Ghibli Web Store

USER MANUAL

**Contents**

1. Overview

About Website

1. Getting Started

Registration and Log-in

Adding to Cart and Checking-out

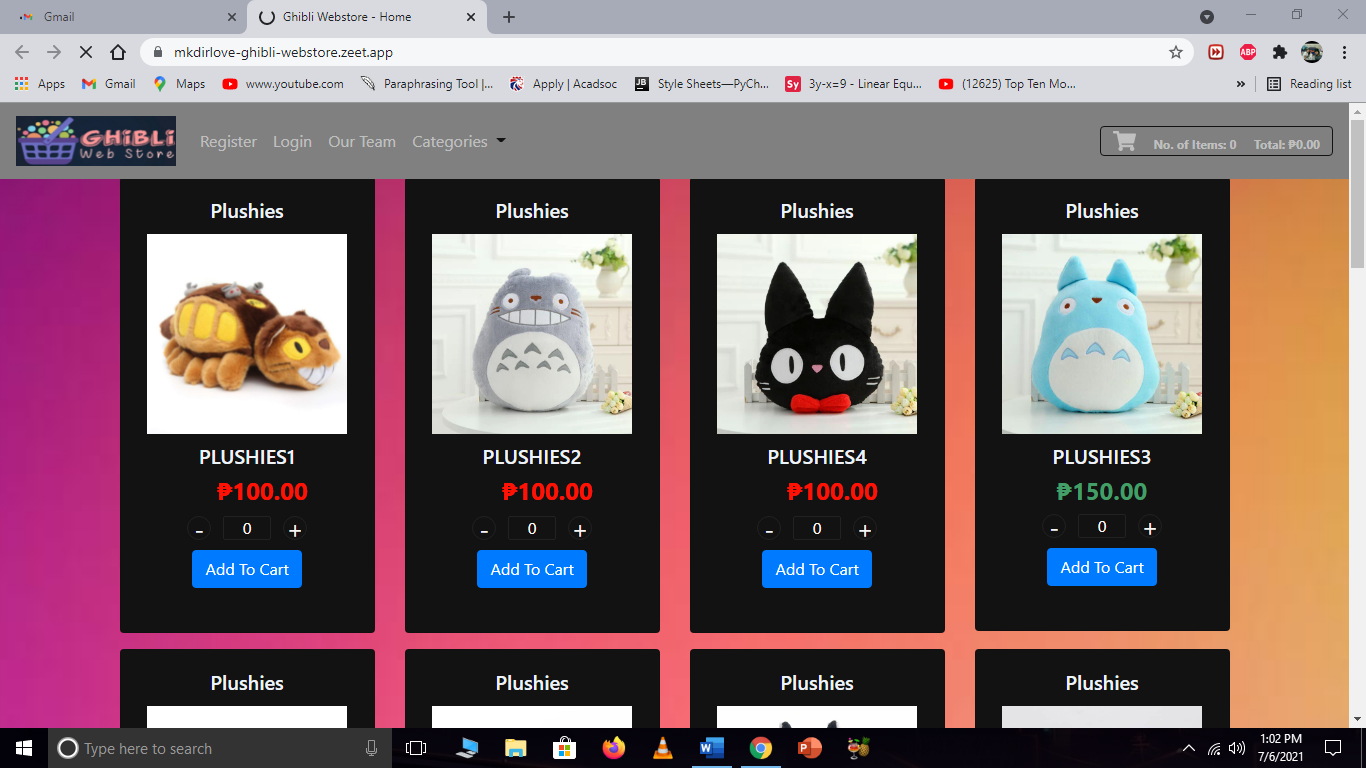
1. Setting Up Database

Installing for Windows

Installing for Linux

Installing for Mac OS

1. Connecting Python Flask Framework with Sqlite

**Overview**

Ghibli Web Store, a shopping website made for audience of all ages but specifically has a special place for Studio Ghibli fans. This website offers high quality merchandises at a customer-friendly price. Each merch was inspired from everything you love about Studio Ghibli, from the sceneries to the characters itself.

About Website

Developer: Jayson San Buenaventura

Designer: Abigail De Guia

Content and Details: Leomar Mangubat, Jasper Abella, John Maynard Elec

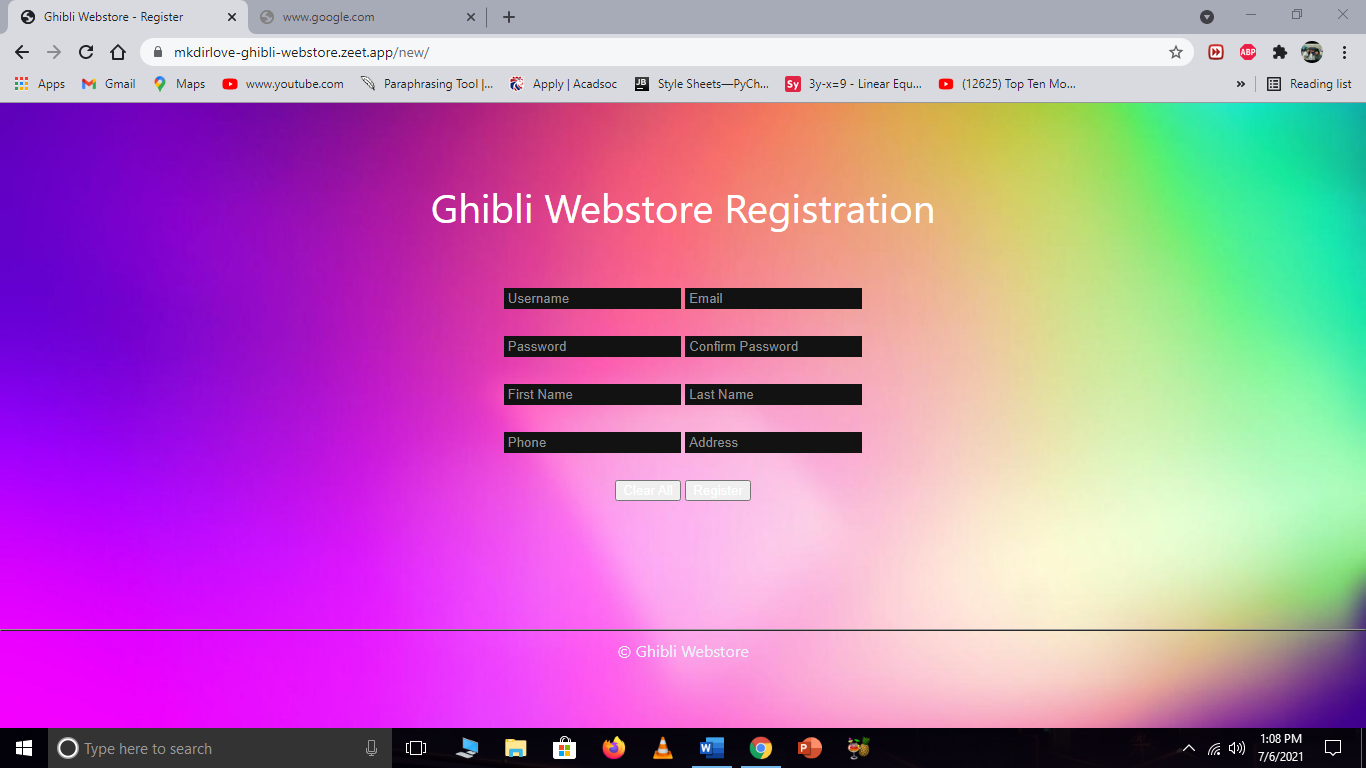
Date Started & Finished: June 2021 - July 2021

**Getting Started**

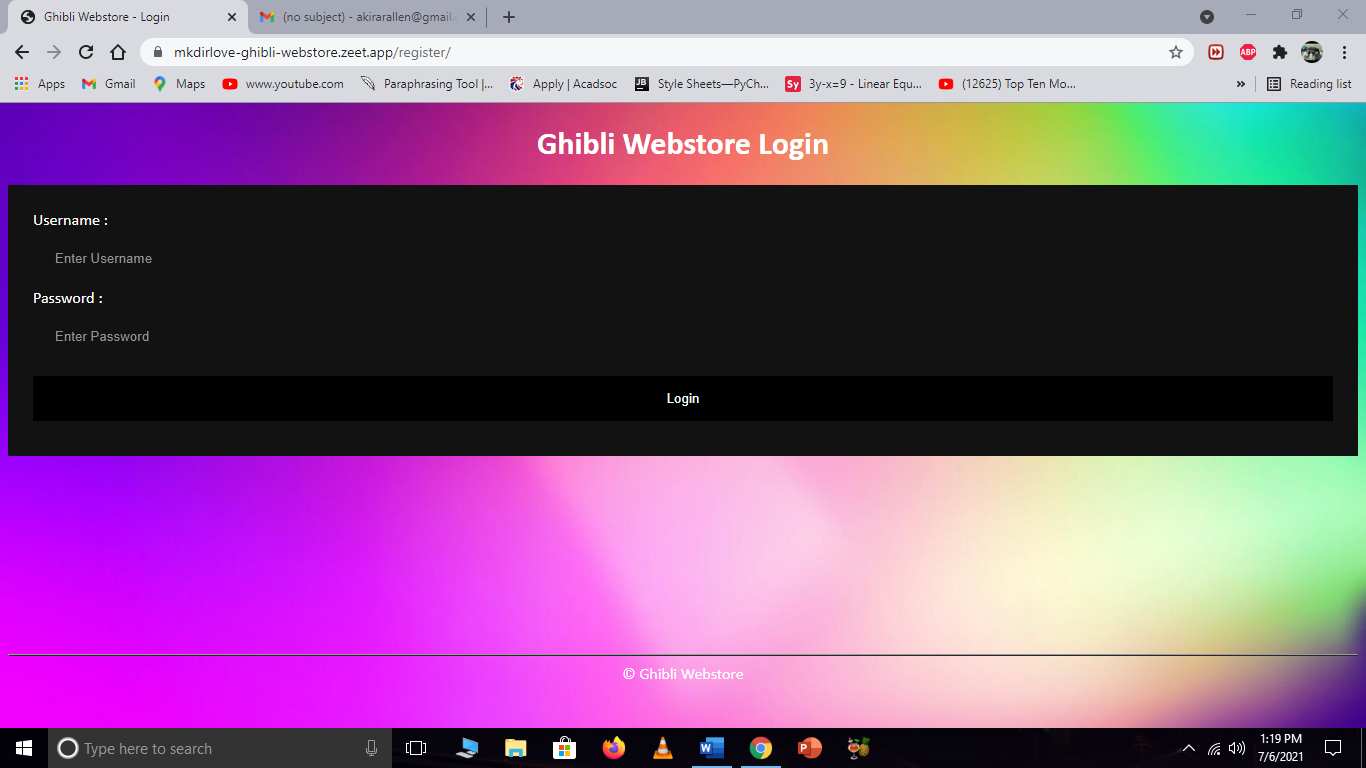
**Registration & Log in**

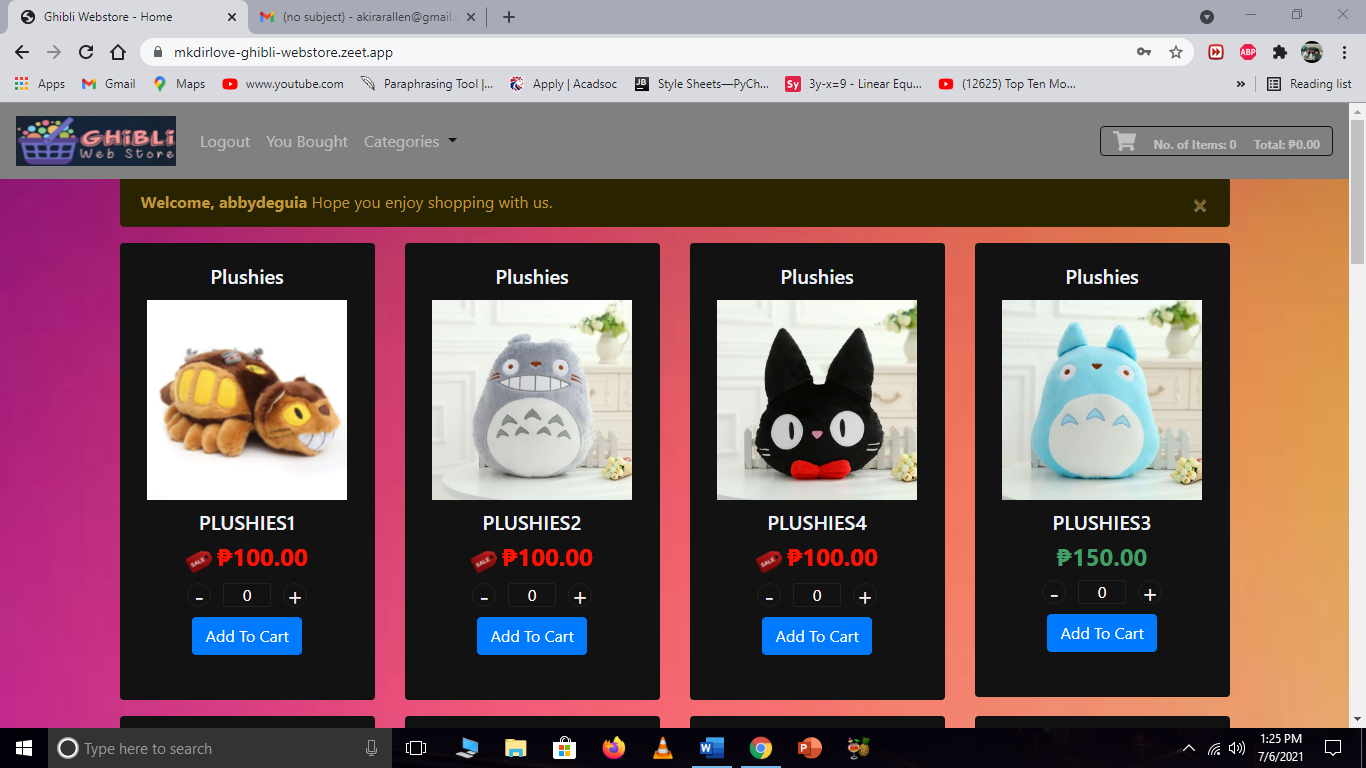
To be able to obtain an account to this website, you should create one first. To do it, follow the procedure;

1. First, you direct to the Registration webpage and fill up the necessary informations.



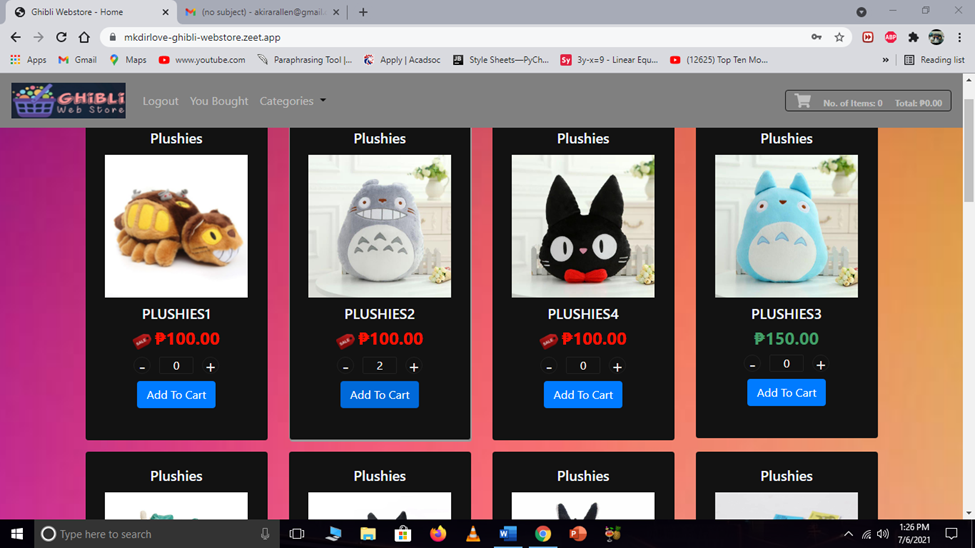
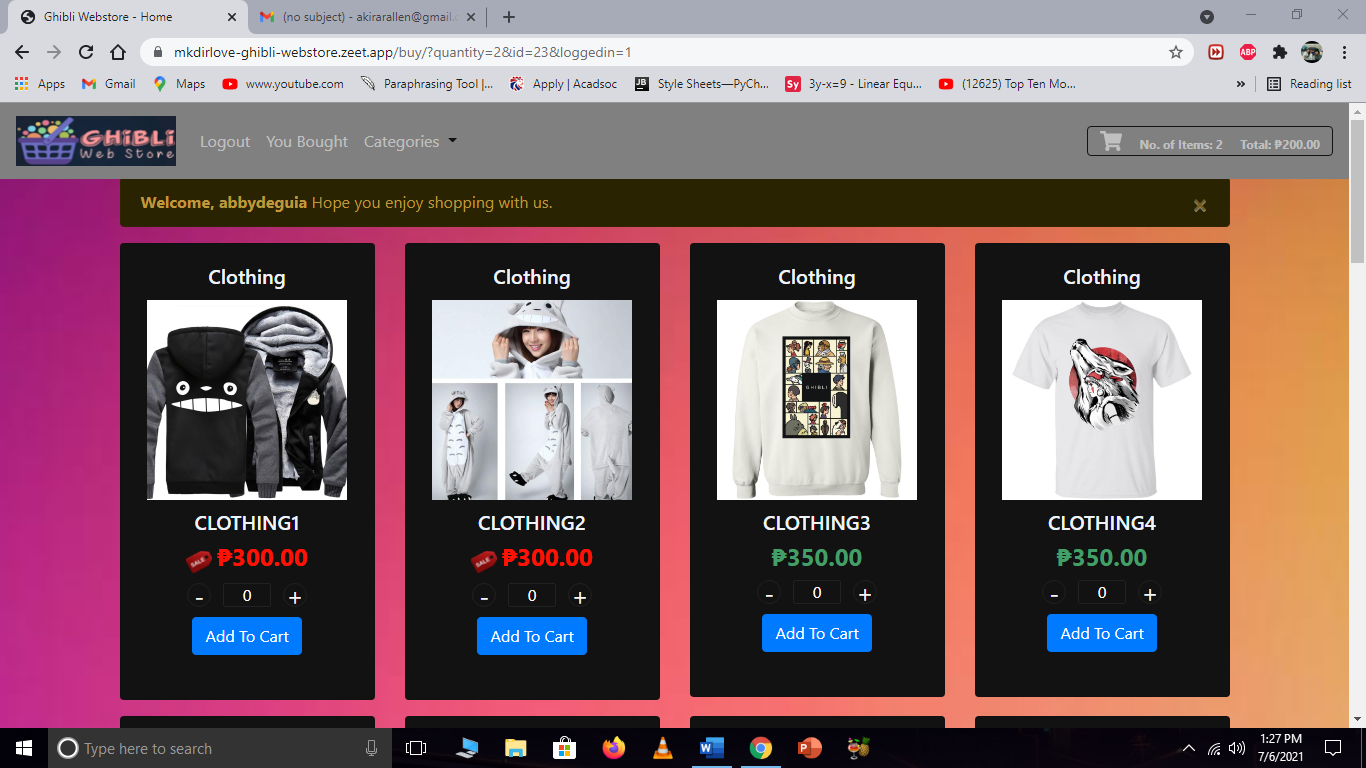
1. Remember your username and password.
2. Then click Log In.
3. After clicking the Log In button, it will direct you to another webpage called Log In. In this webpage you should fill up the username and password that you have filled up before on the creation of your account

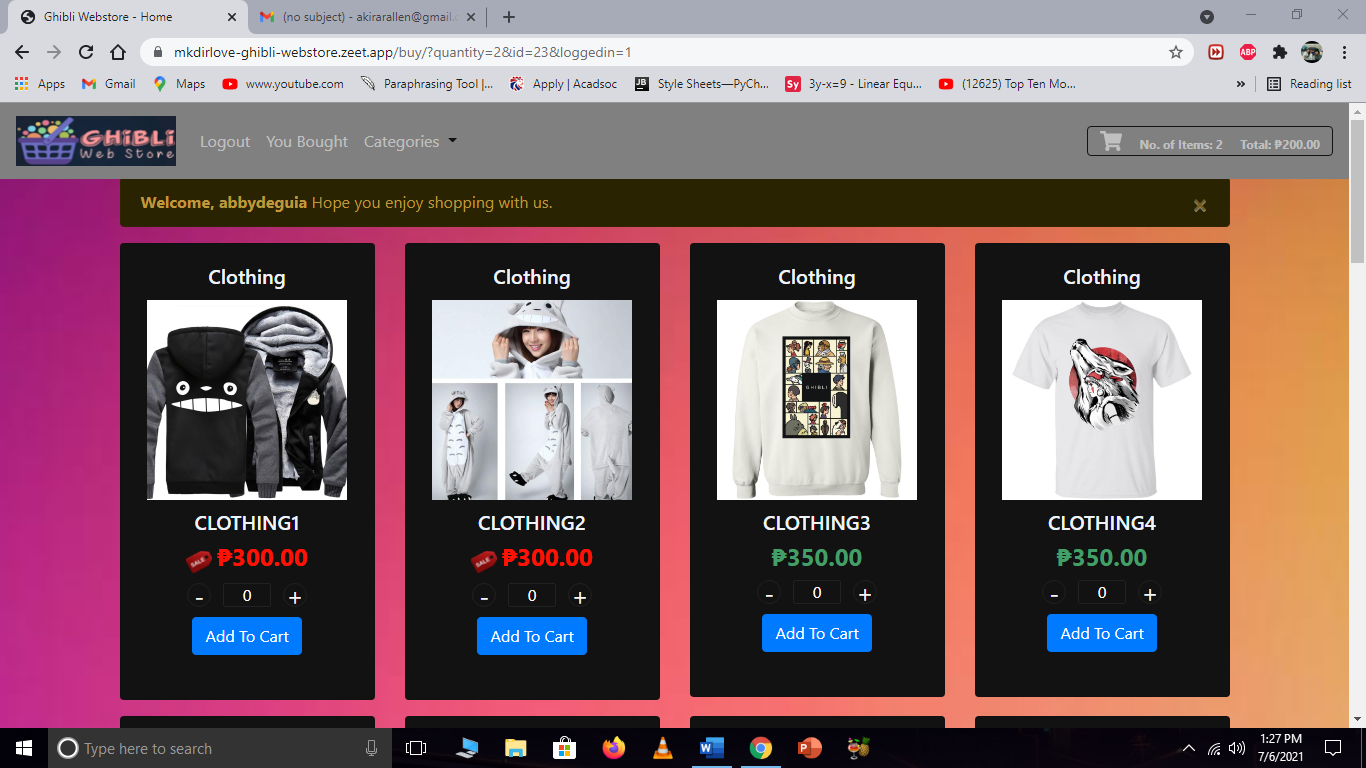


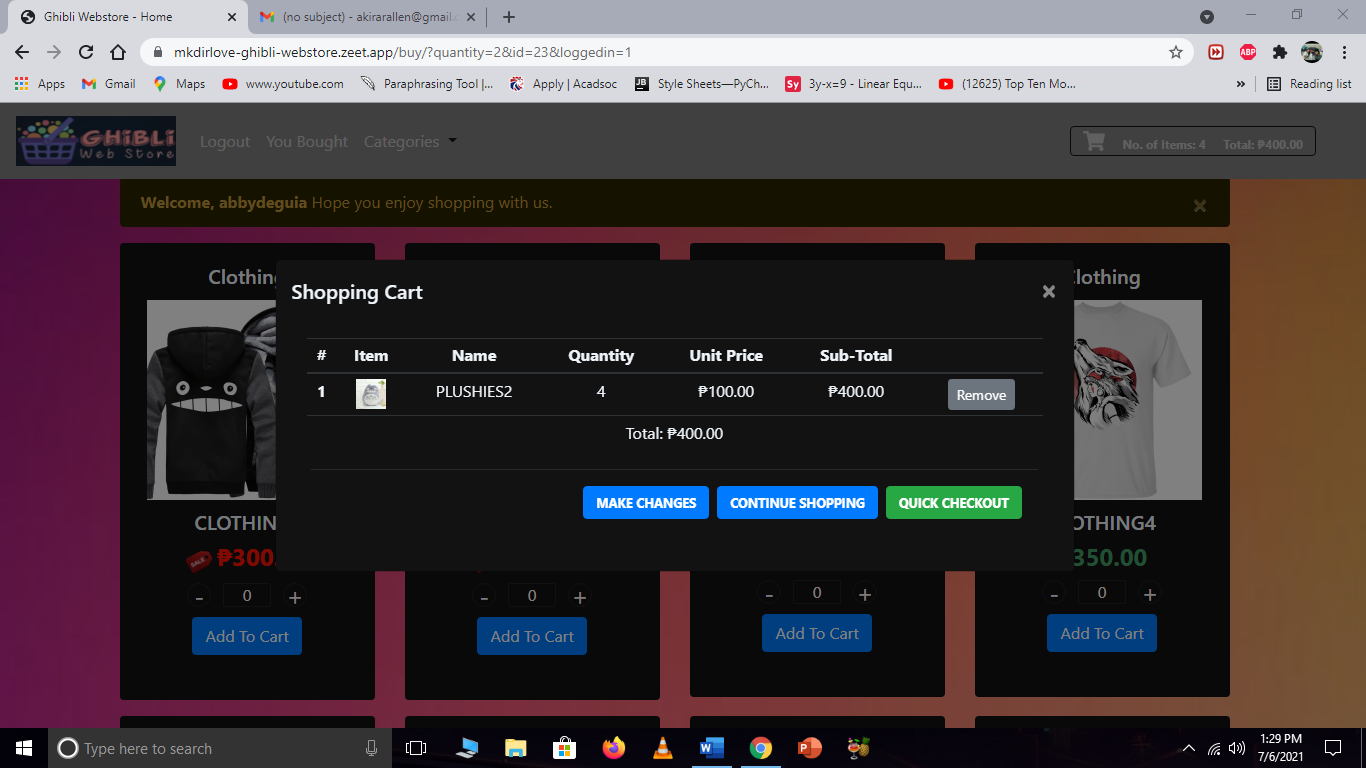
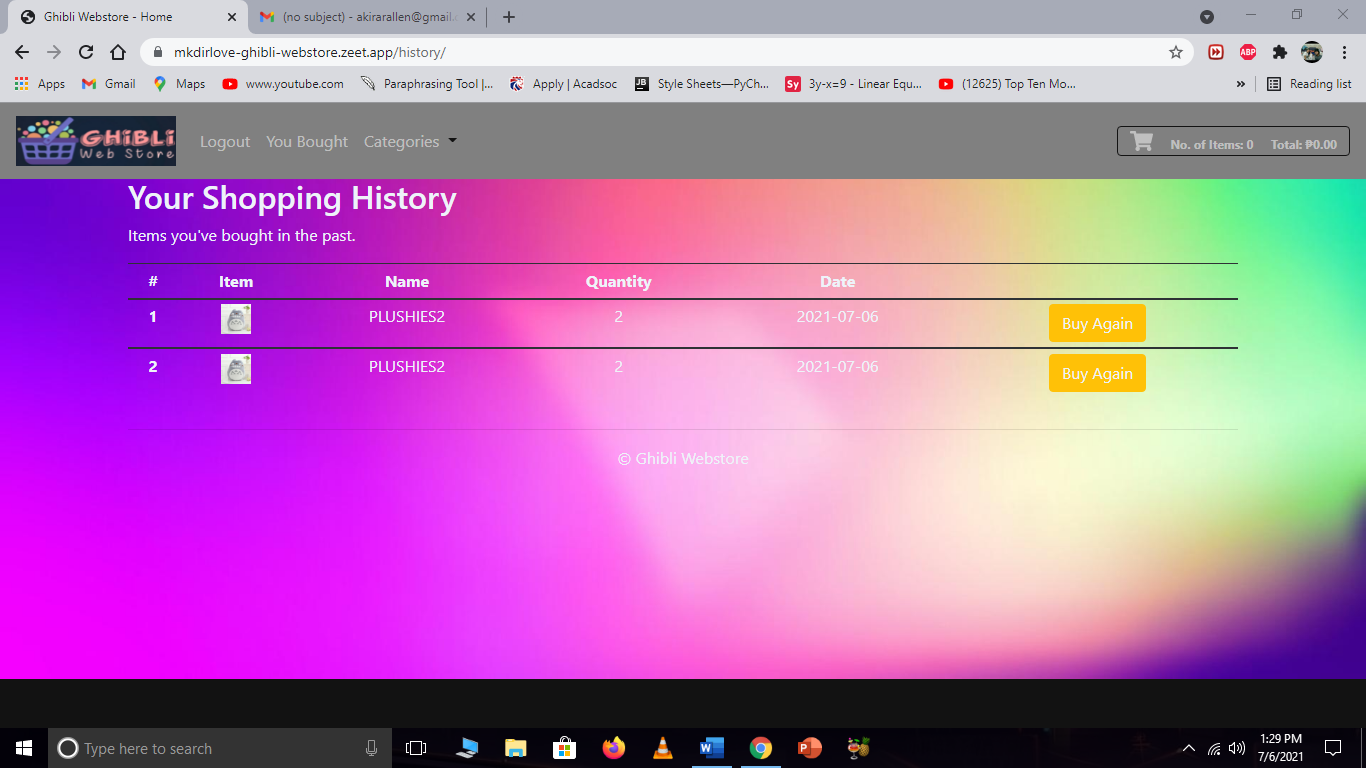
1. After providing needed informations, click the Log In button and it will direct you to the Shopper’s webpage.

**Adding to Cart and Checking Out**

If you have chosen your desired product and wanted to buy it, follow the procedures below;

1. Go to your desired products.
2. Click the plus sign **(+)** to select how many of that product you like.
3. Click on the Add to Cart button after.
4. After you click the button you will receive a notification.
5. Upon receiving the notification, click on to the upper right side of the site where the cart can be found.



1. After clicking it, you’ll be redirected to the Cart webpage.
2. Check if the order number and product is correct before deciding to checkout.
3. After this, click the checkout button and it will direct you to the Shopping History.
4. In shopping history, you can see your ordered item and you can click buy again if you want to search for more items to buy.

**Setting up Sqlite on Linux, Windows & Mac OS**

**Installing Sqlite for Windows**

Step 1. Go to SQLite download page: <https://www.sqlite.org/download.html> and download precompiled binaries from Windows section.

Step 2. Download sqlite-shell-win32-*.zip and sqlite-dll-win32-*.zip zipped files.

Step 3. Create a folder with name C:>sqlite and unzip above two zipped files in this folder, which will give you sqlite3.def, sqlite3.dll and sqlite3.exe files.

Step 4. Add C:>sqlite in your PATH environment variable and finally go to the command prompt and issue sqlite3 command, which should display the following result.

C:\>sqlite3

SQLite version 3.7.15.2 2013-01-09 11:53:05

Enter ".help" for instructions

Enter SQL statements terminated with a ";"

sqlite>

**Installing Sqlite for Linux**

Today, almost all the flavors of Linux OS are being shipped with SQLite. So, to make sure if Sqlite is already in your computer, follow the procedures.

$sqlite3

SQLite version 3.7.15.2 2013-01-09 11:53:05

Enter ".help" for instructions

Enter SQL statements terminated with a ";"

sqlite>

If you do not see the above result, then it means you do not have SQLite installed on your Linux machine.

Following are the steps to install SQLite –

Step 1. Open your Terminal Emulator.

Step 2. Run the following command –

$sudo apt update && sudo apt install sqlite3

**Installing Sqlite for Linux from Source**

In this part, you just issue the following command to check if you already have SQLite installed on your machine.

$sqlite3

SQLite version 3.7.15.2 2013-01-09 11:53:05

Enter ".help" for instructions

Enter SQL statements terminated with a ";"

sqlite>

If you do not see the above result, then it means you do not have SQLite installed on your Linux machine. Below are the following steps to install SQLite −

Step 1 − Go to SQLite download page: <https://www.sqlite.org/download.html> and download sqlite-autoconf-\*.tar.gz from source code section.

Step 2 − Run the following command −

$tar xvfz sqlite-autoconf-3071502.tar.gz

$cd sqlite-autoconf-3071502

$./configure --prefix=/usr/local

$make

$make install

The above command will end with SQLite installation on your Linux machine.

**Installing Sqlite on Mac OS**

Though the latest version of Mac OS X comes pre-installed with SQLite but if you do not have installation available then just follow these following steps –

Step 1. Go to SQLite download page: <https://www.sqlite.org/download.html> and download sqlite-autoconf-\*.tar.gz from source code section.

Step 2. Run the following command −

$tar xvfz sqlite-autoconf-3071502.tar.gz

$cd sqlite-autoconf-3071502

$./configure --prefix=/usr/local

$make

$make install

The above procedure will end with SQLite installation on your Mac OS X machine. Which you can verify by issuing the following command −

$sqlite3

SQLite version 3.7.15.2 2013-01-09 11:53:05

Enter ".help" for instructions

Enter SQL statements terminated with a ";"

sqlite>

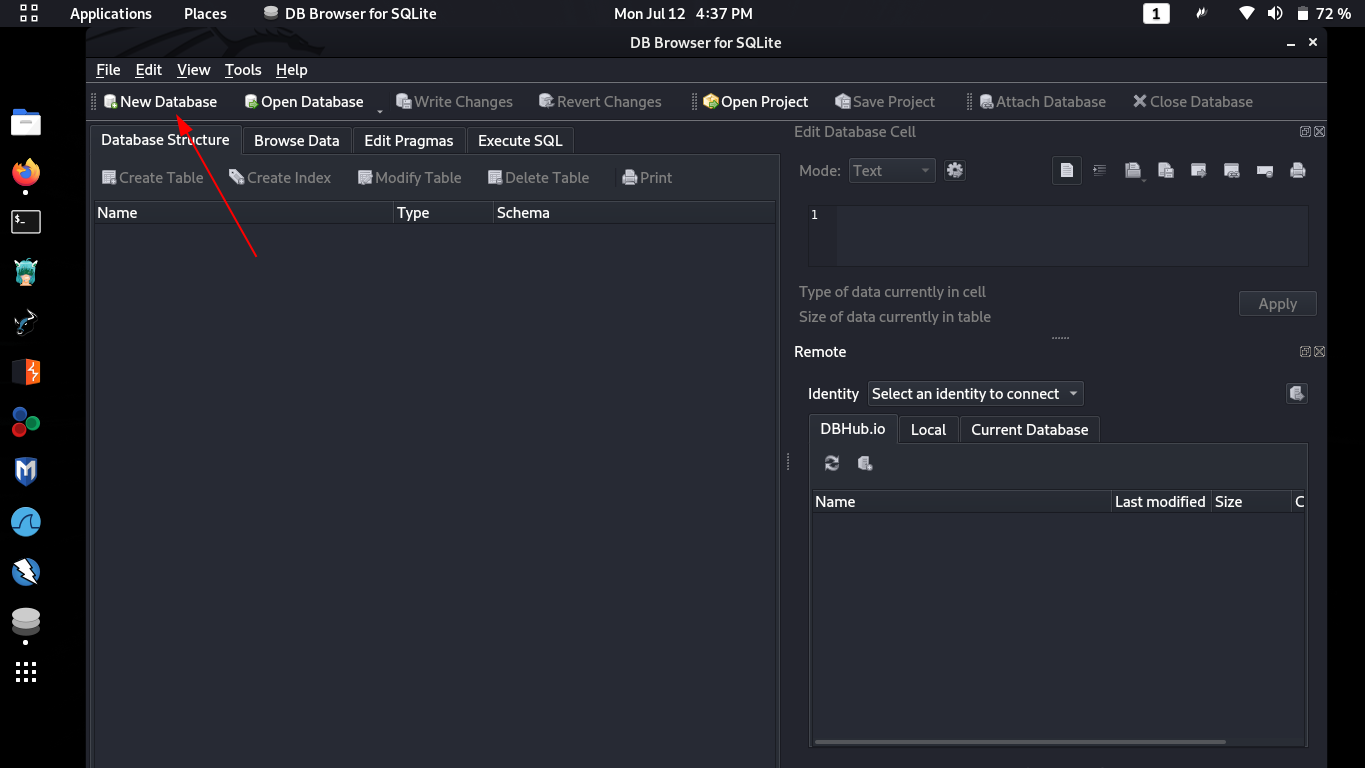
Finally, you have SQLite command prompt where you can issue SQLite commands for your exercises.

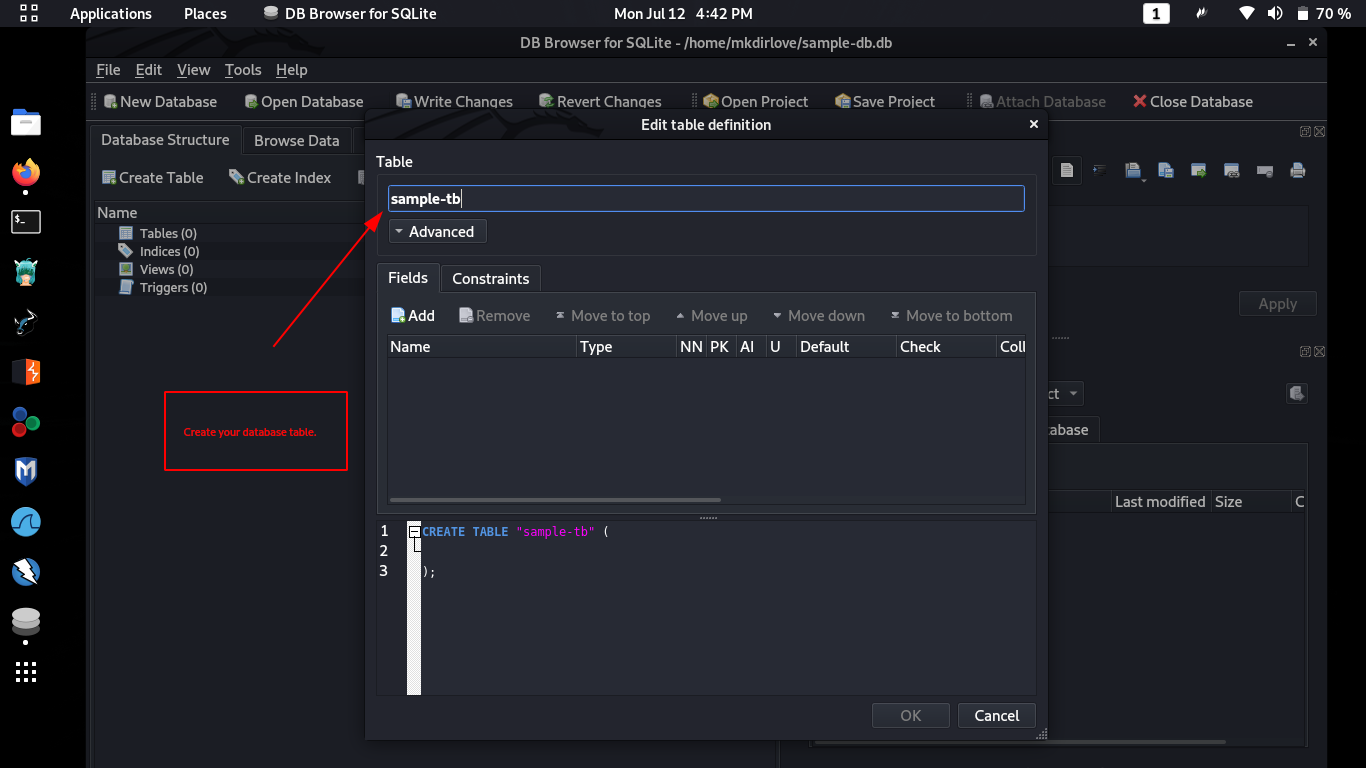
**Connecting Python Flask Framework with Sqlite**

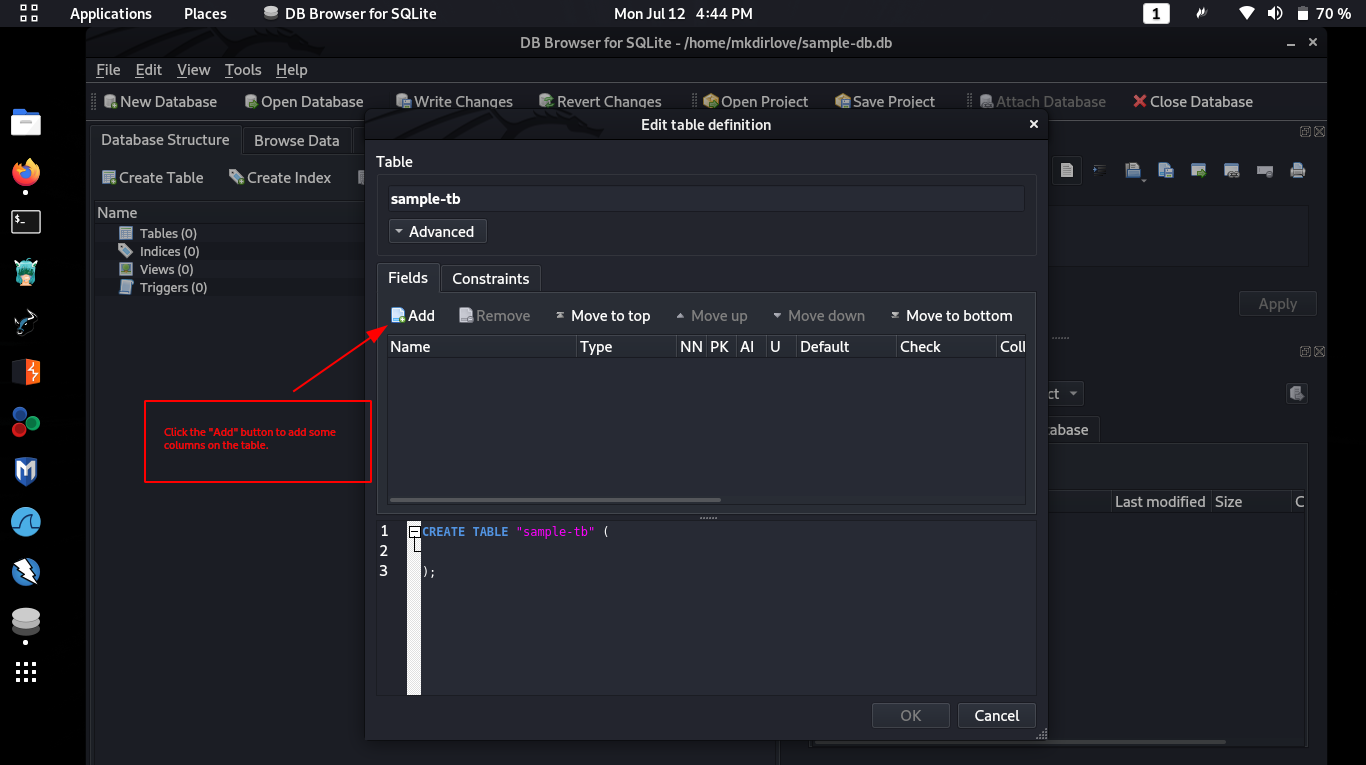
Step 1. Install sqlitebrowser in your operating system using the following commands (In this case: The developer used Linux cause it's pre-installed).

$sudo apt update

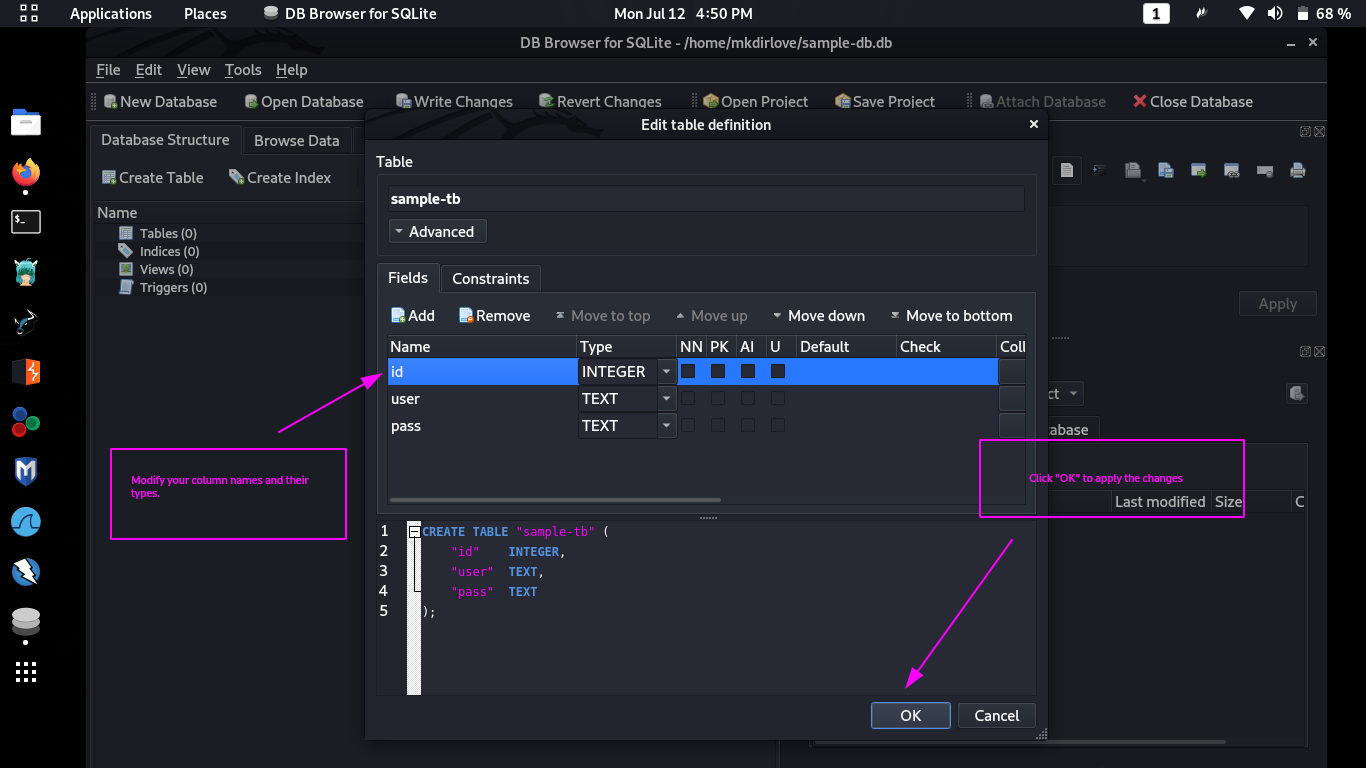
$sudo apt install sqlitebrowser

Step 2. Open your sqlitebrowser and create a new database file.

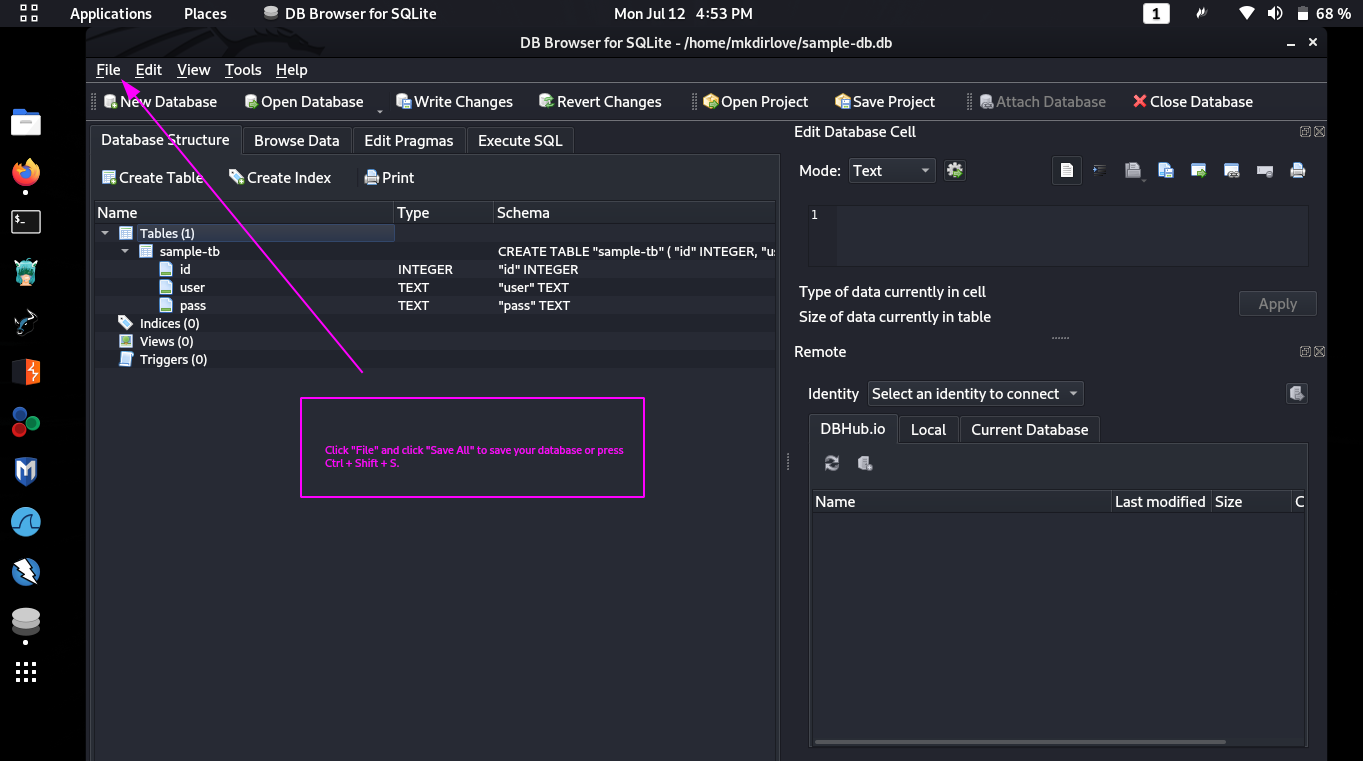
Step 3. Create tables.

Step 4. Create columns.

Step 5. Edit columns.



Step 6. Save Database



Step 7. Connecting Python Flask Framework to your database.

You can connect it using the following lines of code.

from cs50 import SQL

from flask\_session import Session

from flask import Flask, render\_template, redirect, request, session, jsonify

from datetime import datetime

# # Instantiate Flask object named app

app = Flask(\_\_name\_\_)

# # Configure sessions

app.config["SESSION\_PERMANENT"] = False

app.config["SESSION\_TYPE"] = "filesystem"

Session(app)

# Creates a connection to the database

db = SQL ( "sqlite:///data.db" )

Step 8. Creating the routing function.

I used the checkout function on the code as an example.

@app.route("/checkout/")

def checkout():

order = db.execute("SELECT \* from cart")

# Update purchase history of current customer

for item in order:

db.execute("INSERT INTO purchases (uid, id, samplename, image, quantity) VALUES(:uid, :id, :samplename, :image, :quantity)", uid=session["uid"], id=item["id"], samplename=item["samplename"], image=item["image"], quantity=item["qty"] )

# Clear shopping cart

db.execute("DELETE from cart")

shoppingCart = []

shopLen = len(shoppingCart)

totItems, total, display = 0, 0, 0

# Redirect to home page

return redirect('/')

That's all and now you have been created your own database file and routing function that is connected to your database.