ATSS's

Institute of Industrial and Computer Management and Research, Nigdi Pune

MCA Department

Academic Year: 2022-23

Practical Journal on
IT31L- Mobile Application Development
(SEM-III)

Submitted By:

Name: Ashutosh Deshmukh

Roll No: 15

Seat No:

Date: 14-12-2022

Course Outcome:

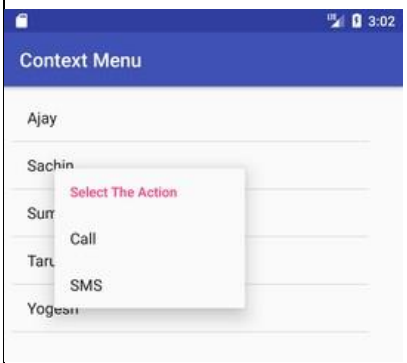
Student will be able to:

CO1: Develop mobile application. (Apply)

ATSS's
Institute of Industrial and Computer Management and Research, Nigdi Pune
MCA Department

INDEX

Students Name: Ashutosh Chandrakant Deshmukh **Roll No. 15**

Sr. No	Program Title	Course Outcome	Page No.	Teacher's Sign with Date	Remarks
1.	Create a rating bar application, where user will rate a product. Display the rating using Toast.	CO1			
2.	Create an app to accept package delivery method from given radio button options. On Clicking of button, display the selected option using Toast.	CO1			
3.	Write a code to display "Do you want to close this application? "AlertDialog box. If user clicks on Yes, close the application and if clicks No, display "you choose no action for alertbox" message	CO1			
4.	Create an option menu with options New Group, New Broadcast, Payments, Settings & search. Display search option as an menu icon.	CO1			
5.	Create following application using ContextMenu. When user long clicks on name, display menu options as Call, SMS. As per the option selected, display proper message. 	CO1			

6.	Write an application to accept a favourite programming language from user. Autocomplete the answer by using AutoCompleteTextView & ArrayAdapter	**CO1**			
7.	Write an android code to turn ON /OFF the Wi-Fi.	**CO1**			
8.	Create a fragment that has its own UI and enable your activities to communicate with fragments.	**CO1**			
9.	Create an application using two activities. In first activity- accept user name, pass the same to next activity & display a “Hello & Welcome <username>” using Intent.	**CO1**			
10.	Write an android code to make a phone call using Intent.	**CO1**			
11.	Write an android application using SQLite to create table and perform CRUD operations Consider a COURSE table with fields C_ID, C_Name, C_Duration, C_Description, perform ADD, UPDATE, DELETE and READ operations.	**CO1**			
12.	Create an Android app to manage STUDENT data, powered by Firebase Realtime database that supports: Adding Data to Firebase Realtime database, Retrieving Data from Firebase, Update a record and Deleting data from firebase data.	**CO1**			
13.	Write an android app to write JSON data into a file and read JSON data from created file.	**CO1**			
14.	Write a ReactNative code to display an image, which is stored in images folder of an app to the user.	**CO1**			
15.	Develop simple flutter application to open a browser using Android SDK	**CO1**			

1. Create a rating bar application, where user will rate a product. Display the rating using Toast

Code:

MainActivity.java –

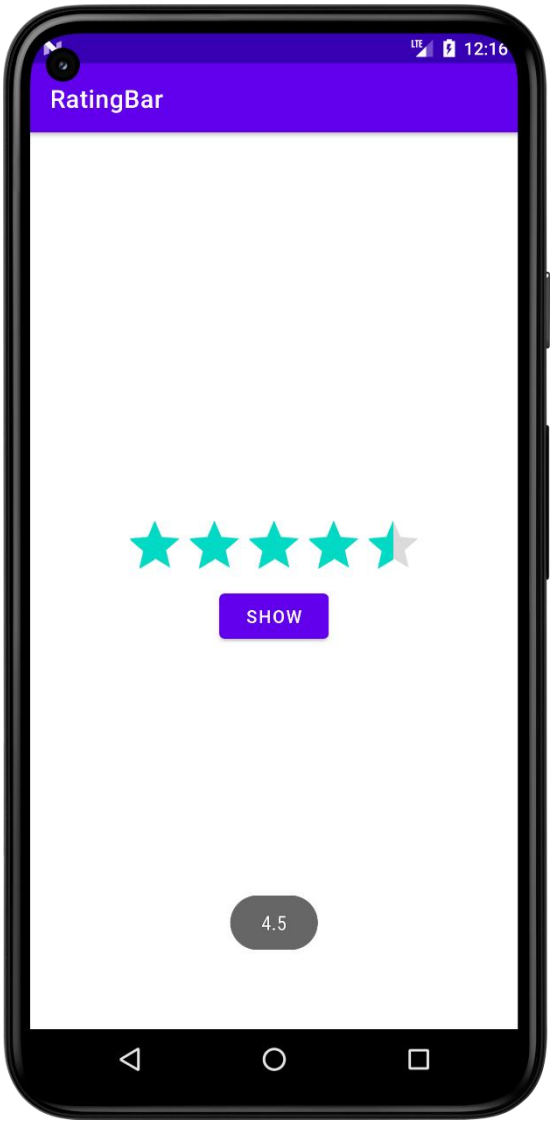
```
package com.example.ratingbar;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.RatingBar;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    Button btn1;
    RatingBar rbar;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

//        Access button and rating bar from XML
        rbar = findViewById(R.id.rate);
        btn1 = findViewById(R.id.btnShow);

//        handling button click
        btn1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
//                Get The Rating Given By User
                float f = rbar.getRating();

//                Display the rating
                Toast.makeText(MainActivity.this, String.valueOf(f),
Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

Main_Activity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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"
    android:orientation="vertical"

    android:gravity="center" >
```

```
    <RatingBar
        android:id="@+id/rate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

    <Button
        android:id="@+id/btnShow"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show" />

</LinearLayout>
```

Output:



2. Create an app to accept package delivery method from given radio button options. On Clicking of button, display the selected option using Toast.

Code:

MainActivity.java –

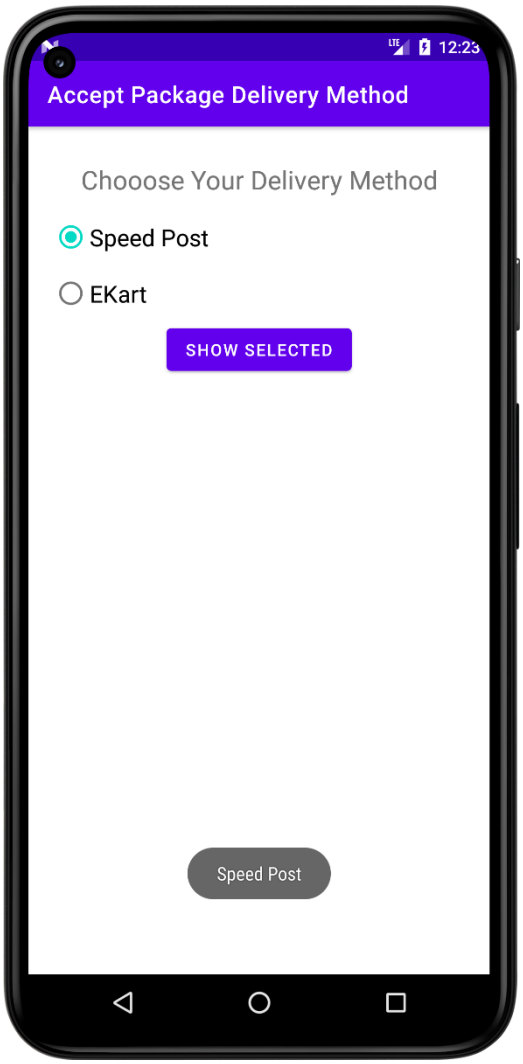
```
package com.example.acceptpackagedeliverymethod;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    Button button;
    RadioButton genderradioButton;
    RadioGroup radioGroup;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        radioGroup=(RadioGroup)findViewById(R.id.radioGroup);
    }

    public void onclickbuttonMethod(View v){
        int selectedId = radioGroup.getCheckedRadioButtonId();
        genderradioButton = (RadioButton) findViewById(selectedId);
        if(selectedId==-1){
            Toast.makeText(MainActivity.this,"Nothing selected",
Toast.LENGTH_SHORT).show();
        }
        else{
            Toast.makeText(MainActivity.this,genderradioButton.getText(),
Toast.LENGTH_SHORT).show();
        }

    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">
```

```
    <TextView
        android:id="@+id/textView2"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:gravity="center_horizontal"
        android:textSize="22sp"
        android:text="Chooose Your Delivery Method" />

    <RadioGroup
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/radioGroup"
        android:paddingStart="20dp" >

        <RadioButton
            android:id="@+id/radioMale"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Speed Post"
            android:layout_marginTop="10dp"
            android:checked="false"
            android:textSize="20sp" />

        <RadioButton
            android:id="@+id/radioFemale"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="EKart"
            android:layout_marginTop="0dp"
            android:checked="false"
            android:textSize="20sp" />
    </RadioGroup>

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Selected"
        android:id="@+id/button"
        android:onClick="onclickbuttonMethod"
        android:layout_gravity="center_horizontal" />

</LinearLayout>
```

Output:



3. **Write a code to display "Do you want to close this application? "Alert Dialog box. If user clicks on Yes, close the application and if clicks No, display "you choose no action for alert box" message**

Code:

MainActivity.java –

```
package com.example.alertdialogdemo;

import androidx.appcompat.app.AppCompatActivity;

import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
```

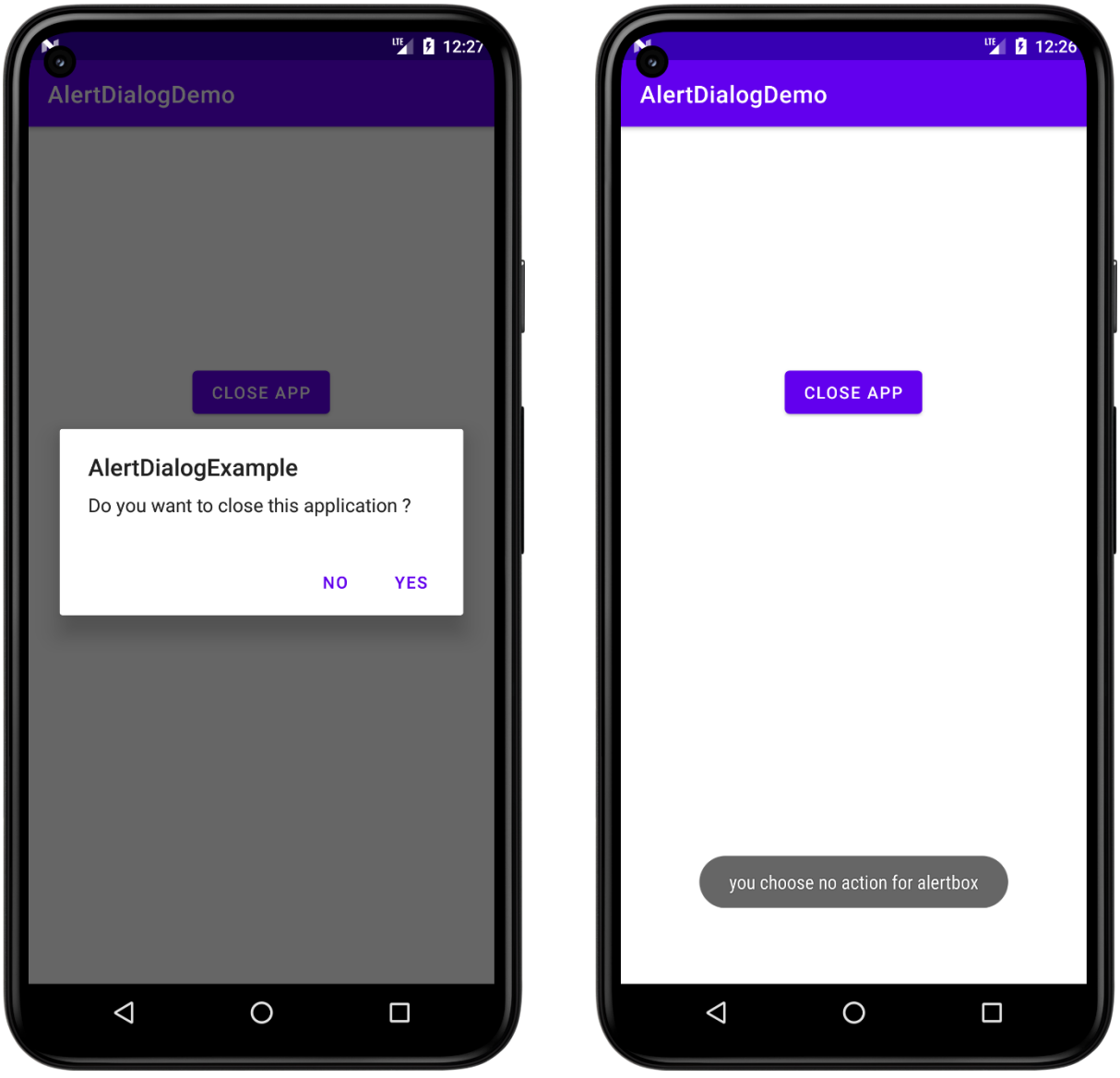
```
public class MainActivity extends AppCompatActivity {
    Button closeButton;
    AlertDialog.Builder builder;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        closeButton = (Button) findViewById(R.id.button);
        builder = new AlertDialog.Builder(this);

        closeButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                //Setting message manually and performing action on button
click
                builder.setMessage("Do you want to close this application
?")
                        .setCancelable(false)
                        .setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
                            public void onClick(DialogInterface dialog, int
id) {
                                finish();
                                Toast.makeText(getApplicationContext(),"you
choose yes action for alertbox",
                                        Toast.LENGTH_SHORT).show();
                            }
                        })
                        .setNegativeButton("No", new
DialogInterface.OnClickListener() {
                            public void onClick(DialogInterface dialog, int
id) {
                                //  Action for 'NO' Button
                                dialog.cancel();
                                Toast.makeText(getApplicationContext(),"you
choose no action for alertbox",
                                        Toast.LENGTH_SHORT).show();
                            }
                        });
                //Creating dialog box
                AlertDialog alert = builder.create();
                //Setting the title manually
                alert.setTitle("AlertDialogExample");
                alert.show();
            }
        });
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/button"
        android:text="Close app"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"/>

</LinearLayout>
```

Output:



4. Create an option menu with options New Group, New Broadcast, Payments, Settings & search. Display search option as a menu icon

Code:

MainActivity.java –

```
package com.example.optionmenuwithicon;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is
present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();
        switch (id) {
            case R.id.option1:
                Toast.makeText(getApplicationContext(),
                        "New Group Selected",
                        Toast.LENGTH_LONG).show();
                return true;
            case R.id.option2:
                Toast.makeText(getApplicationContext(),
                        "New Broadcast Selected",
                        Toast.LENGTH_LONG).show();
                return true;
            case R.id.option3:
                Toast.makeText(getApplicationContext(),
                        "Payments Selected ",
                        Toast.LENGTH_LONG).show();
                return true;
            case R.id.option4:
                Toast.makeText(getApplicationContext(),
                        "Settings Selected",
                        Toast.LENGTH_LONG).show();

                return true;
```

```
            case R.id.option5:
                Toast.makeText(getApplicationContext(),
                        "Search Selected",
                        Toast.LENGTH_LONG).show();
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }
}
```

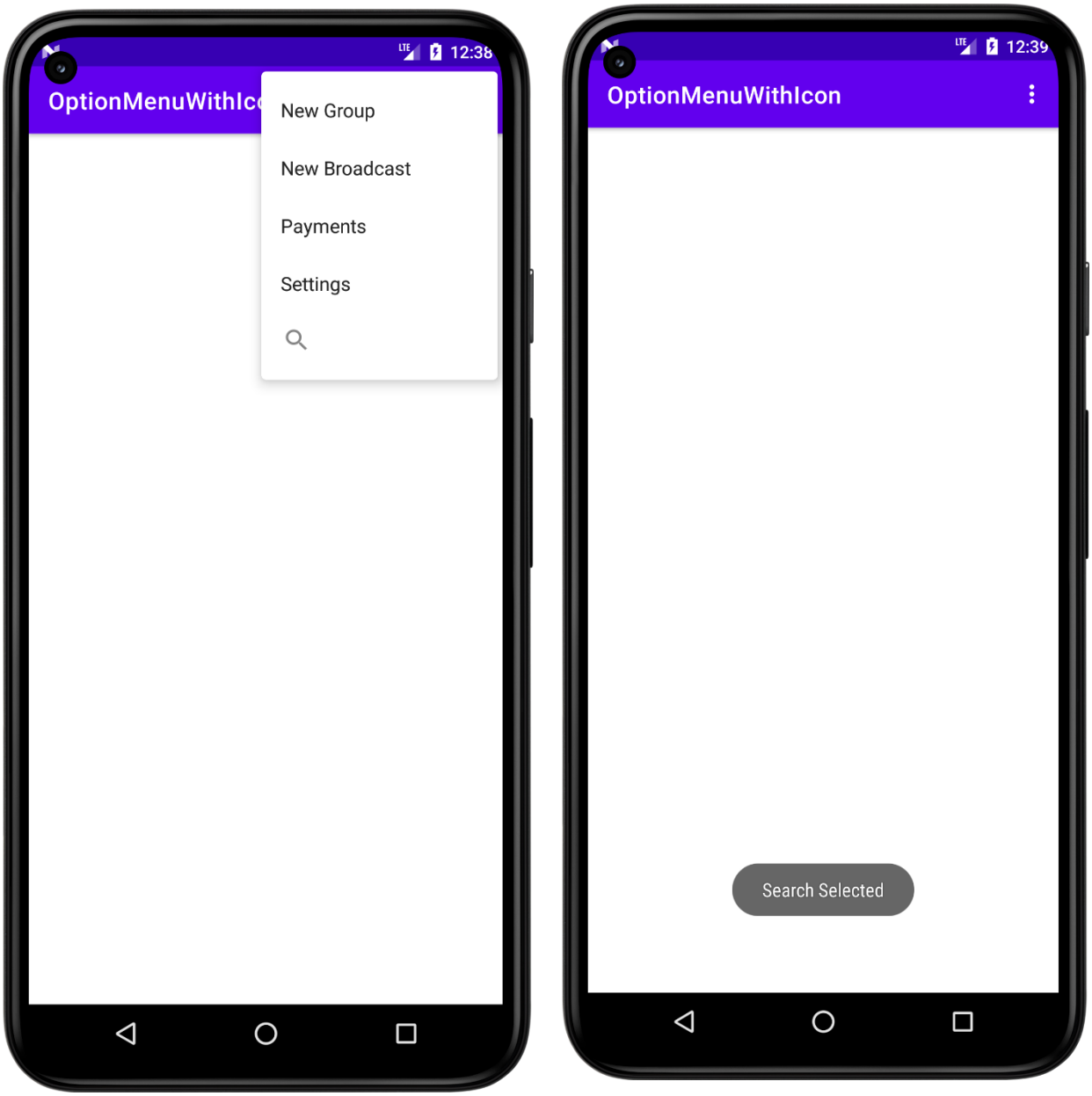
MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns: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">


</androidx.constraintlayout.widget.ConstraintLayout>
```

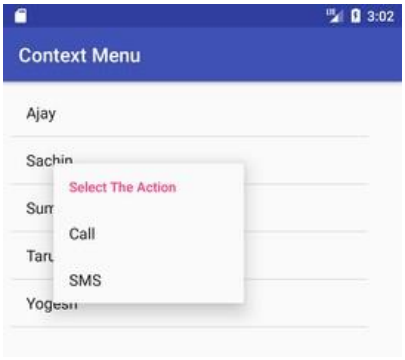
Menu_main.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/option1"
        android:title="@string/Option1" />
    <item
        android:id="@+id/option2"
        android:title="@string/Option2" />
    <item
        android:id="@+id/option3"
        android:title="@string/Option3" />
    <item
        android:id="@+id/option4"
        android:title="@string/Option4" />
    <item
        android:id="@+id/option5"
        android:title="@string/Option5"
        android:icon="@drawable/ic_baseline_search_24" />
</menu>
```

Output:


12:38
OptionMenuWithIc
New Group
New Broadcast
Payments
Settings
12:39
OptionMenuWithIcon
Search Selected

5. Create following application using Context Menu. When user long clicks on name, display menu options as Call, SMS. As per the option selected, display proper message



Code:

MainActivity.java –

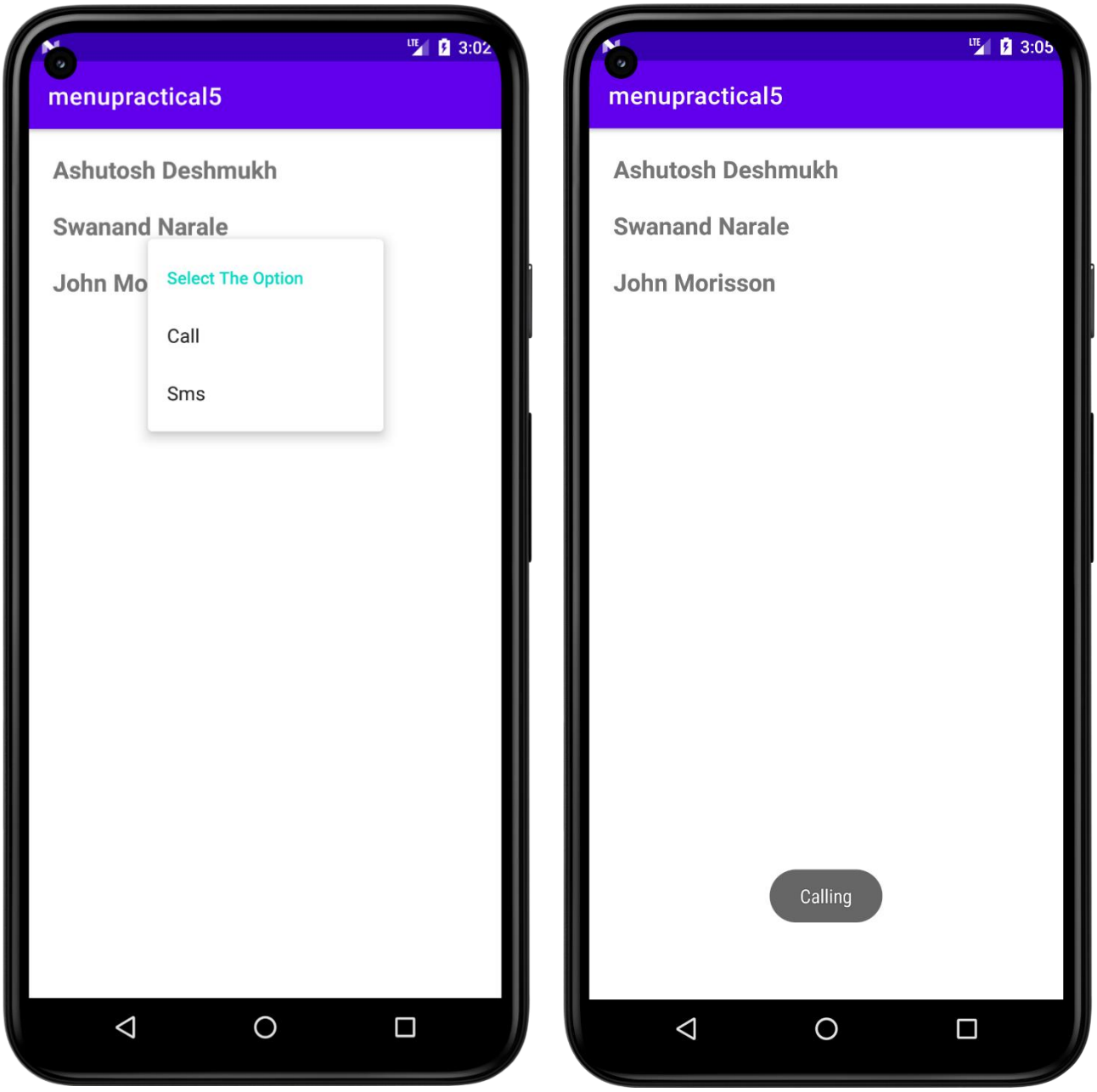
```
package com.example.menupractical5;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    TextView textView, textView1, textView2;
    Button button;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        textView = (TextView) findViewById((R.id.item1));
        textView1 = (TextView) findViewById((R.id.item2));
        textView2 = (TextView) findViewById((R.id.item3));
        registerForContextMenu(textView);
        registerForContextMenu(textView1);
        registerForContextMenu(textView2);
    }
    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
ContextMenu.ContextMenuInfo menuInfo) {
        super.onCreateContextMenu(menu, v, menuInfo);

        menu.setHeaderTitle("Select The Option");
//          add menu items
        menu.add(0, v.getId(),0,"Call");
        menu.add(0, v.getId(),0,"Sms");
```

```
    }
    @Override
    public boolean onContextItemSelected(@NonNull MenuItem item) {
        if(item.getTitle() == "Call"){
            Toast.makeText(getApplicationContext(), "Calling",
Toast.LENGTH_SHORT).show();
        } else if(item.getTitle() == "Sms"){
            Toast.makeText(getApplicationContext(), "Sending Sms",
Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(getApplicationContext(), "Invalid Option
Clicked", Toast.LENGTH_SHORT).show();
        }
        return true;
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/item1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="20dp"
        android:text="Ashutosh Deshmukh"
        android:textSize="20sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/item2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="20dp"
        android:text="Swanand Narale"
        android:textSize="20sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/item3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="20dp"
        android:text="John Morisson"
        android:textSize="20sp"
        android:textStyle="bold" />
</LinearLayout>
```

Output:


3:02
menupractical5
Ashutosh Deshmukh
Swanand Narale
John Mo
Select The Option
Call
Sms
3:05
menupractical5
Ashutosh Deshmukh
Swanand Narale
John Morisson
Calling

6. Write an application to accept a favourite programming language from user. Autocomplete the answer by using Auto Complete Text View & Array Adapter

Code:

MainActivity.java –

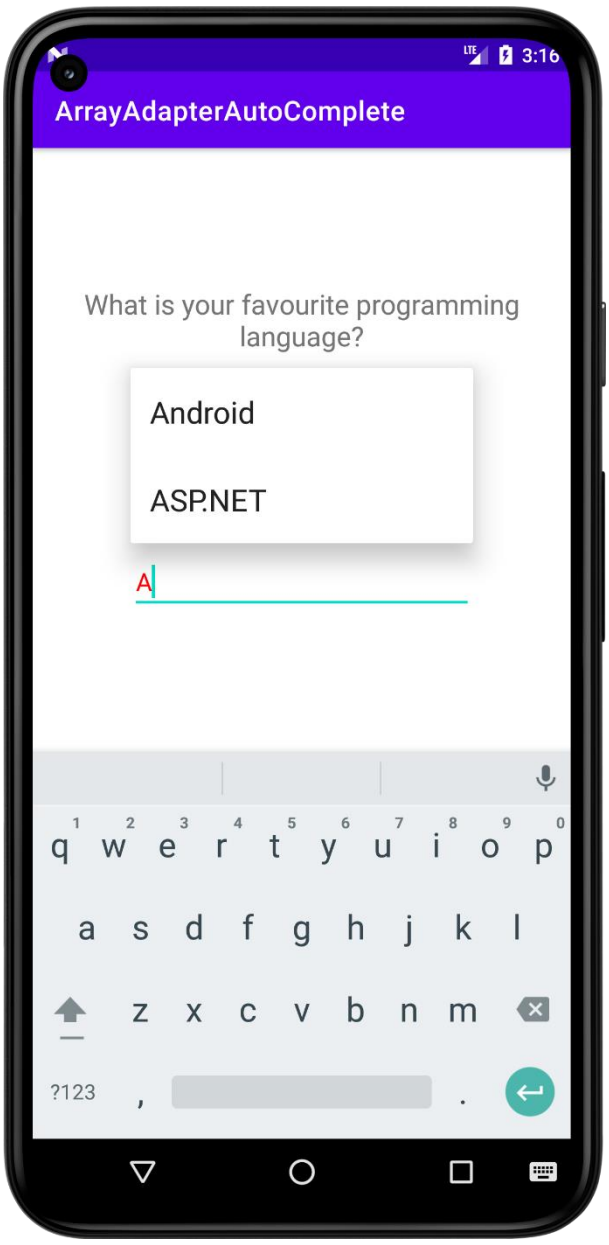
```
package com.example.arrayadapterautocomplete;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.AutoCompleteTextView;

public class MainActivity extends AppCompatActivity {
    String[] language
={"C","C++","Java",".NET","iPhone","Android","ASP.NET","PHP", "Flutter",
"React Native"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        AutoCompleteTextView actv = findViewById
(R.id.autoCompleteTextView);
        //Creating the instance of ArrayAdapter containing list of language
names
        ArrayAdapter<String> adapter = new ArrayAdapter <String>
                (this, android.R.layout.select_dialog_item, language);
        //Getting the instance of AutoCompleteTextView
        actv.setThreshold(1); //will start working from first character
        //setting the adapter data into the AutoCompleteTextView
        actv.setAdapter (adapter);
        actv.setTextColor (Color.RED);
    }
}
```

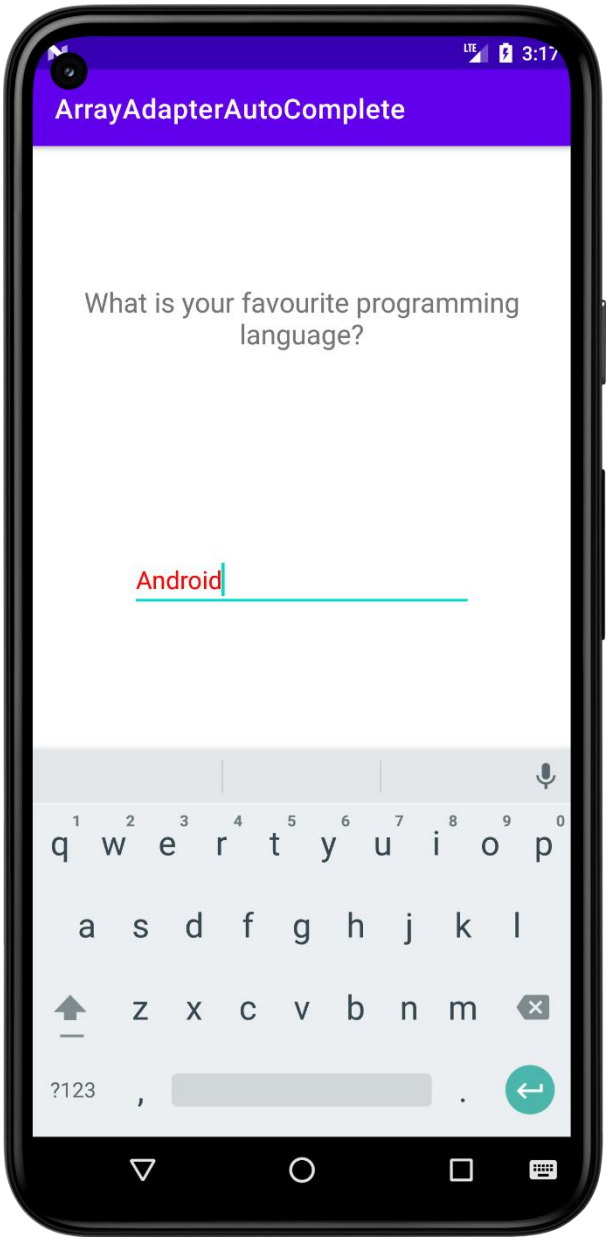
MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="What is your favourite programming language?"

        android:layout_marginTop="100dp"
```

```
        android:textSize="20sp"
        android:textAlignment="center"/>

    <AutoCompleteTextView
        android:id="@+id/autoCompleteTextView"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="144dp"
        android:layout_gravity="center"
        android:text=""
 />

</LinearLayout>
```

Output:





7. Write an android code to turn ON /OFF the Wi-Fi.

Code:

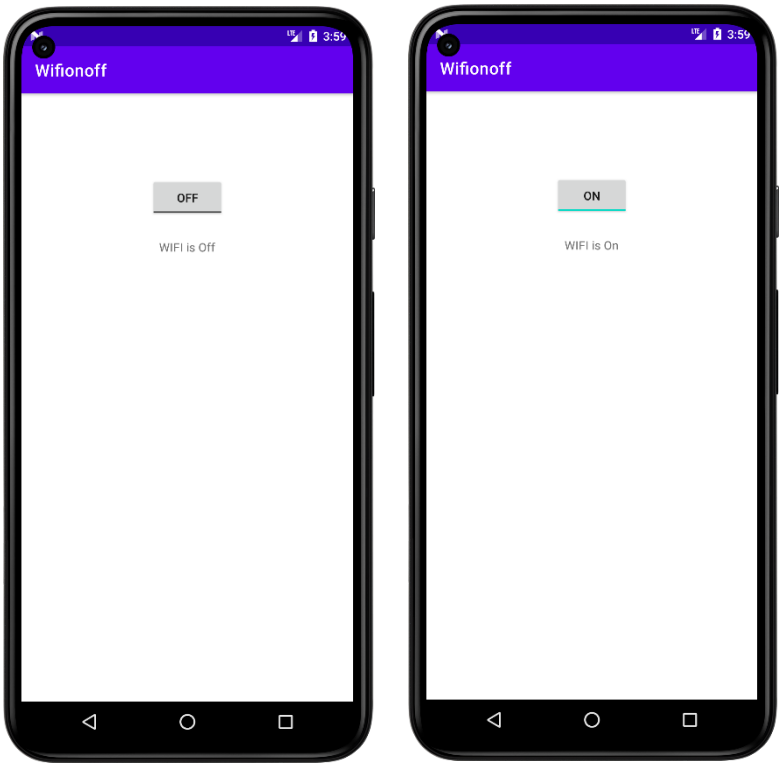
MainActivity.java –

```
package com.example.wifionoff;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.widget.CompoundButton;
import android.widget.TextView;
import android.widget.ToggleButton;

public class MainActivity extends AppCompatActivity {
    ToggleButton toggleButton;
    TextView textView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Getting toggle button and textView from activity_main
        toggleButton = findViewById(R.id.toggleButton);
        textView = findViewById(R.id.textView);

        toggleButton.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton compoundButton,
boolean checked) {
                if(checked){
                    textView.setText("WIFI is On");
                    WifiManager wifi = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);
                    wifi.setWifiEnabled(true);
                }else {
                    textView.setText("WIFI is Off");
                    WifiManager wifi = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);
                    wifi.setWifiEnabled(false);
                }
            }
        });
//        For initial setting
        if(toggleButton.isChecked()){
            textView.setText("WiFi is ON");
            WifiManager wifi = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);
            wifi.setWifiEnabled(true);
        } else {
            textView.setText("WiFi is OFF");
            WifiManager wifi = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);
            wifi.setWifiEnabled(false);
        }
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <ToggleButton
        android:id="@+id/toggleButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="100dp"
        android:checked="false" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="25dp"
        android:layout_gravity="center"/>

</LinearLayout>
```

Output:



8. Create a fragment that has its own UI and enable your activities to communicate with fragments.

Code:

MainActivity.java –

```
package com.example.fragments;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;

import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {
    // creating object for Fragment
    Fragment selectedFragment;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    // method for displaying the appropriate
    // fragment according to the clicked button
    public void selectFragment(View view) {
        if (view == findViewById(R.id.button1)) {
            selectedFragment = new FirstFragment();
        } else {
            selectedFragment = new SecondFragment();
        }
        getSupportFragmentManager()
                .beginTransaction()
                .replace(R.id.fragment_container, selectedFragment)
                .commit();
    }
}
```

FirstFragment.java –

```
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
import com.example.fragments.R;

public class FirstFragment extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable
ViewGroup container, @Nullable Bundle savedInstanceState) {
//return super.onCreateView(inflater, container, savedInstanceState);
        return inflater.inflate(R.layout.first_fragment,container,false);
    }

}
```

SecondFragment.java –

```
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
import com.example.fragments.R;

public class SecondFragment extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable
ViewGroup container,
                             @Nullable Bundle savedInstanceState) {
        //return super.onCreateView(inflater, container,
savedInstanceState);
        return inflater.inflate(R.layout.second_fragment,container,false);
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <!-- Heading of the activity -->
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:layout_marginBottom="20dp"
        android:text="@string/heading"
        android:textAlignment="center"
        android:textColor="@android:color/holo_green_light"
        android:textSize="24sp"
        android:textStyle="bold" />

    <!-- Button to display first fragment -->
    <Button
        android:id="@+id/button1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:background="#4CAF50"
        android:onClick="selectFragment"
        android:text="@string/fragment1_button"

        android:textColor="@android:color/background_light"
```

```
        android:textSize="18sp"
        android:textStyle="bold" />

    <!-- Button to display second fragment -->
    <Button
        android:id="@+id/button2"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:background="#4CAF50"
        android:onClick="selectFragment"
        android:text="@string/fragment2_button"
        android:textColor="@android:color/background_light"
        android:textSize="18sp"
        android:textStyle="bold" />

    <!-- Adding Fragment element in the activity -->
    <FrameLayout
        android:id="@+id/fragment_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</LinearLayout>
```

First_Fragment.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/first"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Hello, First Fragment"
        android:textSize="30sp"
        android:gravity="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Second_Fragment.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/second"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Hello, Second Fragment"
```

```
        android:textSize="30sp"
        android:gravity="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:

9. Create an application using two activities. In first activity- accept user name, pass the same to next activity & display a "Hello & Welcome <username>" using Intent.

Code:

MainActivity.java –

```
package com.example.activitiesdemo;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    public static final String EXTRA_MESSAGE = "MSG";
    EditText mMessageEditText;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mMessageEditText = findViewById(R.id.editText_main);
    }
    public void launchSecondActivity(View view) {
        Intent intent = new Intent(this, SecondActivity.class);
        String message = mMessageEditText.getText().toString();
        intent.putExtra(EXTRA_MESSAGE, message);
        startActivityForResult(intent, 1);
        mMessageEditText.setText("");
    }
}
```

SecondActivity.java –

```
package com.example.activitiesdemo;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;

public class SecondActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);

        Intent intent = getIntent();
        String message = intent.getStringExtra(MainActivity.EXTRA_MESSAGE);
        TextView textView = findViewById(R.id.text_message);
        textView.setText("Hello & Welcome " +message);
    }
}
```

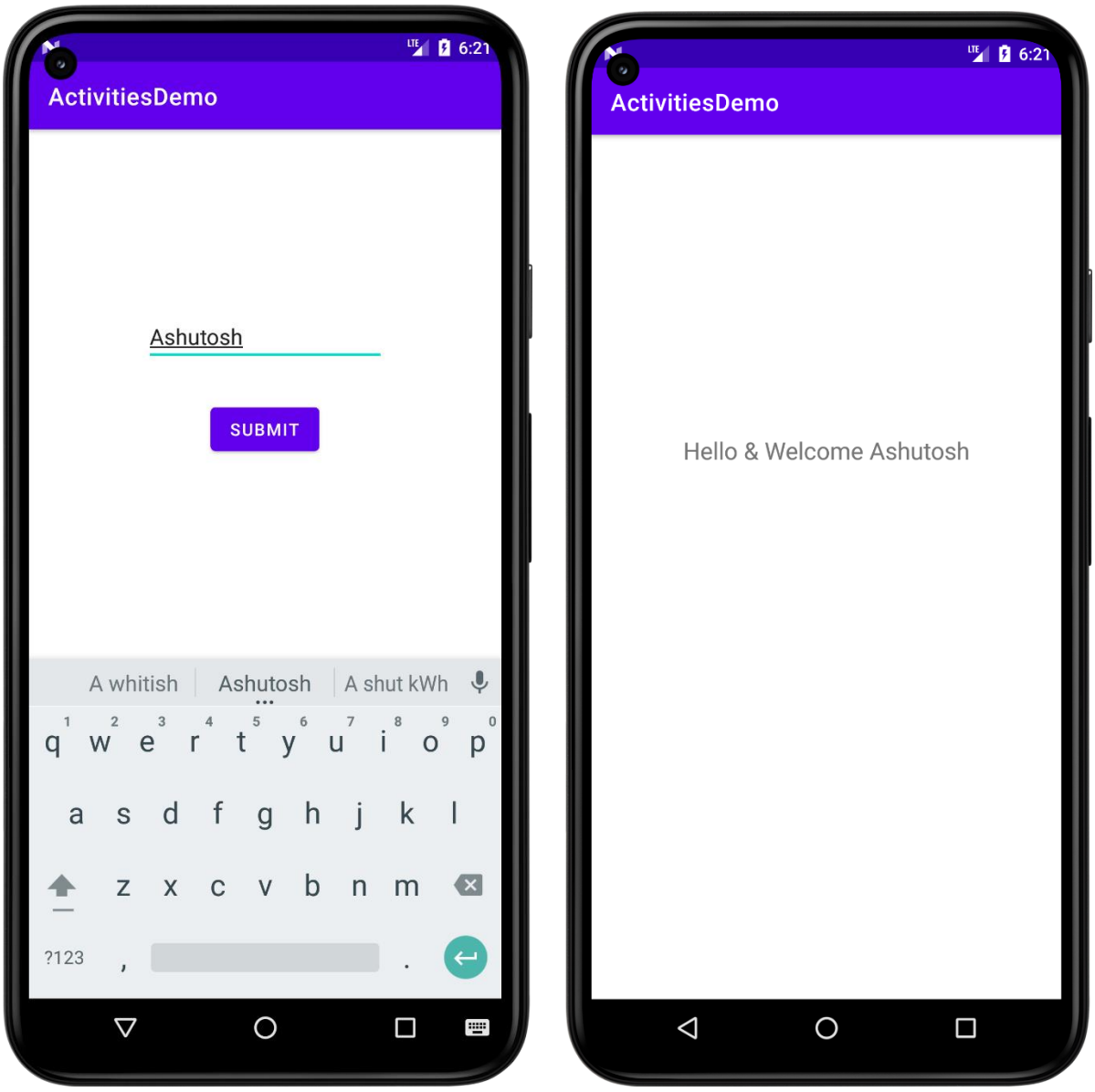
MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editText_main"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:layout_gravity="center"
        android:layout_marginBottom="30dp"
        android:hint="Enter Username" />

    <Button
        android:id="@+id/button_main"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="launchSecondActivity"
        android:layout_gravity="center"
        android:text="Submit" />

</LinearLayout>
```

SecondActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".SecondActivity">

    <TextView
        android:id="@+id/text_message"
        android:layout_width="250dp"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textAlignment="center"
        android:layout_marginTop="250dp"
        android:textSize="20sp"/>

</LinearLayout>
```

Output:


6:21
ActivitiesDemo
Ashutosh
SUBMIT
A whitish
Ashutosh
A shut kWh
6:21
ActivitiesDemo
Hello & Welcome Ashutosh

10. Write an android code to make a phone call using Intent.

Code:

MainActivity.java –

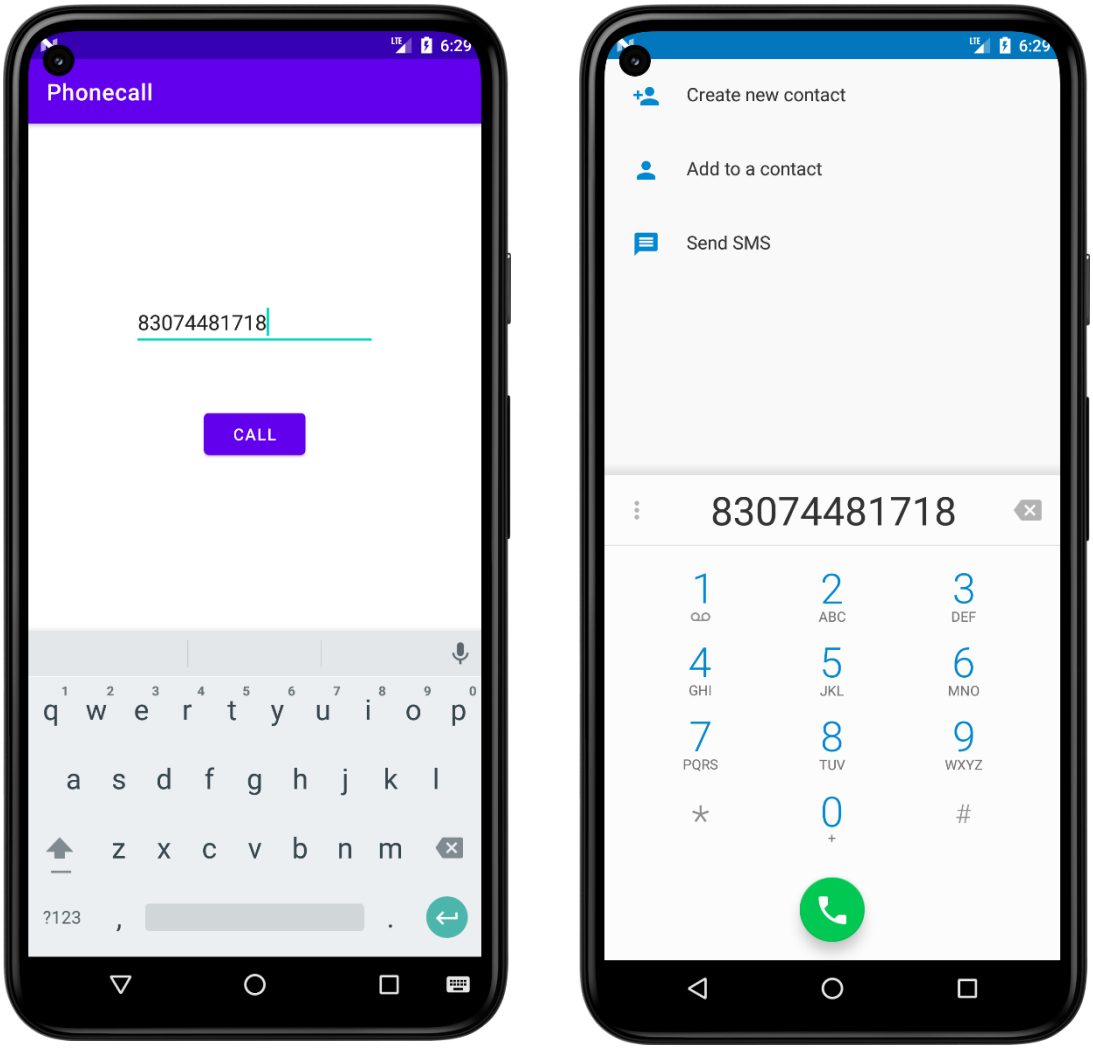
```
package com.example.phonecall;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    EditText edittext1;
    Button callbutton;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //Getting the edittext and button instance
        edittext1 = findViewById(R.id.editText1);
        callbutton = findViewById(R.id.button1);

        //Performing action on button click
        callbutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // getting phone number from edit text and changing it to
String
                String phone_number = edittext1.getText().toString();
                // Getting instance of Intent with action as ACTION_DIAL
                Intent phone_intent = new Intent(Intent.ACTION_DIAL);
                // Set data of Intent through Uri by parsing phone number
                phone_intent.setData(Uri.parse("tel:" + phone_number));
                startActivity(phone_intent);
            }
        });
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <EditText
```

```
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:layout_gravity="center"
        android:ems="10" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="50dp"
        android:text="Call" />

</LinearLayout>
```

Output:



11. Write an android application using SQLite to create table and perform CRUD operations Consider a COURSE table with fields C_ID, C_Name, C_Duration, C_Description, perform ADD, UPDATE, DELETE and READ operations.

Code:

MainActivity.java –

```
package com.example.crudsqlite;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText id, name, ph;
    Button saveBtn, lstBtn, showBtn, updateBtn, deleteBtn;
    Intent intent;
    DatabaseHandler db;
    String uid, uname, uph;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        db = new DatabaseHandler(this);
        id = findViewById(R.id.txtId);
        name = findViewById(R.id.txtName);
        ph = findViewById(R.id.txtPhno);
        saveBtn = findViewById(R.id.btnSave);
        lstBtn = findViewById(R.id.btnList);
        updateBtn = findViewById(R.id.btnUpdate);
        deleteBtn=findViewById(R.id.btnDelete);
        showBtn = findViewById(R.id.btnShowRec);

        //Save record
        saveBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                uid = id.getText().toString();
                uname = name.getText().toString();
                uph = ph.getText().toString();
                try {
                    db.addContact(new Contacts(Integer.parseInt(uid),
uname, uph));
                    Toast.makeText(getApplicationContext(), "Record added",
                            Toast.LENGTH_LONG).show();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }

        });
```

```
        updateBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                uid = id.getText().toString();
                uname = name.getText().toString();
                uph = ph.getText().toString();
                try {
                    db.updateContact(new Contacts(Integer.parseInt(uid),
uname, uph));
                    Toast.makeText(getApplicationContext(), "Record
Updated",
                            Toast.LENGTH_LONG).show();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });

        deleteBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                uid = id.getText().toString();
                try {
                    db.deleteContact(Integer.parseInt(uid));
                    Toast.makeText(getApplicationContext(), "Record
Deleted",
                            Toast.LENGTH_LONG).show();
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });

        // To Display all records from table
        lstBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                intent = new Intent(MainActivity.this,
DetailsActivity.class);
                startActivity(intent);
            }
        });

        // To Display a single record from table
        showBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                uid = id.getText().toString();
                intent = new Intent(MainActivity.this,
DetailsActivity.class);
                intent.putExtra("userid", uid);
                startActivity(intent);
            }
        });
    }
}
```

DetailsActivity.java –

```
package com.example.crudsqlite;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import java.util.ArrayList;
import java.util.List;

public class DetailsActivity extends AppCompatActivity {
    DatabaseHandler db;
    ListView lv;
    int i = 0,uid;
    ArrayList<Contacts> contacts;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_details);

        db = new DatabaseHandler(this);
        lv = findViewById(R.id.user_list);
        Intent intent=getIntent();
        if (intent.hasExtra("userid")) {
            uid = Integer.parseInt(intent.getStringExtra("userid"));
            contacts = db.GetUserByUserId(uid);
        }
        else {
            //code to read all contacts
            contacts = db.getAllContacts();
        }
        List<String> ls = new ArrayList<String>();
        for (Contacts cn : contacts) {
            String s1 = " Id: " + cn.get_id() + "\n Name: " + cn.get_name()
+ "\n Phone: " +
                    cn.get_phone_number();
            ls.add(s1);
        }
        ArrayAdapter<String> adapter = new ArrayAdapter<String>
                (DetailsActivity.this,
                        android.R.layout.simple_expandable_list_item_1,
                        ls);
        lv.setAdapter(adapter);
    }
}
```

DatabaseHandler.java –

```
package com.example.crudsqlite;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import com.example.crudsqlite.Contacts;
```

```
import java.util.ArrayList; import java.util.List;
public class DatabaseHandler extends SQLiteOpenHelper {
    private static final int DATABASE_VERSION = 1;
    private static final String DATABASE_NAME = "contactsManager";
    private static final String TABLE_NAME = "contacts";
    SQLiteDatabase db = null;
    public DatabaseHandler(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        Log.d(DATABASE_NAME, "created");
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_CONTACTS_TABLE =
                "CREATE TABLE IF NOT EXISTS contacts "
                        + "(ID INTEGER PRIMARY KEY, NAME TEXT, PH_NO
TEXT)";

        db.execSQL(CREATE_CONTACTS_TABLE);
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {
        // Drop older table if exists
        db.execSQL("DROP TABLE IF EXISTS contacts");
        // Create tables again
        onCreate(db);
    }
    //add record
    public void addContact(Contacts contacts) {
        try {
            db = this.getWritableDatabase();
            ContentValues values = new ContentValues();
            values.put("ID", contacts.get_id()); //contact id
            values.put("NAME", contacts.get_name()); // Contact Name
            values.put("PH_NO", contacts.get_phone_number()); // Contact
Phone
            // insert(String table, String nullColumnHack, ContentValues
values)
            db.insert("contacts", null, values);
            // Inserting Contacts
            Log.d("Insert: ", "Record Added ..");
            // Closing database connection
            db.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    //update record
    public void updateContact(Contacts contacts) {
        SQLiteDatabase db = this.getWritableDatabase();
        int contact_id=contacts.get_id();
        ContentValues cVals = new ContentValues();
        cVals.put("ID", contacts.get_id());
        cVals.put("NAME", contacts.get_name());
        cVals.put("PH_NO", contacts.get_phone_number());
        int count = db.update("contacts",
                cVals,
                "ID = ?",
                new String[]{String.valueOf(contact_id)});
```

```
    }
    // Delete User Details
    public void deleteContact(int userid) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_NAME, "ID = ?", new
String[]{String.valueOf(userid)});
        db.close();
    }
    // code to get all contacts in a list view
    public ArrayList<Contacts> getAllContacts() {
        SQLiteDatabase db = this.getWritableDatabase();
        ArrayList<Contacts> contactList = new ArrayList<Contacts>();
        // Select All Query
        String selectQuery = "SELECT * FROM contacts";
        // rawQuery(String sql, String[] selectionArgs)
        Cursor cursor = db.rawQuery(selectQuery, null);
        cursor.moveToFirst();
        // looping through all rows and adding to list if
(cursor.moveToFirst()) {
        do {
            Contacts contact = new Contacts();
            contact.set_id(Integer.parseInt(cursor.getString(0)));
            contact.set_name(cursor.getString(1));
            contact.set_phone_number(cursor.getString(2));
            // Adding contact to list
            contactList.add(contact);
        } while (cursor.moveToNext());
        // return contact list
        return contactList;
    }
    // Get User Details based on userid
    public ArrayList<Contacts> GetUserByUserId(int userid) {
        SQLiteDatabase db = this.getWritableDatabase();
        ArrayList<Contacts> contactList = new ArrayList<Contacts>();
        Cursor cursor = db.query
                (TABLE_NAME, new String[]{"id", "name", "ph_no"},
                        "id = ?", new String[]{String.valueOf(userid)},
                        null, null, null, null);
        if (cursor.moveToNext()) {
            Contacts contact = new Contacts();
            contact.set_id(Integer.parseInt(cursor.getString(0)));
            contact.set_name(cursor.getString(1));
            contact.set_phone_number(cursor.getString(2));
            // Adding contact to list
            contactList.add(contact);
        }
        return contactList;
    }
}
```

Contacts.java –

```
package com.example.crudsqlite;

public class Contacts {
    int _id;
    String _name;
    String _phone_number;
    public Contacts() {
```

```
    }
    public Contacts(int _id, String _name, String _phone_number) {
        this._id = _id;
        this._name = _name;
        this._phone_number = _phone_number;
    }
    public int get_id() {
        return _id;
    }
    public void set_id(int _id) {
        this._id = _id;
    }
    public String get_name() {
        return _name;
    }
    public void set_name(String _name) {
        this._name = _name;
    }
    public String get_phone_number() {
        return _phone_number;
    }
    public void set_phone_number(String _phone_number) {
        this._phone_number = _phone_number;
    }
}
```

MainActivity.xml –

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns: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="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/Txt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="100dp"
        android:text="Enter Details:" />

    <TextView
        android:id="@+id/fstTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="50dp"
        android:text="ID" />

    <EditText
        android:id="@+id/txtId"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:ems="10" />
```

```
<TextView
    android:id="@+id/secTxt"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:text="Name" />

<EditText
    android:id="@+id/txtName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:ems="10" />

<TextView
    android:id="@+id/thirdTxt"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:text="Phone Number:" />

<EditText
    android:id="@+id/txtPhno"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="30dp"
    android:ems="10" />

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:text="Save" />

    <Button
        android:id="@+id/btnUpdate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:text="Modify" />

    <Button
        android:id="@+id/btnDelete"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:text="Delete" />

</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
        android:orientation="horizontal">

        <Button
            android:id="@+id/btnShowRec"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="Show Record" />

        <Button
            android:id="@+id/btnList"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="List all" />

    </LinearLayout>

</LinearLayout>
```

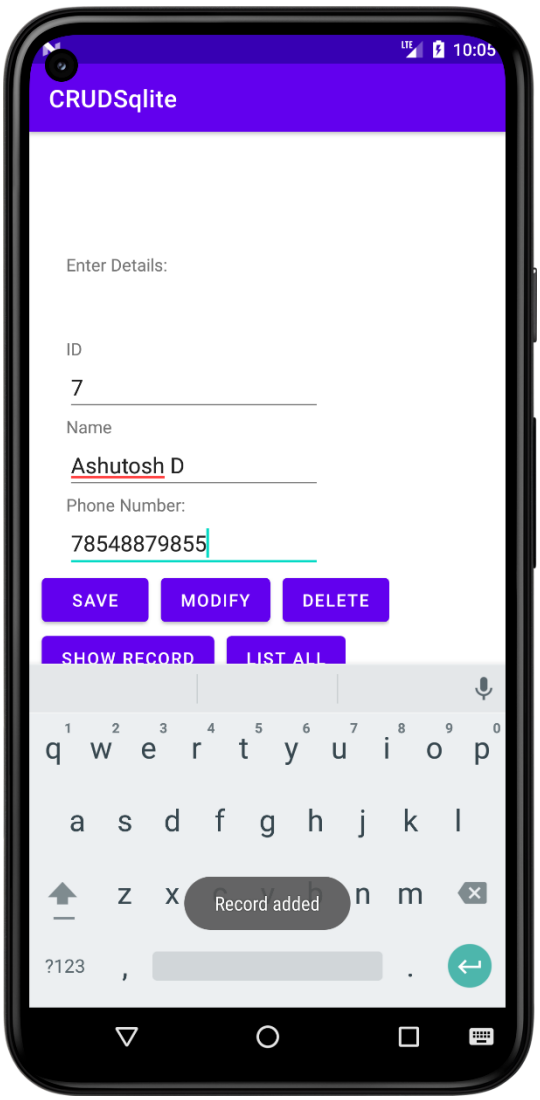
ActivityDetails.xml –

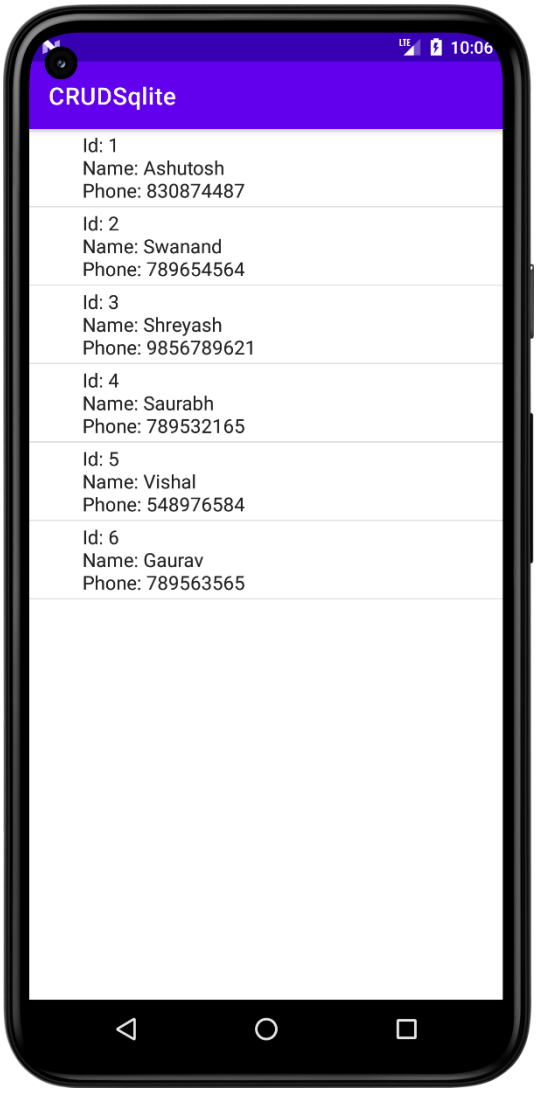
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/Txt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="100dp"
        android:text="Enter Details:" />

    <TextView
        android:id="@+id/fstTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:layout_marginTop="50dp"
        android:text="ID" />

    <EditText
        android:id="@+id/txtId"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:ems="10" />

    <TextView
        android:id="@+id/secTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
```

```
        android:text="Name" />

    <EditText
        android:id="@+id/txtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:ems="10" />

    <TextView
        android:id="@+id/thirdTxt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:text="Phone Number:" />

    <EditText
        android:id="@+id/txtPhno"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="30dp"
        android:ems="10" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btnSave"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="Save" />

        <Button
            android:id="@+id/btnUpdate"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="Modify" />

        <Button
            android:id="@+id/btnDelete"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="Delete" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btnShowRec"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

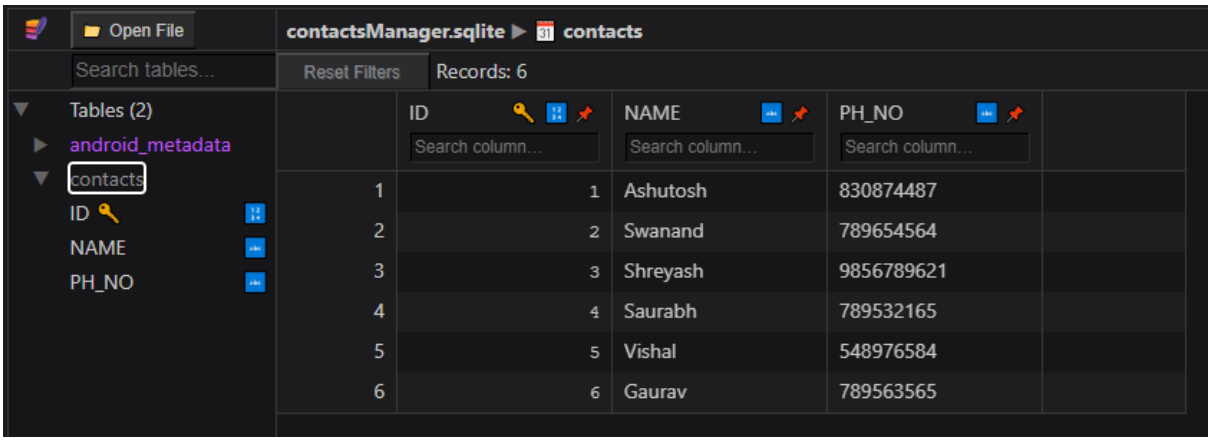
```
            android:layout_marginLeft="10dp"
            android:text="Show Record" />

        <Button
            android:id="@+id/btnList"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="10dp"
            android:text="List all" />

    </LinearLayout>

</LinearLayout>
```

Output:





Database View –



12. Create an Android app to manage STUDENT data, powered by Firebase Realtime database that supports: Adding Data to Firebase Realtime database, Retrieving Data from Firebase, Update a record and Deleting data from firebase data.

Code –

Output –