

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

**AI23431 – WEB TECHNOLOGY AND MOBILE APPLICATION**

**(REGULATION 2023)**

# RAJALAKSHMI ENGINEERING COLLEGE

**Thandalam, Chennai-602015**

**Name: MONISH S**

**Register No: 231501103**

**Year / Branch / Section: 2nd / AIML / A Semester: IV**

**Academic Year: 2024 - 2025**

|  |  |
| --- | --- |
| **EX.NO: 1** | **HTML - WEB PAGE TO EMBED A MAP ALONG WITH HOTSPOT, FRAMES AND LINKS** |

# PROGRAM:-

## index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>India Map</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<img src="india.png" alt="india\_map" id="in\_map" usemap="#in\_map" >

<map name="in\_map">

<area shape="rect" coords="191, 592, 242, 604" href="tn.html" alt="tamilnadu" target="\_blank">

</map>

<a href="https://en.wikipedia.org/wiki/India">INDIA</a>

<iframe src="https://en.wikipedia.org/wiki/India" frameborder="0"></iframe>

</body>

</html>

## tn.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Tamil Nadu</title>

<style>

#tn\_map{ width: 500px; height: auto;

}

</style>

</head>

<body>

<h1>TAMIL NADU</h1>

<p>

Tamil Nadu is a southern state in India.

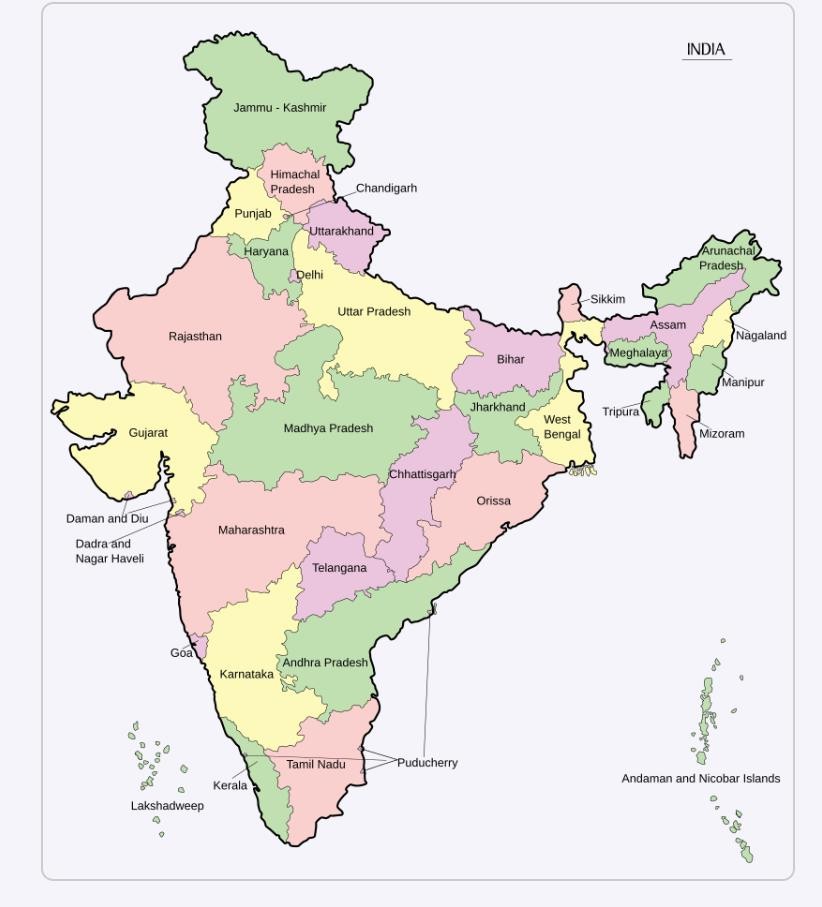
</p>

<img src="tamilnadu.png" alt="" id="tn\_map">

</body>

</html>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 2** | **CSS - WEB PAGE USING INTERNAL, EXTERNAL AND INLINE CSS** |

# PROGRAM:-

## index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>WT\_EX.2</title>

<style>

.internal{

background-color: aquamarine; color: brown;

} #p2{

width: 400px; height: auto;

}

</style>

<link rel="stylesheet" href="style.css">

</head>

<body>

<p style="background-color:black;">

<h1 style="color: blue; background-color:burlywood;">Inline CSS</h1>

<img src="P1.jpg" alt="img1" style="width: 400px; height:auto;">

</p>

<p>

<h1 class="internal">Internal CSS</h1>

<img src="P2.jpg" alt="" id="p2">

</p>

<p>

<h1 class="external">External CSS</h1>

<img src="P3.jpg" alt="" id="p3">

</p>

</body>

</html>

## style.css

.external{

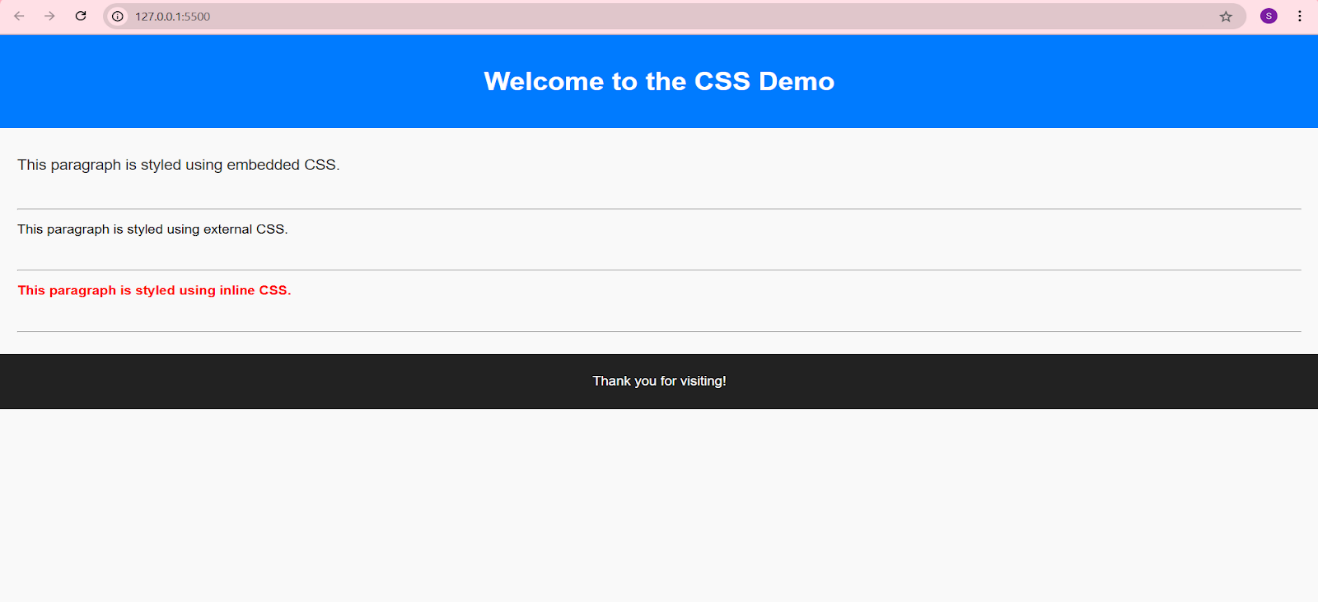
background-color:blueviolet; color:antiquewhite;

} #p3{

width: 400px; height: auto;

}

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 3** | **JAVASCRIPT TO VALIDATE REGISTRATION FORM** |

# PROGRAM:-

## index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Registration form</title>

</head>

<body>

<script>

function validateForm(){

const name=document.getElementById("fname").value; const fn\_error=document.getElementById("fn\_error"); const alphabets = /^[A-Za-z]+$/;

const password=document.getElementById("pswd").value; const p\_error=document.getElementById("p\_error");

const email=document.getElementById("email").value;

const email\_error=document.getElementById("email\_error");

const emailPattern = /^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;

const mobile=document.getElementById("mob").value; const mob\_error=document.getElementById("mob\_error");

const lname=document.getElementById("lname").value; const ln\_error=document.getElementById("ln\_error");

const address=document.getElementById("adrs").value; const adrs\_error=document.getElementById("adrs\_error");

let isValid=true;

fn\_error.innerHTML = ""; p\_error.innerHTML = ""; email\_error.innerHTML = ""; mob\_error.innerHTML = ""; ln\_error.innerHTML = ""; adrs\_error.innerHTML = "";

if(name.length <6 ){

fn\_error.innerHTML="Name must be at least 6 characters long."; isValid= false;

}

else if (!name.match(alphabets)) {

fn\_error.innerHTML = "Name must contain only alphabets."; isValid= false;

}

if(password.length<6){

p\_error.innerHTML="Password must be at least 6 characters long."; isValid= false;

}

if(!email.match(emailPattern)){ email\_error.innerHTML="Invalid format."; isValid=false;

}

if (mobile.length !== 10 || isNaN(mobile)) {

mob\_error.innerHTML = "Mobile number must be exactly 10 digits.";

isValid = false;

}

if(lname.trim()===""){

ln\_error.innerHTML="Last Name cannot be empty."; isValid=false;

}

if(address.trim()===""){ adrs\_error.innerHTML="Address cannot be empty."; isValid=false;

}

if(isValid){ alert("Submitted!"); return true;

}

return false;

}

</script>

<h1 style="text-align: center;">Registration Form</h1>

<form action="" onsubmit="return validateForm()">

<label for="fname">First Name:</label>

<input type="text" id="fname" name="First name" placeholder="Name">

<span style="color: red;" id="fn\_error"></span>

<br><br>

<label for="pswd">Password:</label>

<input type="text" id="pswd" name="Password" placeholder="Password">

<span style="color: red;" id="p\_error"></span>

<br><br>

<label for="E-mail">E-mail:</label>

<input type="text" name="E-mail id" id="email" placeholder="E-mail">

<span style="color: red;" id="email\_error"></span>

<br><br>

<label for="Mob.number">Mobile Number:</label>

<input type="number" name="Mobile number" id="mob" placeholder="Mobile Number">

<span style="color: red;" id="mob\_error"></span>

<br><br>

<label for="lname">Last Name:</label>

<input type="text" name="Last Name" id="lname" placeholder="Last Name">

<span style="color: red;" id="ln\_error"></span>

<br><br>

<label for="address">Address:</label>

<input type="text" name="Address" id="adrs" placeholder="Address">

<span style="color: red;" id="adrs\_error"></span>

<br><br>

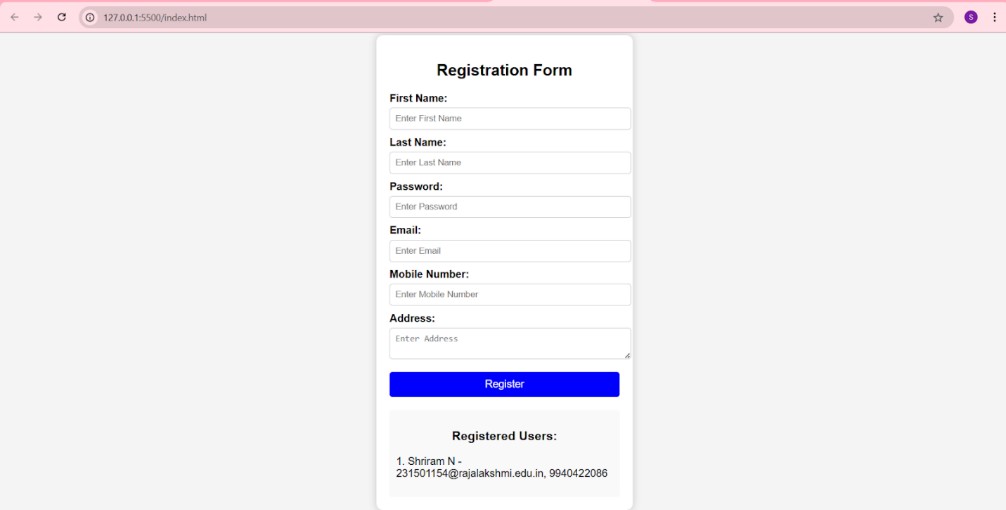
<button>Submit</button>

</form>

</body>

</html>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 4** | **SERVLET TO PRINT “Hello World!”** |

# PROGRAM:-

## HelloWorldServlet.java

import java.io.IOException; import java.io.PrintWriter;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse;

@WebServlet("/hello")

public class HelloWorldServlet extends HttpServlet { private static final long serialVersionUID = 1L;

@Override

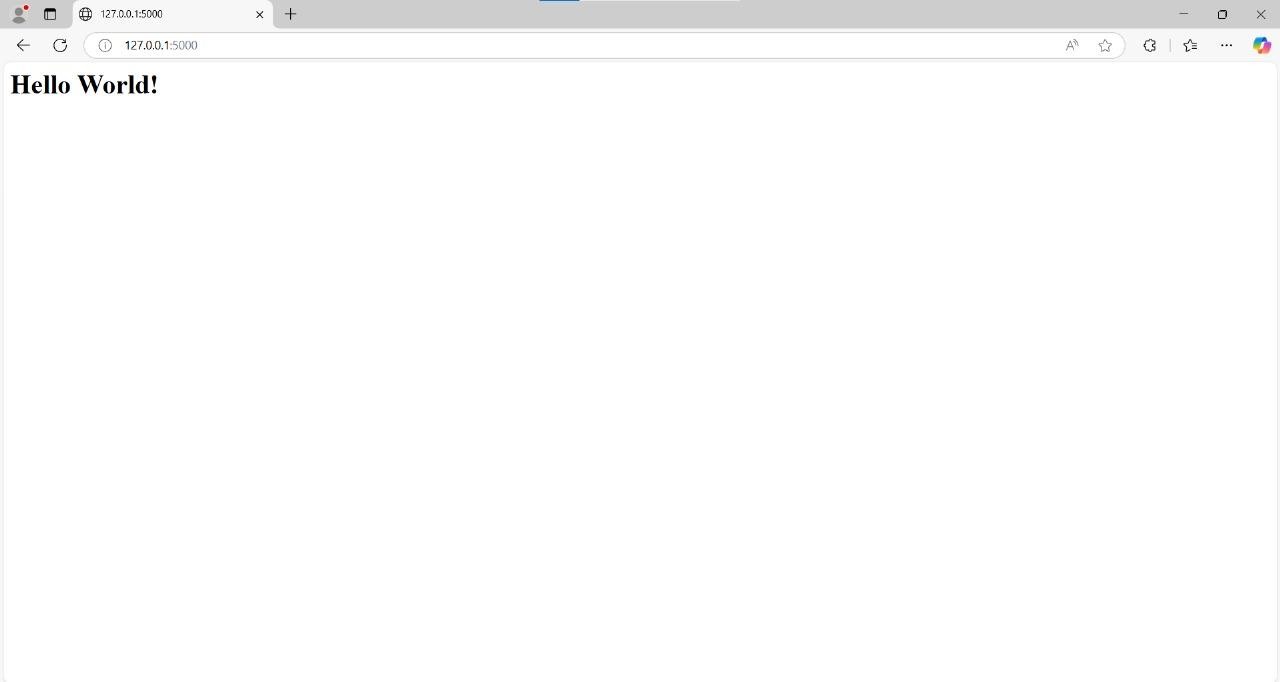
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

response.setContentType("text/html"); PrintWriter out = response.getWriter(); out.println("<html><body>"); out.println("<h1>Hello World</h1>"); out.println("</body></html>");

}

}

# OUTPUT:-

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 5** | **SERVLET TO PROCESS FORM DATA AND DISPLAY ON BROWSER** |

# PROGRAM:-

## FormServlet.java

package com.example; import java.io.IOException;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; @WebServlet("/form")

public class FormServlet extends HttpServlet {

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String name = request.getParameter("name"); String email = request.getParameter("email"); response.setContentType("text/html");

response.getWriter().println("<h1>Form Submitted</h1>"); response.getWriter().println("<p>Name: " + name + "</p>"); response.getWriter().println("<p>Email: " + email + "</p>");

}

}

## index.html

<!DOCTYPE html>

<html>

<head>

<title>Form Example</title>

</head>

<body>

<h1>Submit Your Information</h1>

<form action="form" method="post">

<label for="name">Name:</label><br>

<input type="text" id="name" name="name"><br><br>

<label for="email">Email:</label><br>

<input type="email" id="email" name="email"><br><br>

<input type="submit" value="Submit">

</form>

</body>

</html>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 6** | **SERVLET TO DIFFERENTIATE BETWEEN HTTP GET AND POST** |

# PROGRAM:-

## Form.java

import java.io.IOException; import java.io.PrintWriter;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse;

@WebServlet("/formDemo")

public class Form2 extends HttpServlet {

private static final long serialVersionUID = 1L;

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

response.setContentType("text/html"); PrintWriter out = response.getWriter();

// Handle GET request

String name = request.getParameter("name"); if (name != null) {

out.println("<h3>You submitted via GET: " + name + "</h3>");

}

out.close();

}

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

response.setContentType("text/html"); PrintWriter out = response.getWriter();

// Handle POST request

String name = request.getParameter("name"); if (name != null) {

out.println("<h3>You submitted via POST: " + name + "</h3>");

}

out.close();

}

}

## index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Form Demo</title>

</head>

<body>

<h2>Submit Data Using GET Method</h2>

<form method="GET" action="formDemo"> Name: <input type="text" name="name"><br>

<input type="submit" value="Submit via GET">

</form>

<h2>Submit Data Using POST Method</h2>

<form method="POST" action="formDemo"> Name: <input type="text" name="name"><br>

<input type="submit" value="Submit via POST">

</form>

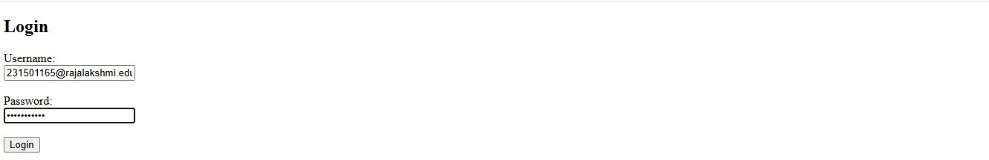
</body>

</html>

# OUTPUT:-

****

****

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 7** | **SERVLET TO DEMONSTRATE SESSION TRACKING**  **USING HttpSession** |

# PROGRAM:-

## index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Login</title>

</head>

<body>

<h2>Login Form</h2>

<form action="login" method="post">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required>

<br><br>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required>

<br><br>

<input type="submit" value="Login">

</form>

</body>

</html>

## LoginServlet.java

import java.io.IOException;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession; @WebServlet("/login")

public class LoginServlet extends HttpServlet { private static final long serialVersionUID = 1L;

protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String username = request.getParameter("username"); String password = request.getParameter("password");

if ("admin".equals(username) && "password".equals(password)) { HttpSession session = request.getSession(); session.setAttribute("username", username); response.sendRedirect("welcome.jsp");

} else {

response.sendRedirect("index.html?error=Invalid credentials");

}

}

}

## LogoutServlet.java

import java.io.IOException;

import javax.servlet.ServletException; import javax.servlet.annotation.WebServlet; import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import javax.servlet.http.HttpSession; @WebServlet("/logout")

public class LogoutServlet extends HttpServlet { private static final long serialVersionUID = 1L;

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

HttpSession session = request.getSession(false);

if (session != null) { session.invalidate();

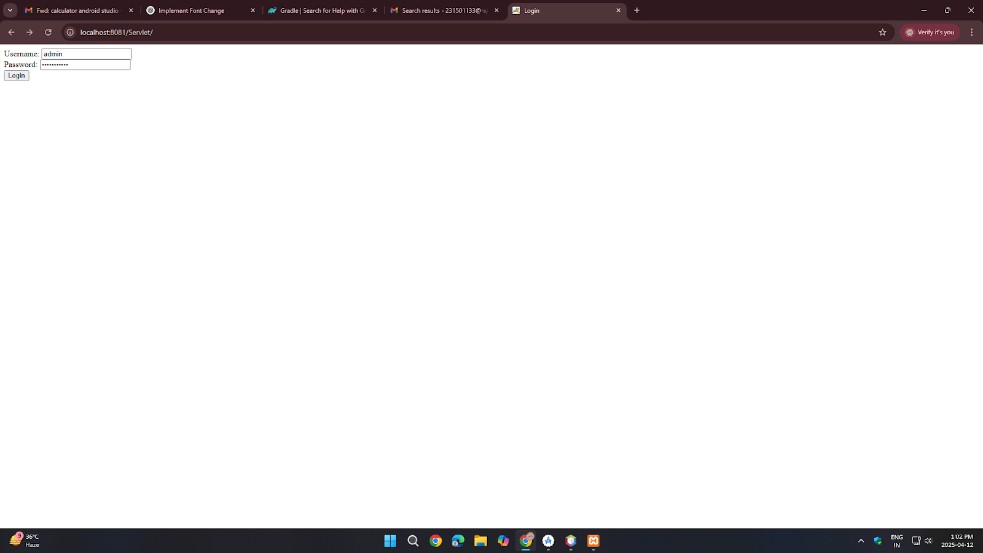
}

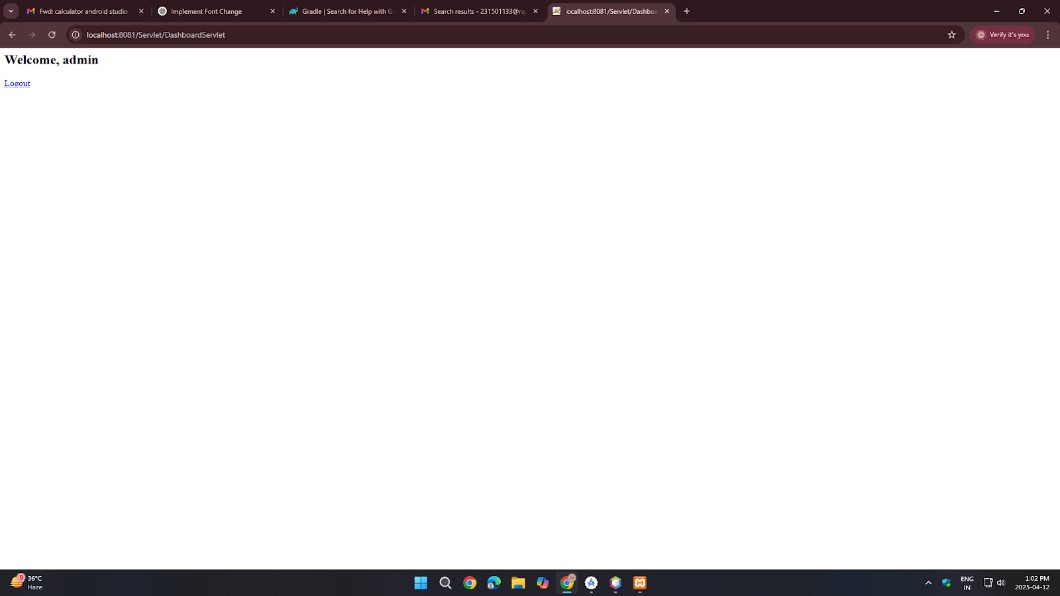
response.sendRedirect("index.html");

}

}

# OUTPUT:-

****

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 8** | **ANDROID APPLICATION - BASIC CALCULATOR** |

# PROGRAM:-

## MainActivity.kt

package com.example.calculatorapp

import androidx.appcompat.app.AppCompatActivity import android.os.Bundle

import android.widget.\*

class MainActivity : AppCompatActivity() { lateinit var num1: EditText

lateinit var num2: EditText lateinit var resultView: TextView lateinit var addBtn: Button lateinit var subBtn: Button lateinit var mulBtn: Button lateinit var divBtn: Button

override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState) setContentView(R.layout.activity\_main)

num1 = findViewById(R.id.num1) num2 = findViewById(R.id.num2)

resultView = findViewById(R.id.resultView) addBtn = findViewById(R.id.addBtn) subBtn = findViewById(R.id.subBtn) mulBtn = findViewById(R.id.mulBtn) divBtn = findViewById(R.id.divBtn)

addBtn.setOnClickListener { calculate('+') }

subBtn.setOnClickListener { calculate('-') } mulBtn.setOnClickListener { calculate('\*') } divBtn.setOnClickListener { calculate('/') }

}

private fun calculate(operator: Char) { val input1 = num1.text.toString() val input2 = num2.text.toString()

if (input1.isEmpty() || input2.isEmpty()) { resultView.text = "Please enter both numbers." return

}

val a = input1.toDouble() val b = input2.toDouble()

val result = when (operator) { '+' -> a + b

'-' -> a - b

'\*' -> a \* b

'/' -> {

if (b == 0.0) {

resultView.text = "Cannot divide by zero." return

} else a / b

}

else -> 0.0

}

resultView.text = "Result: $result"

}

}

## activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="[http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical"

android:padding="20dp">

<EditText android:id="@+id/num1"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:hint="Enter first number" android:inputType="numberDecimal"/>

<EditText android:id="@+id/num2"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:hint="Enter second number" android:inputType="numberDecimal"/>

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:orientation="horizontal" android:gravity="center" android:layout\_marginTop="20dp">

<Button

android:id="@+id/addBtn" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" android:text="+" />

<Button

android:id="@+id/subBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="-" />

<Button

android:id="@+id/mulBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="×" />

<Button

android:id="@+id/divBtn" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="÷" />

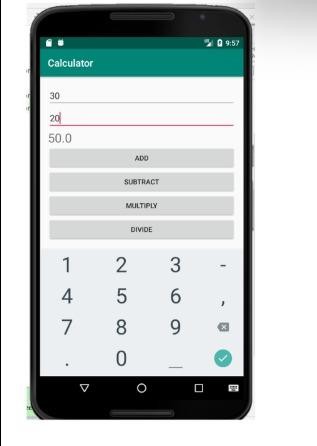
</LinearLayout>

<TextView android:id="@+id/resultView" android:layout\_width="match\_parent" android:layout\_height="wrap\_content"

android:text="Result will be shown here" android:textSize="18sp" android:layout\_marginTop="30dp"/>

</LinearLayout>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 10** | **ANDROID APPLICATION TO CHANGE FONT AND COLOR OF TEXT** |

# PROGRAM:-

## MainActivity.kt

package com.example.fontchange

import android.graphics.Typeface import android.os.Bundle

import android.widget.Button import android.widget.TextView import android.widget.Toast

import androidx.activity.ComponentActivity import androidx.core.content.ContextCompat

class MainActivity : ComponentActivity() {

private lateinit var textView: TextView private lateinit var buttonChange: Button

override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState) setContentView(R.layout.activity\_main)

textView = findViewById(R.id.textView) buttonChange = findViewById(R.id.buttonChange) buttonChange.setOnClickListener { changeTextStyle() showToastMessage()

} }

private fun changeTextStyle() {

textView.typeface = Typeface.create("sans-serif-medium", Typeface.NORMAL) textView.setTextColor(ContextCompat.getColor(this, android.R.color.holo\_blue\_light))

}

private fun showToastMessage() {

Toast.makeText(this, "Text style changed!", Toast.LENGTH\_SHORT).show()

}

}

## activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="[http://schemas.android.com/apk/res/android&quot;](http://schemas.android.com/apk/res/android%26quot%3B) android:layout\_width="match\_parent" android:layout\_height="match\_parent">

<TextView

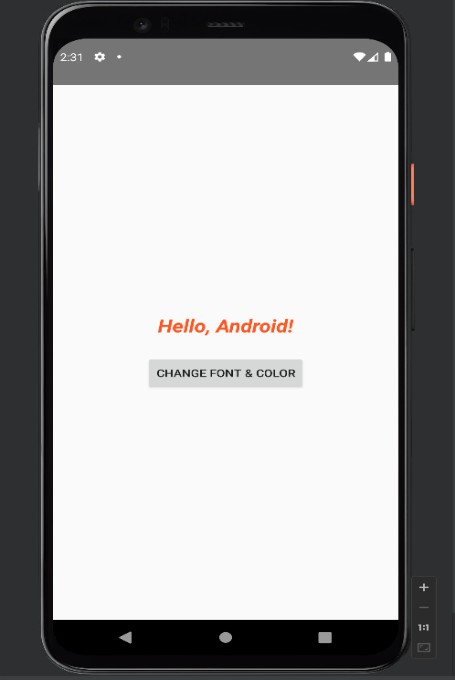
android:id="@+id/textView" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Hello, World!" android:textSize="24sp" android:layout\_centerInParent="true" android:textColor="@android:color/black”/>

<Button

android:id="@+id/buttonChange" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Change Font and Color" android:layout\_below="@id/textView" android:layout\_centerHorizontal="true" android:layout\_marginTop="20dp”/>

</RelativeLayout>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified

|  |  |
| --- | --- |
| **EX.NO: 10** | **ANDROID APPLICATION - SD CARD WRITER** |

# PROGRAM:-

## MainActivity.kt

package com.example.sdcard

import android.content.ContentValues import android.net.Uri

import android.os.Bundle

import android.provider.MediaStore import android.widget.Button import android.widget.Toast

import androidx.activity.ComponentActivity class MainActivity : ComponentActivity() {

override fun onCreate(savedInstanceState: Bundle?) { super.onCreate(savedInstanceState) setContentView(R.layout.activity\_main)

val writeButton = findViewById<Button>(R.id.buttonWrite)

writeButton.setOnClickListener { writeToExternalStorage("Hello World!")

}

}

private fun writeToExternalStorage(data: String) { val values = ContentValues().apply {

put(MediaStore.Files.FileColumns.DISPLAY\_NAME, "sample.txt")

put(MediaStore.Files.FileColumns.MIME\_TYPE, "text/plain") put(MediaStore.Files.FileColumns.RELATIVE\_PATH, "Documents/MyAppFolder")

}

val uri: Uri? = contentResolver.insert(MediaStore.Files.getContentUri("external"), values)

uri?.let { try {

val outputStream = contentResolver.openOutputStream(it) outputStream?.write(data.toByteArray()) outputStream?.close()

Toast.makeText(this, "Data written to $it", Toast.LENGTH\_LONG).show()

} catch (e: Exception) {

Toast.makeText(this, "Error: ${e.message}", Toast.LENGTH\_LONG).show()

}

} ?: run {

Toast.makeText(this, "Error creating file", Toast.LENGTH\_LONG).show()

}

}

}

## activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="[http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:orientation="vertical" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:gravity="center"

android:padding="16dp">

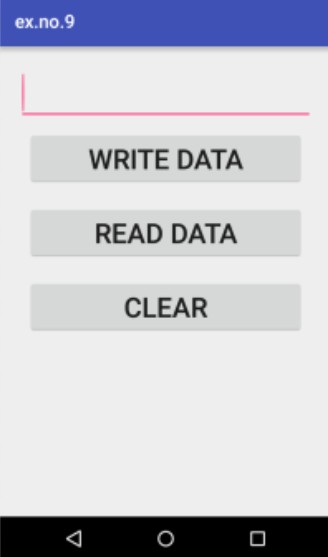
<Button

android:id="@+id/buttonWrite"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Write to SD Card" />

</LinearLayout>

# OUTPUT:-

****

**RESULT:-**

Thus the given program is executed successfully and output is verified