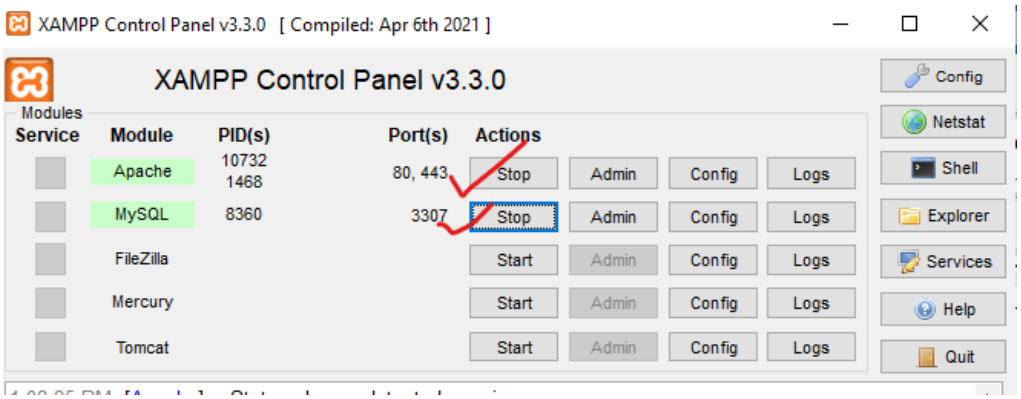
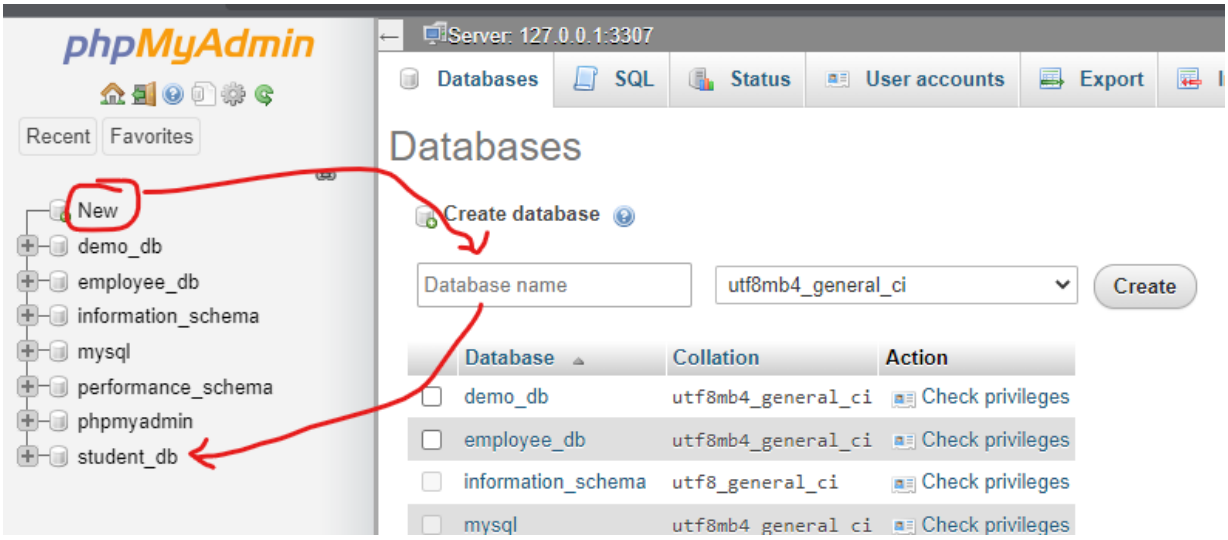
Working with the XAMPP :

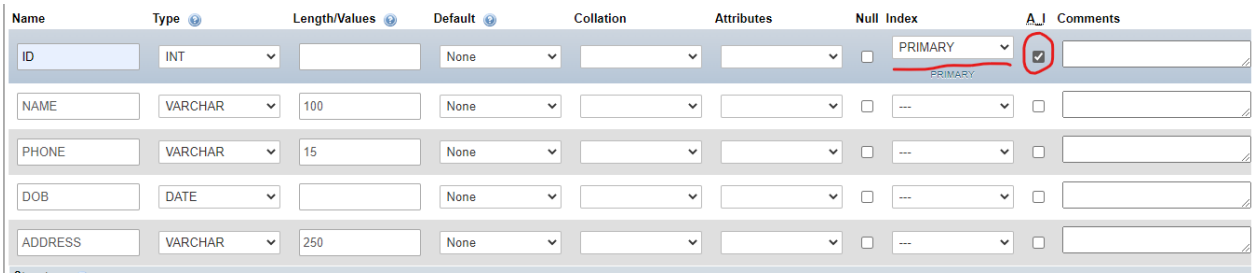
1. Open xampp control panel
2. Start Apache and mysql
   a. 

3. Open browser and type localhost or 127.0.0.1
4. Click on phpMyAdmin
   a. 

Working with database

1. Create database with name – student_db
   a. 

Working with table

1. Create table with name – student_info with 5 cols –
   a. id(AI), name(VARCHAR), phone(VARCHAR), dob, address(VARCHAR)
   b. 

2. Add data into table

Creating virtual environment

1. Create folder for your project
2. Navigate cmd to that folder
3. Run command → python -m venv . (. For current folder)
    a. D:\Flask Projects>python -m venv .
    b. It will take few mins
4. Activate venv
    a. D:\Flask Projects>**.\Scripts\activate**
    b. If activated successfully, you will get cmd like → (Flask Projects) D:\Flask Projects>
5. Deactivate venv
    a. (Flask Projects) D:\Flask Projects>.**\Scripts\deactivate.bat**

Install required libraries

Important : activate venv to install libraries.

1. Flask → pip install flask
    a. (Flask Projects) D:\Flask Projects>pip install flask
2. Pymysql → pip install pymysql
    a. (Flask Projects) D:\Flask Projects>pip install pymysql
3. Check if all installation is done properly
    a. (Flask Projects) D:\Flask Projects>pip freeze

Open VS code

1. Open project folder
2. Create python file →app.py
3. Inside app.py
    a. Import Flask →

```
from flask import Flask
```

    b. #create object of Flask class

```
app = Flask(__name__)
```

    c. #to execute the code

```
if __name__=='__main__':

        app.run(debug=True)
```

    d. #a method that envoked at server execution

```
@app.route("/")
def index():
      return "Hello World"
```

4. Run app.py from cmd
    a. python app.py or you can use → py app.py
    b. copy url and paste in browser
    c. Done !
5. Final code in app.py

```
from flask import Flask
app = Flask(__name__)
@app.route("/")
def index():
   return "Hello World"
if __name__ =="__main__":
   app.run(debug=True)
```

Creating a template

1. Create a **templates** folder (name same)
2. Create **index.html** inside templates folder with h1 tag

Connecting template with flask

1. Import **render_template**
   a. `from flask import Flask, render_template`
2. Inside def index(), return render_template("index.html")

```
@app.route("/")
def index():
  return render_template("index.html")
```

using boostrap

1. Go to browser and search for bootstrap
2. Open official website- getbootstrap.com
3. From navigation bar, click on docs
4. Scroll down and copy code from starter template and paste in index.html
5. save→ and refresh browser

DATABASE OPERATIONS

Basics imports –

```
import pymysql
db_connection = None
tb_cursor = None
```

now open xampp and start mysql and apache

function to connect to db

```
# function to connect to databse
def connectToDb():
    global db_connection, tb_cursor
    db_connection=pymysql.connect(host="localhost",user="root",
    passwd="",database="student_db",port=3307)
    if(db_connection):
        print("Done!!!")
        tb_cursor=db_connection.cursor()

    else:
        print("Not done")


# function to dicconnect from databse
def disconnectDb():
    db_connection.close()
    tb_cursor.close()

# function to get data from databse
def getAllStudentData():
    connectToDb()
    selectQuery = "SELECT * FROM student_info;"
    tb_cursor.execute(selectQuery)
    allData = tb_cursor.fetchall()
    disconnectDb()
    return allData
```

Passing received data to index.html for disaplay . so in def index() –

```
#a method that envoked at server execution
@app.route("/")
def index():
    #return "Hello flask"
    #return render_template("index.html")
    allData = getAllStudentData()
    return render_template("index.html",data = allData)
```

creating table inside index.html template

```
<div class="container">
        <table class="table table-striped table-hover">
            <thead>
                <tr>
                    <th scope="col">#</th>
                    <th scope="col">Name</th>
                    <th scope="col">Phone</th>
                    <th scope="col">Birth Date</th>
                    <th scope="col">Address</th>
                </tr>
            </thead>
            <tbody>
                {% for student in data %}
                <tr>
                    <td>{{student[0]}}</td>
                    <td>{{student[1]}}</td>
                    <td>{{student[2]}}</td>
                    <td>{{student[3]}}</td>
                    <td>{{student[4]}}</td>
                </tr>
                {% endfor %}
            </tbody>
          </table>
    </div>
```

Output

#	Name	Phone	Birth Date	Address
1	Arjun Kumar	9856523589	2021-11-04	Mumbai
2	Aniket Kadam	9856235475	2021-11-17	Mumbai
3	Pranit Jadhav	9854575236	2021-11-30	Delhi

Adding student data :

1. create button in index.html to add students

```
    <div class="container">
    <a href="http://127.0.0.1:5000/add" class="btn btn-primary">Add Students</a>
    </div>
```

   a.

2. Create add.html file in templates folder
   a. In add.html write boostrap template
3. Create function in app.py to display add.html

```
a.  @app.route("/add",methods=["GET","POST"])
b.  def addStudent():
c.      return render_template("add.html")
```

4. Create form in add.html to add data into the table

```
<div class="container">
      <h1>Add new student</h1>
      <div class="col-6">
        <form method="post">
          <input type="text" class="form-control" id="txtName" name="txtName" placeholder="Name " required>
          <br>
          <input type="number" class="form-control" id="txtPhone" name="txtPhone" placeholder="Phone " required>
          <br>
          <input type="date" class="form-control" id="txtDob" name="txtDob" placeholder="Birth Date " required>
          <br>
          <input type="text" class="form-control" id="txtAddress" name="txtAddress" placeholder="Address " required>
          <br>
          <input type="submit" value="Add Student" class="btn btn-primary">
        </form>
      </div>
    </div>
```

1. Create function in app.py to insert data into table

```
def insertIntoTable(name,phone,dob,address):
    connectToDb()
    inserQuery = "INSERT INTO student_info(NAME,PHONE,DOB,ADDRESS) VALUES(%s, %s, %s, %s);"
    tb_cursor.execute(inserQuery,(name,phone,dob,address))
    db_connection.commit()
    disconnectDb()
    return True
```

1.

9. Import request module in app.py to check what kind of request made
    1. `from flask import Flask, render_template, request`
10. writing method to collect data from form and inser onto table

```
@app.route("/add/",methods=["GET","POST"])
def addStudent():
    if request.method == "POST":
        data = request.form
        isiInserted = insertIntoTable(data['txtName'],data['txtPhone'],data['txtDob'],data['txtAddress'])
        if(isiInserted):
            message = "Insertion sucess"
        else:
            message = "Insertion Error"
        return render_template("add.html",message = message)
    return render_template("add.html")
```

1.

22. displaying received (inserted or not) message to user in add.html after submit button

```
{% if message %}
            <div class="alert alert-info">
              {{message}}
            </div>
            {% endif %}
```

23. Adding button in add.html to show all students

```
<a href="http://127.0.0.1:5000" class="btn btn-primary">Show Students</a>
```

1.

Updating record :

1. Create function to fetch id of a student
    -
2. Create function to update record
    -
    -