VELALAR COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

REGULATION 2018

18ITL61-MOBILEAPPLICATION DEVELOPMENT LABORATORY



|  |  |  |  |
| --- | --- | --- | --- |
| Name | :………………………………………………... | Course | :……….. |
| Register No. | :………………………………………………… | Branch | :……….. |
| Class | :………………………………………………… | Semester | :……….. |
| Course Name | :………………………………………………… | Course Code | :……….. |

*Certified that this is a bonafide record of workdone by the*

*above student during the academic year 20    - 20*

Faculty in-charge                         Head of the Department

Submitted for the Practical Examination held on :……………………………..

Internal Examiner        External Examiner

INDEX

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S. No | Date | Experiments | Page No | Marks | Signature |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S. No | Date | Experiments | Page No | Marks | Signature |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| Ex.no:01 |  | Roll.no: |
| Date: |  | Page.no: |
| SETTING UP ANDROID PLATFORM, ANDROID VIRTUAL DEVICE AND CREATE APPLICATION TO DISPLAY WELCOME MESSAGE | | |

AIM:

To set up Android Platform, Android Virtual Device and to create application to display welcome message

PROCEDURE:

STEP 1: Open the Android Studio

STEP 2:Click on the file menu

STEP 3:Click on the New button

STEP 4:Now, click on New project

STEP 5:Now, our first android project is ready, we can run it by clicking on

the green run button in [android studio](https://developer.android.com/studio)

STEP6:Once we run the android hello world app, it will be installed

automatically on the [Android Virtual Device](https://data-flair.training/blogs/android-virtual-device-setup/) (AVD).

PROGRAM:

activity\_main.xml

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

<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android=”http://schemas.android.com/apk/res/android”

xmlns:app=”http://schemas.android.com/apk/res-auto”

xmlns:tools=”http://schemas.android.com/tools”

android:layout\_width=”match\_parent”

android:layout\_height=”match\_parent”

tools:context=”.MainActivity”>

<TextView

android:layout\_width=”wrap\_content”

android:layout\_height=”wrap\_content”

android:text=”Welcome”

app:layout\_constraintBottom\_toBottomOf=”parent”

app:layout\_constraintLeft\_toLeftOf=”parent”

app:layout\_constraintRight\_toRightOf=”parent”

app:layout\_constraintTop\_toTopOf=”parent” />

<TextView

android:id=”@+id/textView”

android:layout\_width=”151dp”

android:layout\_height=”55dp”

android:text=”welcome to android”

app:layout\_constraintBottom\_toTopOf=”@+id/textView2”

app:layout\_constraintEnd\_toEndOf=”parent”

app:layout\_constraintStart\_toStartOf=”parent”

app:layout\_constraintTop\_toTopOf=”parent” />

</androidx.constraintlayout.widget.ConstraintLayout>

MainActivity.java

package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

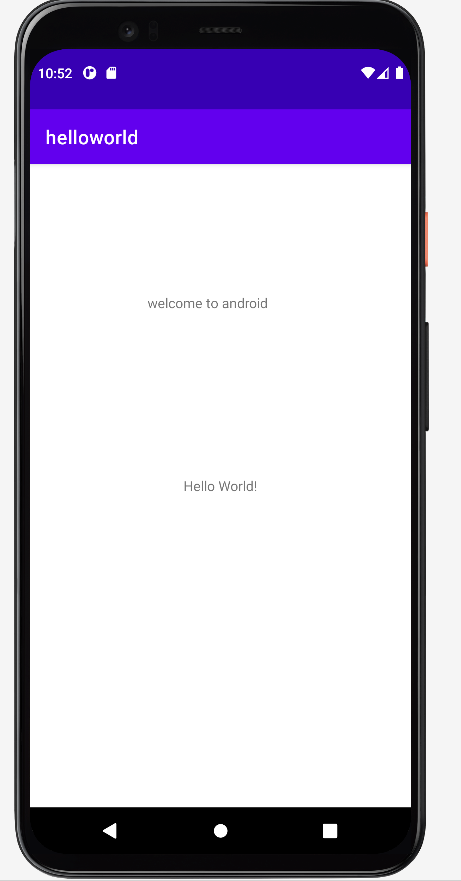
super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

}

OUTPUT:



RESULT:

Thus the implementation of Android Application to display the welcome message has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:02 |  | Roll.no: |
| Date: |  | Page.no: |
| ANDROID SCORE KEEPER APP TO KEEP TRACK OF THE SCORE OF TWO DIFFERENT TEAMS | | |

AIM:

To develop a score keeper app that gives the score of two different teams playing a games.

PROCEDURE:

STEP 1: Open eclipse or android studio and select new android project .

STEP 2:Give application name,select blank activity and design application icon

SETP 3:Create view layout for user interface.

SETP 4:Write a java function for view layout widgets.

SETP 5:Execute the android application.

SETP 6:Use android emulator to show the created application.

PROGRAM:

Activity\_main.xml

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

<androidx.constraintlayout.widget.ConstraintLayoutxmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".activity\_result">

<TextView

android:id="@+id/result\_TeamA"

android:layout\_width="182dp"

android:layout\_height="70dp"

android:layout\_marginStart="16dp"

android:layout\_marginTop="28dp"

android:text="@string/team\_a"

android:textSize="50sp"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<TextView

android:id="@+id/result\_TeamB"

android:layout\_width="182dp"

android:layout\_height="70dp"

android:layout\_marginStart="16dp"

android:layout\_marginTop="68dp"

android:text="@string/team\_b"

android:textSize="50sp"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toBottomOf="@+id/result\_TeamA" />

<TextView

android:id="@+id/result\_TeamAscore"

android:layout\_width="151dp"

android:layout\_height="61dp"

android:layout\_marginStart="40dp"

android:layout\_marginTop="28dp"

android:text="@string/score\_a"

android:textAlignment="center"

android:textSize="50sp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toEndOf="@+id/result\_TeamA"

app:layout\_constraintTop\_toTopOf="parent"

tools:ignore="MissingConstraints" />

<TextView

android:id="@+id/result\_TeamBscore"

android:layout\_width="150dp"

android:layout\_height="72dp"

android:layout\_marginStart="32dp"

android:layout\_marginTop="72dp"

android:text="@string/score\_b"

android:textAlignment="center"

android:textSize="50sp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.258"

app:layout\_constraintStart\_toEndOf="@+id/result\_TeamB"

app:layout\_constraintTop\_toBottomOf="@+id/result\_TeamAscore" />

<TextView

android:id="@+id/result\_ResultString"

android:layout\_width="match\_parent"

android:layout\_height="153dp"

android:layout\_marginTop="336dp"

android:text="@string/result\_page"

android:textAlignment="center"

android:textSize="40sp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.0"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

tools:ignore="InvalidId,MissingConstraints" />

</androidx.constraintlayout.widget.ConstraintLayout>

Main Activity.java

package com.example.scoreboard;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

private TextView score1;

private TextView score2;

private int s1=0,s2=0;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

TextViewToNumber();

addListenerOnButton();

Reset();

}

private void Reset() {

Button reset=findViewById(R.id.reset\_button);

reset.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

s1=0;

s2=0;

score1.setText(String.valueOf(s1));

score2.setText(String.valueOf(s2));

}

});

}

private void TextViewToNumber() {

score1= findViewById(R.id.Score1);

score2= findViewById((R.id.score2));

s1=Integer.parseInt(score1.getText().toString());

s2 = Integer.parseInt(score2.getText().toString());

}

public void CallResultActivity()

{

Intent intent=new Intent(this,activity\_result.class);

intent.putExtra("Ascore",s1);

intent.putExtra("Bscore",s2);

startActivity(intent);

}

private void addListenerOnButton() {

Button button1= findViewById(R.id.button1);

Button button2= findViewById(R.id.button2);

Button result\_button= findViewById(R.id.button\_result);

button1.setOnClickListener(view -> {

s1 = s1 + 1;

score1.setText(String.valueOf(s1));

});

button2.setOnClickListener(view -> {

s2 = s2 + 1;

score2.setText(String.valueOf(s2));

});

//result\_button.setOnClickListener(view ->CallResultActivity());

result\_button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

CallResultActivity();

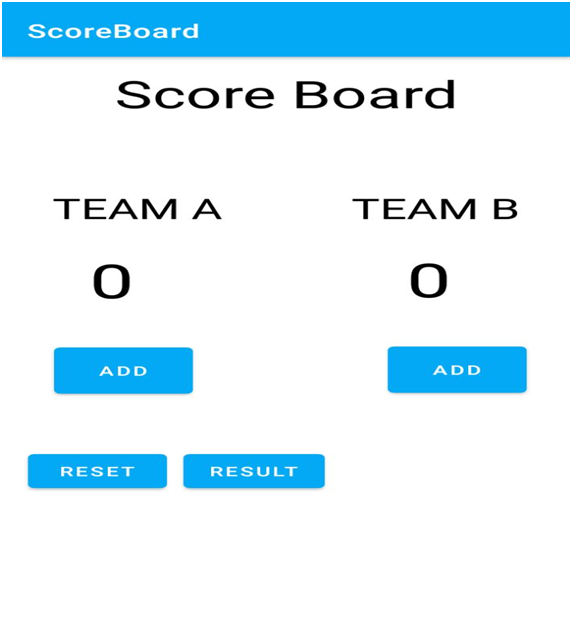
}

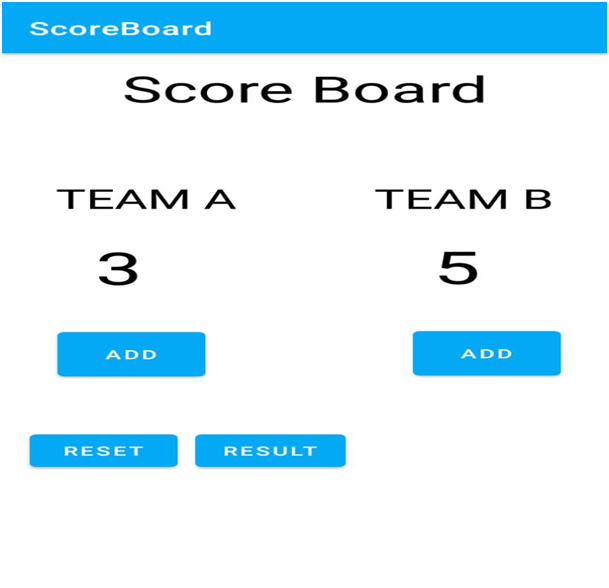
});

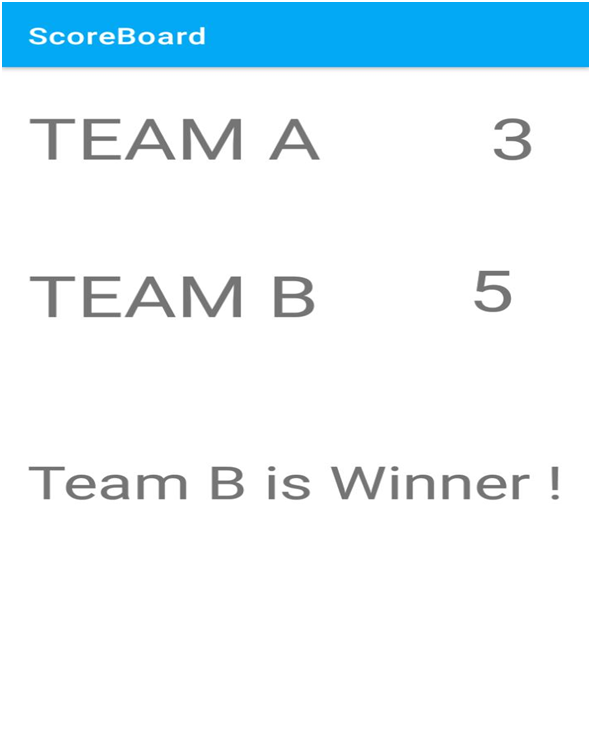
}

}

OUTPUT:







RESULT:

Thus the implementation of Android score keeper app has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:03 |  | Roll.no: |
| Date: |  | Page.no: |
| MUSICAL STRUCTURE APP TO STORE AND PRESENT THE LIBRARY OF SONGS | | |

AIM:

To create a Musical Structure App to store and present the user with the library of songs available indifferent categories of classical and Melodies.

PROCEDURE:

STEP1: Start the Android Studio Application

STEP 2: Create an empty Activity project

STEP 3: Create a raw source folder under the res folder and copy one of the

.mp3 file extension

STEP 4: In activity\_main.xml file create the layout of application consist of

three buttons PLAY, PAUSE and STOP

STEP 5: Create the media player instance needs and the attribute needs to be set

in the MainActivity.java file before playing audio.

STEP 6: Stop the program.

PROGRAM:

activity\_main.xml

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

<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="horizontal"

android:gravity="center"

tools:context=".MainActivity">

<Button

android:id="@+id/play"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:text="Play"/>

<Button

android:id="@+id/pause"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:text="Pause"/>

</LinearLayout>

MainActivity.java

package com.arulesh.musicplayer;

import androidx.appcompat.app.AppCompatActivity;

import android.media.MediaPlayer;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

public class MainActivity extends AppCompatActivity {

Button play,pause;

MediaPlayer player;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

play = findViewById(R.id.play);

pause = findViewById(R.id.pause);

player = MediaPlayer.create(this, R.raw.music);

play.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

player.start();

}

});

pause.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

player.pause();

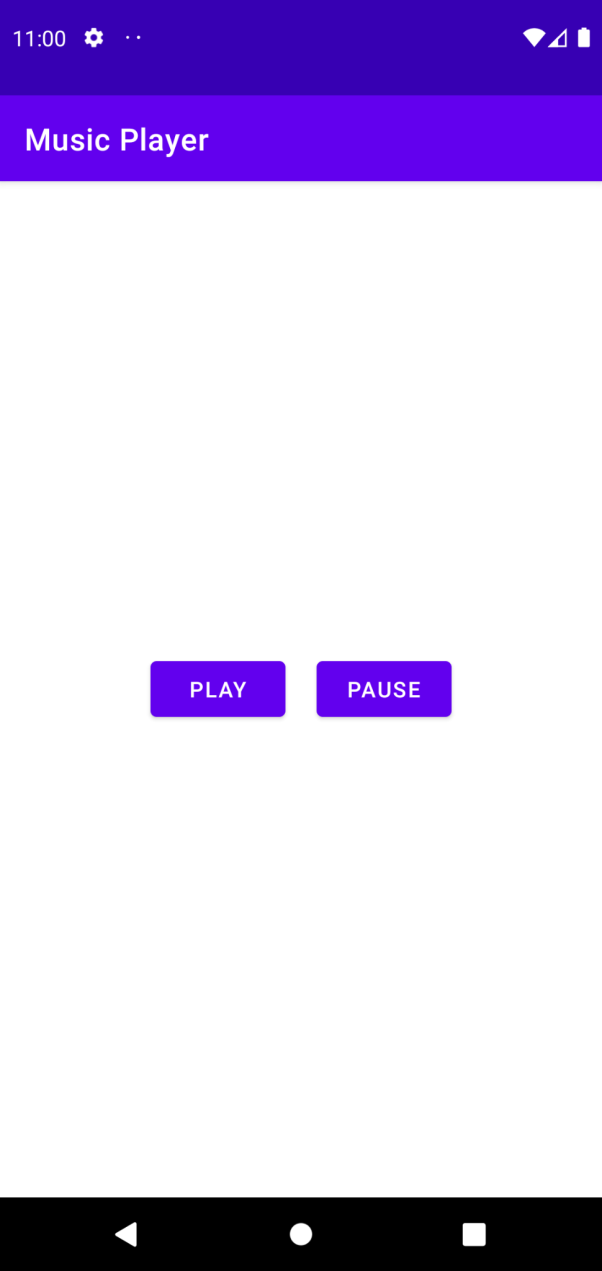
}

});

}

}

OUTPUT:



RESULT:

Thus the implementation of Musical Structure App has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:04 |  | Roll.no: |
| Date: |  | Page.no: |
| ELECTRICITY BILL CALCULATOR AND SMS TO THE MOBILE | | |

AIM:

To create an application to calculate the electricity bill and create an appropriate alert message as well assend the value to the given mobile number using SMS.

PROCEDURE:

STEP1:  Start the process.

STEP2:  In Eclipse, create a new Android application project by clicking the

menu File -> New -> Android Application Project.

STEP 3:  Give the application name, project name, package name and click

Next.

STEP4:  Click Next in the Configure project and Configure the attributes of

theicon set wizard.

STEP5:  Select Blank activity and click Next.

STEP6:  Specify the Activity and Layout name. Click Finish.

STEP7:  Choose the version of emulator as Android 10.

STEP8:  Design the screen layout by placing the appropriate TextViews, Edit

Texts and Button in the activity\_main.xml and save the file.

STEP9:  Write the coding in MainActivity.java and save the file.

STEP 10:  Create an emulator using Window -> Android Virtual Device

Manager -> New Android Virtual Device.

STEP 11:  Select the Android Virtual Device and give start and launch.

STEP 12:  Now run the project on emulator.

STEP 13:  Stop the process.

PROGRAM:

activity\_main.xml

<RelativeLayoutxmlns: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"

tools:context=".MainActivity" >

<TextView

android:id="@+id/tvTitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="50dp"

android:text="Eb Bill Calculator"

android:textColor="@color/purple\_700"

android:textSize="20sp" />

<TextView

android:id="@+id/tvUnits"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/tvTitle"

android:layout\_marginLeft="15dp"

android:layout\_marginTop="50dp"

android:text="Units Consumed" />

<TextView

android:id="@+id/tvMobNo"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/tvUnits"

android:layout\_marginLeft="15dp"

android:layout\_marginTop="50dp"

android:text="MobileNumber" />

<EditText

android:id="@+id/etUnitsCon"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvUnits"

android:layout\_alignLeft="@+id/btnCalculate"

android:layout\_marginLeft="20dp"

android:ems="10"

android:inputType="number" >

<requestFocus />

</EditText>

<EditText

android:id="@+id/etMobNo"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvMobNo"

android:layout\_alignLeft="@+id/btnCalculate"

android:layout\_marginLeft="20dp"

android:ems="10"

android:inputType="number" />

<Button

android:id="@+id/btnCalculate"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/tvMobNo"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="50dp"

android:text="Calculate" />

</RelativeLayout>

MainActivity.java

package com.madlab.ebbill;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import android.app.Activity;

public class MainActivity extends Activity {

EditText et1, et2;

Button b1;

String s1, s2;

Double units, bill\_amount;

android.telephony.SmsManager manager;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

et1 = (EditText)findViewById(R.id.etUnitsCon);

et2 = (EditText)findViewById(R.id.etMobNo);

b1=(Button)findViewById(R.id.btnCalculate);

b1.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

s1 = et1.getText().toString();

s2 = et2.getText().toString();

units = Double.parseDouble(s1);

if(units <= 100)

{

bill\_amount = 0.00;

}

else if(units <= 200)

{

bill\_amount = (units - 100) \* 1.50;

}

else if(units <= 500)

{

bill\_amount = 100 \* 2.00 + (units - 200) \* 3.00;

}

else if(units > 500)

{

bill\_amount = 100 \* 3.50 + 300 \* 4.60 + (units - 500) \* 6.60;

}

Toast.makeText(getApplicationContext(), "The bill amount is " +

bill\_amount, Toast.LENGTH\_LONG).show();

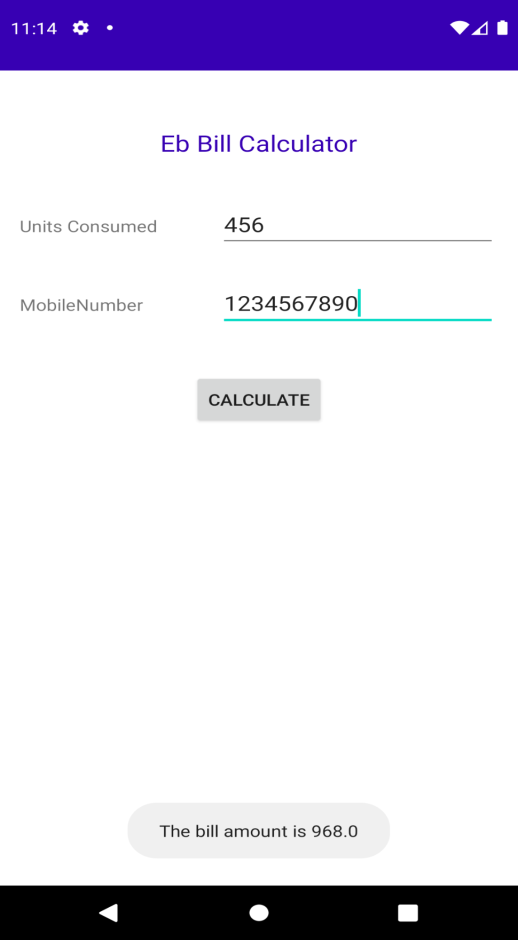
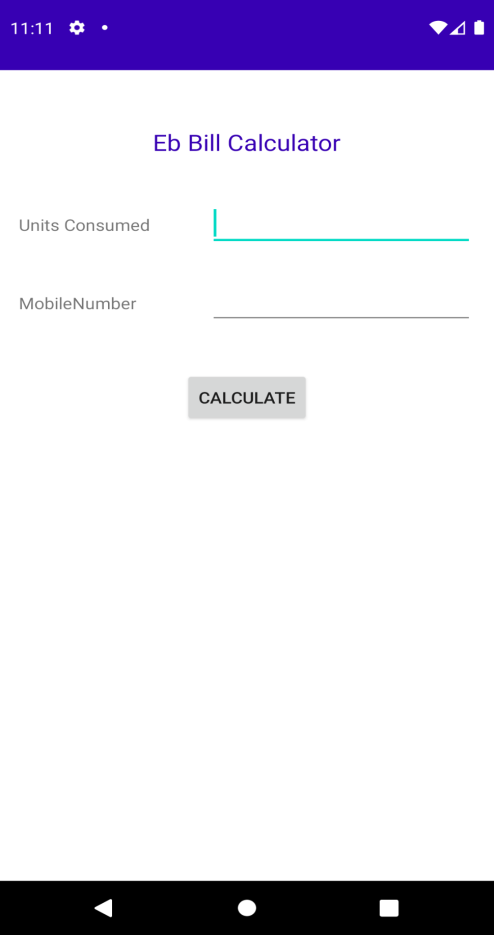
}

});

}

}

OUTPUT:



RESULT:

Thus the implementation of Electricity Bill Calculator and to send the SMS to the given mobile number has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:05 |  | Roll.no: |
| Date: |  | Page.no: |
| ANDROID PROGRAM FOR MENU FILE WITH NEW AND OPEN AS MENU ITEMS | | |

AIM:

To develop a simple application that shows how to create a Menu “File” with “New” and “Open” as Menu items.

PROCEDURE:

STEP1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menu File -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click

Next.

STEP 4: Click Next in the Configure project and Configure the attributes of

theicon set wizard.

STEP 5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP 7: Choose the version of emulator as Android 10.

STEP 8: Write the coding in main.xml and save the file.

STEP9: Write the coding in MainActivity.java and save the file.

STEP 10: Create an emulator using Window -> Android Virtual Device

Manager -> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

activity\_main.xml:

<RelativeLayoutxmlns: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"

tools:context=".MainActivity" >

</RelativeLayout>

strings.xml:

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

<resources>

<string name="app\_name">Menu</string>

<string name="strFile">File</string>

<string name="strNew">New</string>

<string name="strOpen">Open</string>

</resources>

main.xml:

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

<menu xmlns:android="http://schemas.android.com/apk/res/android" >

<item

android:id="@+id/filemenu"

android:orderInCategory="1"

android:showAsAction="never"

android:title="@string/strFile">

<menu>

<item

android:id="@+id/newmenu"

android:orderInCategory="1"

android:showAsAction="never"

android:title="@string/strNew"/>

<item

android:id="@+id/openmenu"

android:orderInCategory="2"

android:showAsAction="never"

android:title="@string/strOpen"/>

</menu>

</item>

</menu>

MainActivity.java:

package com.madlab.menu;

import android.os.Bundle;

import android.app.Activity;

import android.view.Menu;

import android.view.MenuInflater;

import android.view.MenuItem;

import android.widget.Toast;

public class MainActivity extends Activity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

@Override

public booleanonCreateOptionsMenu(Menu menu) {

MenuInflater inflater = getMenuInflater();

inflater.inflate(R.menu.main, menu);

return true;

}

@Override

public booleanonOptionsItemSelected(MenuItem item) {

switch (item.getItemId()){

case R.id.newmenu:

Toast.makeText(getApplicationContext(),"New File Menu is

Selected",Toast.LENGTH\_LONG).show();

return true;

case R.id.openmenu:

Toast.makeText(getApplicationContext(),"Open File Menu is

Selected",Toast.LENGTH\_LONG).show();

return true;

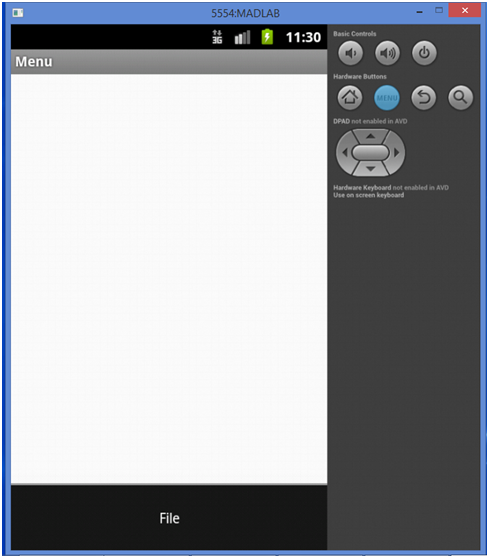
default:

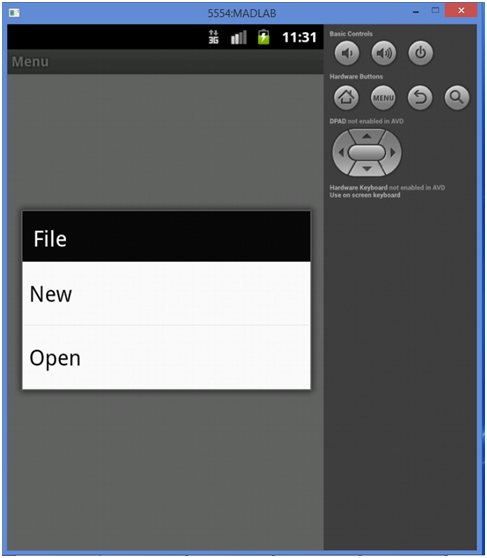
return super.onOptionsItemSelected(item);

}

}}

OUTPUT:





RESULT:

Thus the implementation of simple application of menu files with new and open as menu item has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:06 |  | Roll.no: |
| Date: |  | Page.no: |
| **SIMPLE APPLICATION TO DISPLAY CURRENT LOCATION INFORMATION USING AN ALERT MESSAGE** | | |

**AIM:**

To develop a simple application that displays the current location information (Latitude and Longitude) using an alert message.

**PROCEDURE:**

STEP 1:  Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menu File -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click

Next.

STEP 4: Click Next in the Configure project and Configure the attributes of the

icon set wizard.

STEP 5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP 7: Choose the version of emulator as Android 10.

STEP 8: Write the coding in activity\_main.xml and save the file.

STEP 9: Write the coding in LocationActivity.java, GPSTracker.java and save

the file.

STEP 10: Create an emulator using Window -> Android Virtual Device

Manager -> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

**PROGRAM:**

Add permission in manifest

AndriodManifest.xml

<manifest >

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />

<uses-permissionandroid:name="android.permission. ACCESS\_COARSE\_LOCATION" />

<uses-permission android:name="android.permission.INTERNET" />

</manifest>

Activity\_main.xml

<RelativeLayoutxmlns: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"

tools:context=".MainActivity" >

<TextView

android:id="@+id/textview1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_centerVertical="true"

android:text="@string/hello\_world" />

</RelativeLayout>

MainActivity.java

package com.javapapers.android.geolocationfinder;

import android.os.Bundle;

import android.app.Activity;

import android.content.Context;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.widget.TextView;

import android.util.Log;

public class MainActivity extends Activity implements LocationListener{

protected LocationManagerlocationManager;

protected LocationListenerlocationListener;

protected Context context;

TextViewtxtLat;

String lat;

String provider;

protected String latitude,longitude;

protected booleangps\_enabled,network\_enabled;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

txtLat = (TextView) findViewById(R.id.textview1);

locationManager=(LocationManager) getSystemService(Context.LOCATION\_SERVICE);

locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, 0, 0, this);

}

@Override

public void onLocationChanged(Location location) {

txtLat = (TextView) findViewById(R.id.textview1);

txtLat.setText("Latitude:" + location.getLatitude() + ", Longitude:" + location.getLongitude());

}

@Override

public void onProviderDisabled(String provider) {

Log.d("Latitude","disable");

}

@Override

public void onProviderEnabled(String provider) {

Log.d("Latitude","enable");

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

Log.d("Latitude","status");

}

}

OUTPUT:



RESULT:

Thus the implementation of simple application to display the current location and send an alert message has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:07 |  | Roll.no: |
| Date: |  | Page.no: |
| DEVELOP AN APPLICATION TO IMPLEMENT CRUD OPERATIONS USING SQLITE DATABASE | | |

AIM:

To develop a simple application for demonstrating the CRUD operations using SQLite database.

PROCEDURE:

STEP1: Start the process.

STEP2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP3: Give the application name, project name, package name and click Next.

STEP4: Click Next in the Configure project and Configure the attributes of the

icon set wizard.

STEP5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP7: Choose the version of emulator as Android 10.

STEP8: Write the coding in corresponding layout files and save them.

STEP9: Write the coding in corresponding Activity files and save them.

STEP 10: Create an emulator using Window -> Android Virtual Device Manager

-> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

activity\_main.xml:

<RelativeLayoutxmlns: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:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context=".MainActivity" >

<Button

android:id="@+id/button1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="29dp"

android:text="Login" />

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/button1"

android:layout\_below="@+id/button1"

android:layout\_marginTop="69dp"

android:text="Register" />

</RelativeLayout>

activity\_login.xml:

<RelativeLayoutxmlns: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:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context=".LoginActivity" >

<TextView

android:id="@+id/tvName1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="25dp"

android:text="@string/name" />

<EditText

android:id="@+id/etName1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvName1"

android:layout\_alignBottom="@+id/tvName1"

android:layout\_alignParentRight="true"

android:layout\_marginRight="10dp"

android:ems="7"

android:inputType="text" >

<requestFocus />

</EditText>

<TextView

android:id="@+id/tvPassword1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/tvName1"

android:layout\_below="@+id/etName1"

android:layout\_marginTop="50dp"

android:text="@string/pass" />

<EditText

android:id="@+id/etPassword1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvPassword1"

android:layout\_alignBottom="@+id/tvPassword1"

android:layout\_alignLeft="@+id/etName1"

android:ems="7"

android:inputType="textPassword" />

<Button

android:id="@+id/btnAdd"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignRight="@+id/etPassword1"

android:layout\_below="@+id/etPassword1"

android:layout\_marginRight="24dp"

android:layout\_marginTop="40dp"

android:text="@string/login" />

<Button

android:id="@+id/btnClear"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/btnAdd"

android:layout\_alignBottom="@+id/btnAdd"

android:layout\_alignRight="@+id/tvName1"

android:text="@string/clear" />

</RelativeLayout>

activity\_user\_registration.xml:

<RelativeLayoutxmlns:tools="http://schemas.android.com/tools"

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context=".UserRegistrationActivity" >

<TextView

android:id="@+id/tvName1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="25dp"

android:text="@string/name" />

<EditText

android:id="@+id/etName1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvName1"

android:layout\_alignBottom="@+id/tvName1"

android:layout\_alignParentRight="true"

android:layout\_marginRight="10dp"

android:ems="7"

android:inputType="text">

<requestFocus />

</EditText>

<TextView

android:id="@+id/tvPassword1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/tvName1"

android:layout\_below="@+id/etName1"

android:layout\_marginTop="50dp"

android:text="@string/pass" />

<EditText

android:id="@+id/etPassword1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/tvPassword1"

android:layout\_alignBottom="@+id/tvPassword1"

android:layout\_alignLeft="@+id/etName1"

android:ems="7"

android:inputType="textPassword"/>

<TextView

android:id="@+id/textView1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/tvPassword1"

android:layout\_alignLeft="@+id/tvPassword1"

android:layout\_marginTop="60dp"

android:text="@string/gender" />

<RadioGroup

android:id="@+id/tv1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/etPassword1"

android:layout\_below="@+id/etPassword1"

android:layout\_marginTop="28dp" >

<RadioButton

android:id="@+id/radio0"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:checked="true"

android:text="@string/male" />

<RadioButton

android:id="@+id/radio1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/female" />

</RadioGroup>

<TextView

android:id="@+id/tvCity"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/tvPassword1"

android:layout\_below="@+id/tv1"

android:layout\_marginTop="20dp"

android:text="@string/city" />

<Spinner

android:id="@+id/spinner1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/tv1"

android:layout\_alignParentRight="true"

android:layout\_below="@+id/tv1"

android:entries="@array/city"/>

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/tvCity"

android:layout\_below="@+id/spinner1"

android:layout\_marginTop="22dp"

android:text="@string/hobbies" />

<CheckBox

android:id="@+id/checkBox1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignBaseline="@+id/textView2"

android:layout\_alignBottom="@+id/textView2"

android:layout\_alignLeft="@+id/spinner1"

android:text="@string/reading" />

<CheckBox

android:id="@+id/checkBox2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/checkBox1"

android:layout\_below="@+id/checkBox1"

android:text="@string/play" />

<Button

android:id="@+id/btnAdd"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:text="@string/adduser" />

</RelativeLayout>

activity\_welcome.xml:

<RelativeLayoutxmlns: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:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context=".WelcomeActivity" >

<TextView

android:id="@id/tv1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/wel" />

</RelativeLayout>

MainActivity.java:

package com.madlab.mysimpledb;

import android.os.Bundle;

import android.app.Activity;

import android.content.Intent;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

public class MainActivity extends Activity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Button b1 = (Button)findViewById(R.id.button1);

Button b2 = (Button)findViewById(R.id.button2);

b1.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

Intent in = new Intent(getApplicationContext(), LoginActivity.class);

startActivity(in);

}

});

b2.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

Intent in = new Intent(getApplicationContext(), UserRegistrationActivity.class);

startActivity(in);

}

});

}

}

LoginActivity.java

package com.madlab.mysimpledb;

import android.os.Bundle;

import android.app.Activity;

import android.content.Intent;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

public class LoginActivity extends Activity {

DBAdapterdbAdapter;

Button b1, b2;

EditText et1, et2;

String s1, s2;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_login);

b1 = (Button)findViewById(R.id.btnAdd);

b2 = (Button)findViewById(R.id.btnClear);

et1 = (EditText)findViewById(R.id.etName1);

et2 = (EditText)findViewById(R.id.etPassword1);

dbAdapter = new DBAdapter(getApplicationContext());

dbAdapter.open();

b1.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

s1 = et1.getText().toString();

s2 = et2.getText().toString();

//if((s1!=null)&&(s2!=null)){

String p = dbAdapter.getUserPass(s1);

if(p.equals(s2)){

Intent in = new Intent(getApplicationContext(), WelcomeActivity.class);

in.putExtra("name", s1);

startActivity(in);

}

/\*}\*/

else{

Toast.makeText(LoginActivity.this, "Enter Username/Password", Toast.LENGTH\_LONG).show();

}

}

});

b2.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

et1.setText("");

et2.setText("");

}

});

}

}

UserRegistrationActivity.java

package com.madlab.mysimpledb;

import android.os.Bundle;

import android.app.Activity;

import android.view.Menu;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.CheckBox;

import android.widget.EditText;

import android.widget.RadioButton;

import android.widget.RadioGroup;

import android.widget.Spinner;

import android.widget.Toast;

public class UserRegistrationActivity extends Activity {

DBAdapterloginDB;

EditText name, pass;

CheckBox c1, c2;

RadioGrouprg;

RadioButton r;

Spinner s;

Button btnAdd;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_user\_registration);

loginDB = new DBAdapter(this);

name = (EditText)findViewById(R.id.etName1);

pass = (EditText)findViewById(R.id.etPassword1);

btnAdd = (Button)findViewById(R.id.btnAdd);

c1 = (CheckBox)findViewById(R.id.checkBox1);

c2 = (CheckBox)findViewById(R.id.checkBox2);

rg = (RadioGroup)findViewById(R.id.tv1);

s=(Spinner)findViewById(R.id.spinner1);

btnAdd.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

int rs = rg.getCheckedRadioButtonId();

r=(RadioButton)findViewById(rs);

StringBuffersb = new StringBuffer();

if(c1.isChecked())

sb.append(c1.getText().toString());

if(c2.isChecked())

sb.append(", "+c2.getText().toString());

booleanisInserted = loginDB.addStudent(name.getText().toString(), pass.getText().toString(), r.getText().toString(),s.getSelectedItem().toString(), sb.toString());

if(isInserted)

Toast.makeText(UserRegistrationActivity.this, "New Student Added Successfully", Toast.LENGTH\_LONG).show();

else

Toast.makeText(UserRegistrationActivity.this, "New Student Not Added", Toast.LENGTH\_LONG).show();

}

});

}

}

WelcomeActivity.java

package com.madlab.mysimpledb;

import android.os.Bundle;

import android.app.Activity;

import android.content.Intent;

import android.view.Menu;

import android.widget.TextView;

public class WelcomeActivity extends Activity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_welcome);

Intent in = getIntent();

String s = in.getStringExtra("name");

TextView tv = (TextView)findViewById(R.id.tv1);

tv.append(s);

}

}

DBAdapter.java

package com.madlab.mysimpledb;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteDatabase.CursorFactory;

import android.database.sqlite.SQLiteOpenHelper;

import android.util.Log;

public class DBAdapter {

private static final String DB\_NAME = "User\_Details.db";

private static final int DB\_VERSION = 1;

private static final String TABLE\_NAME = "User\_Table";

private static final String COL\_SNO = "sno";

private static final String COL\_NAME = "name";

private static final String COL\_PASS = "password";

private static final String COL\_GENDER = "gender";

private static final String COL\_CITY = "city";

private static final String COL\_HOBBIES = "hobbies";

private static final String TABLE\_CREATE = "create table " + TABLE\_NAME + "(" + COL\_SNO + " integer primary key autoincrement, " + COL\_NAME + " text not null, " + COL\_PASS + " text not null, " + COL\_GENDER + " text not null, " + COL\_CITY + " text not null, " + COL\_HOBBIES + " text not null);";

private static final String DROP\_TABLE = "drop table if exists " +TABLE\_NAME +";";

private SQLiteDatabaseuserDatabase;

private final Context context;

private MyDBHelper helper;

public DBAdapter(Context context){

this.context = context;

helper = new MyDBHelper(context, DB\_NAME, null, DB\_VERSION);

}

private class MyDBHelper extends SQLiteOpenHelper{

public MyDBHelper(Context context, String name, CursorFactoryfactory,int version) {

super(context, name, factory, version);

//userDatabase = this.getWritableDatabase();

//userDatabase.execSQL(DROP\_TABLE);

onCreate(userDatabase);

}

@Override

public void onCreate(SQLiteDatabasedb) {

Log.i("Table Creation", "Table is being created");

db.execSQL(TABLE\_CREATE);

}

@Override

public void onUpgrade(SQLiteDatabasedb, int oldVersion, int newVersion) {

Log.w("Updation", "Database version is being updated");

db.execSQL(DROP\_TABLE);

onCreate(db);

}

}

public DBAdapteropen(){

userDatabase=helper.getWritableDatabase();

return this;

}

public void close() {

userDatabase.close();

}

public booleanaddStudent(String name, String pass, String gender, String city, String hobbies){

ContentValues cv = new ContentValues();

cv.put(COL\_NAME, name);

cv.put(COL\_PASS, pass);

cv.put(COL\_GENDER, gender);

cv.put(COL\_CITY, city);

cv.put(COL\_HOBBIES, hobbies);

long res = userDatabase.insert(TABLE\_NAME, null, cv);

if(res ==-1)

return false;

else

return true;

}

public String getUserPass(String u){

Cursor cursor = userDatabase.rawQuery("SELECT \* FROM User\_Table where name='" + u + "'", null);

if(cursor !=null &&cursor.moveToFirst())

return cursor.getString(2);

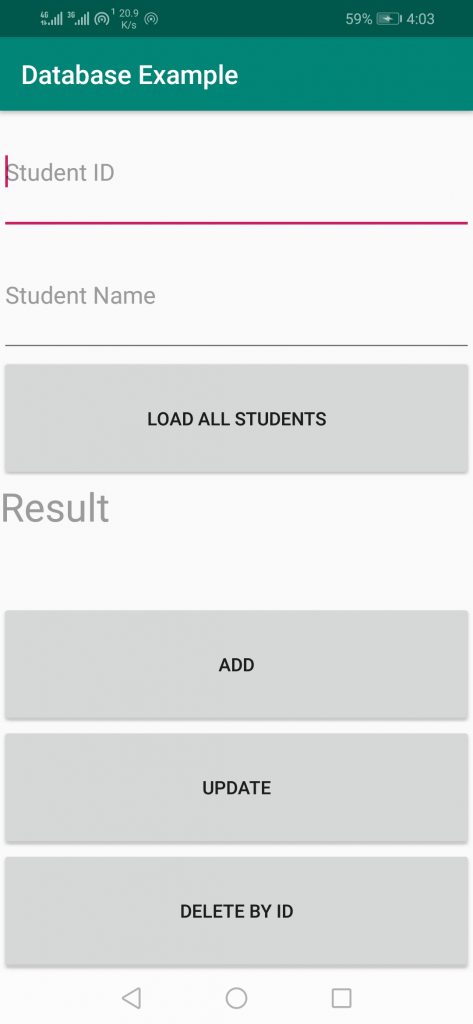
else

return "";

}

}

OUTPUT :



RESULT :

Thus the program to develop a simple application for demonstrating the CRUD operations using SQLite database has been built successfully

|  |  |  |
| --- | --- | --- |
| Ex.no:08 |  | Roll.no: |
| Date: |  | Page.no: |
| ANDROID APPLICATION FOR  PHONE NUMBER VERIFICATION BY OTP USING FIREBASE | | |

AIM:

To develop an application to implement phone number Verification by OTP using Firebase in Android.

PROCEDURE:

STEP 1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP 3: Connect your app to Firebase by navigate to Tools ->Firebase ->

Authentication -> email and password authentication.

STEP 4: Verify that dependency for Firebase authentication is added inside

your app.

STEP 5: work with activity\_main.xml file.

STEP 6: Add permission for the internet in our Manifest.xml file by navigate

to the app -> AndroidManifest.xml

STEP 7: Create a new activity four our home by navigating to app -> java ->

your app’s package name and click a new -> activity -> empty activity.

STEP 8: Enter the necessary codes in the MainActivity.java and the

HomeActivity files.

STEP 9: Enable Firebase Phone Authentication in our Firebase Console by

clicking on go to console -> your project -> authentication -> sig-in

method -> phone -> enable.

STEP 10: Stop the Program.

PROGRAM:

MainActivity.java

package com.example.phonenumberotp;

import android.content.Intent;

import android.os.Bundle;

import android.text.TextUtils;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.annotation.NonNull;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

import com.google.android.gms.tasks.TaskExecutors;

import com.google.firebase.FirebaseException;

import com.google.firebase.auth.AuthResult;

import com.google.firebase.auth.FirebaseAuth;

import com.google.firebase.auth.PhoneAuthCredential;

import com.google.firebase.auth.PhoneAuthProvider;

import java.util.concurrent.TimeUnit;

public class MainActivity extends AppCompatActivity {

private FirebaseAuthmAuth;

private EditTextedtPhone, edtOTP;

private Button verifyOTPBtn, generateOTPBtn;

private String verificationId;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

mAuth = FirebaseAuth.getInstance();

edtPhone = findViewById(R.id.idEdtPhoneNumber);

edtOTP = findViewById(R.id.idEdtOtp);

verifyOTPBtn = findViewById(R.id.idBtnVerify);

generateOTPBtn = findViewById(R.id.idBtnGetOtp);

generateOTPBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (TextUtils.isEmpty(edtPhone.getText().toString())) {

Toast.makeText(MainActivity.this, "Please enter a valid phone number.", Toast.LENGTH\_SHORT).show();

} else {

String phone = "+91" + edtPhone.getText().toString();

sendVerificationCode(phone);

}

}

});

verifyOTPBtn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (TextUtils.isEmpty(edtOTP.getText().toString())) {

Toast.makeText(MainActivity.this, "Please enter OTP", Toast.LENGTH\_SHORT).show();

} else {

verifyCode(edtOTP.getText().toString());

}

}

});

}

private void signInWithCredential(PhoneAuthCredential credential) {

mAuth.signInWithCredential(credential)

.addOnCompleteListener(new OnCompleteListener<AuthResult>() {

@Override

public void onComplete(@NonNull Task<AuthResult> task) {

if (task.isSuccessful()) {

Intent i = new Intent(MainActivity.this, HomeActivity.class);

startActivity(i);

finish();

} else {

Toast.makeText(MainActivity.this, task.getException().getMessage(), Toast.LENGTH\_LONG).show();

}

}

});

}

private void sendVerificationCode(String number) {

PhoneAuthOptions options =

PhoneAuthOptions.newBuilder(mAuth)

.setPhoneNumber(number)

.setTimeout(60L, TimeUnit.SECONDS)

.setActivity(this)

.setCallbacks(mCallBack)

.build();

PhoneAuthProvider.verifyPhoneNumber(options);

}

private PhoneAuthProvider.OnVerificationStateChangedCallbacks

mCallBack = new PhoneAuthProvider.OnVerificationStateChangedCallbacks() {

@Override

public void onCodeSent(String s, PhoneAuthProvider.ForceResendingTokenforceResendingToken) {

super.onCodeSent(s, forceResendingToken);

verificationId = s;

}

@Override

public void onVerificationCompleted(PhoneAuthCredentialphoneAuthCredential) {

final String code = phoneAuthCredential.getSmsCode();

if (code != null) {

edtOTP.setText(code);

verifyCode(code);

}

}

@Override

public void onVerificationFailed(FirebaseException e) {

Toast.makeText(MainActivity.this, e.getMessage(), Toast.LENGTH\_LONG).show();

}

};

private void verifyCode(String code) {

PhoneAuthCredential credential = PhoneAuthProvider.getCredential(verificationId, code);

signInWithCredential(credential);

}

}

HomeAcitivity.java

package com.example.phonenumberotp;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class HomeActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_home);

}

}

Activity\_main.xml

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

<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"

tools:context=".MainActivity">

<!--Edittext for getting users phone number-->

<EditText

android:id="@+id/idEdtPhoneNumber"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_margin="10dp"

android:hint="Enter your phone"

android:importantForAutofill="no"

android:inputType="phone" />

<!--Button for getting OTP-->

<Button

android:id="@+id/idBtnGetOtp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/idEdtPhoneNumber"

android:layout\_margin="10dp"

android:text="Get OTP"

android:textAllCaps="false" />

<!--Edittext for getting otp from user-->

<EditText

android:id="@+id/idEdtOtp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/idBtnGetOtp"

android:layout\_margin="10dp"

android:hint="Enter OTP"

android:importantForAutofill="no"

android:inputType="phone" />

<!--button for verifying user OTP-->

<Button

android:id="@+id/idBtnVerify"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/idEdtOtp"

android:layout\_margin="10dp"

android:text="Verify OTP"

android:textAllCaps="false" />

</RelativeLayout>

Activity\_home.xml

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

<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"

tools:context=".HomeActivity">

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_centerInParent="true"

android:padding="10dp"

android:text="Geeks for Geeks \n Welcome to Home Screen"

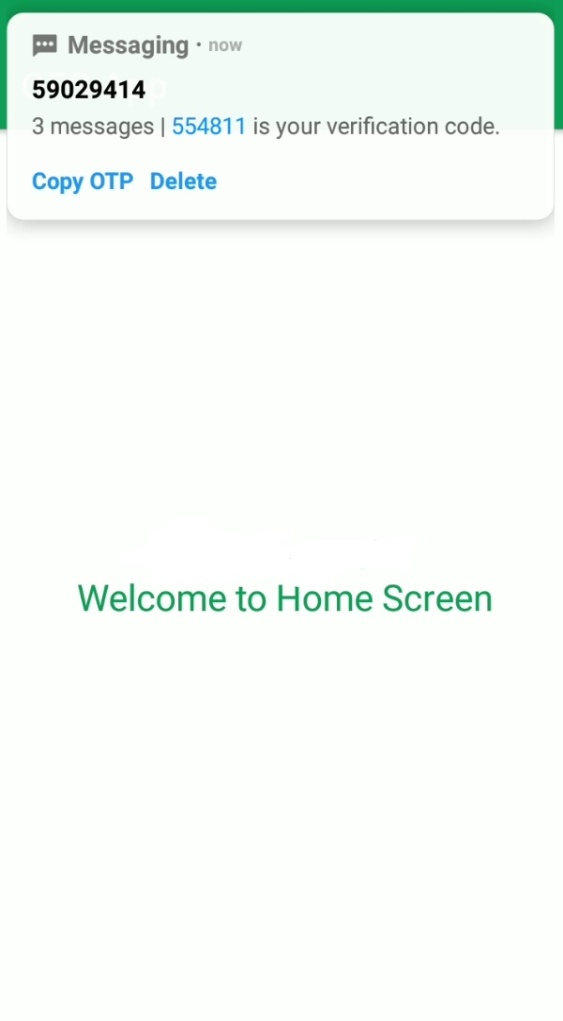
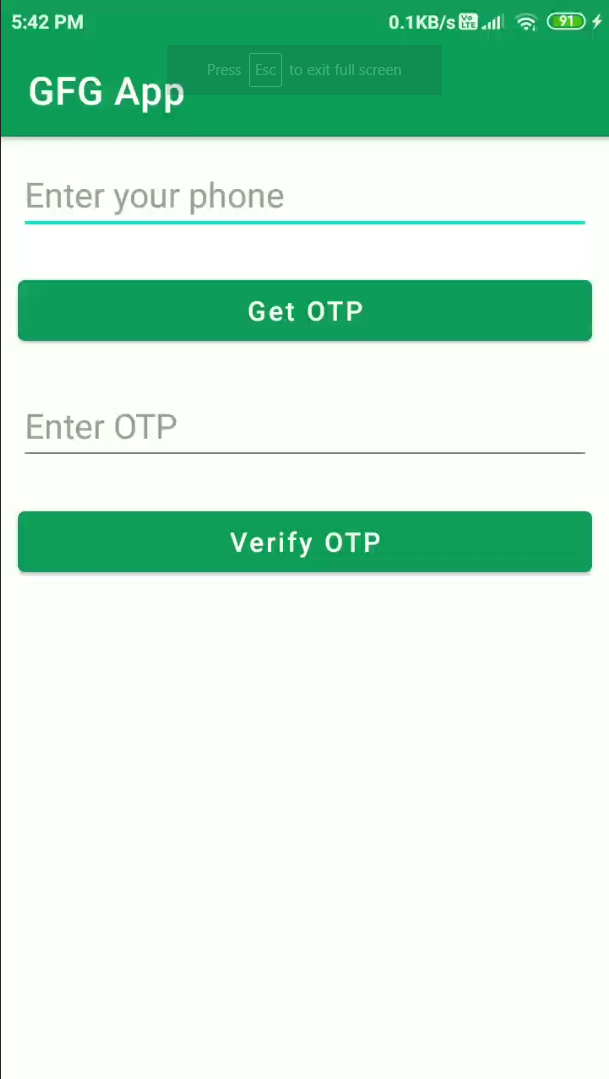
android:textAlignment="center"

android:textColor="@color/purple\_500"

android:textSize="20sp" />

</RelativeLayout>

OUTPUT:



RESULT:

Thus the android app for phone number verification by OTP using firebase has been created and executed successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:09 |  | Roll.no: |
| Date: |  | Page.no: |
| ANDROID APPLICATION FOR  EXTRACTING INFORMATION FOR JSON | | |

AIM:

To create an android application to extract employee information using JSON messages and load it in UI.

PROCEDURE:

STEP 1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click Next.

STEP 4: Add the dependency in your build.gradle file by navigate to the app ->

gradle scripts ->build.grade(app) and add the dependency code to your

file

STEP 5: Add permission to the internet in the AndroidManifest.xml file

STEP 6: Work with avtivity\_main.xml file and add the necessary variable in it.

STEP 7: Create a model class for storing our date by navigating to app -> java

–> your app’s package name -> right click on it -> new -> java class.

STEP 8: Write the coding in corresponding layout files for each item of our

RecyclerView.

STEP 9: Creating an Adapter class for setting data to our RecyclerView item.

STEP 10: Create an emulator using Window -> Android Virtual Device Manager

-> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

Activity\_main.xml

<RelativeLayoutxmlns:androclass="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity" >

<TextView

android:id="@+id/textView1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentLeft="true"

android:layout\_alignParentTop="true"

android:layout\_marginLeft="75dp"

android:layout\_marginTop="46dp"

android:text="TextView" />

</RelativeLayout>

MainActivity.java

import org.json.JSONException;

import org.json.JSONObject;

import android.app.Activity;

import android.os.Bundle;

import android.widget.TextView;

public class MainActivity extends Activity {

public static final String JSON\_STRING="{\"employee\":{\"name\":\"Sachin\",\"salary\":56000}}";

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

TextView textView1=(TextView)findViewById(R.id.textView1);

try{

JSONObject emp=(new JSONObject(JSON\_STRING)).getJSONObject("employee");

String empname=emp.getString("name");

int empsalary=emp.getInt("salary");

String str="Employee Name:"+empname+"\n"+"Employee Salary:"+empsalary;

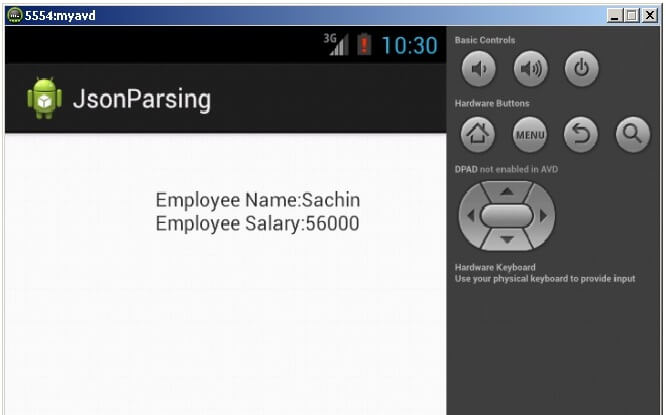
textView1.setText(str);

}catch (Exception e) {e.printStackTrace();}

}

}

OUTPUT:



RESULT:

Thus the android application for extracting information from JSON messages has been created and executed successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:10a |  | Roll.no: |
| Date: |  | Page.no: |
| MOBILE APPLICATION USING ANDROID  EDUCATIONAL QUIZ APP | | |

AIM :

To create an educational quiz application using android.

PROCEDURE:

STEP 1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click Next.

STEP 4: Click Next in the Configure project and Configure the attributes of the

icon set wizard.

STEP 5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP 7: Choose the version of emulator as Android 10.

STEP 8: Write the coding in corresponding layout files and save them.

STEP 9: Write the coding in corresponding Activity files and save them.

STEP 10: Create an emulator using Window -> Android Virtual Device Manager

-> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

Activity\_main.xml

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

<!--Using linear layout with vertical orientation and center gravity -->

<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:background="#FFFFFF"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:gravity="center"

tools:context=".MainActivity">

<!--ImageView used for showing pictures along with questions-->

<ImageView

android:id="@+id/myimage"

android:layout\_width="wrap\_content"

android:src="@drawable/f1"

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

<!--TextView used for showing questions on screen-->

<TextView

android:id="@+id/answer\_text\_view"

android:text="@string/a"

android:textColor="@android:color/black"

android:textSize="30sp"

android:padding="10dp"

android:layout\_width="wrap\_content"

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

<!--Using another LinearLayout for showing buttons

in horizontal orientation-->

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<!--TrueButton-->

<Button

android:id="@+id/true\_button"

android:layout\_marginRight="20dp"

android:backgroundTint="#5BD91B"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:textSize="20sp"

android:text="@string/true\_text" />

<!--FalseButton-->

<Button

android:id="@+id/false\_button"

android:layout\_marginLeft="20dp"

android:layout\_width="wrap\_content"

android:backgroundTint="#E33328"

android:layout\_height="wrap\_content"

android:textSize="20sp"

android:text="@string/false\_text" />

</LinearLayout>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<!--PreviousButton-->

<ImageButton

android:id="@+id/prev\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/baseline\_keyboard\_arrow\_left\_black\_18dp"

android:backgroundTint="#DFD2D1"

android:text="@string/prev\_text" />

<!--NextButton-->

<ImageButton

android:id="@+id/next\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:backgroundTint="#DFD2D1"

android:src="@drawable/baseline\_keyboard\_arrow\_right\_black\_18dp"

android:text="@string/next\_text" />

</LinearLayout>

</LinearLayout>

Questions.java

package org.geeksforgeeks.quizapp;

public class Question

{

// answerResId will store question

private int answerResId;

// answerTrue will store correct answer

// of the question provided

private booleananswerTrue;

public Question(int answerResId, booleananswerTrue)

{

// setting the values through

// arguments passed in constructor

this.answerResId = answerResId;

this.answerTrue = answerTrue;

}

// returning the question passed

public int getAnswerResId()

{

return answerResId;

}

// setting the question passed

public void setAnswerResId(int answerResId)

{

this.answerResId = answerResId;

}

// returning the correct answer

// of question

public booleanisAnswerTrue()

{

return answerTrue;

}

// setting the correct

// ans of question

public void setAnswerTrue(booleananswerTrue)

{

this.answerTrue = answerTrue;

}

}

Strings.xml

<resources>

<!--All the string resources come here including Questions and title -->

<string name="app\_name">GFG | HOW WELL DO YOU KNOW SIMRAN?</string>

<string name="correct"><b>CORRECTNESS IS</b> \n

<b>%1$d</b> OUT OF 6</string>

<string name="true\_text">true</string>

<string name="false\_text">false</string>

<string name="correct\_answer">That\'s correct</string>

<string name="wrong\_answer">That\'s incorrect</string>

<string name="a">Simran loves Chocolates.</string>

<string name="b">Simran Knows Following Skills:\n \t

<b>Ballet</b>\n \t

<b>HipHop</b></string>

<string name="c">Do You Think Simran Believes In:\n \t

<b>Luck!!!</b></string>

<string name="d">Do You Think Simran Wants To Visit <b>Italy</b></string>

<string name="e">Simran Loves Loyalty.</string>

<string name="f">Simran Sleeps Less</string>

<string name="next\_text">next</string>

<string name="prev\_text">previous</string>

</resources>

Main\_Activity.java

import android.annotation.SuppressLint;

import android.os.Build;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import android.widget.ImageButton;

import android.widget.ImageView;

import android.widget.TextView;

import android.widget.Toast;

import androidx.annotation.RequiresApi;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity

implements View.OnClickListener {

// setting up things

private Button falseButton;

private Button trueButton;

private ImageButtonnextButton;

private ImageButtonprevButton;

private ImageView Image;

private TextViewquestionTextView;

private int correct = 0;

// to keep current question track

private int currentQuestionIndex = 0;

private Question[] questionBank = new Question[] {

// array of objects of class Question

// providing questions from string

// resource and the correct ans

new Question(R.string.a, true),

new Question(R.string.b, false),

new Question(R.string.c, true),

new Question(R.string.d, true),

new Question(R.string.e, true),

new Question(R.string.f, false),

};

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// setting up the buttons

// associated with id

falseButton = findViewById(R.id.false\_button);

trueButton = findViewById(R.id.true\_button);

nextButton = findViewById(R.id.next\_button);

prevButton = findViewById(R.id.prev\_button);

// register our buttons to listen to

// click events

questionTextView

= findViewById(R.id.answer\_text\_view);

Image = findViewById(R.id.myimage);

falseButton.setOnClickListener(this);

trueButton.setOnClickListener(this);

nextButton.setOnClickListener(this);

prevButton.setOnClickListener(this);

}

@SuppressLint("SetTextI18n")

@RequiresApi(api = Build.VERSION\_CODES.LOLLIPOP)

@Override

public void onClick(View v)

{

// checking which button is

// clicked by user

// in this case user choose false

switch (v.getId()) {

case R.id.false\_button:

checkAnswer(false);

break;

case R.id.true\_button:

checkAnswer(true);

break;

case R.id.next\_button:

// go to next question

// limiting question bank range

if (currentQuestionIndex< 7) {

currentQuestionIndex

= currentQuestionIndex + 1;

// we are safe now!

// last question reached

// making buttons

// invisible

if (currentQuestionIndex == 6) {

questionTextView.setText(getString(

R.string.correct, correct));

nextButton.setVisibility(

View.INVISIBLE);

prevButton.setVisibility(

View.INVISIBLE);

trueButton.setVisibility(

View.INVISIBLE);

falseButton.setVisibility(

View.INVISIBLE);

if (correct > 3)

questionTextView.setText(

"CORRECTNESS IS " + correct

+ " "

+ "OUT OF 6");

// showing correctness

else

Image.setImageResource(

R.drawable.resu);

// if correctness<3 showing sad emoji

}

else {

updateQuestion();

}

}

break;

case R.id.prev\_button:

if (currentQuestionIndex> 0) {

currentQuestionIndex

= (currentQuestionIndex - 1)

% questionBank.length;

updateQuestion();

}

}

}

@RequiresApi(api = Build.VERSION\_CODES.LOLLIPOP)

private void updateQuestion()

{

Log.d("Current",

"onClick: " + currentQuestionIndex);

questionTextView.setText(

questionBank[currentQuestionIndex]

.getAnswerResId());

// setting the textview with new question

switch (currentQuestionIndex) {

case 1:

// setting up image for each

// question

Image.setImageResource(R.drawable.f2);

break;

case 2:

Image.setImageResource(R.drawable.f3);

break;

case 3:

Image.setImageResource(R.drawable.f4);

break;

case 4:

Image.setImageResource(R.drawable.f5);

break;

case 5:

Image.setImageResource(R.drawable.f6);

break;

case 6:

Image.setImageResource(R.drawable.f7);

break;

case 7:

Image.setImageResource(R.drawable.f1);

break;

}

}

private void checkAnswer(booleanuserChooseCorrect)

{

booleananswerIsTrue

= questionBank[currentQuestionIndex]

.isAnswerTrue();

// getting correct ans of current question

int toastMessageId;

// if ans matches with the

// button clicked

if (userChooseCorrect == answerIsTrue) {

toastMessageId = R.string.correct\_answer;

correct++;

}

else {

// showing toast

// message correct

toastMessageId = R.string.wrong\_answer;

}

Toast

.makeText(MainActivity.this, toastMessageId,

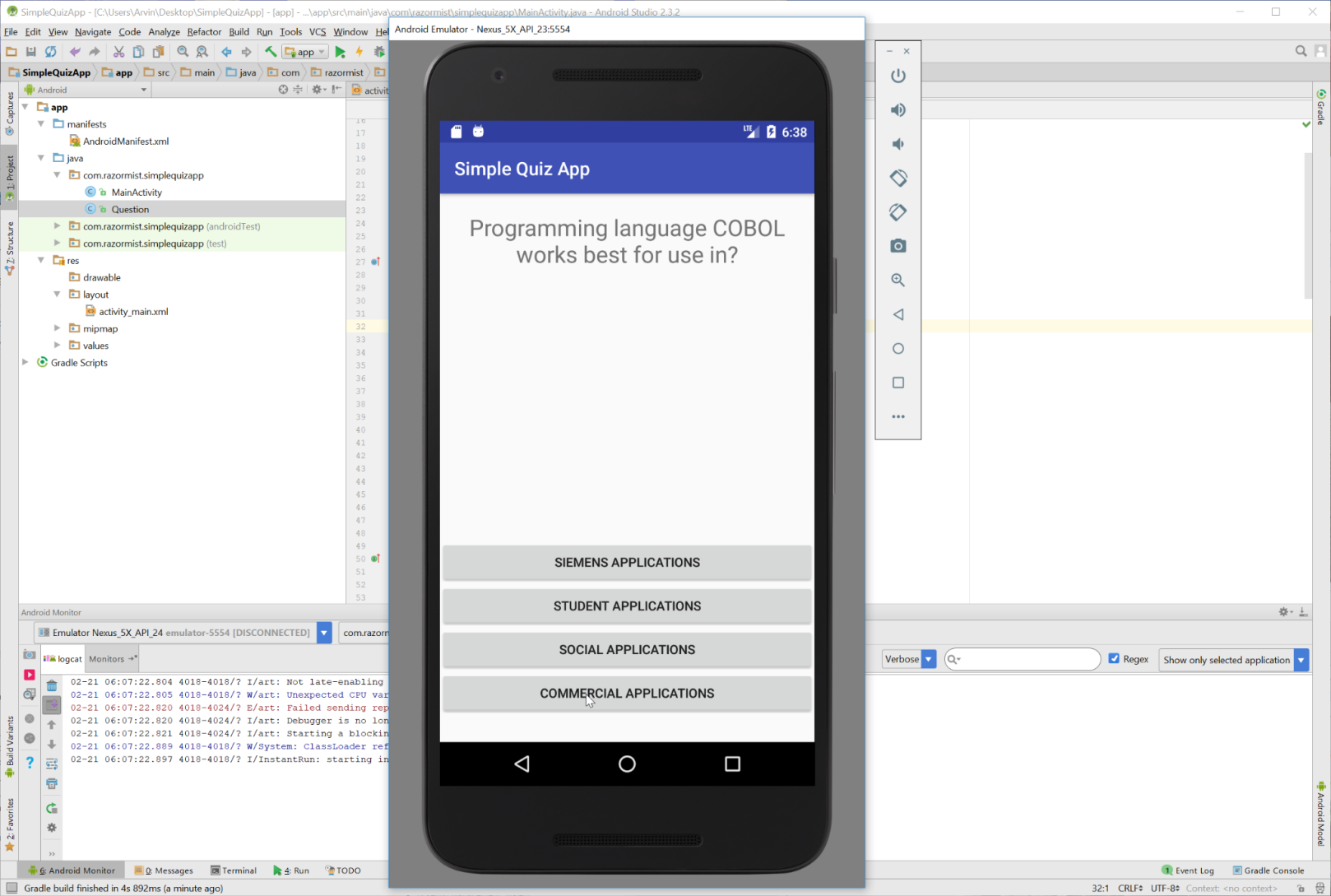
Toast.LENGTH\_SHORT)

.show();

}

}

OUTPUT:



RESULT:

Thus the android application for an educational quiz application has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:10b |  | Roll.no: |
| Date: |  | Page.no: |
| MOBILE APPLICATION USING ANDROID  TOUR GUIDE APP | | |

**AIM:**

PROCEDURE:

STEP 1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click Next.

STEP 4: Click Next in the Configure project and Configure the attributes of the

icon set wizard.

STEP 5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP 7: Choose the version of emulator as Android 10.

STEP 8: Write the coding in corresponding layout files and save them.

STEP 9: Write the coding in corresponding Activity files and save them.

STEP 10: Create an emulator using Window -> Android Virtual Device Manager

-> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

Activity\_main.xml

<LinearLayoutxmlns: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"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="vertical"

tools:context=".MainActivity">

<androidx.cardview.widget.CardView

android:id="@+id/history"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Erode History"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/shopping"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Shopping Places"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/restarunts"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Famous Restaurants"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/park"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Famous Parks"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

</LinearLayout>

MainActivity.java

package com.madlab.tour;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import android.app.Activity;

import androidx.cardview.widget.CardView;

public class MainActivity extends Activity {

CardViewhistory,shopping,park,restaurant;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

history = findViewById(R.id.history);

restaurant = findViewById(R.id.restarunts);

park = findViewById(R.id.park);

shopping = findViewById(R.id.shopping);

history.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.history));

startActivity(intent);

}

});

restaurant.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.restaurant));

startActivity(intent);

}

});

park.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.park));

startActivity(intent);

}

});

shopping.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.shopping));

startActivity(intent);

}

});

}

}

Activtiy\_second.xml

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

<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center"

tools:context=".SecondActivity">

<TextView

android:id="@+id/text"

android:layout\_width="wrap\_content"

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

</LinearLayout>

SecondActivity.java

package com.madlab.tour;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.widget.TextView;

import org.w3c.dom.Text;

public class SecondActivity extends AppCompatActivity {

TextViewtextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_second);

Intent intent = getIntent();

textView = findViewById(R.id.text);

textView.setText(intent.getExtras().getString("text"));

} }

Strings.xml

<resources>

<string name="app\_name">Tour Guide</string>

<string name="history">Erode District was a part of Coimbatore has its history intervened with that of Coimbatore and because of its close linkage with the erstwhile Coimbatore district. It is very difficult to separately deal with the history of Erode region. Together with the area comprised in the Coimbatore district, it formed part of the ancient Kongu region. It is found that in the early days, this area was occupied by tribes, most prominent among them being the “Kosars” reportedly having their headquarters at ‘Kosamputhur’ which is believed to have in due course become Coimbatore. </string>

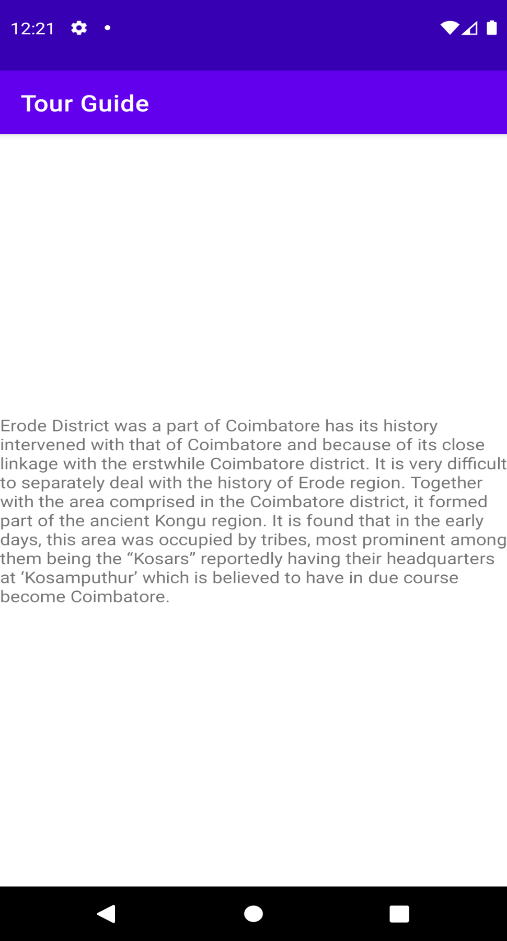
<string name="park">pannirSelvam park VOC Park</string>

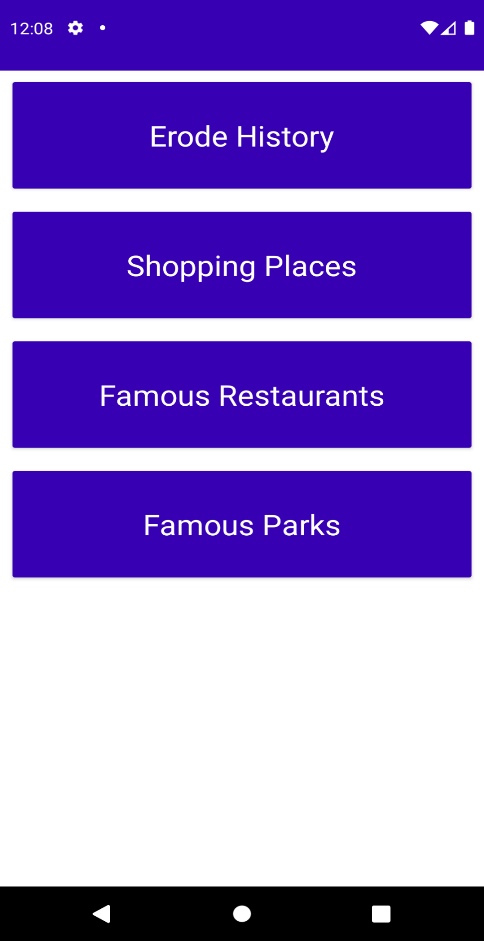
<string name="shopping">Reliance Mall</string>

<string name="restaurant">Kfc , Star Briyani</string>

</resources

OUTPUT:





RESULT:

Thus the android application for Tour Guide Application has been executed and verified successfully.

|  |  |  |
| --- | --- | --- |
| Ex.no:10c |  | Roll.no: |
| Date: |  | Page.no: |
| MOBILE APPLICATION USING ANDROID  NEWS FEED APP | | |

AIM:

PROCEDURE:

STEP 1: Start the process.

STEP 2: In Eclipse, create a new Android application project by clicking the

menuFile -> New -> Android Application Project.

STEP 3: Give the application name, project name, package name and click Next.

STEP 4: Click Next in the Configure project and Configure the attributes of the

icon set wizard.

STEP 5: Select Blank activity and click Next.

STEP 6: Specify the Activity and Layout name. Click Finish.

STEP 7: Choose the version of emulator as Android 10.

STEP 8: Write the coding in corresponding layout files and save them.

STEP 9: Write the coding in corresponding Activity files and save them.

STEP 10: Create an emulator using Window -> Android Virtual Device Manager

-> New Android Virtual Device.

STEP 11: Select the Android Virtual Device and give start and launch.

STEP 12: Now run the project on emulator.

STEP 13: Stop the process.

PROGRAM:

Activty\_main.xml

<LinearLayoutxmlns: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"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="vertical"

tools:context=".MainActivity">

<androidx.cardview.widget.CardView

android:id="@+id/news"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="News HeadLines"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/sports"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Sports"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/cinema"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Cinema"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

<androidx.cardview.widget.CardView

android:id="@+id/science"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

app:cardBackgroundColor="@color/purple\_700">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Science"

android:padding="30dp"

android:textSize="25dp"

android:textColor="@color/white"

android:layout\_gravity="center"/>

</androidx.cardview.widget.CardView>

</LinearLayout>

MainActivtiy.java

package com.madlab.newsfeed;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import android.app.Activity;

import androidx.cardview.widget.CardView;

public class MainActivity extends Activity {

CardViewnews,cinema,sports,science;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

news = findViewById(R.id.news);

science = findViewById(R.id.science);

sports = findViewById(R.id.sports);

cinema = findViewById(R.id.cinema);

news.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.news));

startActivity(intent);

}

});

science.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.science));

startActivity(intent);

}

});

sports.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.sports));

startActivity(intent);

}

});

cinema.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

Intent intent = new Intent(MainActivity.this,SecondActivity.class);

intent.putExtra("text",getResources().getString(R.string.cinema));

startActivity(intent);

}

});

}

}

Activtiy\_second.xml

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

<LinearLayoutxmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:gravity="center"

tools:context=".SecondActivity">

<TextView

android:id="@+id/text"

android:layout\_width="wrap\_content"

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

</LinearLayout>

SecondActivity.java

package com.madlab.newsfeed;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.widget.TextView;

import org.w3c.dom.Text;

public class SecondActivity extends AppCompatActivity {

TextViewtextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_second);

Intent intent = getIntent();

textView = findViewById(R.id.text);

textView.setText(intent.getExtras().getString("text"));

}

}

Strings.xml

<resources>

<string name="app\_name">News Feeder</string>

<string name="news">It feels good to get so much love: Nikhat Zareen on winning Gold Medal</string>

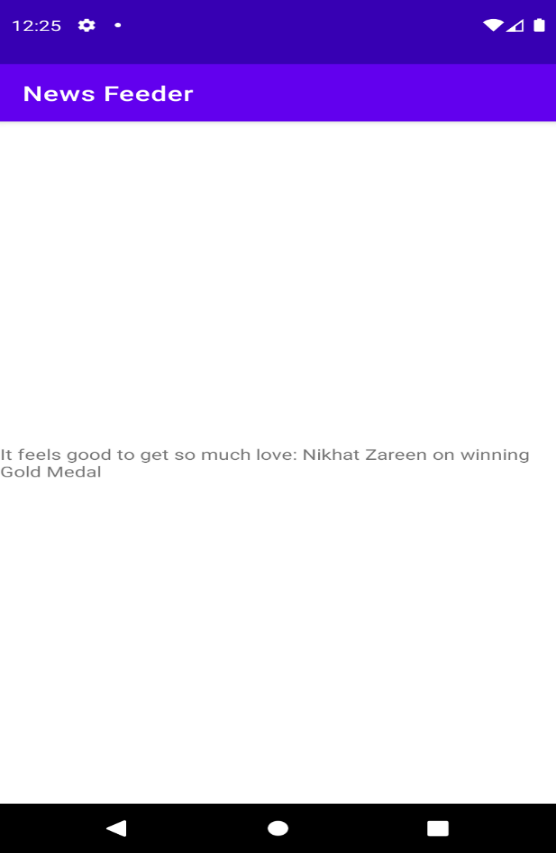
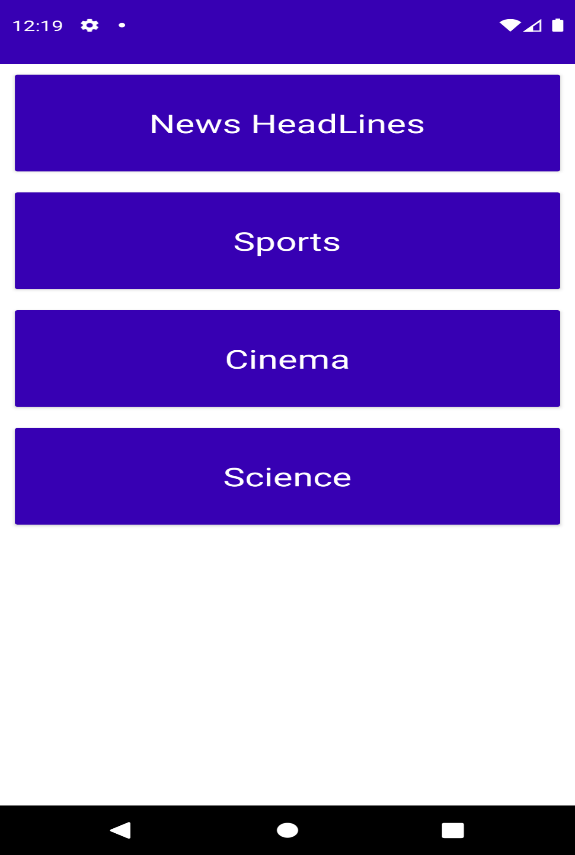
<string name="cinema">Thor: Love And Thunder Trailer - Chris Hemsworth-Natalie Portman Double Whammy Vs Christian Bale</string>

<string name="science">NASA\'s diffractive solar sailing project could take science to new heights</string>

<string name="sports">IPL 2022, GT vs RR Live Score Updates: Jos Buttler Scores 89 To Set 189-Run Target For GT | Cricket...</string>

</resources>

OUTPUT:



RESULT:

Thus the android application for News Feed application has been executed and verified successfully.