browser. Open a browser and type in your http://IP address.



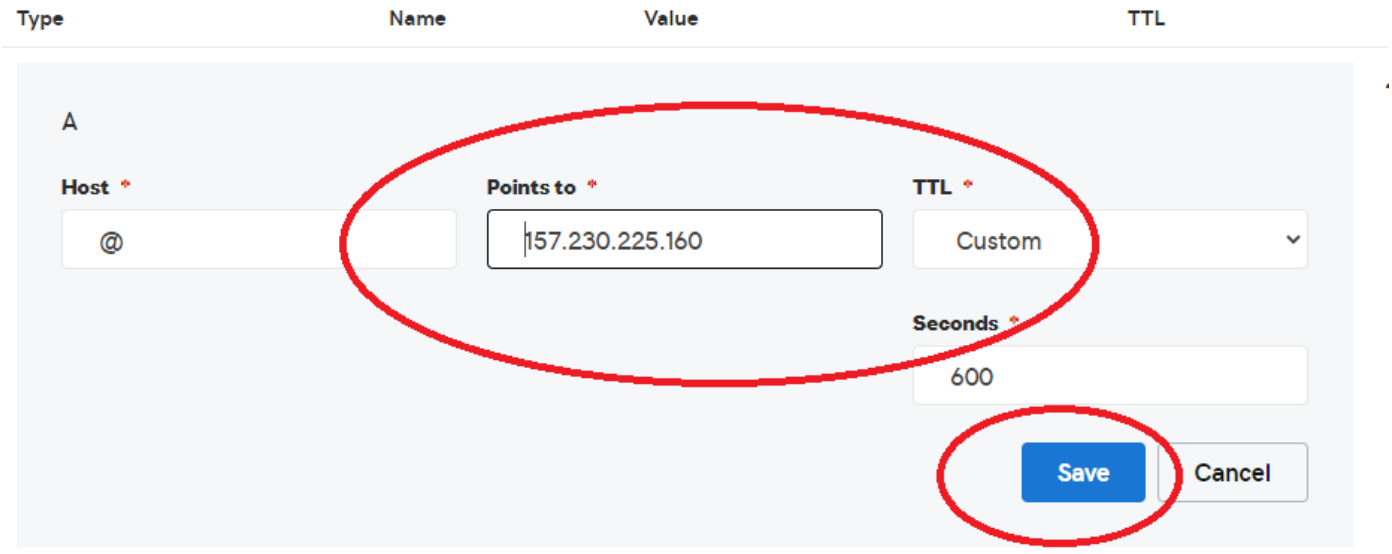
Now for a domain. You cannot buy a domain through digital ocean (that I know of, but that may have changed). Choose another domain registrant. I use GoDaddy, but there are lots of them. Purchase a domain and point the domain to your digital ocean applet. Below are the steps I took on GoDaddy.

1. I already had this domain purchased:

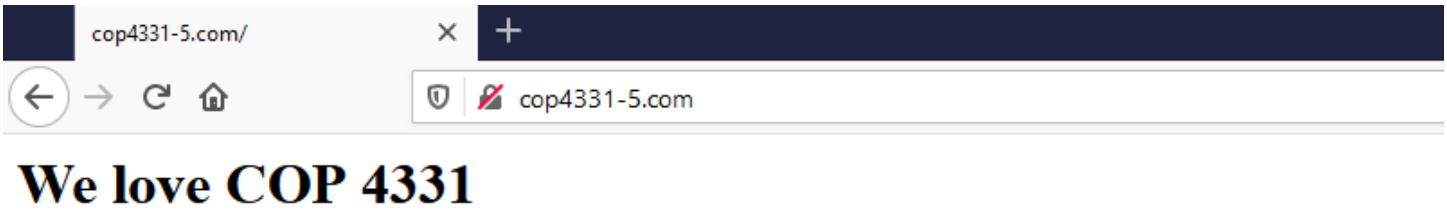
2. Navigate to the DNS manager:



3. Edit the IP address and save:



4. Test with a browser. It might take a few minutes to propagate. (On Windows it is helpful to go to a command prompt and type **ipconfig /flushdns**) You might also want to use Ctrl-F5 to hard reset the web content.



Phase 2 (Database):
Create the MySQL database used by the COLORS app and populate it.

Option: Connect to MySQL: **mysql -u root -p** (then enter your password)

Note: There are three levels of password you will need to think about. This is the first, which gives you access to MySQL from the command line.

Here are the steps to create the database, tables, and working data.

1. Create database

create database COP4331;

use COP4331;

2. Create tables

CREATE TABLE `COP4331`.`Users` (`ID` INT NOT NULL AUTO_INCREMENT , `FirstName` VARCHAR(50) NOT NULL DEFAULT '' , `LastName` VARCHAR(50) NOT NULL DEFAULT '' , `Login` VARCHAR(50) NOT NULL DEFAULT '' , `Password` VARCHAR(50) NOT NULL DEFAULT '' , PRIMARY KEY (`ID`)) ENGINE = InnoDB;

```
CREATE TABLE `COP4331`.`Users`
(
	`ID` INT NOT NULL AUTO_INCREMENT ,
	`FirstName` VARCHAR(50) NOT NULL DEFAULT '' ,
	`LastName` VARCHAR(50) NOT NULL DEFAULT '' ,
	`Login` VARCHAR(50) NOT NULL DEFAULT '' ,
	`Password` VARCHAR(50) NOT NULL DEFAULT '' ,
	PRIMARY KEY (`ID`)
) ENGINE = InnoDB;
```

CREATE TABLE `COP4331`.`Colors` (`ID` INT NOT NULL AUTO_INCREMENT , `Name` VARCHAR(50) NOT NULL DEFAULT '' , `UserID` INT NOT NULL DEFAULT '0' , PRIMARY KEY (`ID`)) ENGINE = InnoDB;

CREATE TABLE `COP4331`.`Colors`

```
(
                `ID` INT NOT NULL AUTO_INCREMENT ,
                `Name` VARCHAR(50) NOT NULL DEFAULT '' ,
                `UserID` INT NOT NULL DEFAULT '0' ,
                PRIMARY KEY (`ID`)
) ENGINE = InnoDB;
```

```
CREATE TABLE `COP4331`.`Contacts`
(
                `ID` INT NOT NULL AUTO_INCREMENT ,
                `FirstName` VARCHAR(50) NOT NULL DEFAULT '' ,
                `LastName` VARCHAR(50) NOT NULL DEFAULT '' ,
                `Phone` VARCHAR(50) NOT NULL DEFAULT '' ,
                `Email` VARCHAR(50) NOT NULL DEFAULT '' ,
                `UserID` INT NOT NULL DEFAULT '0' ,
                PRIMARY KEY (`ID`)
) ENGINE = InnoDB;
```

3. Populate working data rows

```
USE COP4331;

insert into Users (FirstName,LastName,Login,Password) VALUES
('Aashish','Yadavally','AYadavally','COP4331');

insert into Users (FirstName,LastName,Login,Password) VALUES ('Sam','Hill','SamH','Test');

insert into Users (FirstName,LastName,Login,Password) VALUES
('Aashish','Yadavally','AYadavally','5832a71366768098cceb7095efb774f2');

insert into Users (FirstName,LastName,Login,Password) VALUES
('Sam','Hill','SamH','0cbc6611f5540bd0809a388dc95a615b');

insert into Colors (Name,UserID) VALUES ('Blue',1);

insert into Colors (Name,UserID) VALUES ('White',1);
```

```
insert into Colors (Name,UserID) VALUES ('Black',1);
insert into Colors (Name,UserID) VALUES ('gray',1);
insert into Colors (Name,UserID) VALUES ('Magenta',1);
insert into Colors (Name,UserID) VALUES ('Yellow',1);
insert into Colors (Name,UserID) VALUES ('Cyan',1);
insert into Colors (Name,UserID) VALUES ('Salmon',1);
insert into Colors (Name,UserID) VALUES ('Chartreuse',1);
insert into Colors (Name,UserID) VALUES ('Lime',1);
insert into Colors (Name,UserID) VALUES ('Light Blue',1);
insert into Colors (Name,UserID) VALUES ('Light Gray',1);
insert into Colors (Name,UserID) VALUES ('Light Red',1);
insert into Colors (Name,UserID) VALUES ('Light Green',1);
insert into Colors (Name,UserID) VALUES ('Chiffon',1);
insert into Colors (Name,UserID) VALUES ('Fuscia',1);
insert into Colors (Name,UserID) VALUES ('Brown',1);
insert into Colors (Name,UserID) VALUES ('Beige',1);

insert into Colors (Name,UserID) VALUES ('Blue',3);
insert into Colors (Name,UserID) VALUES ('White',3);
insert into Colors (Name,UserID) VALUES ('Black',3);
insert into Colors (Name,UserID) VALUES ('gray',3);
insert into Colors (Name,UserID) VALUES ('Magenta',3);
insert into Colors (Name,UserID) VALUES ('Yellow',3);
insert into Colors (Name,UserID) VALUES ('Cyan',3);
insert into Colors (Name,UserID) VALUES ('Salmon',3);
insert into Colors (Name,UserID) VALUES ('Chartreuse',3);
insert into Colors (Name,UserID) VALUES ('Lime',3);
insert into Colors (Name,UserID) VALUES ('Light Blue',3);
insert into Colors (Name,UserID) VALUES ('Light Gray',3);
insert into Colors (Name,UserID) VALUES ('Light Red',3);
insert into Colors (Name,UserID) VALUES ('Light Green',3);
```

insert into Colors (Name,UserID) VALUES ('Chiffon',3);

insert into Colors (Name,UserID) VALUES ('Fuscia',3);

insert into Colors (Name,UserID) VALUES ('Brown',3);

insert into Colors (Name,UserID) VALUES ('Beige',3);

Test with:

select * from Users;

select * from Colors;

also:

select * from Colors where UserID=1;

select * from Colors where UserID=3;

We will create a user:

Note: Remember, there are three levels of password you will need to think about. This is the second, which allows the webapp (the API code) to run queries against MySQL.

Use COP4331;

create user 'TheBeast' identified by 'WeLoveCOP4331';

Now we need to grant permissions to the database for that user:

grant all privileges on COP4331.* to 'TheBeast'@'%';

The database is ready to use.

Here: Talk about primary and foreign keys

The web directory hierarchy is as follows:

root (/var/www/html)

- css (/var/www/ html /css)
- images (/var/www/ html /images)

js (/var/www/ html /js)

LAMPAPI (/var/www/ html /LAMPAPI)

index.html

color.html

Navigate to /var/www/html

Create the directories

mkdir css

mkdir images

mkdir js

mkdir LAMPAPI

Phase 3 (Create API endpoints):
Implement the backend logic that enables full application functionality

There will be three API endpoints: **AddColor**, **Login**, and **SearchColors**. They each have a single .php file that is contained in the LAMPAPI directory.

Please note that there is a php statement that must be changed with your database username, password, and database name.

$conn = new mysqli("localhost", "username", "password", "database");

For our example that becomes

$conn = new mysqli("localhost", "TheBeast", "WeLoveCOP4331", "COP4331");

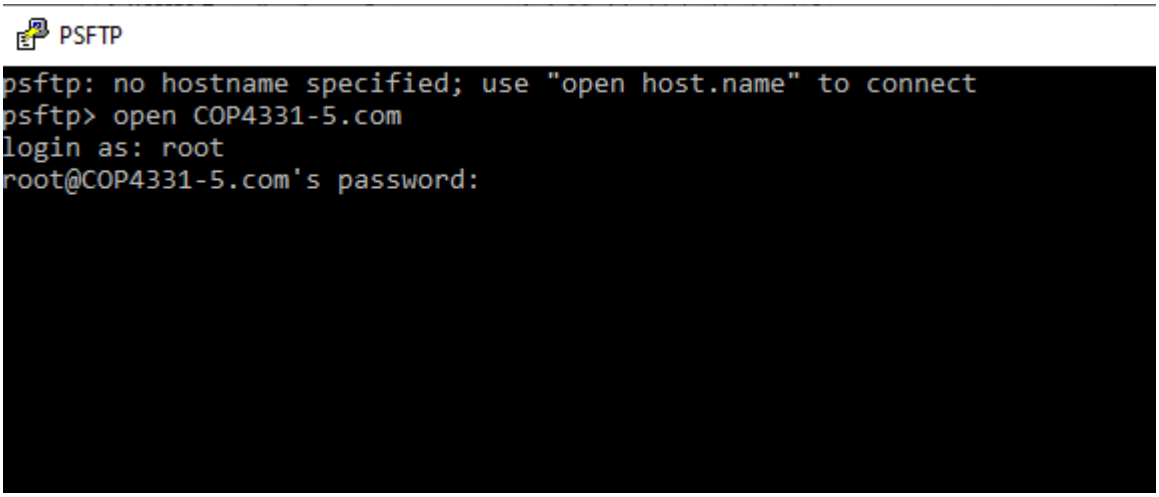
There are three example .php files in the stacks\LAMP\LAMPAPI path of the file on the webcourse.

Upload the .php files to the server. They will be placed into LAMPAPI.

Run PuTTY FTP. (PSFTP)

Type open COP4331-5.com (your domain name)

Enter username and password



Type cd /var/www/html

You can type ls to see the subdirectories

Enter the LAMPAPI directory (**cd LAMPAPI**)

Now you will upload the API endpoint files with the following:

put "C:\work\LAMP Stack\LAMPAPI\AddColor.php"

put "C:\work\LAMP Stack\LAMPAPI\Login.php"

put "C:\work\LAMP Stack\LAMPAPI\SearchColors.php"

Use the ls command to double check that the files have been uploaded

Remember that Linux is case sensitive for file names and directories

You can consider using FileZilla to do the FTP transfers.

Here: analysis of .php API endpoint files.

Now the API endpoints can be tested.

Use ARC or Postman or CURL or Swagger

(Make sure you are set for POST method and application/json content type)

http://cop4331-5.com/LAMPAPI/Login.php
login
password

http://cop4331-5.com/LAMPAPI/AddColor.php
userId
color

http://cop4331-5.com/LAMPAPI/SearchColors.php
userId
search

For your contacts you will want to return an array of JSON objects. I have a video here: https://www.youtube.com/watch?v=G7GTKjTLCSI that explains this.

Front End

Upload css, images, js directories. Also upload color.html and index.html

cd css

put "C:\work\LAMP Stack\css\styles.css"

cd ../images

put "C:\work\LAMP Stack\images\ background.png"

cd ../js

put "C:\work\LAMP Stack\js\code.js"

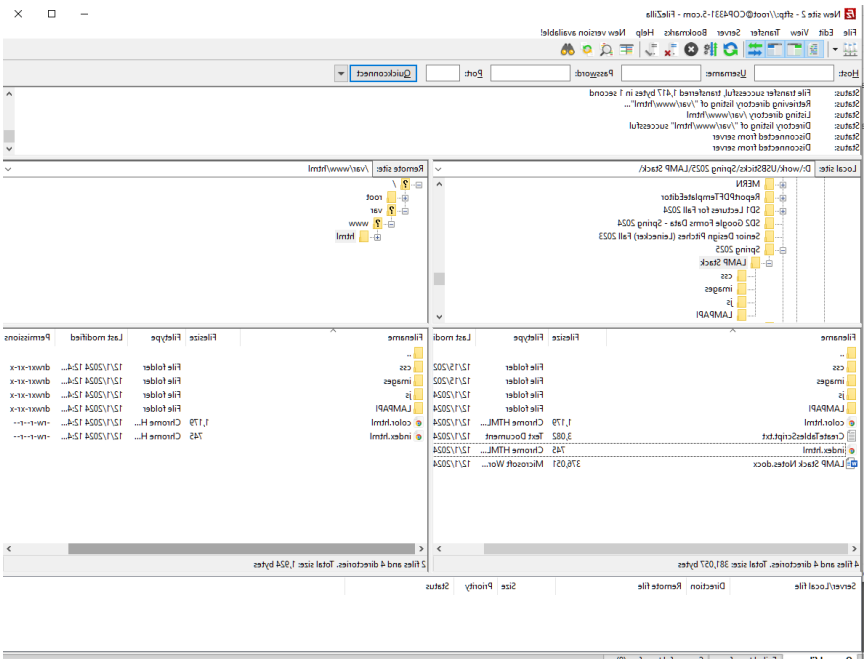
put "C:\work\LAMP Stack\js\md5.js"

cd ..

put "C:\work\LAMP Stack\index.html"

put "C:\work\LAMP Stack\color.html"

FileZilla is way easier.



Here: Analysis of all source code

Hashing passwords

insert into Users (FirstName,LastName,Login,Password) VALUES ('Aashish','Yadavally','AYadavally','5832a71366768098cceb7095efb774f2');

insert into Users (FirstName,LastName,Login,Password) VALUES ('Sam','Hill','SamH','0cbc6611f5540bd0809a388dc95a615b');

For an example of a previous project visit http://4331paradise.com/