**PROGRAM:-**

const fs=require('fs');

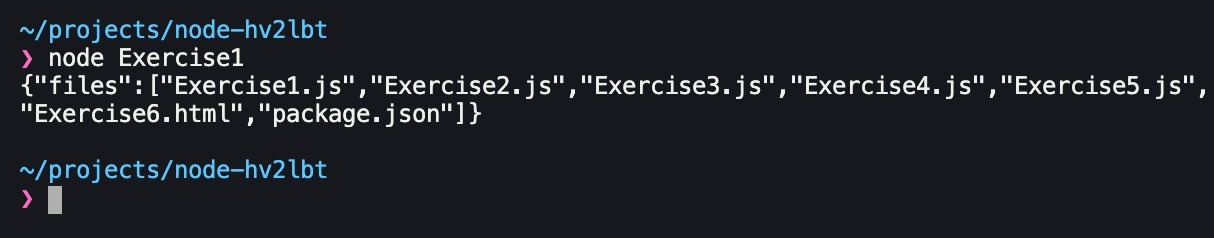
fs.readdir('./', (err, files) => {

let files\_json = JSON.stringify({files})

console.log(files\_json)

});

**OUTPUT:-**



**PROGRAM:-**

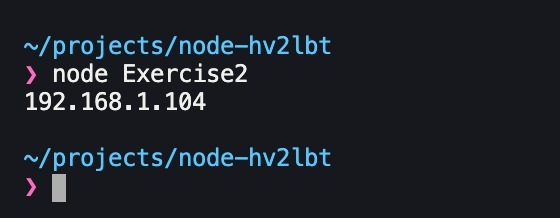
var os = require( 'os' );

var networkInterfaces = os.networkInterfaces();

var arr = networkInterfaces['en1']

console.log(arr[1].address)

**OUTPUT:-**



**PROGRAM:-**

var http = require('http');

function checkCookie (request) {

rc = request.headers.cookie;

if(rc){

return true

}

else{

return false

}

}

http.createServer(function (request, response) {

response.writeHead(200, {'Content-Type': 'text/plain'});

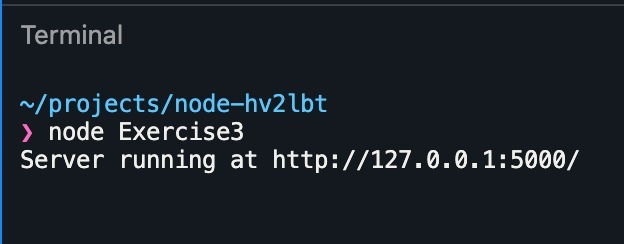
response.end('Cookie Present : '+checkCookie(request));

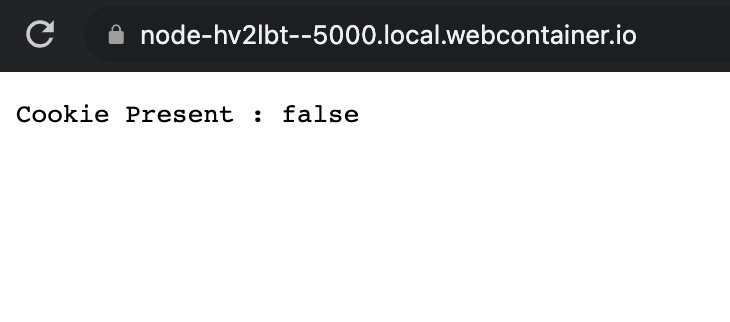
//ßconsole.log(checkCookie(request))

}).listen(5000);

console.log('Server running at <http://127.0.0.1:5000/>');

**OUTPUT:-**





**PROGRAM:-**

const express = require('express'); const mysql = require('mysql');

// Create connection

const db = mysql.createConnection({ host : 'localhost',

user : 'root', password : 'gokulraj', database : 'time'

});

// Connect db.connect((err) => {

if(err){

throw err;

});

}

console.log('MySql Connected...');

const app = express();

// Create DB

app.get('/createdb/:txt', (req, res) => { var tx=req.param('txt');

let sql = 'CREATE DATABASE '+tx;

db.query(sql, (err, result) => { if(err) throw err; console.log(result); res.send('Database created...');

});

});

// Insert data for table 1 app.get('/addcardata/:carname/:companyname', (req, res) => {

let post = {name:req.param('carname')+'', companyname:req.param('companyname')+''};

let sql = 'INSERT INTO cars SET ?';

let query = db.query(sql, post, (err, result) => { if(err) throw err;

console.log(result); res.send('Post added...');

});

});

// Insert data for table 1 app.get('/adduserdata/:uname', (req, res) => {

let post = {User\_name:req.param('uname')+''}; let sql = 'INSERT INTO user SET ?';

let query = db.query(sql, post, (err, result) => { if(err) throw err;

console.log(result); res.send('Post added...');

});

});

// Create table1 app.get('/createtable1', (req, res) => {

let sql = 'CREATE TABLE cars(modelno int AUTO\_INCREMENT, name VARCHAR(255), companyname VARCHAR(255), PRIMARY KEY(modelno))';

db.query(sql, (err, result) => { if(err) throw err; console.log(result); res.send('Car table created...');

});

});

// Create table2 app.get('/createtable2', (req, res) => {

let sql = 'CREATE TABLE user(modelno int AUTO\_INCREMENT, User\_name VARCHAR(255), PRIMARY KEY(modelno))';

db.query(sql, (err, result) => { if(err) throw err; console.log(result); res.send('user table created...');

});

});

// Select result by joining the table... app.get('/selecttables', (req, res) => {

let sql = 'select \* from cars Natural join user'; db.query(sql, (err, result) => {

if(err) throw err; console.log('printed'); res.send(result);

});

});

// Select result by carname... app.get('/searchbycar/:txt', (req, res) => {

var tx=req.param('txt');

let sql = 'select \* from cars Natural join user where name

='+mysql.escape(tx); db.query(sql, (err, result) => {

if(err) throw err; console.log('printed'); res.send(result);

});

});

// Select result by model number... app.get('/searchbycarnumber/:txt', (req, res) => {

var tx=req.param('txt');

let sql = 'select \* from cars Natural join user where modelno

='+mysql.escape(tx); db.query(sql, (err, result) => {

if(err) throw err; console.log('printed'); res.send(result);

});

});

// Select result by user... app.get('/searchbyusername/:txt', (req, res) => {

let sql = 'select \* from cars Natural join user where User\_name

='+mysql.escape(tx); db.query(sql, (err, result) => {

if(err) throw err; console.log('printed'); res.send(result);

});

});

// Display first 5 rows... app.get('/displayfirst5', (req, res) => {

var tx=req.param('txt');

let sql = 'select \* from cars Natural join user limit 5'; db.query(sql, (err, result) => {

if(err) throw err; console.log('printed'); res.send(result);

});

});

// Display recently added 5 rows... app.get('/displayrescently5', (req, res) => {

var tx=req.param('txt');

let sql = 'select \* from cars Natural join user order by cars.modelno desc limit 5';

db.query(sql, (err, result) => { if(err) throw err; console.log('printed'); res.send(result);

});

});

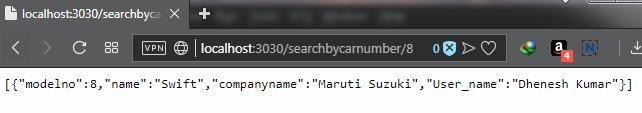
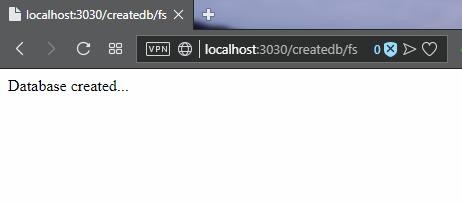
app.listen('3030', () => {

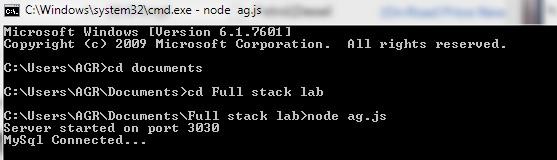
console.log('Server started on port 3030');

});

**OUTPUT:-**







**PROGRAM:-**

var MongoClient = require('mongodb').MongoClient; const express = require('express');

var url="mongodb://localhost:27017/"; var dbo;

var dbase='fs';

// Connect to mongodb MongoClient.connect("mongodb://localhost:27017/", function(err, db) {

if(!err) {

console.log("We are connected");

}

});

const app = express();

// Create DB and Collection app.get('/createdb/:txt/:txtt', (req, res) => {

dbase = req.param('txt'); var txx = req.param('txtt');

MongoClient.connect(url, function(err, db) { if (err) throw err;

dbo = db.db(dbase); dbo.createCollection(txx, function(err, res1) {

if (err) throw err;

res.send("Database and Collection created!");

});

});

});

// Insert data for Document 1 app.get('/addcardata/:modelno/:carname/:companyname', (req, res) => {

MongoClient.connect(url, function(err, db) { if (err) throw err;

var dbo = db.db(dbase); var myobj =

{ modelno:req.param('modelno')+'',name:req.param('carname')+'', companyname:req.param('companyname')+''};

dbo.collection("cars").insertOne(myobj, function(err, res1) { if (err) throw err;

res.send(" document inserted"); db.close();

});

if (err) throw err;

var dbo = db.db(dbase); var myobj =

{ modelno:req.param('modelno')+'',uname:req.param('username')+''}; dbo.collection("user").insertOne(myobj, function(err, res1) {

if (err) throw err;

res.send(" document inserted"); db.close();

});

});

});

// Select result by joining the table... app.get('/selecttables', (req, res) => {

MongoClient.connect(url, function(err, db) { if (err) throw err;

var dbo = db.db(dbase); dbo.collection('cars').aggregate([

{ $lookup:

{

from: 'user', localField: 'modelno',

foreignField: 'modelno', as: 'cars'

}

}

]).toArray(function(err, res1) { if (err) throw err;

res.send(JSON.stringify(res1)); db.close();

});

});

});

// Select result by carname... app.get('/searchbycarname/:txt', (req, res) => {

var tx=req.param('txt')+''; MongoClient.connect(url, function(err, db) {

if (err) throw err;

var dbo = db.db(dbase); dbo.collection("cars").find({name:tx}).toArray(function(err, result) {

if (err) throw err; res.send(result); db.close();

});

});

});

// Select result by carno... app.get('/searchbycarno/:txt', (req, res) => {

var tx=req.param('txt')+''; MongoClient.connect(url, function(err, db) {

});

if (err) throw err;

var dbo = db.db(dbase); dbo.collection("cars").find(

{modelno:tx}).toArray(function(err, result) {

if (err) throw err; res.send(result); db.close();

});

});

});

})

// Select result by username... app.get('/searchbyusername/:txt', (req, res) => {

var tx=req.param('txt')+''; MongoClient.connect(url, function(err, db) {

if (err) throw err;

var dbo = db.db(dbase); dbo.collection("user").find({uname:tx}).toArray(function(err, result) {

if (err) throw err; res.send(result); db.close();

});

});

});

// Display first 5 rows... app.get('/displayfirst5', (req, res) => {

MongoClient.connect(url, function(err, db) { if (err) throw err;

var dbo = db.db(dbase); dbo.collection('cars').find().limit(5).toArray(function(err, result) {

if (err) throw err; res.send(result); db.close();

**OUTPUT:-**



**PROGRAM:-**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Exercise 6</title>

<style>

.avatar{

height: 150px;

width: 150px;

border-radius: 50%;

border: solid grey;

border-width: 2px;

display: block;

margin-left: auto;

margin-right: auto;

}

</style>

</head>

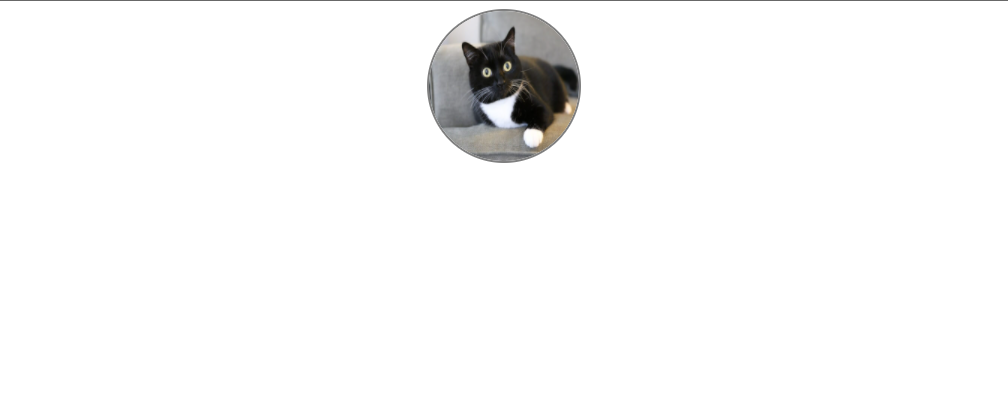
<body>

<img src="./cat.jpeg" class="avatar">

</body>

</html>

**OUTPUT:-**



**PROGRAM:-**

**Marks.sql:-**

-- phpMyAdmin SQL Dump

-- version 4.8.0.1

-- https://www.phpmyadmin.net/

--

-- Host: 127.0.0.1

-- Generation Time: Nov 04, 2021 at 08:40 PM

-- Server version: 10.1.32-MariaDB

-- PHP Version: 5.6.36

SET SQL\_MODE = "NO\_AUTO\_VALUE\_ON\_ZERO";

SET AUTOCOMMIT = 0;

START TRANSACTION;

SET time\_zone = "+00:00";

/\*!40101 SET @OLD\_CHARACTER\_SET\_CLIENT=@@CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET @OLD\_CHARACTER\_SET\_RESULTS=@@CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET @OLD\_COLLATION\_CONNECTION=@@COLLATION\_CONNECTION \*/;

/\*!40101 SET NAMES utf8mb4 \*/;

--

-- Database: `FS-Lab`

--

-- --------------------------------------------------------

--

-- Table structure for table `students`

--

CREATE TABLE `students` (

`fname` varchar(50) COLLATE latin1\_general\_ci NOT NULL,

`lname` varchar(50) COLLATE latin1\_general\_ci NOT NULL,

`DOB` date NOT NULL,

`branch` varchar(10) COLLATE latin1\_general\_ci NOT NULL,

`score` int(11) NOT NULL,

`regno` int(11) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1\_general\_ci;

--

-- Dumping data for table `students`

--

INSERT INTO `students` (`fname`, `lname`, `DOB`, `branch`, `score`, `regno`) VALUES

('Bill', 'Gates', '1987-03-05', 'CSE-C', 92, 15108006),

('Steve', 'Jobs', '1978-02-16', 'CSE-B', 98, 15108007),

('James', 'Bond', '1976-01-26', 'CSE-A', 90, 15108008),

('Elon ', 'Musk', '1989-06-01', 'CSE-A', 100, 15108009),

('Tony ', 'stark', '1980-08-10', 'Mech-D', 63, 151080010),

('Will ', 'Smith', '1977-05-30', 'FT-B', 87, 151080011);

--

-- Indexes for dumped tables

--

--

-- Indexes for table `students`

--

ALTER TABLE `students`

ADD PRIMARY KEY (`regno`);

COMMIT;

/\*!40101 SET CHARACTER\_SET\_CLIENT=@OLD\_CHARACTER\_SET\_CLIENT \*/;

/\*!40101 SET CHARACTER\_SET\_RESULTS=@OLD\_CHARACTER\_SET\_RESULTS \*/;

/\*!40101 SET COLLATION\_CONNECTION=@OLD\_COLLATION\_CONNECTION \*/;

**Marks.php:-**

<?php

$conn = new mysqli('localhost', 'root', '', 'Results');

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

else{

$sql="SELECT \* FROM marks";

$result=$conn->query($sql);

while($row = $result->fetch\_assoc()) {

$records[] = $row;

}

echo json\_encode($records);

}

?>

**Results.htm**

<!DOCTYPE html>

<html>

<script src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.5.11/angular.min.js"></script>

<body>

<h1>Welcome</h1>

<div ng-app="myApp" ng-controller="customersCtrl" ng-init="searchstd()">

<select ng-model="SortColumn">

<option value="fname">Name Asc</option>

<option value="-fname">Name Desc</option>

<option value="lname">last name Asc</option>

<option value="-lname">last name Desc</option>

<option value="DOB">DOB ASC</option>

<option value="-DOB">DOB Desc</option>

<option value="branch">branch ASC</option>

<option value="-branch">branch DESC</option>

<option value="score">score Asc</option>

<option value="-score">score Desc</option>

</select>

<table border="1">

<tr>

<th>First Name</th>

<th>Last Name</th>

<th>Date Of Birth</th>

<th>Branch & Sec</th>

<th>Score</th>

<th>Register Number</th>

</tr>

<tr ng-repeat="x in getdetails|orderBy:SortColumn">

<td>{{ x.fname }}</td>

<td>{{x.lname}}</td>

<td>{{x.DOB}}</td>

<td>{{x.branch}}</td>

<td>{{ x.score }}</td>

<td>{{x.regno}}</td>

</tr>

</table>

</div>

<script>

var app = angular.module('myApp', []);

app.controller('customersCtrl', function($scope, $http) {

$scope.searchstd=function()

{

$http.get('marks.php')

.success(function (data)

{

$scope.getdetails = data;

});

}

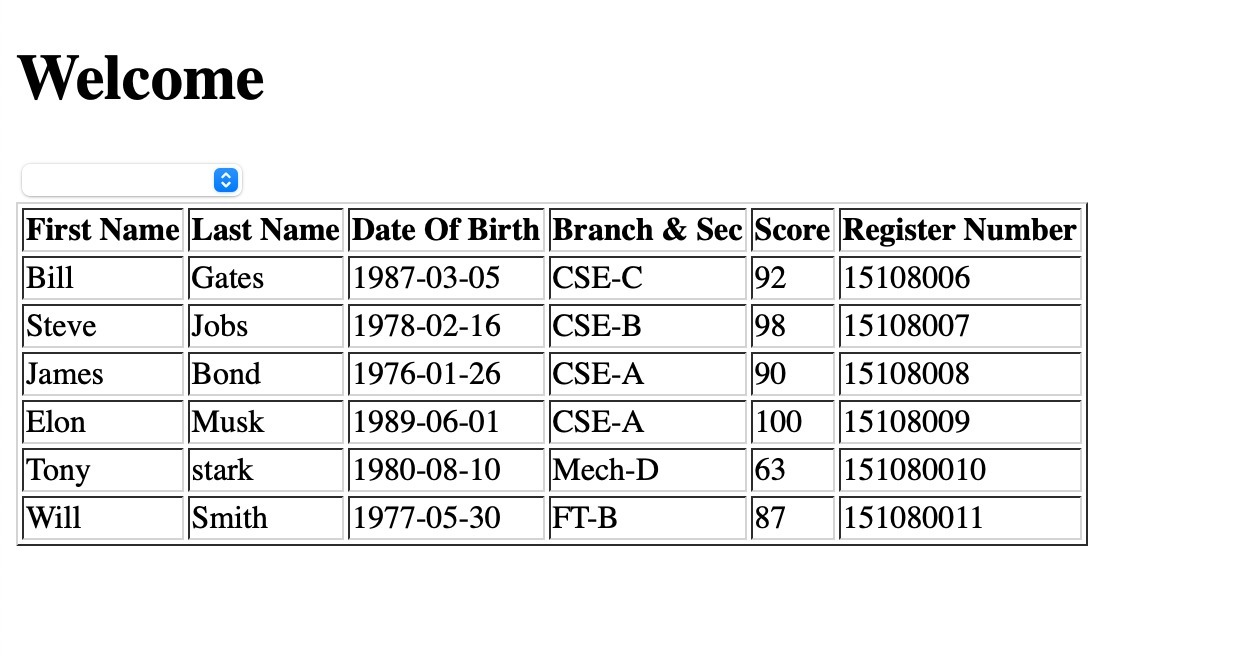
});

</script>

</body>

</html

**OUTPUT:-**



**PROGRAM:-**

**Login.php:-**

<!DOCTYPE html>

<html>

<body align="center">

<?php

try {

// Create a Redis Instance

$redis = new Redis();

$redis->connect('127.0.0.1', 6379);

} catch (Exception $ex) {

echo $ex->getMessage();

}

session\_start();

$conn=mysqli\_connect('localhost','root','','users');

if(isset($\_POST['submit'])){

if(isset($\_POST['email']) && isset($\_POST['password'])){

$email=$\_POST['email'];

$password=$\_POST['password'];

$sql = "SELECT userid,username FROM login WHERE email = '$email' and passcode = '$password'";

$result = mysqli\_query($conn,$sql);

$count = mysqli\_num\_rows($result);

$row = mysqli\_fetch\_array($result);

if(is\_array($row)) {

$\_SESSION["id"] = $row['userid'];

$\_SESSION["username"] = $row['username'];

}else {

$message = "Invalid Username or Password!";

}

if($\_SESSION["username"]){

echo 'here';

$redis->set('E-mail', $email);

$name = $redis->get('E-mail');

echo $name;

header('location:home.php');

}else{

echo 'Nope';

}

}

}

if(isset($\_COOKIE["Email"])&&isset($\_COOKIE["Password"])){

$sql = "SELECT userid FROM login WHERE email = '".$\_COOKIE["Email"]."' and passcode = '".$\_COOKIE["Password"]."'";

$result = mysqli\_query($conn,$sql);

$count = mysqli\_num\_rows($result);

if($count==1){

echo 'here';

header('location:home.php');

}else{

echo 'Username or Password is wrong';

}

}

?>

<form method="post">

<h1>Welcome To My Web Page</h1>

<input type="email" placeholder="Enter Your Email ID" name="email"/><br><br>

<input type="Password" placeholder="Enter Your Password" name="password"/><br><br>

<input type="submit" name="submit"/><br><br>

<a href="register.php">New User? Click here</a>

</form>

</body>

</html>

**Register.php:-**

<!DOCTYPE html>

<html>

<body align="center">

<?php

$conn=mysqli\_connect('localhost','root','','users');

if(isset($\_POST['submit'])){

session\_start();

if(isset($\_POST['email']) && $\_POST['password'] && isset($\_POST['name']) && $\_POST['phone']){

$email=$\_POST['email'];

$password=$\_POST['password'];

$name=$\_POST['name'];

$phone=$\_POST['phone'];

$sql = "INSERT INTO login (email,passcode,username,phone) VALUES('$email','$password','$name','$phone')";

if ($conn->query($sql) === TRUE) {

setcookie("Email", $email);

setcookie("Password", $password);

$\_SESSION["username"]=$name;

echo "New record created successfully";

header('location:home.php');

} else {

echo "Error: " . $sql . "<br>" . $conn->error;

}

}

}

?>

<form method="post">

<h1>Welcome To My Web Page</h1>

<input type="text" placeholder="Enter Your Name" name="name"/><br><br>

<input type="email" placeholder="Enter Your Email ID" name="email"/><br><br>

<input type="Password" placeholder="Enter Your Password" name="password"/><br><br>

<input type="phone" placeholder="Enter Your Phone Number" name="phone"/><br><br>

<input type="submit" name="submit"/><br><br>

<a href="login.php">Already have an account? Click here</a>

</form>

</body>

</html>

**Home.php:-**

<?php

session\_start();

if(isset($\_POST["logout"])){

unset($\_SESSION["id"]);

unset($\_SESSION["username"]);

header("Location:login.php");

}

$redis = new Redis();

$redis->connect('127.0.0.1', 6379);

$email = $redis->get('E-mail');

?>

<!DOCTYPE html>

<html>

<body>

<style>

h1{

color:#fff;

}

body{

background-color:#000;

}

Button{

color:white;

background-color:red;

cursor: pointer;

text-align: center;

align:center;

border-radius:5px;

left: 50%;

width:10%;

height:5%;

position: absolute;

}

</style>

<h1>Hello, <?php echo $\_SESSION["username"]; ?></h1>

<h1><?php echo $email; ?></h1>

<marquee width="100%" direction="left" height="100px">

<h1>Welcome to My Home Page :)</h1>

</marquee>

<form method="post">

<Button name="logout">logout</Button>

</form>

</body>

</html>

**Email.js:-**

const redis = require('redis');

var nodemailer = require('nodemailer');

const client = redis.createClient();

client.on('connect', function() {

console.log('Connected!');

});

client.get('E-mail', function(err, reply) {

console.log(reply);

var transporter = nodemailer.createTransport({

service: 'gmail',

auth: {

user: 'unknown@gmail.com',

pass: ''

}

});

var mailOptions = {

from: 'unknown@gmail.com',

to: reply,

subject: 'Hello User',

text: 'Welcome User thank you for using our services good luck'

};

transporter.sendMail(mailOptions, function(error, info){

if (error) {

console.log(error);

} else {

console.log('Email sent: ' + info.response);

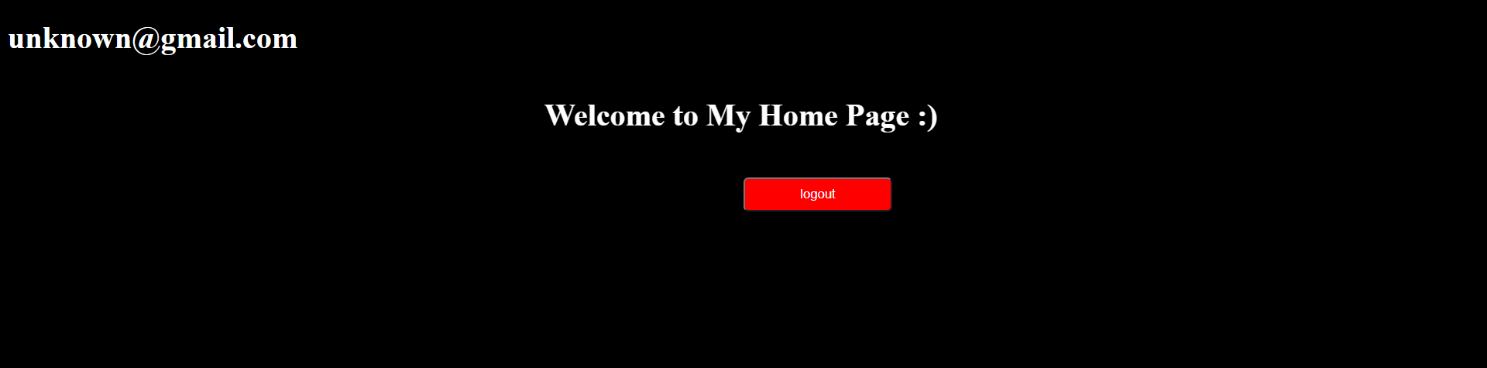
}

}); });

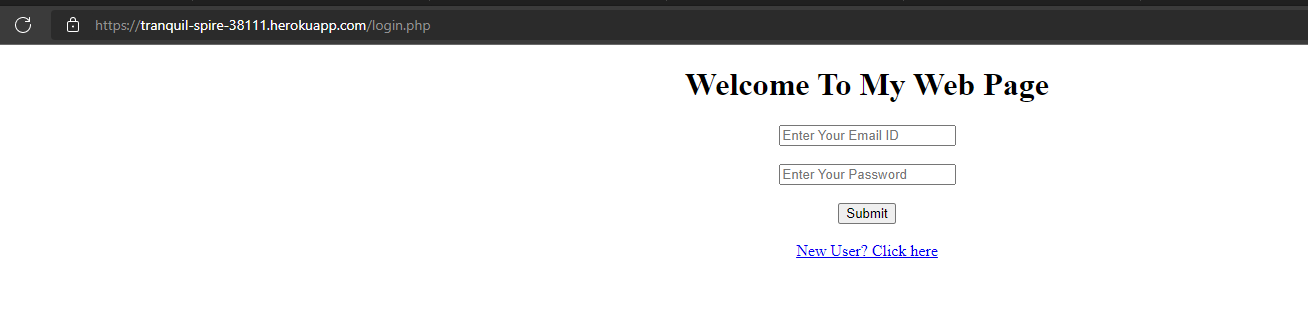
**OUTPUT:-**







**OUTPUT:-**



**PROGRAM:-**

**MainActivity.java**

import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.view.Menu; import android.view.MenuItem; import android.view.View; import android.widget.Button; import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

Button button0 , button1 , button2 , button3 , button4 , button5 , button6 , button7 , button8 , button9 , buttonAdd , buttonSub , buttonDivision , buttonMul , button10 , buttonC , buttonEqual ;

EditText edt1 ;

float mValueOne , mValueTwo ;

boolean mAddition , mSubtract ,mMultiplication ,mDivision ; @Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

button0 = (Button) findViewById(R.id.button0); button1 = (Button) findViewById(R.id.button1); button2 = (Button) findViewById(R.id.button2); button3 = (Button) findViewById(R.id.button3); button4 = (Button) findViewById(R.id.button4); button5 = (Button) findViewById(R.id.button5); button6 = (Button) findViewById(R.id.button6); button7 = (Button) findViewById(R.id.button7); button8 = (Button) findViewById(R.id.button8); button9 = (Button) findViewById(R.id.button9); button10 = (Button) findViewById(R.id.button10); buttonAdd = (Button) findViewById(R.id.buttonadd); buttonSub = (Button) findViewById(R.id.buttonsub); buttonMul = (Button) findViewById(R.id.buttonmul);

buttonDivision = (Button) findViewById(R.id.buttondiv); buttonC = (Button) findViewById(R.id.buttonC); buttonEqual = (Button) findViewById(R.id.buttoneql); edt1 = (EditText) findViewById(R.id.edt1); button1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) { edt1.setText(edt1.getText()+"1");

}

});

button2.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"2");

}});

button3.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"3");

}

});

button4.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"4");

}

});

button5.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"5");

}

});

button6.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"6");

}

});

button7.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"7");

}

});

button8.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"8");

}

});

button9.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"9");

}

});

button0.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+"0");

}

});

buttonAdd.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { if (edt1 == null){

edt1.setText("");

}else {

mValueOne = Float.parseFloat(edt1.getText() + ""); mAddition = true;

edt1.setText(null);

}

}

});

buttonSub.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

mValueOne = Float.parseFloat(edt1.getText() + ""); mSubtract = true ;

edt1.setText(null);

}

});

buttonMul.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

mValueOne = Float.parseFloat(edt1.getText() + ""); mMultiplication = true ;

edt1.setText(null);

}

});

buttonDivision.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

mValueOne = Float.parseFloat(edt1.getText()+""); mDivision = true ;

edt1.setText(null);

}

});

buttonEqual.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

mValueTwo = Float.parseFloat(edt1.getText() + ""); if (mAddition == true){

edt1.setText(mValueOne + mValueTwo +""); mAddition=false;

}

if (mSubtract == true){ edt1.setText(mValueOne - mValueTwo+""); mSubtract=false;

}

if (mMultiplication == true){ edt1.setText(mValueOne \* mValueTwo+"");

mMultiplication=false;

}

if (mDivision == true){ edt1.setText(mValueOne / mValueTwo+""); mDivision=false;

}

}

});

buttonC.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText("");

}

});

button10.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { edt1.setText(edt1.getText()+".");

}

});

}

}

**activity\_main.xml**

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:paddingLeft="@dimen/activity\_horizontal\_margin" android:paddingRight="@dimen/activity\_horizontal\_margin" android:paddingTop="@dimen/activity\_vertical\_margin" android:paddingBottom="@dimen/activity\_vertical\_margin"

tools:context=".MainActivity" android:id="@+id/relative1">

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/edt1"/>

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" android:text="1" android:id="@+id/button1" android:layout\_marginTop="94dp"

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:paddingLeft="@dimen/activity\_horizontal\_margin" android:paddingRight="@dimen/activity\_horizontal\_margin" android:paddingTop="@dimen/activity\_vertical\_margin" android:paddingBottom="@dimen/activity\_vertical\_margin"

tools:context=".MainActivity" android:id="@+id/relative1">

<EditText android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:id="@+id/edt1"/>

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="1" android:id="@+id/button1" android:layout\_marginTop="94dp" android:layout\_below="@+id/edt1" android:layout\_toStartOf="@+id/button4" android:layout\_alignRight="@+id/button4"

android:layout\_alignEnd="@+id/button4" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="2" android:id="@+id/button2" android:layout\_alignTop="@+id/button1" android:layout\_toLeftOf="@+id/button3"

android:layout\_toStartOf="@+id/button3" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content"

android:text="3" android:id="@+id/button3" android:layout\_alignTop="@+id/button2" android:layout\_centerHorizontal="true" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="4" android:id="@+id/button4" android:layout\_below="@+id/button1" android:layout\_toLeftOf="@+id/button2" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="5" android:id="@+id/button5"

android:layout\_alignBottom="@+id/button4" android:layout\_alignLeft="@+id/button2" android:layout\_alignStart="@+id/button2" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="6" android:id="@+id/button6" android:layout\_below="@+id/button3" android:layout\_alignLeft="@+id/button3"

android:layout\_alignStart="@+id/button3" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="7" android:id="@+id/button7" android:layout\_below="@+id/button4" android:layout\_toLeftOf="@+id/button2" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="8" android:id="@+id/button8" android:layout\_below="@+id/button5" android:layout\_alignLeft="@+id/button5"

android:layout\_alignStart="@+id/button5" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="9" android:id="@+id/button9" android:layout\_below="@+id/button6" android:layout\_alignLeft="@+id/button6"

android:layout\_alignStart="@+id/button6" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="+" android:id="@+id/buttonadd" android:layout\_alignTop="@+id/button3" android:layout\_toRightOf="@+id/button3" android:layout\_marginLeft="46dp" android:layout\_marginStart="46dp" android:layout\_alignRight="@+id/edt1" android:layout\_alignEnd="@+id/edt1" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="-" android:id="@+id/buttonsub" android:layout\_below="@+id/buttonadd" android:layout\_alignLeft="@+id/buttonadd" android:layout\_alignStart="@+id/buttonadd" android:layout\_alignRight="@+id/buttonadd"

android:layout\_alignEnd="@+id/buttonadd" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="\*" android:id="@+id/buttonmul" android:layout\_below="@+id/buttonsub" android:layout\_alignLeft="@+id/buttonsub" android:layout\_alignStart="@+id/buttonsub" android:layout\_alignParentRight="true" android:layout\_alignParentEnd="true" />

<Button style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="." android:id="@+id/button10" android:layout\_below="@+id/button7" android:layout\_toLeftOf="@+id/button2" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="0" android:id="@+id/button0" android:layout\_below="@+id/button8" android:layout\_alignLeft="@+id/button8"

android:layout\_alignStart="@+id/button8" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="C" android:id="@+id/buttonC" android:layout\_below="@+id/button9" android:layout\_alignLeft="@+id/button9"

android:layout\_alignStart="@+id/button9" />

<Button

style="?android:attr/buttonStyleSmall" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="/" android:id="@+id/buttondiv" android:layout\_below="@+id/buttonmul" android:layout\_alignLeft="@+id/buttonmul" android:layout\_alignStart="@+id/buttonmul"

android:layout\_alignRight="@+id/buttonmul" android:layout\_alignEnd="@+id/buttonmul" />

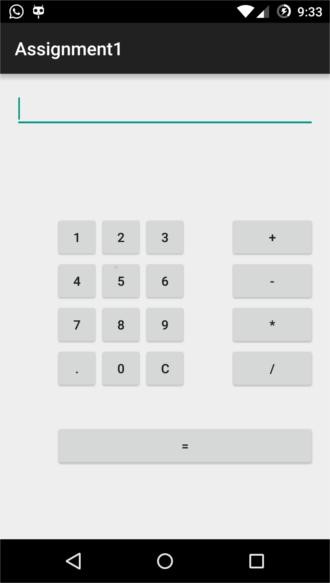
<Button

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="=" android:id="@+id/buttoneql" android:layout\_below="@+id/button0" android:layout\_marginTop="37dp" android:layout\_alignRight="@+id/buttondiv"

android:layout\_alignEnd="@+id/buttondiv" android:layout\_alignLeft="@+id/button10" android:layout\_alignStart="@+id/button10" />

</RelativeLayout>

**OUTPUT:-**



**PROGRAM:-**

import UIKit

class ViewController: UIViewController { var numberOnScreen: Double = 0;

var previousNumber: Double = 0; var performingMath = false;

var operation = 0;

@IBAction func numbers(\_ sender: UIButton) { if performingMath == true {

label.text = String(sender.tag-1) numberOnScreen = Double(label.text!)! performingMath = false

}

else {

label.text = label.text! + String(sender.tag-1) numberOnScreen = Double(label.text!)!}

}

@IBOutlet weak var label: UILabel! @IBAction func buttons(\_ sender: UIButton) {

if label.text != "" && sender.tag != 11 && sender.tag != 16{ previousNumber = Double(label.text!)!

if sender.tag == 12 { //Divide label.text = "/";

}

if sender.tag == 13 { //Multiply label.text = "x";

}

if sender.tag == 14 { //Subtract label.text = "-";

}

if sender.tag == 15 { //Add label.text = "+";

}

operation = sender.tag performingMath = true;

}

else if sender.tag == 16 { if operation == 12{ //Divide

label.text = String(previousNumber / numberOnScreen)

}

else if operation == 13{ //Multiply

label.text = String(previousNumber \* numberOnScreen)

}

else if operation == 14{ //Subtract

label.text = String(previousNumber - numberOnScreen)

}

else if operation == 15{ //Add

label.text = String(previousNumber + numberOnScreen)}}

else if sender.tag == 11{ label.text = "" previousNumber = 0;

numberOnScreen = 0;

operation = 0;

}

}

override func viewDidLoad() { super.viewDidLoad()

// Do any additional setup after loading the view, typically from a nib.

}

override func didReceiveMemoryWarning() { super.didReceiveMemoryWarning()

// Dispose of any resources that can be recreated.

}

}

**OUTPUT:-**

