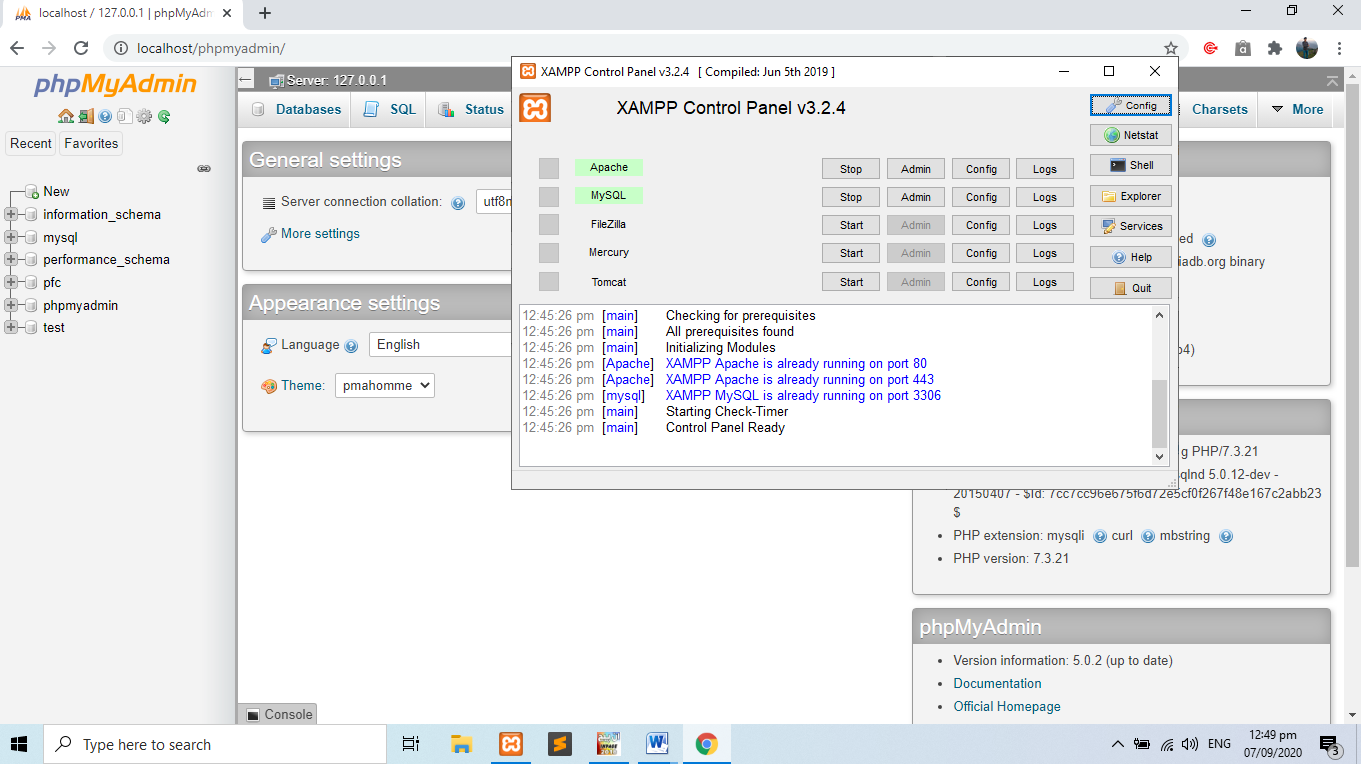
**Database**

**Task 1 Intro to database**

Database is usually used to store, edit, delete and fetch the data. We use the database for this purpose, and all this function perform the some specific software is called database management system (DBMS).

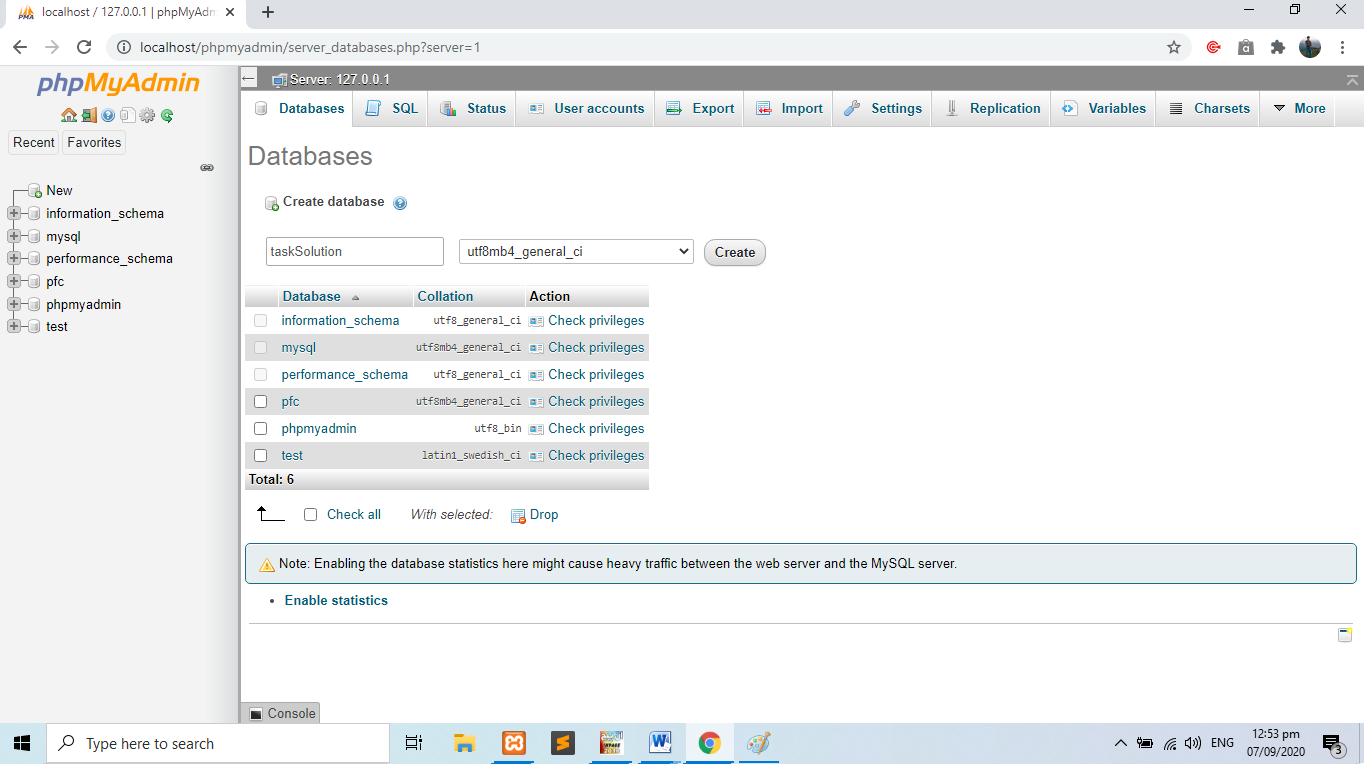
**Task 2 Starting MySQL**

First of all install XAMPP server and MySQL database is built in in this server. After install the Xampp server opens this and clicks the start button of MySQL. Then type this command or query in Google chrome “localhost/phpmyadmin”. After this open the MySQL homepage.

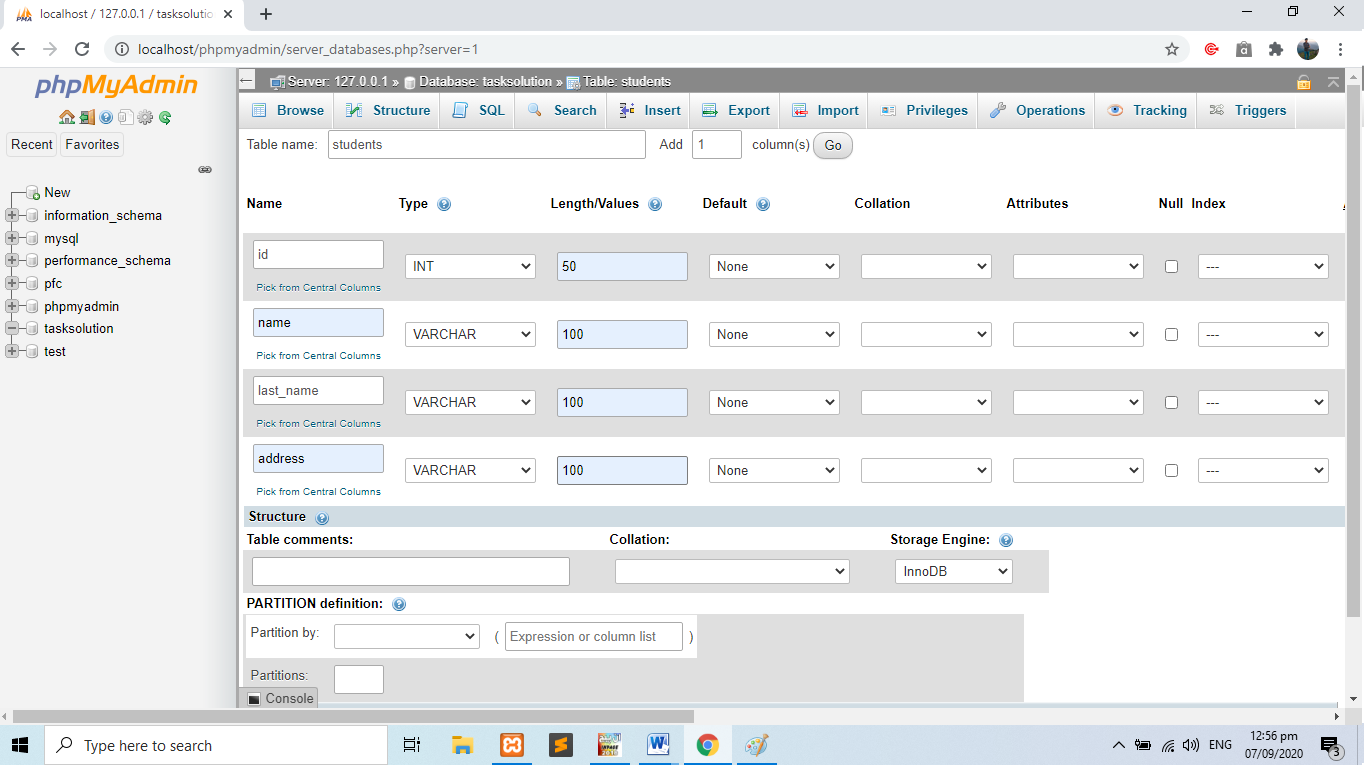


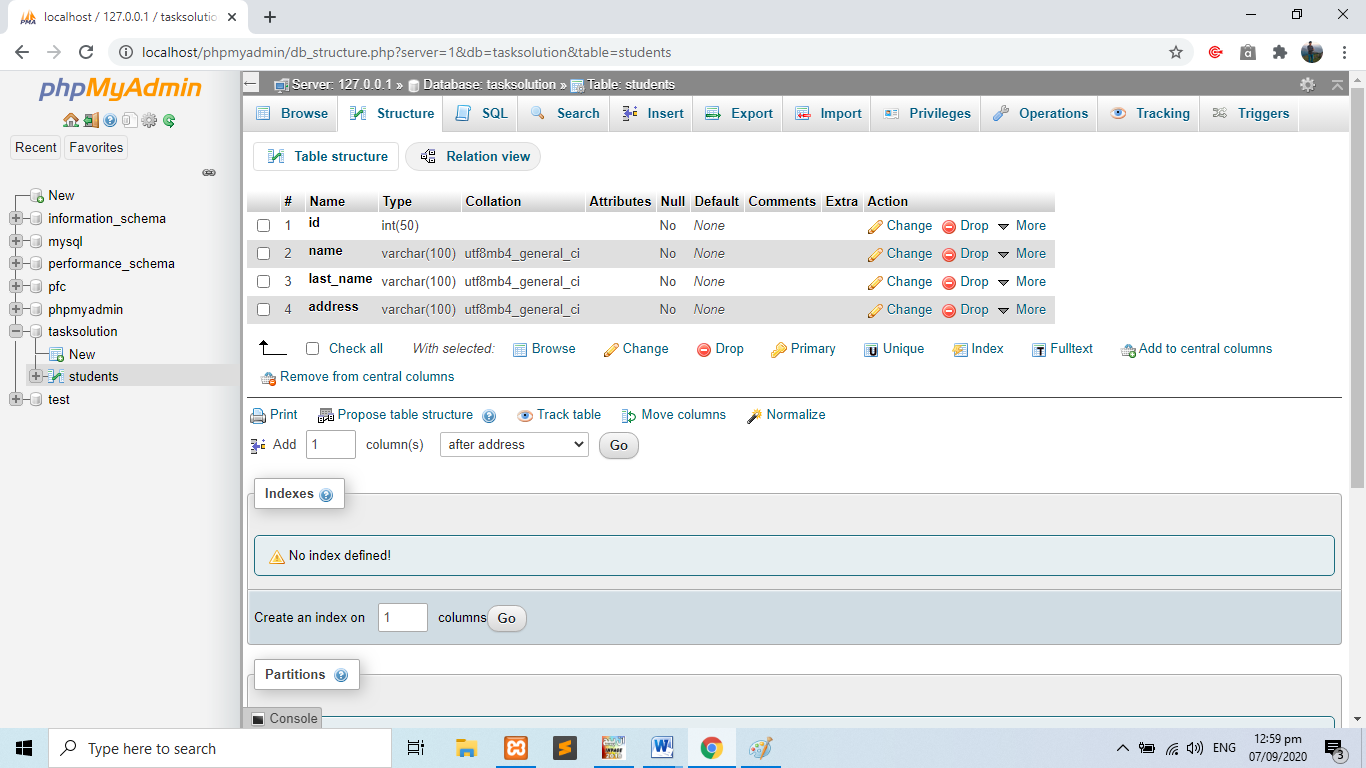
**Task 3 Creating Database and Table**

Click the new link left top side in MySQL homepage. Then write the database name and click the create button.



Enter the table name and quantity of the column in a table and click go button. After this write down the column name, data type, length and click the save button and then table will created.





**Task 4 Connecting with the database**

.env file used to connecting database

DB\_CONNECTION=mysql (database name)

DB\_HOST=127.0.0.1 (Xampp localhost name)

DB\_PORT=3306

DB\_DATABASE=PFC (database name that we are created in MySQL)

DB\_USERNAME=root (User name is always root except special case)

DB\_PASSWORD= (write here password if apply on database my case is empty)

**Task 5 Inserting data to database**

1. Make html form and make some input field
2. When click the submit button system move the controller and execute the controller code
3. After this data was successfully inserted into the database

Html form code

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>insert data</title>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

</head>

<body>

    <div class="container">

<form class="form-design" action="{{action('insertData@store')}}" method="post" enctype="multipart/form-data">

<center><h2>Insert data into database</h2></center>

<br><br>

<br>

@if(count($errors) > 0)

<ul>

@foreach($errors->all() as $error)

<div class="alert alert-danger" role="alert">{{$error}}</div>

@endforeach

</ul>

@endif

@if(\Session::has('success'))

<div class="alert alert-success" role="alert">

<p>{{\Session::get('success')}} </p>

</div>

@endif

@csrf

  <div class="form-group">

    <label for="exampleInputEmail1">Applicant First Name</label>

    <input type="text" name="name" class="form-control"  placeholder="Enter Applicant first name">

  </div>

  <div class="form-group">

    <label for="exampleInputPassword1">Applicant Last Name</label>

    <input type="text" name="last\_name" class="form-control"  placeholder="Enter Applicant last name">

  </div>

  <div class="form-group">

    <label for="exampleInputPassword1">Applicant Address</label>

    <input type="text" name="address" class="form-control"  placeholder="Enter Applicant address">

  </div>

  <div class="form-group">

    <label for="exampleInputPassword1">Applicant Picture</label>

    <input type="file" name="pic" class="form-control"  >

  </div>

  <button type="submit" class="btn btn-primary">Submit</button>

</form>

</div>

</body>

</html>

Inserting data controller code

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\insertModel;

class insertData extends Controller

{

    /\*\*

     \* Display a listing of the resource.

     \*

     \* @return \Illuminate\Http\Response

     \*/

    public function index()

    {

        //

    }

    /\*\*

     \* Show the form for creating a new resource.

     \*

     \* @return \Illuminate\Http\Response

     \*/

    public function create()

    {

        return view('insert');

    }

    /\*\*

     \* Store a newly created resource in storage.

     \*

     \* @param  \Illuminate\Http\Request  $request

     \* @return \Illuminate\Http\Response

     \*/

    public function store(Request $request)

    {

        $this->validate($request, [

            'name' => 'required|alpha',

           'last\_name' => 'required|alpha',

            'address' => 'required|string',

             'pic' => 'required|image|mimes:jpeg,png,jpg,gif|max:2048'

        ],[

            'name.alpha' => 'Applicant Name Must be Alphabetic Character Only',

            'name.required' => 'Please Enter Applicant Name',

            'last\_name.required' => 'Applicant Last Name Must be Alphabetic Character Only',

             'address.required' => 'Please Enter Applicant Address',

            'address.string' =>' Applicant Address must be a string',

                  'pic.required' => 'Please Upload CNIC Pic'

        ]);

        $student = new insertModel([

            'name' => $request->get('name'),

            'last\_name' => $request->get('last\_name'),

            'address' => $request->get('address')

]);

// single image upload

if($request->hasfile('pic')){

    $file = $request->file('pic');

    $extension= $file->getClientOriginalExtension();

    $filename= time() . '.' . $extension;

    $file->move('taskSolution/Applicant\_pic/', $filename);

    $student->pic= $filename;

}

else{

    echo "Pic was not save";

}

$student->save();

return redirect()->back()->with('success','Data was Successfully Added!');

}

Model code

<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class insertModel extends Model

{

    public $timestamps = false;

    protected $table='students';

    protected $fillable =['name','last\_name','address','pic'

    ];

}

Route

route::view('insert','insert');

route::resource('insert','insertData');

**Task 6 Retrieving data from database**

View page code where all the data retrieve

<!DOCTYPE html>

<html lang="en">

<head>

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm" crossorigin="anonymous">

    <title>show data</title>

</head>

<body>

<div class="row">

<div class="col-md-12">

<br>

<h3 align="center">Show Data</h3><br>

<div class="container">

@if(\Session::has('success'))

<div class="alert alert-danger">

<p>{{\Session::get('success')}} </p>

</div>

@endif

</div>

<center><table class="table table-hover table-dark" style="width:auto"  height="auto"></center>

  <thead>

    <tr>

    <th scope="col">#</th>

    <th scope="col">First Name</th>

    <th scope="col">Last Name</th>

    <th scope="col">Address</th>

      <th scope="col">Picture</th>

      <th scope="col">Action</th>

    </tr>

  </thead>

  <tbody>

@foreach($students as $row)

<tr>

<td>{{$row['id']}}</td>

<td>{{$row['name']}}</td>

<td>{{$row['last\_name']}}</td>

<td>{{$row['address']}}</td>

<th> <img src="{{ asset('taskSolution/Applicant\_pic/' . $row['pic']) }}" alt="pic" width="90px" height="60px">   </th>

<td>

<form method="post" action="{{action('insertData@destroy', $row['id'])}}">

@csrf

<input type="hidden" name="\_method" value="DELETE">

<button type="submit" class="btn btn-danger" >Delete</button>

</form>

</td>

<td>

<a href="{{action('insertData@edit', $row ['id'])}}"><button type="submit" class="btn btn-success" >Update</button></a>

</td>

</tr>

@endforeach

</tbody>

</table>

</div>

</div>

</body>

</html>

Controller code or view code

public function index()

    {

        $students = insertModel::all()->toArray();

        return view('viewData', compact ('students'));

    }

**Task no 07 Delete the data**

Controller or delete code

 public function destroy($id)

    {

        $student = insertModel::find($id);

        $student->delete();

        return redirect()->back()->with('success','Data was Deleted Successfully!');

    }

View page code

<td>

<form method="post" action="{{action('insertData@destroy', $row['id'])}}">

@csrf

<input type="hidden" name="\_method" value="DELETE">

<button type="submit" class="btn btn-danger" >Delete</button>

</form>

</td>

**Task no 08 Where clause in MySQL**

SELECT \* FROM `students` WHERE id=5 (particular data search)

SELECT \* FROM `students` WHERE name='ali' (particular data search)

SELECT \* FROM `students` WHERE address='gujranwala' (particular data search)

SELECT \* FROM `students` WHERE age > 34 (particular data search)

SELECT \* FROM `students` WHERE age < 34 (particular data search)

SELECT \* FROM `students` WHERE age=55 (particular data search)

SELECT last\_name FROM `students` WHERE id='5'

**Task no 09 Retrieving particular data from database**

SELECT \* FROM `students` WHERE id=5 (particular data search)

SELECT \* FROM `students` WHERE name='ali' (particular data search)

SELECT \* FROM `students` WHERE address='gujranwala' (particular data search)

**Task no 10 ORDER BY clause in MySQL**

SELECT \* FROM `students` WHERE id=5 ORDER BY name DES (Name column display ascending order)

SELECT \* FROM `students` WHERE id=5 ORDER BY name DESC (Name column display descending order)

SELECT \* FROM `students` ORDER BY name DESC

SELECT DISTINCT address FROM `students` (Reduce duplicity)

***Insert data query***

INSERT INTO `students`(`id`, `name`, `last\_name`, `address`,) VALUES (1,ali,kamran,gujranwala)

***Update query***

UPDATE `students` SET `id`=1,`name`=[kamran],`last\_name`=ali],`address`=[lahore], WHERE id=1

***Delete query***

DELETE FROM `students` WHERE `students`.`id` = 4