

Workshop on Smart City Applications Using Raspberry Pi

Introduction to Raspberry Pi

A credit card sized computer with endless capabilities!

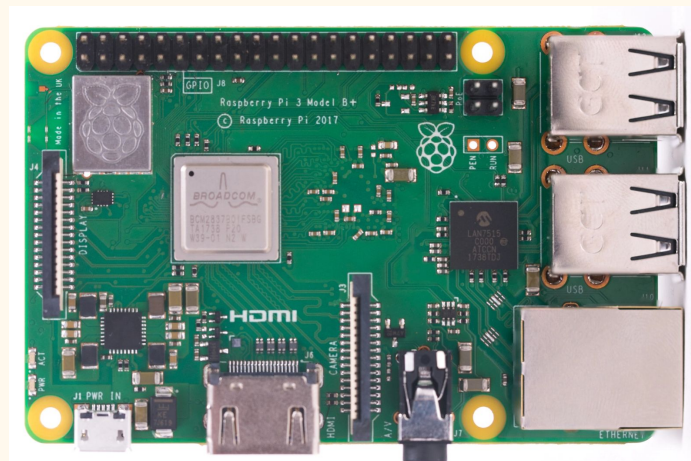


Image 1: Raspberry Pi 3B+

Specification:

Raspberry Pi 3B+

SoC: Broadcom BCM2837

CPU: 4× ARM Cortex-A53, 1.2GHz

GPU: Broadcom VideoCore IV

RAM: 1GB LPDDR2 (900 MHz)

Networking: 10/100 Ethernet, 2GHz & 5GHz 802.11n wireless

Bluetooth: Bluetooth 4.1 Classic, Bluetooth Low Energy

Storage: microSD

GPIO: 40-pin header, populated

Ports: HDMI, 3.5mm analogue audio-video jack, 4× USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)

Source: <https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/>

Workshop on Smart City Applications Using Raspberry Pi

GPIO Pinouts

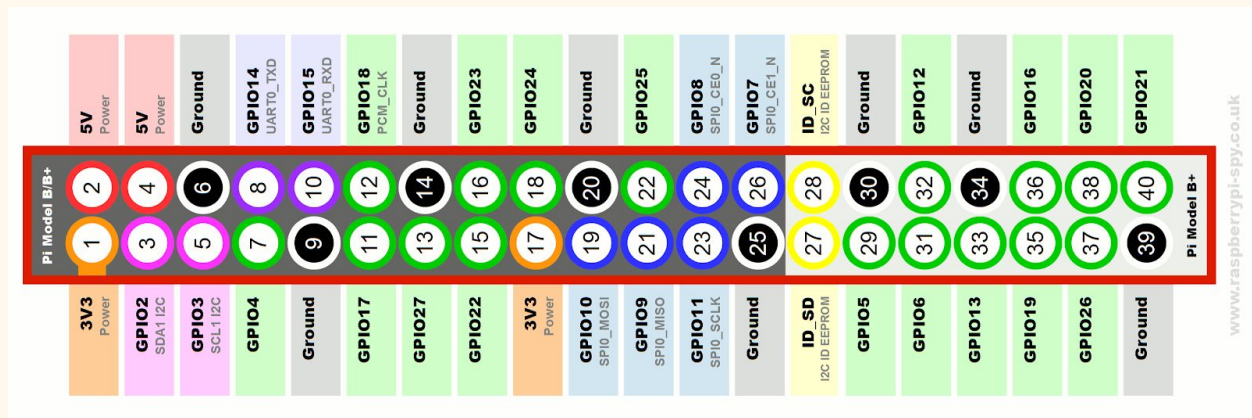


Image 2: Raspberry Pi GPIO Pinouts

Additional Supported Hardware

- GPS over USB
- GPS over UART
- IMU Sensors over I2C
- ADC over I2C and SPI
- Zigbee over USB and UART
- Almost anything!

Use cases at AmritaWNA

Smart Energy Systems and Solutions:

Raspberry Pi is used as the data collection and control nodes in the StabilizE Project. Data from the energy meters are captured by the Raspberry Pi. The Pi then relays it to the control server. The Pi also control different contactors through its GPIO Pins

Workshop on Smart City Applications Using Raspberry Pi

Oceannet Systems:

Raspberry Pi is used in the OceanNet Control Box. The Pi controls the Automatic Antenna Rotator System that helps the equipment in the boat to be in the line of sight with the base station.

IoT based landslide detection systems:

In the landslide deployment, the Raspberry Pi collects data from various sensors and relays the data to the control center at Amritapuri.

Topic 1: Installation of Operating System

The most recent version of Raspbian OS is called Raspbian Stretch. It is available in two forms, Raspbian and Raspbian Lite. Raspbian is the full version of the OS including the Graphical User Interface (GUI) whereas the Lite version is the strip down version and it has only the Command Line Interface (CLI). Ideally, for Beginners, the full version of the OS is recommended. There are two ways to install the Operating System. The simplest way is to use the NOOBS installer provided by RaspberryPi.org. The other method is to download the complete OS image and write it to the SD Card. This workshop follows the NOOBS Method.

Follow the steps to install the OS on the Raspberry Pi

Step1: Open the Browser in the lab computer and visit

Workshop on Smart City Applications Using Raspberry Pi

<https://www.raspberrypi.org/downloads/>

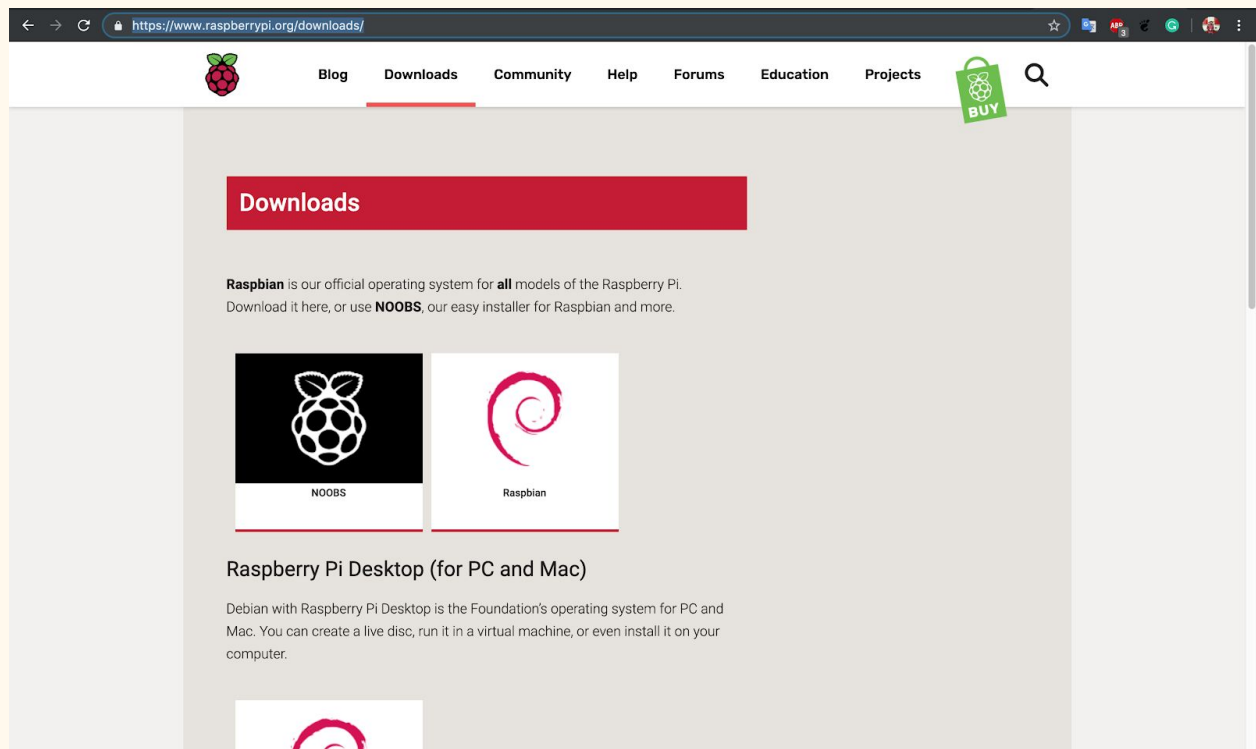


Image 3:Raspbian download page

Step2: Click NOOBS

Workshop on Smart City Applications Using Raspberry Pi

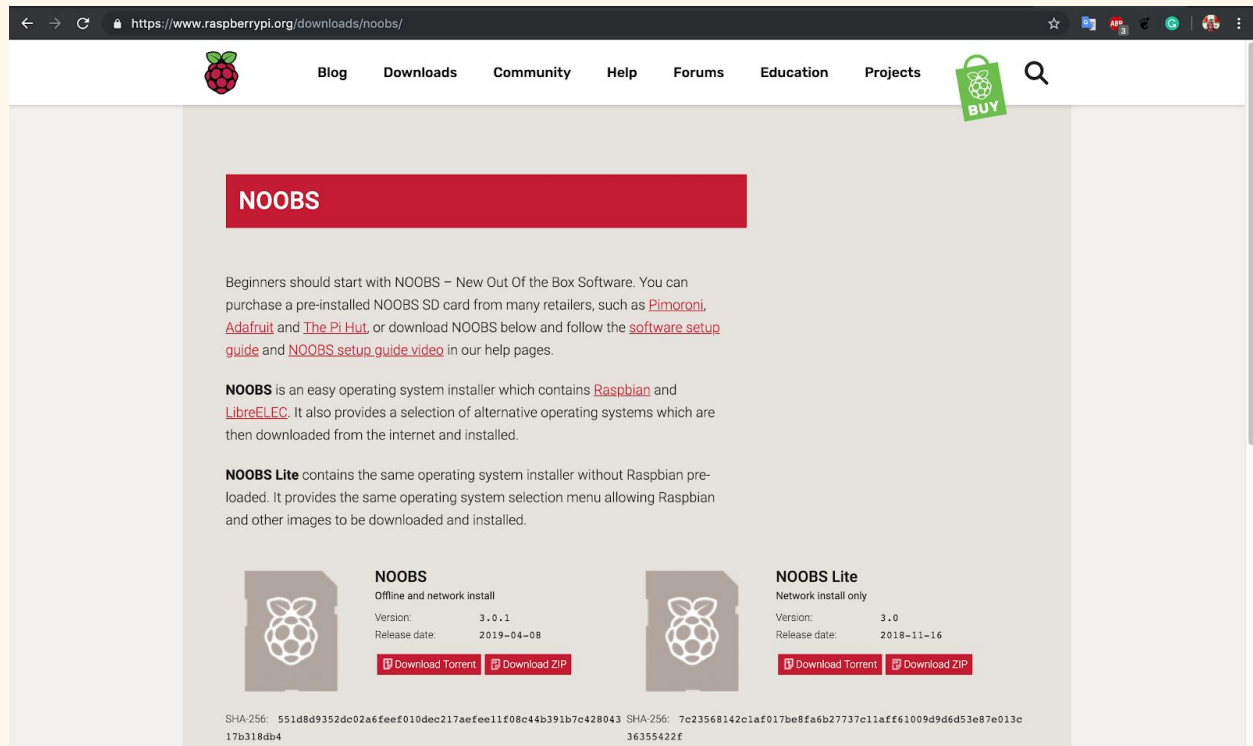


Image 4: Raspbian NOOBS Download Page

Step3: Click Download Zip. The file downloading will be initiated

Step4: Once the download is complete, go to your downloads folder and extract the downloaded file

Workshop on Smart City Applications Using Raspberry Pi

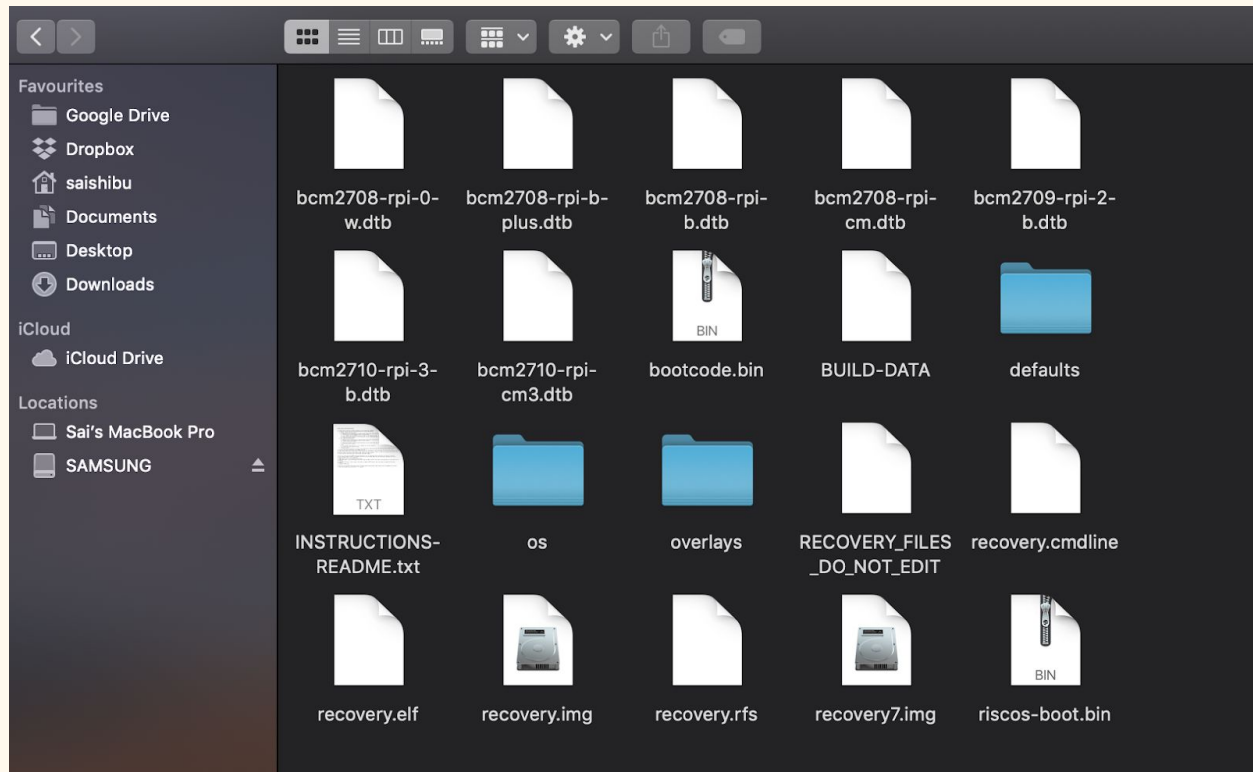


Image5: Files after extraction

Step5: Copy the extracted files to the SD Card

Step6: Safely unmount the SD Cards from the computer and insert it to the Raspberry Pi

Step7: Connect HDMI Cable to a Display, Keyboard and Mouse. Power the Raspberry Pi using a USB Cable and 5V, 2.4A Power Brick. Notice the glowing LEDs.

Workshop on Smart City Applications Using Raspberry Pi

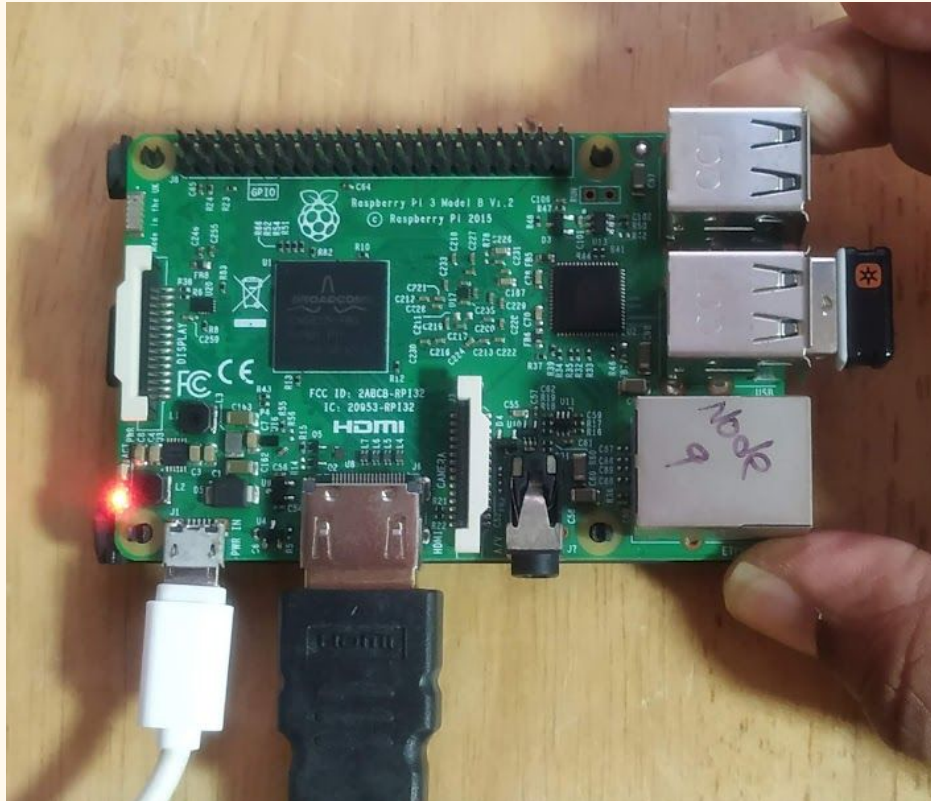


Image6: Raspberry Pi with all peripherals connected

Installation Steps from official Raspberry Pi org.

NOOBS INSTALLATION INSTRUCTIONS

1. Insert an SD card that is 8GB or greater in size into your computer.
2. Format the SD card using the platform-specific instructions below:
 - a. Windows (32GB cards and under)
 - i. Download the SD Association's Formatting Tool from https://www.sdcard.org/downloads/formatter_4/eula_windows/
 - ii. Install and run the Formatting Tool on your machine
 - iii. Check that the SD card you inserted matches the one selected by the Tool
 - iv. Click the "Format" button
 - b. Mac (32GB cards and under)
 - i. Download the SD Association's Formatting Tool from https://www.sdcard.org/downloads/formatter_4/eula_mac/

Workshop on Smart City Applications Using Raspberry Pi

- ii. Install and run the Formatting Tool on your machine
- iii. Select "Overwrite Format"
- iv. Check that the SD card you inserted matches the one selected by the Tool
- v. Click the "Format" button
- c. Linux
 - i. We recommend using gparted (or the command line version parted)
 - ii. Format the entire disk as FAT32
- d. Cards over 32GB
 - i. Follow the instructions on

https://www.raspberrypi.org/documentation/installation/sdxc_formatting.md

3. Extract the files contained in this NOOBS zip file.

4. Copy the extracted files onto the SD card that you just formatted so that this file is at the root directory of the SD card. Please note that in some cases it may extract the files into a folder, if this is the case then please copy across the files from inside the folder rather than the folder itself.

5. Insert the SD card into your Pi and connect the power supply.

Your Pi will now boot into NOOBS and should display a list of operating systems that you can choose to install.

If your display remains blank, you should select the correct output mode for your display by pressing one of the following number keys on your keyboard:

- 1. HDMI mode - this is the default display mode.
- 2. HDMI safe mode - select this mode if you are using the HDMI connector and cannot see anything on screen when the Pi has booted.
- 3. Composite PAL mode - select either this mode or composite NTSC mode if you are using the composite RCA video connector.
- 4. Composite NTSC mode

If you are still having difficulties after following these instructions, then please visit the Raspberry Pi Forums (<http://www.raspberrypi.org/forums/>) for support.

Step8: Booting and Installation of OS

Workshop on Smart City Applications Using Raspberry Pi

Boot screen with Rainbow colours will be displayed on the monitor when the Pi is turned on



Image7: Raspberry Pi boot Screen

Workshop on Smart City Applications Using Raspberry Pi

Step9: Select Raspbian Full and click Install

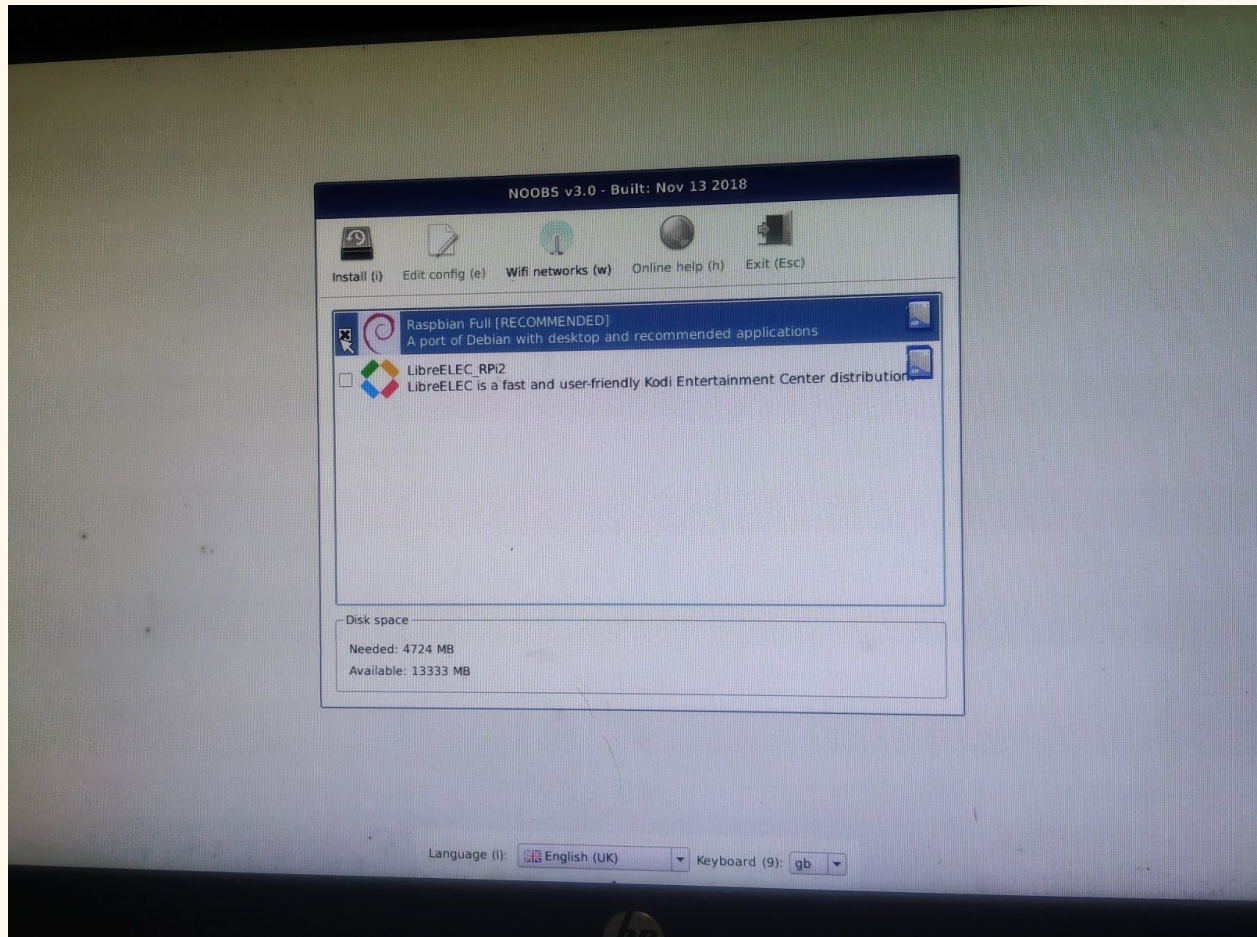


Image8: Operating System installation

Workshop on Smart City Applications Using Raspberry Pi

Step10: Click Yes to begin installation

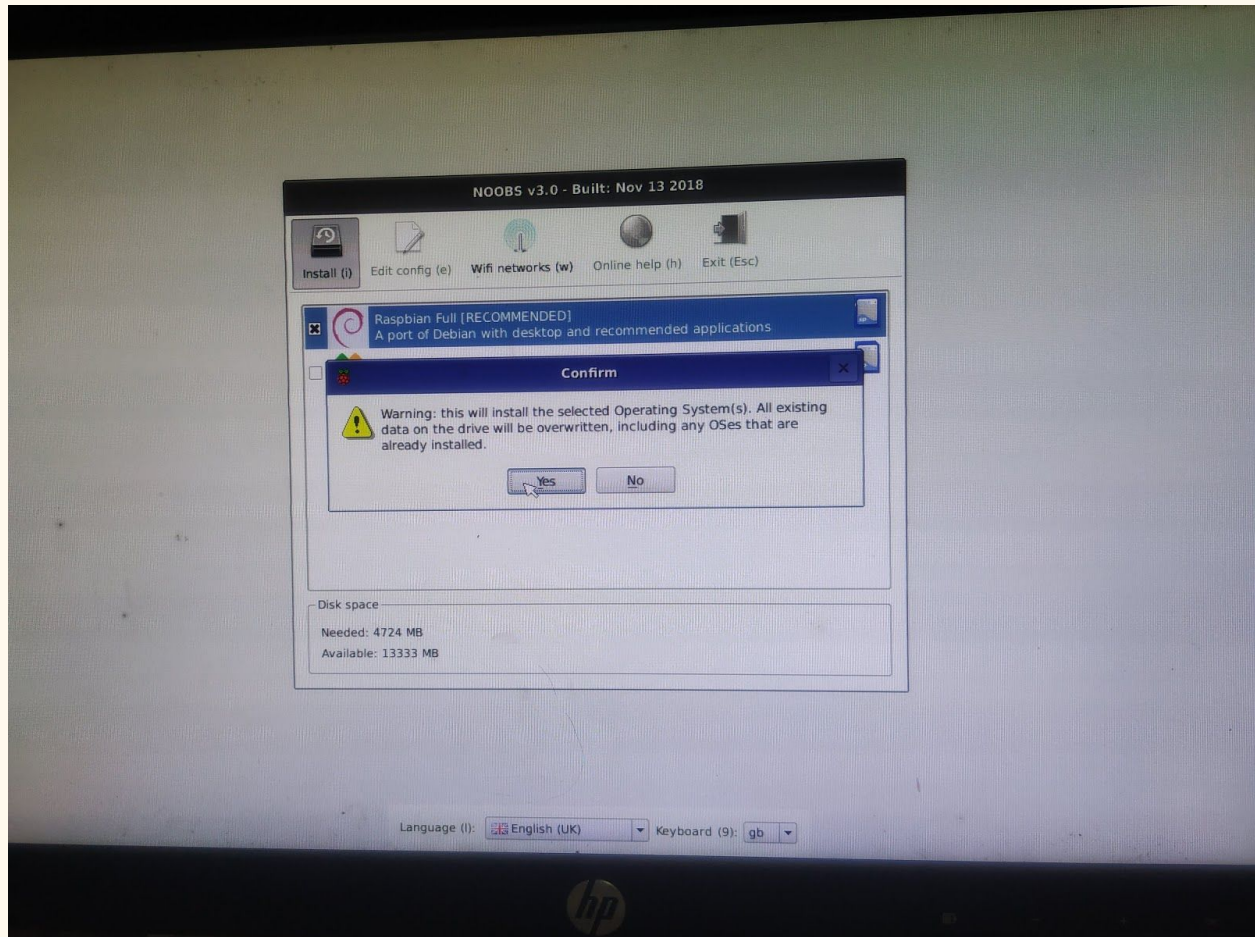


Image9: Operating System Installation - Confirmation Window

Workshop on Smart City Applications Using Raspberry Pi

Step 11: Please wait until the installation process is complete

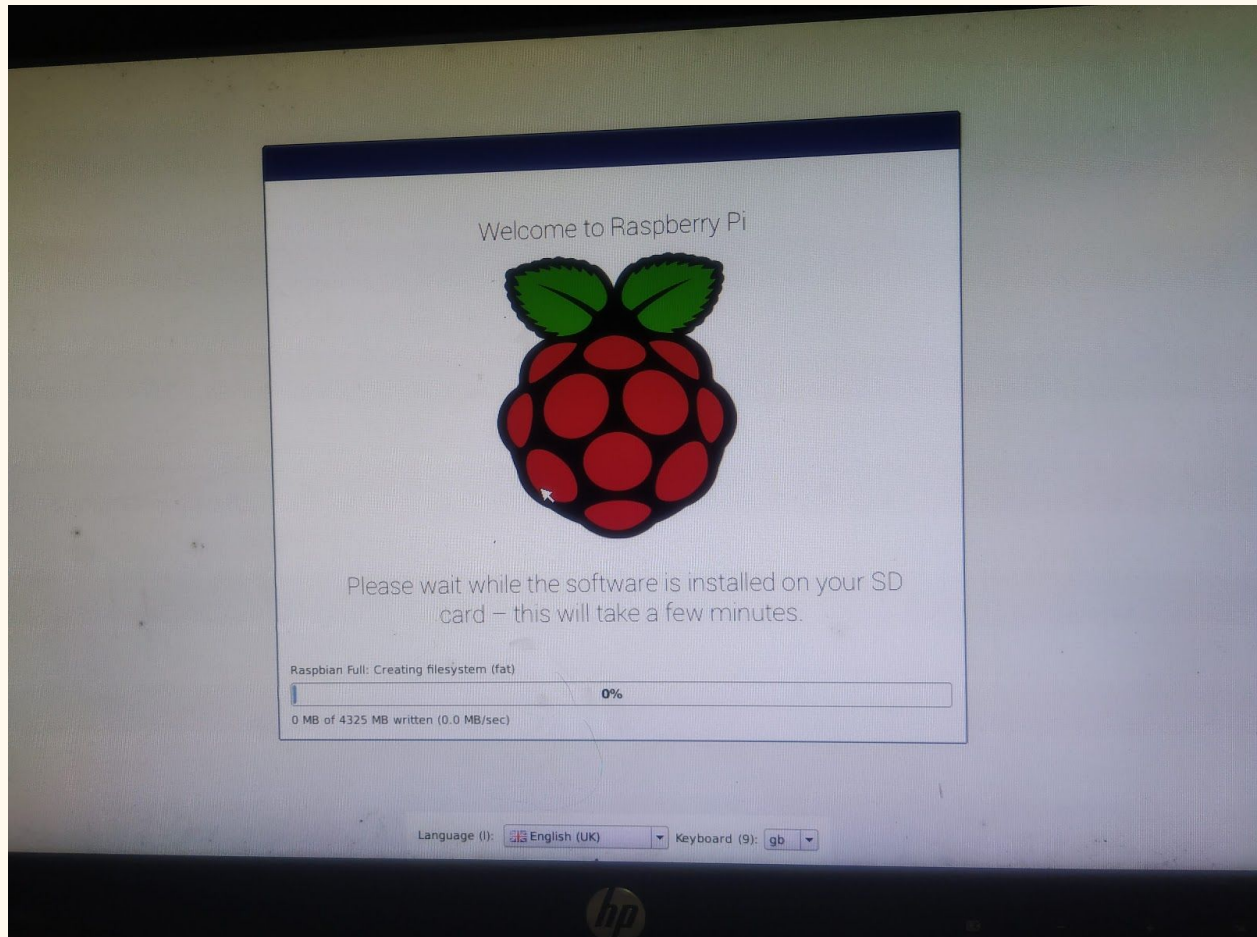


Image10: Operating System installation begins

Workshop on Smart City Applications Using Raspberry Pi

Step 12: Click Ok to finish the installation

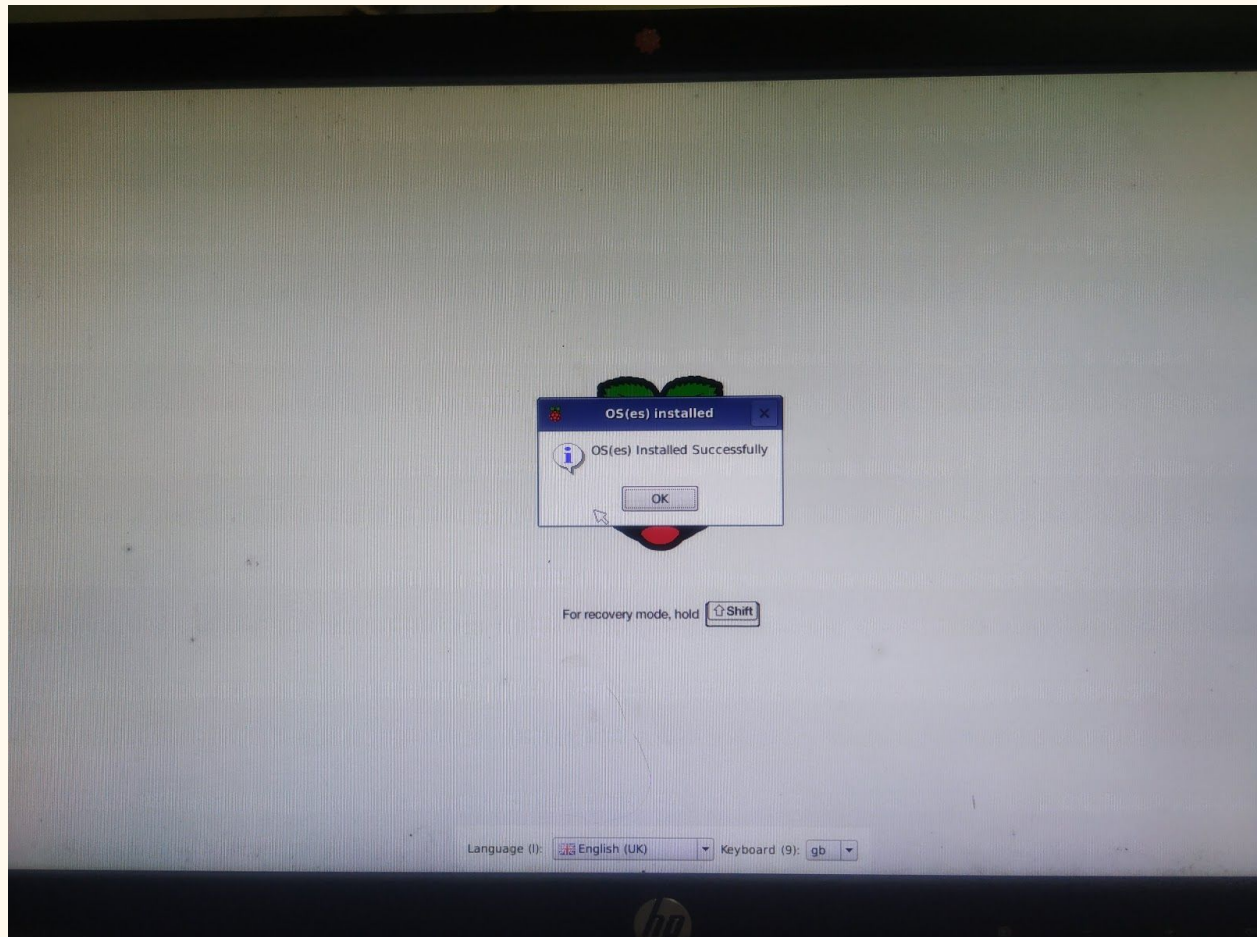


Image11: Operating System installation completed successfully

Workshop on Smart City Applications Using Raspberry Pi

Step 13: Once the Pi boot follow the on-screen setup by clicking Next

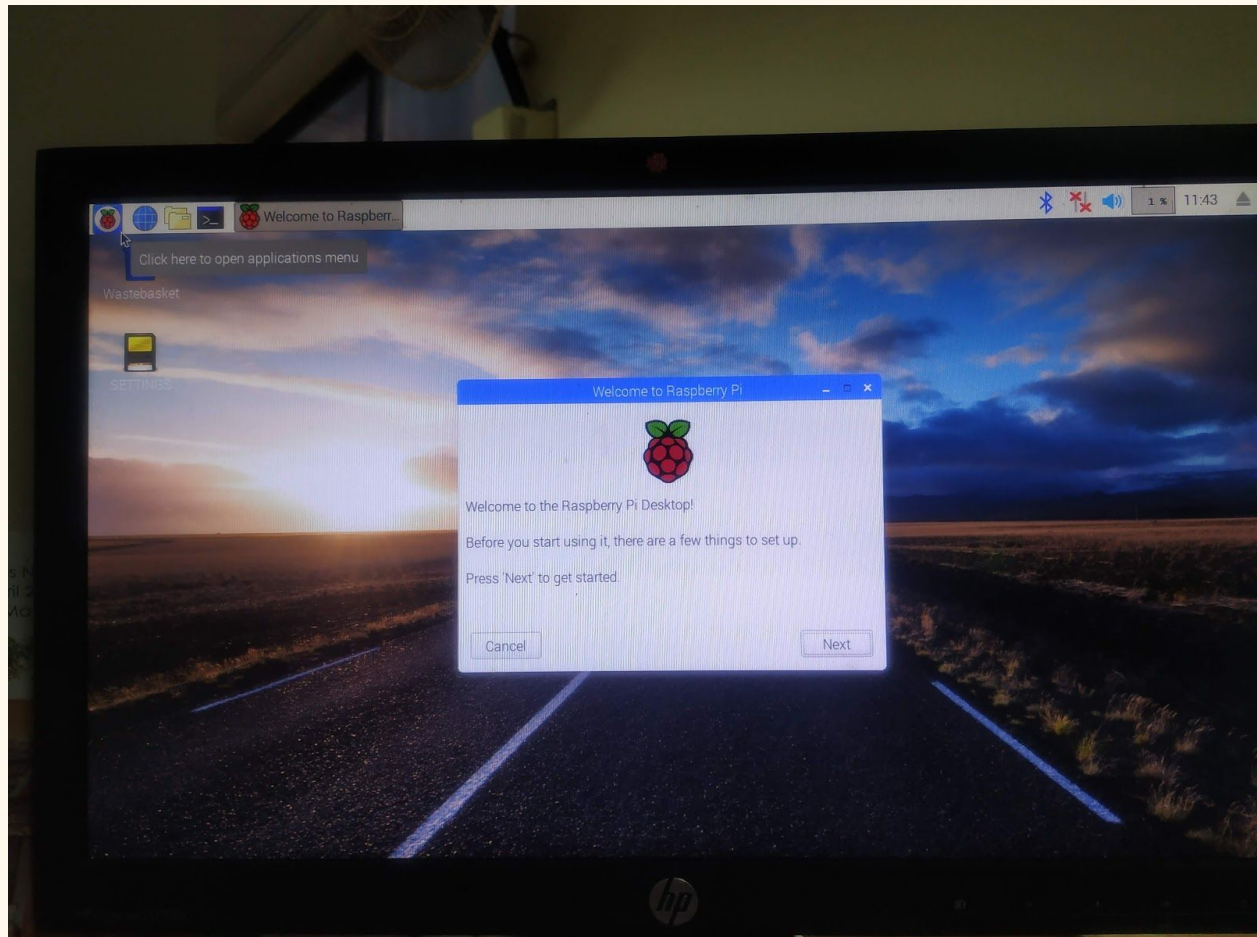


Image12: First time boot setting

Workshop on Smart City Applications Using Raspberry Pi

Step14: Choose the country, Language and Timezone. Then click Next

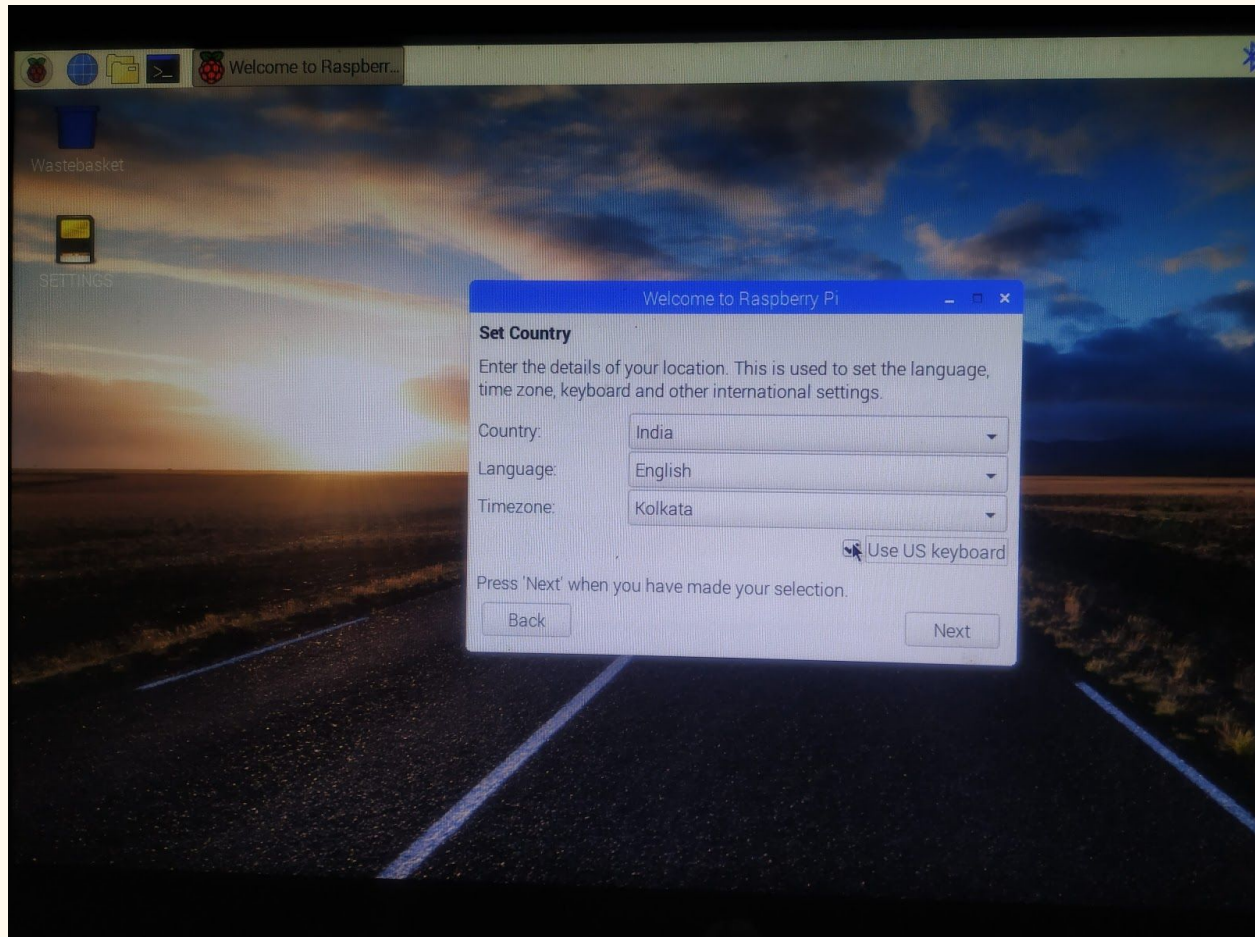


Image13: One time configuration after installation

Workshop on Smart City Applications Using Raspberry Pi

Step 15: Enter the new password and click Next

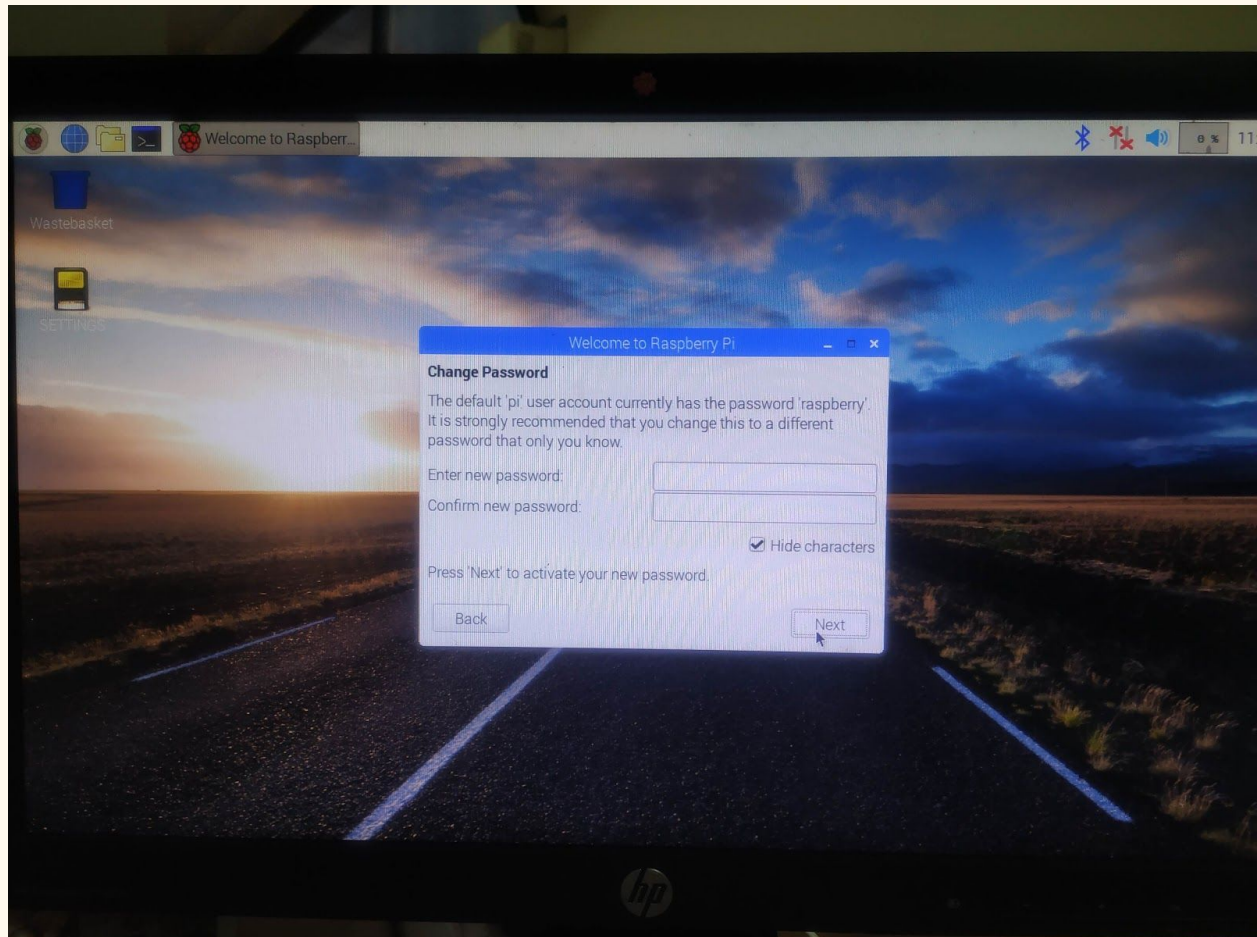


Image 14: Setting up the login password

Workshop on Smart City Applications Using Raspberry Pi

Step16: Rest of the steps can be skipped and wait till the Pi Boots.

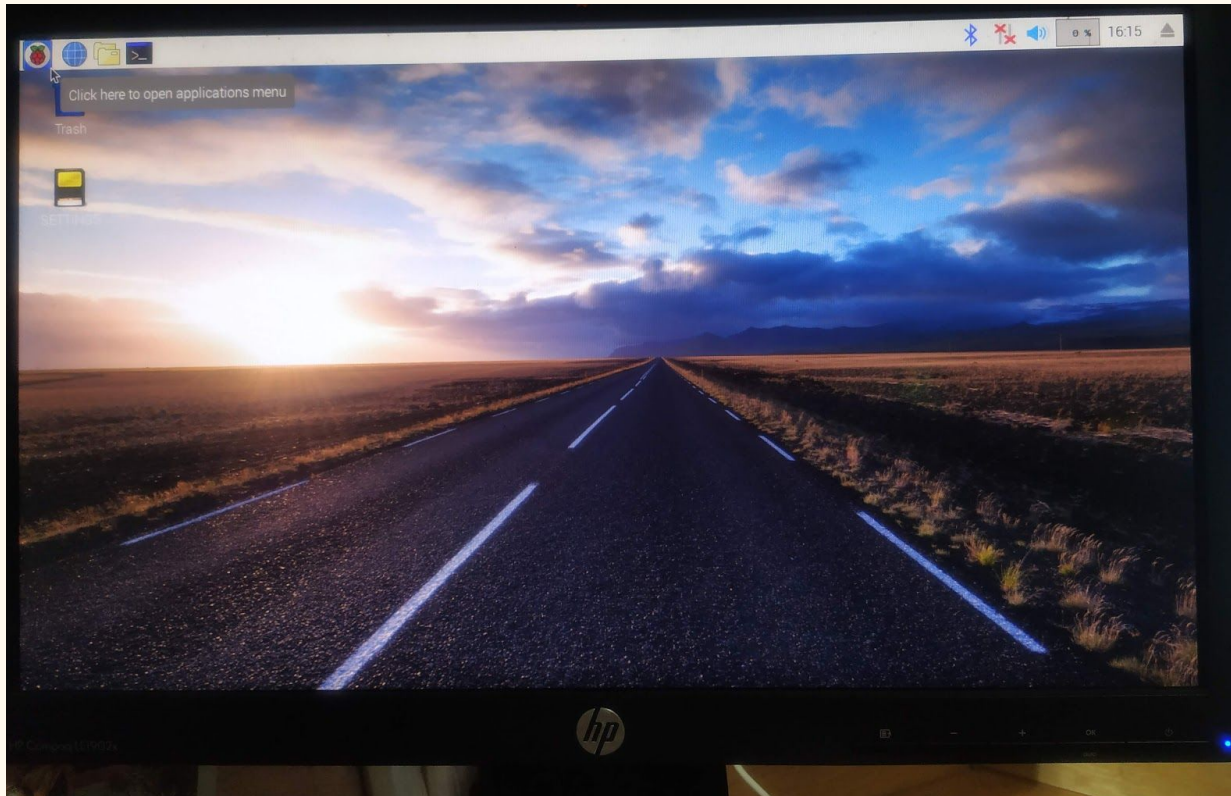


Image15: After all the configuration - Raspbian Pixel Desktop

Step17: Connect to WiFi by clicking the network icon and choose the SSID. Key in the Security Key if needed.

Workshop on Smart City Applications Using Raspberry Pi

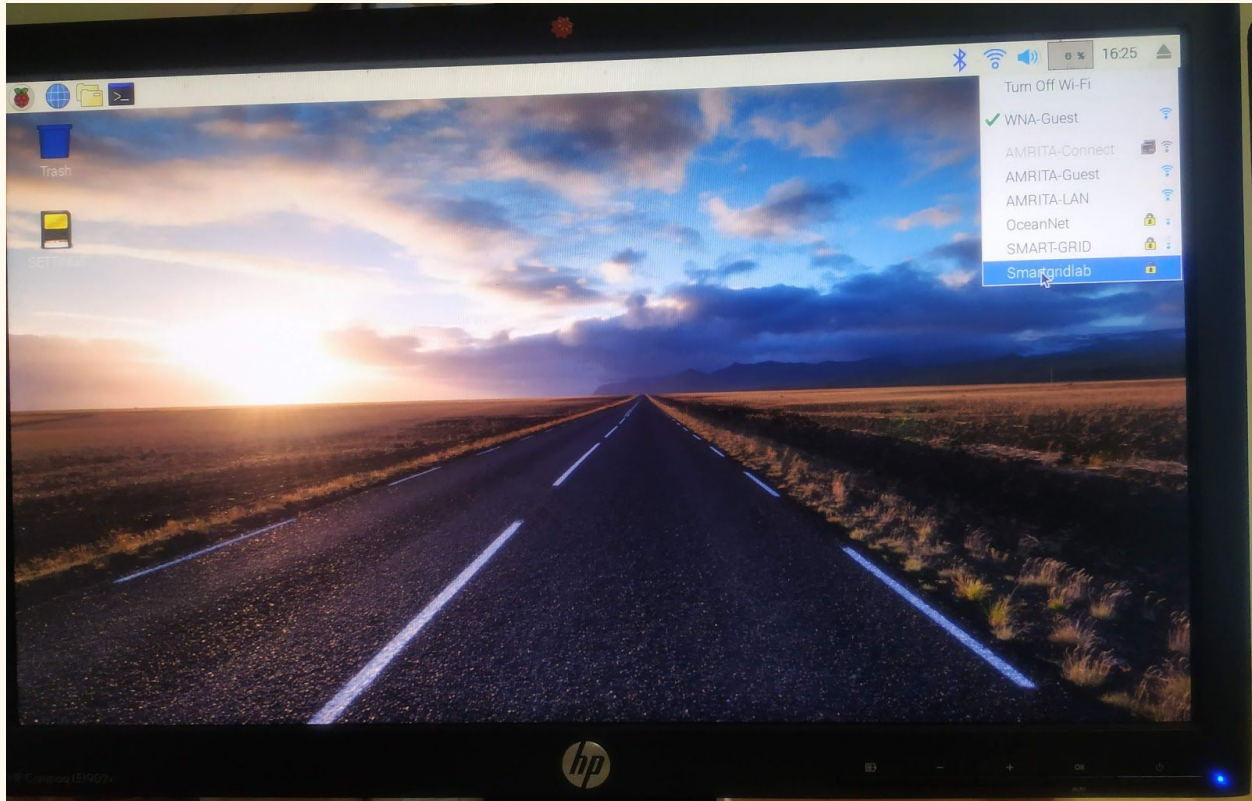


Image16: WiFi configuration

Step18: Update the Repository. Open the terminal and type

```
sudo su #get super user privileges  
apt-get update #Update the apt repository  
apt-get upgrade #install the updates from the repository
```


Workshop on Smart City Applications Using Raspberry Pi

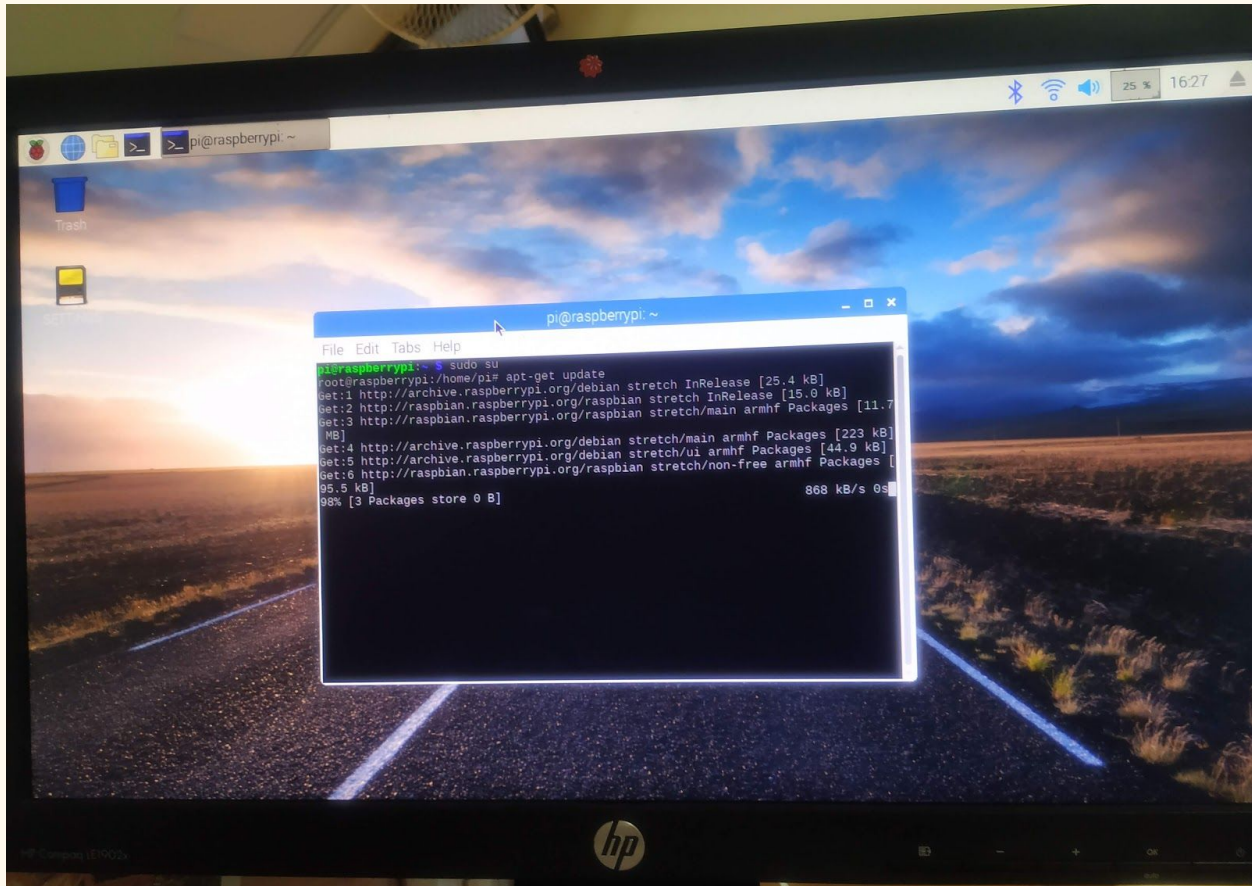


Image 17: Repository updation

Topic 2: Introduction to Git

Git is a version control system that keeps track of changes in files and projects. GitHub provides a cloud based user interface on top of GIT. All codes/documentation related to this workshop is available at the Github repository. To download the files to Raspberry Pi, follow the following step.

Step1: Open Terminal in Raspberry Pi.

Step2: Clone the Repository.

```
git clone https://github.com/saishibu38380/smartcityworkshop2019.git
```

Workshop on Smart City Applications Using Raspberry Pi

Step3: Navigate to the folder.

```
cd smartcityworkshop2019/
```

Step4: List the files inside the folder

```
ls
```

Now the files will be cloned to the local storage of the Raspberry Pi

Topic 3: Introduction to Python - Print "Hello World"

Python is a general purpose programming language.

Step1: Navigate to the folder

```
cd smartcityworkshop2019/
```

Step2: Open the file in text editor nano and see what is in it

```
nano hello.py
```

Step3: Execute the program

```
python hello.py
```

Topic 4: Introduction to LAMP - Apache, MySQL Database and PHPMyAdmin

Step1: Install Apache Server

Step2: Install MySQL Database

```
sudo apt-get install mysql-server
```

Step3: Login to MySQL

```
sudo mysql -u root
```

Workshop on Smart City Applications Using Raspberry Pi

Step4: Create a new user for MySQL

```
Grant all privileges on *.* to "user"@"localhost" identified by "password"
```

Step5: Logout from current MySQL Session and Login with the new username

```
/q  
Mysql -u user -p  
"Enter the password" (it is invisible)
```

Step6: Install phpMyAdmin

```
apt-get install phpmyadmin
```

Step7: Login to phpMyAdmin. Open web browser and type

```
<ip address>/phpmyadmin
```

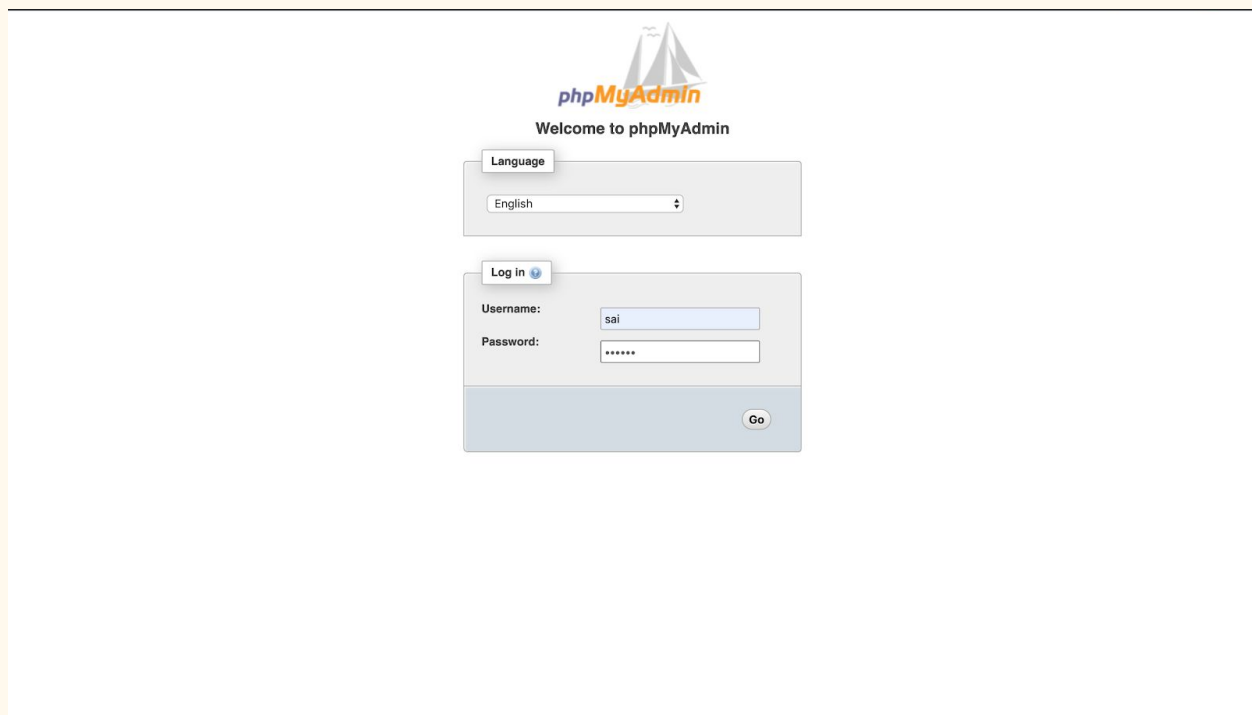


Image18: phpMyAdmin login page

Step8: Input the credentials and click Go

Workshop on Smart City Applications Using Raspberry Pi

Step9: Explore the window. Click New from the left pane to create new databases

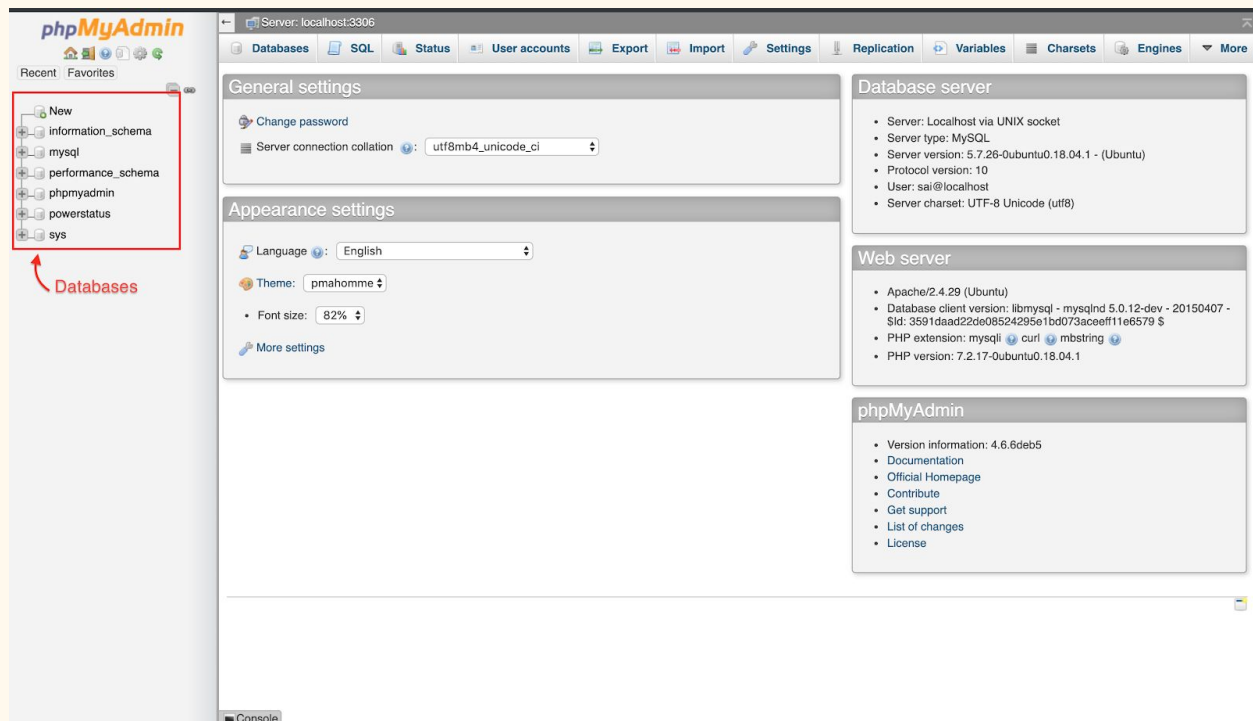


Image19: MySQL Database Window

Workshop on Smart City Applications Using Raspberry Pi

Step10: Give a name to the database and click create

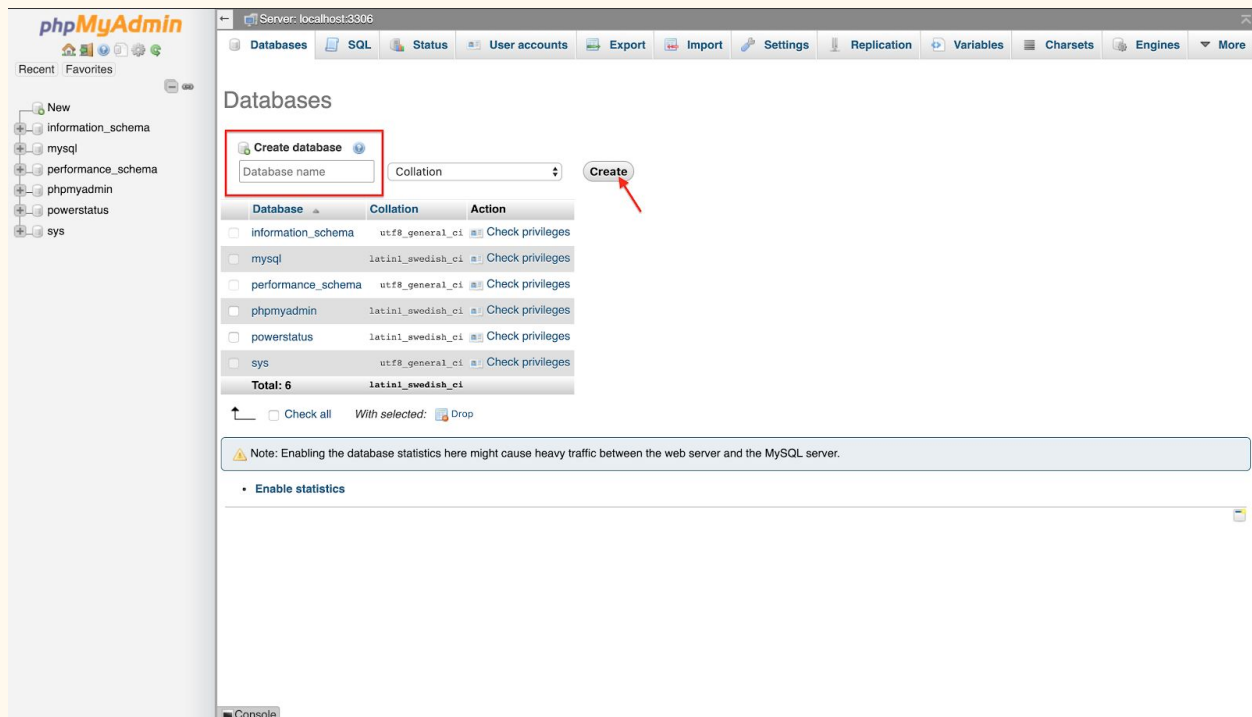


Image20: MySQL Create database window

Step11: Create a table "Participants" and click Go

Workshop on Smart City Applications Using Raspberry Pi

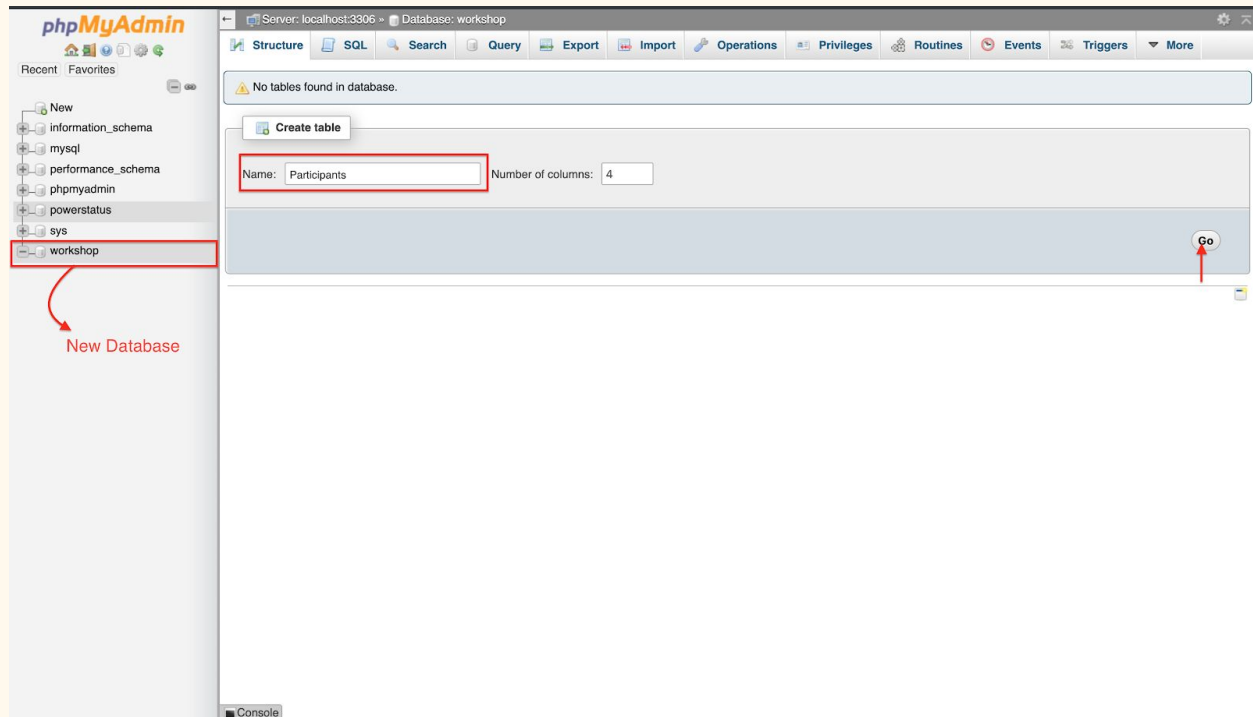


Image21: MySQL Create table window

Workshop on Smart City Applications Using Raspberry Pi

Step12: Create fields in the table and click save once done.

The screenshot shows the phpMyAdmin interface for creating a new table named 'Participants' in the 'workshop' database. The 'Structure' tab is active, showing a table with 1 column(s). The columns are defined as follows:

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	Comments
id	INT		None			<input type="checkbox"/>	PRIMARY	
Name	TEXT		None			<input type="checkbox"/>	---	
Age	INT		None			<input type="checkbox"/>	---	
Location	TEXT		None			<input type="checkbox"/>	---	

Below the column definitions, there are sections for 'Table comments:', 'Collation:', 'Storage Engine:' (set to InnoDB), and 'PARTITION definition:'. At the bottom right, there are buttons for 'Preview SQL' and 'Save'.

Image22: MySQL Create table window

Step13: Add data manually. Click insert on top section. Input values for the field. Click Go once completed.

Workshop on Smart City Applications Using Raspberry Pi

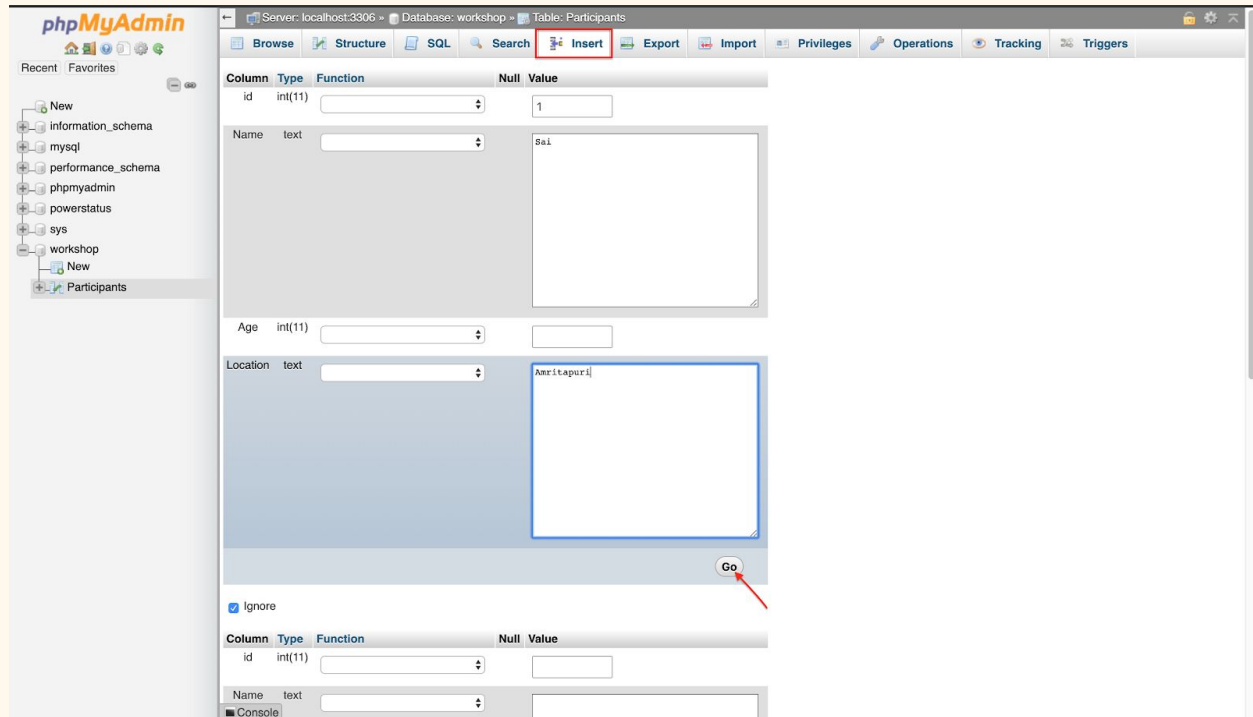


Image23: MySQL Insert data window

Workshop on Smart City Applications Using Raspberry Pi

Step14: View the data stored in the table by clicking Browse

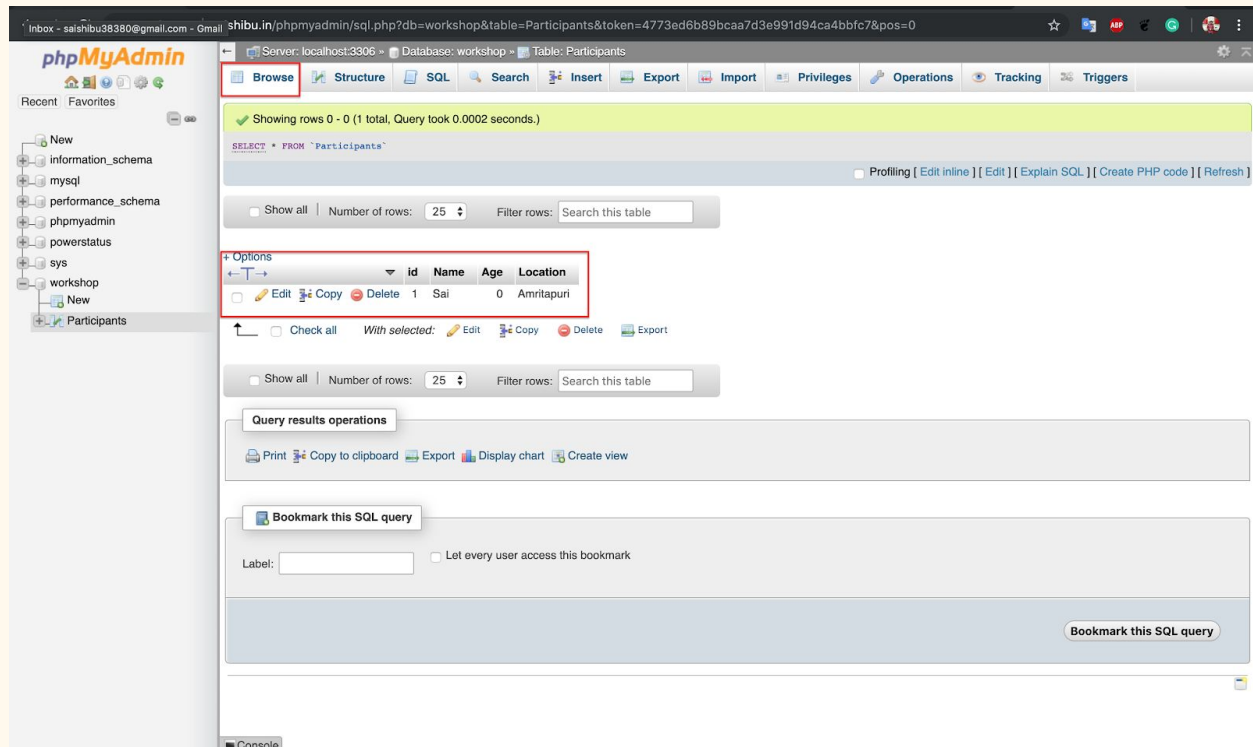


Image 24: MySQL Browse Window

Step15: Insert a few more data to the table by repeating step 13

Step16: Install mysql-python connector (pymysql)

```
pip install pymysql
```

Step17: Python program to add data to the database

Navigate to the workshop directory and explore the file mysqlfeed.py

Workshop on Smart City Applications Using Raspberry Pi

Topic 5: Introduction to TCP/IP Communication on Raspberry Pi

Information exchange between a program running on the Raspberry Pi and a partner program running on a remote computer systems

Experiment: programming with a low- level socket library

Topic 6: Introduction to REST APIs

REST means REpresentational State Transfer. Basically, it represents the state transfer of database. REST is based on the HTTP Protocol. HTTP methods GET, POST, PUT and DELETE are used in REST Architecture.

Setting up REST on Raspberry Pi

Step1: Install Flask python module

```
pip install Flask  
pip install flask-mysql
```

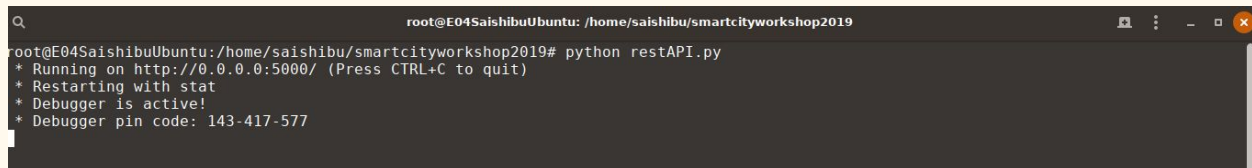
Example program on REST API

Step1: Navigate to workshop directory

Step2: Open rest.py

Step3: Execute the file in the terminal

Workshop on Smart City Applications Using Raspberry Pi



```
root@E04SaishibuUbuntu: /home/saishibu/smartcityworkshop2019
root@E04SaishibuUbuntu:/home/saishibu/smartcityworkshop2019# python restAPI.py
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger pin code: 143-417-577
```

Image 25: REST API execution at Terminal

Step4: Open browser and visit localhost:5000

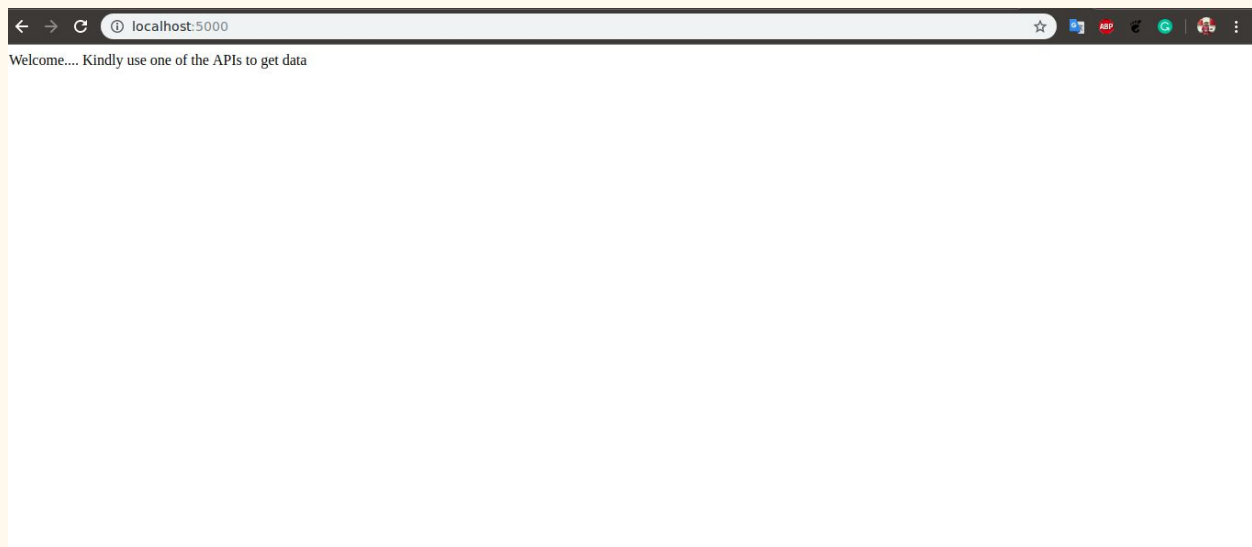


Image 26: REST API output at Terminal

Example program to display data on a database using REST API

Step1: Navigate to workshop directory

Step2: Open rest.py

Step3: Execute the file in the terminal