

## MCAL34 Mobile Computing Lab

### INDEX

Practical No.	Practical List	Date	Sign
1	Android program using various UI components	04/09/23	
2	Android program using different layouts and views	08/09/23	
3	Android program based on Intents	11/09/23	
4	Android program for notifications and alert box	18/09/23	
5	Android program to perform CRUD operation using SQLite DB	26/09/23	
6	Android program using Shared Preferences, Internal and External Storage	26/09/23	
7	Android program to work with graphics and animation	17/10/23	
8	Android program to work with google maps and locations	23/10/23	
9	Android program to work with images and videos	09/11/23	
10	Android program based on RestAPI	20/11/23	
11	Flutter program using layout widgets and state management	25/11/23	

12	Flutter program to work with SQLite Database	20/12/23	
13	Flutter program based on RestAPI	22/12/23	

## 1. Activity Overloading

XML code :

```
<?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="Your Name"
    />
    <EditText
        android:id="@+id/txt1"
        android:layout_width="214dp"
        android:layout_height="wrap_content" />
    <Button
        android:id="@+id/btn1"
        android:layout_width="106dp"
        android:layout_height="wrap_content"
        android:text="OK"/>
    <Button
        android:id="@+id/btn2"
        android:layout_width="106dp"
        android:layout_height="wrap_content"
        android:text="Cancel"/>

</LinearLayout>
```

Java Code

```
package com.tymca.www.actiover;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.widget.Toast;

import org.w3c.dom.Text;
```

```

public class MainActivity extends Activity {

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

    public boolean onKeyDown(int keyCode, KeyEvent event) {
        switch (keyCode) {
            case KeyEvent.KEYCODE_DPAD_CENTER:
                Toast.makeText(getApplicationContext(), "Center Was Clicked",
                    Toast.LENGTH_LONG).show();
                break;
            case KeyEvent.KEYCODE_DPAD_LEFT:
                Toast.makeText(getApplicationContext(), "Left arrow was clicked",
                    Toast.LENGTH_LONG).show();
                break;
            case KeyEvent.KEYCODE_DPAD_RIGHT:
                Toast.makeText(getApplicationContext(), "Right arrow was clicked",
                    Toast.LENGTH_LONG).show();
                break;
            case KeyEvent.KEYCODE_DPAD_UP:
                Toast.makeText(getApplicationContext(), "Up arrow was clicked",
                    Toast.LENGTH_LONG).show();
                break;
            case KeyEvent.KEYCODE_DPAD_DOWN:
                Toast.makeText(getApplicationContext(), "Down arrow was clicked",
                    Toast.LENGTH_LONG).show();
                break;
        }
        return true;
    }
}

```

---

## 2. Activity LifeCycle Demo

- ```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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">

    <TextView
        android:layout_width="wrap_content"

```

```
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

Java File

```
package com.tymca.www.lifecycledemo;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.annotation.SuppressLint;
import android.widget.Toast;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        notify("onCreate");
    }
    protected void onPause()
    {
        super.onPause();
        notify("onPause");
    }
    protected void onResume()
    {
        super.onResume();
        notify("onResume");
    }
    protected void onStop()
    {
        super.onStop();
        notify("onStop");
    }
    protected void onDestroy()
    {
        super.onDestroy();
        notify("onDestroy");
    }
    protected void onRestoreInstanceState(Bundle savedInstanceState)
    {
        super.onRestoreInstanceState(savedInstanceState);
        notify("onRestoreInstanceSate");
    }
}
```

```

}
protected void onSaveInstanceState(Bundle outState)
{
    super.onSaveInstanceState(outState);
    notify("onSaveInstanceState");
}
private void notify(String methodName)
{
    String name = this.getClass().getName();
    String [] strings = name.split("\\.");
    Toast.makeText(getApplicationContext(),methodName+"
"+strings[strings.length - 1],Toast.LENGTH_LONG).show();
}
}

```

---

#### 4. Hello World Program

XML code

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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">

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button"
    tools:layout_editor_absoluteX="16dp"
    tools:layout_editor_absoluteY="266dp" />

</android.support.constraint.ConstraintLayout>

```

Java Code

```

package com.example.vikram.myapplication;

import android.support.v7.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);
    }
}

```

---

## 5. Calculator

### XML Code

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <RelativeLayout android:layout_width="match_parent"
    android:id="@+id/relativeLayout1"
    android:layout_height="match_parent"
    android:background="@color/bgcolor">
        <TextView android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:layout_marginTop="45dp" android:text="Enter User Name
        :"></TextView>
        <EditText android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="textPersonName"
        android:layout_below="@+id/textView1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:id="@+id/txtusername" android:hint="Enter UserName">
            <requestFocus></requestFocus>
        </EditText>
        <TextView android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

android:textAppearance="?android:attr/textAppearanceLarge"
android:layout_below="@+id/txtusername"
android:layout_alignParentLeft="true"
android:layout_marginTop="20dp" android:text="Enter Password
:"></TextView>
    <EditText android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:inputType="textPassword"
android:layout_below="@+id/textView2"
android:layout_alignParentLeft="true"
android:layout_alignParentRight="true"
android:id="@+id/txtpassword" android:hint="Enter
Password"></EditText>
    <Button android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/btnlogin"
android:layout_toRightOf="@+id/textView1" android:text="Clear"
android:id="@+id/btnclear"></Button>
    <Button android:id="@+id/btnlogin"
android:layout_width="wrap_content"
android:layout_height="wrap_content" android:text="Login"
android:layout_below="@+id/txtpassword"
android:layout_alignRight="@+id/textView2"
android:layout_marginRight="33dp"
android:layout_marginTop="21dp"></Button>
</RelativeLayout>
</LinearLayout>

```

Java Code

```

package com.tymca.www.calculator;

```

```

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

```

```

public class MainActivity extends Activity {

```

```

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    final EditText amt = (EditText) findViewById(R.id.bill_amt);
    final EditText tip = (EditText) findViewById(R.id.bill_per);
    final TextView result = (TextView) findViewById(R.id.res);

```

```

    Button calc = (Button) findViewById(R.id.button1);
    calc.setOnClickListener(new View.OnClickListener() {

```



```

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        double amount = Double.parseDouble(amt.toString());
        double tip_per = Double.parseDouble(tip.toString());
        double tip_cal = (amount * tip_per) / 100;
        result.setText("Result : " + Double.toString(tip_cal));
    }
});
}

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

```

---

## 6. Temperature Converter

### XML Code

```

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

    <EditText
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number|numberDecimal"/>

    <RadioGroup
        android:id="@+id/radioGroup1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/editText1">

        <RadioButton
            android:id="@+id/radio0"
            android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:checked="true"
        android:text="celcius"/>
    <RadioButton
        android:id="@+id/radio1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="farenhiet"/>
</RadioGroup>
<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/radioGroup1"
    android:onClick="onClick"
    android:text="cal"/>
</RelativeLayout>

```

Java Code

```

package com.tymca.www.temperature;

```

```

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.Toast;

```

```

public class MainActivity extends AppCompatActivity {

```

```

    EditText text;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        text = (EditText)findViewById(R.id.editText1);
    }
    public float onClick(View view)
    {
        switch(view.getId()) {
            case R.id.button1:
                RadioButton cel = (RadioButton) findViewById(R.id.radio0);
                RadioButton far = (RadioButton) findViewById(R.id.radio1);
                if (text.getText().length() == 0) {
                    Toast.makeText(this, "Enter Valid Number",
Toast.LENGTH_LONG).show();
                }
                return;
            }
        }
    }

```

```

        float input = Float.parseFloat(text.getText().toString());
        if (cel.isChecked()) {
            text.setText(String.valueOf(convertFarToCel(input)));
            cel.setChecked(false);
            far.setChecked(true);
        } else {
            text.setText(String.valueOf(convertCelToFar(input)));
            cel.setChecked(true);
            far.setChecked(false);
        }
        break;
    }
    float convertFarToCel(float fahrenheit)
    {
        return((fahrenheit-32)*5/9);
    }
    float convertCelToFar(float celsius)
    {
        return ((celsius*9/5)+32);
    }
}
}

```

---

## 7. TextDemo

XML code

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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="com.tymca.www.textdemo.MainActivity">

<TextView
    android:id="@+id/text"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    android:textcolor="#ffffff"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true"
    android:textSize="20sp"
    android:textStyle="bold" />

```

</RelativeLayout>

Java Code

```
package com.tymca.www.textdemo;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final TextView txtView = (TextView)findViewById(R.id.text);
        txtView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                txtView.setText("You Clicked the textView");
            }
        });
    }
}
```

---

## 8. Input Text Demo

XML Code

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
android:layout_centerHorizontal="true"
android:hint="Enter your Name"
android:layout_marginTop="150dp"
android:inputType="text"/>
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="75dp"
    android:layout_centerVertical="true"
    android:layout_centerHorizontal="true"
    android:textStyle="bold"
    android:textColor="#58ff55" />
```

```
</RelativeLayout>
```

Java Code

```
package com.tymca.www.inputtext;
```

```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.textEditable;
import android.text.TextWatcher;
import android.widget.EditText;
import android.widget.TextView;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    EditText editText;
```

```
    TextView textView;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        editText = (EditText)findViewById(R.id.editText);
```

```
        textView = (TextView)findViewById(R.id.textView);
```

```
        editText.addTextChangedListener(new TextWatcher() {
```

```
            @Override
```

```
            public void beforeTextChanged(CharSequence charSequence, int i, int i1,
int i2) {
```

```
            }
```

```
            @Override
```

```
            public void onTextChanged(CharSequence charSequence, int i, int i1, int
i2) {
```

```

    }

    @Override
    public void afterTextChanged(Editable editable) {
        String name = editText.getText().toString();
        textView.setText("Hi TYMCA Student " + name);
    }
});
}
}

```

---

## 9. Text Autocorrect Demo

XML code

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
    tools:context=".MainActivity">

    <AutoCompleteTextView
        android:id="@+id/acTextView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:completionThreshold="1"/>

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:textColor="#f6f"
    />

</RelativeLayout>

```

Java Code

```

package com.tymca.www.autocorrect;

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

```

```

import android.widget.AutoCompleteTextView;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    AutoCompleteTextView acTextView;
    TextView textView;
    String [] inputs =
{"India","Indonesia","Isreal","America","Austrilia","Canada","China","
Denmark","France"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        acTextView = (AutoCompleteTextView)findViewById(R.id.acTextView);
        textView = (TextView)findViewById(R.id.textView);
        textView.setText("Text Inputs:\n
India,Indonesia,Isreal,America,Austrilia,Canada,China,Denmark,Fran
ce");
        ArrayAdapter<String> adapter = new
        ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,inputs);
        acTextView.setAdapter(adapter);

    }
}

```

---

## 10. Button Demo

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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:padding="16dp"
tools:context=".MainActivity">
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        android:text="Click Me"/>

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"

```

```

        android:layout_marginBottom="100dp"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        android:layout_above="@+id/button"
        android:textStyle="bold"
        android:textColor="#f6f"
    />

```

</RelativeLayout>

Java Code

```
package com.tymca.www.butdemo;
```

```

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

```

```

public class MainActivity extends AppCompatActivity {
    Button button;
    TextView textView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        button = (Button)findViewById(R.id.button);
        textView = (TextView)findViewById(R.id.textView);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                textView.setText("You click the Button");
            }
        });
    }
}

```

---

#### 11. Radio Button Demo to change background color

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
android:id="@+id/relativeLayout"
tools:context=".MainActivity">
```

```
<RadioGroup
    android:id="@+id/radioGroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true">
```

```
<RadioButton
    android:id="@+id/radiobutton1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="RED"/>
```

```
<RadioButton
    android:id="@+id/radiobutton2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Green"/>
```

```
<RadioButton
    android:id="@+id/radiobutton3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="BLUE"/>
```

```
<RadioButton
    android:id="@+id/radiobutton4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Yellow"/>
```

```
</RadioGroup>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

```
</RelativeLayout>
```

Java Code

```

package com.tymca.www.radiog;

import android.graphics.Color;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.RadioGroup;
import android.widget.RadioButton;
import android.widget.RelativeLayout;

public class MainActivity extends AppCompatActivity {
    RadioGroup radioGroup;
    RelativeLayout relativeLayout;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        radioGroup = (RadioGroup)findViewById(R.id.radioGroup);
        relativeLayout = (RelativeLayout)findViewById(R.id.relativeLayout);
        radioGroup.setOnCheckedChangeListener(new
RadioGroup.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(RadioGroup radioGroup, int i) {
        switch (i)
        {
            case R.id.radiobutton1:

relativeLayout.setBackgroundColor(Color.parseColor("#ff0000"));
                break;
            case R.id.radiobutton2:

relativeLayout.setBackgroundColor(Color.parseColor("#00ff00"));
                break;
            case R.id.radiobutton3:

relativeLayout.setBackgroundColor(Color.parseColor("#0000ff"));
                case R.id.radiobutton4:
                    relativeLayout.setBackgroundColor(Color.parseColor("#00ffff"));
                }
            }
        });
    }
}

```

---

## 12. Switch Case Demo

XML code

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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:padding="16dp"
    tools:context=".MainActivity">

    <Switch
        android:id="@+id/switchButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        android:onClick="onSwitchClick"/>
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="100dp"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:layout_above="@+id/switchButton"
        android:textStyle="bold"
        android:textColor="#ff0000"
    />

</RelativeLayout>

```

Java Code

```

package com.tymca.www.switdemo;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Switch;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    Switch switchButton;
    TextView textView;

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

```

```

switchButton = (Switch)findViewById(R.id.switchButton);
textView = (TextView)findViewById(R.id.textView);

}
public void onSwitchClick(View view)
{
    if(switchButton.isChecked())
    {
        textView.setText("Switch is ON");
    }
    else {
        textView.setText("Switch is OFF");
    }
}
}

```

---

### 13. Rating Bar Demo

#### XML Code

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
    tools:context=".MainActivity">

    <RatingBar
        android:id="@+id/ratingBar"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:stepSize="0.2"
        />

    <TextView
        android:layout_above="@+id/textView1"
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="20dp"
        android:textStyle="bold"
        android:textColor="#ff0000"

```

```

/>
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="80dp"
    android:textStyle="bold"
    android:text="Please Give us rating"
    android:textColor="#000000"
/>
<Button
    android:id="@+id/submitButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/ratingBar"
    android:layout_marginTop="20dp"
    android:layout_centerHorizontal="true"
    android:text="Submit"
    android:onClick="onSubmit"/>

</RelativeLayout>

```

Java Code

```

package com.tymca.www.ratingdemo;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.RatingBar;
import android.widget.Button;
import android.widget.TextView;

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

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ratingBar = (RatingBar)findViewById(R.id.ratingBar);
        textView = (TextView)findViewById(R.id.textView);
        button = (Button)findViewById(R.id.submitButton);
    }

    public void onSubmit(View view)

```

```

{
    float ratingValue = ratingBar.getRating();
    if(ratingValue<2)
    {
        textView.setText("Rating"+ratingValue+"\n is worst");
    }
    else if(ratingValue<=3 && ratingValue>=2)
    {
        textView.setText("Rating"+ratingValue+" we will try better");
    }
    else if(ratingValue>3 && ratingValue<=4)
    {
        textView.setText("Rating"+ratingValue+"\n It is good");
    }
    else if(ratingValue>4)
    {
        textView.setText("Rating"+ratingValue+"\n Excellent");
    }
}
}

```

---

#### 14. Simple Calculator Demo

XML code

```

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

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentTop="true"
        android:text="Simple Calculator"
        android:textColor="@color/colorAccent"
        android:layout_alignParentLeft="true" />

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentStart="true"

```

```

        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true">

</RelativeLayout>
</RelativeLayout>

Java Code

package com.tymca.www.simple_calculator;

import android.support.v7.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);
    }
}

```

---

## 15. Interest Calculator

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_marginTop="50dp"
        android:layout_alignParentRight="true"
        android:text="Enter the Bill Amount"
        android:textStyle="bold"
        />
    <TextView
        android:id="@+id/res"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"

```

```

        android:layout_marginBottom="14dp"
        android:layout_alignParentRight="true"
        android:layout_alignRight="@+id/button1"
        android:text="Result:"
        android:textStyle="bold"
    />
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@id/res"
        android:layout_alignParentLeft="true"
        android:layout_alignRight="@+id/textView2"
        android:text="Calculate"/>
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/button1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:layout_marginBottom="96dp"
        android:text="Enter Percentage"/>
    <EditText
        android:id="@+id/bill_amt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:layout_below="@+id/textView1"
        android:layout_marginTop="41dp"/>
    <EditText
        android:id="@+id/bill_per"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/button1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:layout_marginBottom="22dp"/>

</RelativeLayout>

```

Java Code

```
package com.tymca.www.inetrestcal;
```

```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
```



```

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

public class MainActivity extends AppCompatActivity
{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final EditText amt = (EditText)findViewById(R.id.bill_amt);
        final EditText per = (EditText)findViewById(R.id.bill_per);
        final TextView result = (TextView)findViewById(R.id.res);
        Button cal = (Button)findViewById(R.id.button1);
        cal.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                double amount = Double.parseDouble(amt.toString());
                double percentage = Double.parseDouble(per.toString());
                double res1 = amount*percentage;
                result.setText("Result"+Double.toString(res1));
            }
        });
    }
}

```

---

## 16. Seekbar Demo

### XML Code

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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:padding="16dp"
    tools:context=".MainActivity">

    <SeekBar
        android:id="@+id/seekBar"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"/>

    <TextView
        android:id="@+id/textView"

```

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="100dp"
android:layout_centerVertical="true"
android:layout_centerHorizontal="true"
android:layout_above="@+id/seekBar"
android:textStyle="bold"
android:textColor="#40caff"
/>
```

</RelativeLayout>

Java Code

```
package com.tymca.www.seekbar;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.SeekBar;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    SeekBar seekBar;
    TextView textView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        seekBar=(SeekBar)findViewById(R.id.seekBar);
        textView = (TextView)findViewById(R.id.textView);
        seekBar.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int i, boolean b) {
                textView.setTextSize(i);
                textView.setText(""+(i++));
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {

            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {

            }
        });
    }
}
```

```
}  
}
```

---

## 17. Progress Bar Demo

### XML Code

```
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
    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:padding="16dp"  
    tools:context=".MainActivity">  
    <ProgressBar  
        android:id="@+id/progressBar"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
        android:layout_centerHorizontal="true"  
        android:layout_marginTop="10dp"/>  
    <Button  
        android:id="@+id/button1"  
        android:layout_below="@+id/progressBar"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:text="Click Me"  
        android:layout_marginTop="40dp"  
        android:onClick="progressBarClick"  
        android:layout_centerHorizontal="true"/>  
    <ProgressBar  
        android:id="@+id/progressBarHorizontal"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_centerHorizontal="true"  
        android:layout_below="@+id/button1"  
        android:layout_marginTop="50dp"  
        android:max="100"  
        android:progress="0"  
        android:scrollbarStyle="insideInset"/>  
    <Button  
        android:id="@+id/button2"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_centerHorizontal="true"  
        android:layout_below="@+id/progressBarHorizontal"  
        android:text="Click Me"  
        android:layout_marginTop="100dp"
```

```

        android:onClick="progressBarHorizontal"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</RelativeLayout>

```

Java Code

```

package com.tymca.www.progressBar;

import android.support.v7.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);
    }
}

```

## 18. GridView Demo

XML Code

```

<?xml version="1.0" encoding="utf-8"?>
<GridView
    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:id="@+id/simpleGrid"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnWidth="100dp"
    android:verticalSpacing="10dp"
    android:horizontalSpacing="10dp"
    android:numColumns="auto_fit"
    android:stretchMode="columnWidth"

```

```
android:gravity="center"
tools:context=".MainActivity">
```

</GridView>

Java Code

```
package com.tymca.www.gridview;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Toast;
public class MainActivity extends Activity {
    GridView grid;
    String
    items[]={"Apple","Banana","Orange","Mango","Papaya","Watermelon",
    "Grapes","Pineapple"};
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        grid = (GridView)findViewById(R.id.simpleGrid);
        ArrayAdapter adapter = new
        ArrayAdapter(this,android.R.layout.simple_list_item_1,items);
        grid.setAdapter(adapter);
        grid.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View view, int i,
            long l) {

                Toast.makeText(getApplicationContext(),items[i],Toast.LENGTH_LONG).show
                ();
            }
        });
    }
}
```

---

19. DatePicker Demo

XML Code

```

<?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:paddingBottom="15dp"
android:paddingLeft="15dp"
android:paddingTop="15dp"
android:orientation="vertical"
android:gravity="center"
tools:context=".MainActivity">

    <Button
        android:id="@+id/setDateBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Set Date"
        android:layout_marginBottom="10dp"/>
    <TextView
        android:id="@+id/selectedDateTxt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

</LinearLayout>

```

### Java Code

```

package com.tymca.www.datepick;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import java.util.Calendar;
import android.widget.TextView;
import android.widget.Button;
import android.widget.DatePicker;
import android.view.View;
import android.app.Dialog;
import android.app.DatePickerDialog;

public class MainActivity extends AppCompatActivity {
    Button setDateBtn;
    TextView selectedDateTxt;
    int day,month,year;

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    setDateBtn = (Button)findViewById(R.id.setDateBtn);
    selectedDateTxt = (TextView)findViewById(R.id.selectedDateTxt);
    Calendar c = Calendar.getInstance();
    day = c.get(Calendar.DAY_OF_MONTH);
    month = c.get(Calendar.MONTH);
    year = c.get(Calendar.YEAR);
    displayDate(day, month, year);
    setDateBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            showDialog(111);
        }
    });
}

void displayDate(int day, int month, int year)
{
    selectedDateTxt.setText("Date " + day + "/" + month + "/" + year);
}

protected Dialog onCreateDialog(int id)
{
    if(id==111)
    {
        return new
DatePickerDialog(this, dateLPickerListener, year, month, day);

    }
    return null;
}

private DatePickerDialog.OnDateSetListener dateLPickerListener = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker datePicker, int i, int i1, int i2) {
        displayDate(day, month+1, year);
    }
};
}

```

---

## 20. WebURL Demo

XML Code

```

<?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">
    <EditText
        android:id="@+id/url"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/enter_the_url_to_open"/>
    <Button
        android:id="@+id/openBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/Open"/>
    <WebView
        android:id="@+id/webView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"></WebView>

</LinearLayout>

```

Manifest XML File

```

<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
package="com.tymca.www.weburl">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>
    </application>

```



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

Java Code

```
package com.tymca.www.weburl;
```

```
import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.webkit.WebView;
import android.webkit.WebViewClient;
```

```
public class MainActivity extends Activity {
    Button openBtn;
    EditText url;
    WebView webView;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        openBtn = (Button)findViewById(R.id.openBtn);
        url = (EditText)findViewById(R.id.url);
        webView = (WebView)findViewById(R.id.webView);
        webView.getSettings().setJavaScriptEnabled(true);
        openBtn.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
```

```
            public void onClick(View view) {
                webView.setWebViewClient(new CustomWebClient());
                webView.loadUrl(url.getText().toString());
            }
        });
    }
```

```
public class CustomWebClient extends WebViewClient
{
```

```
    public boolean shouldOverrideUrlLoading(WebView view,String url)
    {
        view.loadUrl(url);
        return true;
    }
```

```

}
}
```

---

## 21. Popup Demo

### XML Code

```
<?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"
android:padding="10dp"
android:gravity="center"
android:background="#34e710"
android:id="@+id/linearLayout1"
tools:context=".MainActivity">
<TextView
    android:id="@+id/txt"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/this_is_popup_window"
    android:textColor="#efebef"/>
<Button
    android:id="@+id/showPopupBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/show"/>

</LinearLayout>
```

### Java Code

```
package com.tymca.www.popdemo;

import android.content.Context;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Gravity;
import android.view.LayoutInflater;
import android.widget.Button;
import android.widget.TextView;
import android.widget.LinearLayout;
import android.view.View;
import android.view.ViewGroup.LayoutParams;
import android.widget.PopupWindow;
```

```

public class MainActivity extends AppCompatActivity {
    Button showPopupBtn;
    Button closePopupBtn;
    PopupWindow popupWindow;
    LinearLayout linearLayout;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        showPopupBtn = (Button)findViewById(R.id.showPopupBtn);
        closePopupBtn = (Button)findViewById(R.id.closePopupBtn);
        linearLayout = (LinearLayout)findViewById(R.id.linearLayout1);
        showPopupBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                LayoutInflater inflater =
                (LayoutInflater)MainActivity.this.getSystemService(Context.LAYOUT_INFLATE
R_SERVICE);
                View customView = inflater.inflate(R.layout.activity_main,null);
                popupWindow = new
                PopupWindow(customView,LayoutParams.WRAP_CONTENT,LayoutParams.WR
AP_CONTENT);
                popupWindow.showAtLocation(linearLayout, Gravity.CENTER,0,0);
                closePopupBtn.setOnClickListener(new View.OnClickListener() {
                    @Override
                    public void onClick(View view) {
                        popupWindow.dismiss();
                    }
                });
            }
        });
    }
}

```

---

## 22. Check Internet Connection

### XML Code

```

<?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">

    <Button

```

```

    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Check Internet Connectivity"
    android:onClick="buttonAction"/>

```

</LinearLayout>

Manifest XML File

```

<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tymca.www.internetconnect">
    <uses-permission
        android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
            />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

Java File

```

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.Button;
import android.widget.Toast;
import android.view.View;
import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;

public class MainActivity extends Activity {
    Button button;

    @Override

```

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    button = (Button)findViewById(R.id.button);
}
public void buttonAction(View view)
{
    ConnectivityManager cm =
    (ConnectivityManager)getApplicationContext().getSystemService(Context.CONNE
CTIVITY_SERVICE);
    NetworkInfo networkInfo[] = cm.getAllNetworkInfo();
    int i;
    for(i=0;i<networkInfo.length;++i)
    {
        if(networkInfo[i].getState()==NetworkInfo.State.CONNECTED)
        {
            Toast.makeText(getApplicationContext(),"Internet
Connected",Toast.LENGTH_LONG).show();
            break;
        }
    }
    if (i==networkInfo.length)
    {
        Toast.makeText(getApplicationContext(),"Internet Not
Connected",Toast.LENGTH_LONG).show();
    }
}
}

```

---

## 23. Database Connection Demo

### XML Code

```

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

    <EditText
        android:id="@+id/id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter id to update or delete"
        android:onClick="buttonAction"/>

```

```

<EditText
    android:id="@+id/name"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/id"
    android:layout_marginTop="10dp"
    android:hint="Enter Name to update and delete"/>
<LinearLayout
    android:id="@+id/layout1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/name"
    android:orientation="vertical">
    <Button
        android:id="@+id/insert"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Insert"
        android:onClick="buttonAction"/>
    <Button
        android:id="@+id/view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Retrive"
        android:onClick="buttonAction"
    />
    <Button
        android:id="@+id/update"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Update"
        android:onClick="buttonAction"/>
    <Button
        android:id="@+id/delete"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Delete"
        android:onClick="buttonAction"/>

</LinearLayout>
<TextView
    android:id="@+id/text"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"/>
</RelativeLayout>

```

MainActivity.java

```
package com.tymca.www.dbconnect;
```

```
import android.app.Activity;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;
```

```
public class MainActivity extends Activity {  
    EditText id,name;  
    Button insert,view,update,delete;  
    TextView textView;  
    DBHandler db;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        id = (EditText)findViewById(R.id.id);  
        name = (EditText)findViewById(R.id.name);  
        insert = (Button)findViewById(R.id.insert);  
        view = (Button)findViewById(R.id.view);  
        update = (Button)findViewById(R.id.update);  
        delete = (Button)findViewById(R.id.delete);  
        textView = (TextView)findViewById(R.id.text);  
        db = new DBHandler(getApplicationContext());  
    }  
    public void buttonAction(View view)  
    {  
        switch (view.getId())  
        {  
            case R.id.insert:  
                db.insertRecord(name.getText().toString());  
                Toast.makeText(getApplicationContext(), "record  
inserted", Toast.LENGTH_LONG).show();  
                break;  
            case R.id.view:  
                textView.setText(db.getRecords());  
                break;  
            case R.id.update:  
                db.updateRecord(id.getText().toString(), name.getText().toString());  
                Toast.makeText(getApplicationContext(), "record  
update", Toast.LENGTH_LONG).show();  
                break;  
            case R.id.delete:  
                db.deleteRecord(id.getText().toString());  
                Toast.makeText(getApplicationContext(), "record  
deleted", Toast.LENGTH_LONG).show();
```

```

        break;

    }

}
}

```

DBHandler.java

```

package com.tymca.www.dbconnect;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DBHandler extends SQLiteOpenHelper
{
    private static final String DB_NAME = "vikramdb";
    private static final int DB_VERSION = 1;
    private static final String TABLE_NAME = "record";
    private static final String ID_COL = "id";
    private static final String NAME_COL = "name";
    public DBHandler(Context context)
    {
        super(context, DB_NAME, null, DB_VERSION);
    }
    public void onCreate(SQLiteDatabase db)
    {
        String query = "Create TABLE
"+TABLE_NAME+"("+ID_COL+"INTEGER PRIMARY KEY
AUTOINCREMENT,"+NAME_COL+"TEXT)";
        db.execSQL(query);
    }
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion)
    {
        db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME);
        onCreate(db);
    }
    public void insertRecord(String name)
    {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(NAME_COL, name);
        db.insert(TABLE_NAME, null, values);
        db.close();
    }
}

```



```

}
public String getRecords()
{
    String query = "SELECT * FROM "+TABLE_NAME;
    String results="";
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor cursor = db.rawQuery(query,null);
    cursor.moveToFirst();
    while (cursor.isAfterLast()==false)
    {
        results+=cursor.getString(0)+" "+cursor.getString(1)+"\n";
        cursor.moveToNext();
    }
    db.close();
    return results;
}

public void updateRecord(String id,String name)
{
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(NAME_COL,name);
    db.update(TABLE_NAME,values,"id=?",new String[]{id});
    db.close();
}

public void deleteRecord(String id)
{
    SQLiteDatabase db = this.getWritableDatabase();
    db.delete(TABLE_NAME,"id=?",new String[]{id});
    db.close();
}
}

```

---

## 24. SharedPreference Demo

### XML Code

```

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

```

```

<Button
    android:id="@+id/btnSave"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:onClick="Save"
    android:text="Save"/>
<Button
    android:id="@+id/btnRetr"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:onClick="Get"
    android:text="Retrive"/>
<Button
    android:id="@+id/btnClear"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignRight="@+id/etEmail"
    android:layout_centerVertical="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:onClick="clear"
    android:text="Clear"/>
<EditText
    android:id="@+id/etEmail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:layout_below="@+id/etName"
    android:layout_marginTop="20dp"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"/>
<EditText
    android:id="@+id/etName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Name"
    android:inputType="text"
    android:layout_alignParentTop="true"
    android:layout_alignLeft="@+id/etEmail"
    android:layout_alignStart="@+id/etEmail"/>

```

```

</RelativeLayout>

```

## MainActivity.java

```
package com.tymca.www.shpref;

import android.app.Activity;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.content.Context;
import android.content.SharedPreferences;
import android.view.View;
import android.view.Menu;
import android.widget.TextView;

public class MainActivity extends Activity {
    SharedPreferences sharedPreferences;
    TextView name;
    TextView email;
    public static final String mypreference = "mypref";
    public static final String Name = "nameKey";
    public static final String Email = "emailKey";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = (TextView) findViewById(R.id.etName);
        email = (TextView) findViewById(R.id.etEmail);
        sharedPreferences = getSharedPreferences(mypreference,
            Context.MODE_PRIVATE);
        if (sharedPreferences.contains(Name)) {
            name.setText(sharedPreferences.getString(Name, ""));
        }
        if (sharedPreferences.contains(Email)) {
            email.setText(sharedPreferences.getString(Email, ""));
        }
    }

    public void Save(View view) {
        String n = name.getText().toString();
        String e = email.getText().toString();
        SharedPreferences.Editor editor = sharedPreferences.edit();
        editor.putString(Name, n);
        editor.putString(Email, e);
        editor.commit();
    }

    public void clear(View view) {
        name = (TextView) findViewById(R.id.etName);
```

```

        email = (TextView) findViewById(R.id.etEmail);
        name.setText("");
        email.setText("");
    }

    public void Get(View view) {
        name = (TextView) findViewById(R.id.etName);
        email = (TextView) findViewById(R.id.etEmail);
        sharedPreferences = getSharedPreferences(myreference,
            Context.MODE_PRIVATE);

        if (sharedPreferences.contains(Name)) {
            name.setText(sharedPreferences.getString(Name, ""));
        }
        if (sharedPreferences.contains(Email)) {
            email.setText(sharedPreferences.getString(Email, ""));
        }
    }

    @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;
    }
}

```

---

## 25. File Demo for Internal SD card

activity\_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:layout_gravity="center"
        tools:context=".MainActivity" >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="center"

```

```

        android:textAlignment="center"
        android:text="Android Read/Write File" />
    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/fname"
        android:hint="File Name" />
    <EditText
        android:layout_width="fill_parent"
        android:layout_height="100px"
        android:id="@+id/ftext"
        android:hint="File Text" />
    <Button
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btnwrite"
        android:text="Write File" />
    <EditText
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/fnameread"
        android:hint="File Name" />
    <Button
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btnread"
        android:text="Read File" />
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/filecon" />

</LinearLayout>

```

#### FileOperations.java

```

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

import android.util.Log;

public class FileOperations {
    public FileOperations() {

    }
}

```

```

public Boolean write(String fname, String fcontent){
    try {

        String fpath = "/sdcard/"+fname+".txt";

        File file = new File(fpath);

        // If file does not exists, then create it
        if (!file.exists()) {
            file.createNewFile();
        }

        FileWriter fw = new
FileWriter(file.getAbsolutePath());
        BufferedWriter bw = new BufferedWriter(fw);
        bw.write(fcontent);
        bw.close();

        Log.d("Suceess", "Sucess");
        return true;

    } catch (IOException e) {
        e.printStackTrace();
        return false;
    }

}

public String read(String fname){

    BufferedReader br = null;
    String response = null;

    try {

        StringBuffer output = new StringBuffer();
        String fpath = "/sdcard/"+fname+".txt";

        br = new BufferedReader(new FileReader(fpath));
        String line = "";
        while ((line = br.readLine()) != null) {
            output.append(line + "n");
        }
        response = output.toString();

    } catch (IOException e) {
        e.printStackTrace();
        return null;
    }

    return response;
}

```

```
}  
}
```

## MainActivity.java

```
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
import android.app.Activity;  
  
import learn2crack.androidfile.FileOperations;  
  
public class MainActivity extends Activity {  
    EditText fname,fcontent,fnameread;  
    Button write,read;  
    TextView filecon;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        fname = (EditText)findViewById(R.id.fname);  
        fcontent = (EditText)findViewById(R.id.fcontent);  
        fnameread = (EditText)findViewById(R.id.fnameread);  
        write = (Button)findViewById(R.id.btnwrite);  
        read = (Button)findViewById(R.id.btnread);  
        filecon = (TextView)findViewById(R.id.filecon);  
        write.setOnClickListener(new View.OnClickListener() {  
  
            @Override  
            public void onClick(View arg0) {  
                // TODO Auto-generated method stub  
                String filename = fname.getText().toString();  
                String filecontent = fcontent.getText().toString();  
                FileOperations fop = new FileOperations();  
                fop.write(filename, filecontent);  
                if(fop.write(filename, filecontent)){  
                    Toast.makeText(getApplicationContext(), filename+".txt created",  
Toast.LENGTH_SHORT).show();  
                }else{  
                    Toast.makeText(getApplicationContext(), "I/O error",  
Toast.LENGTH_SHORT).show();  
                }  
            }  
        });  
    }  
}
```

```

read.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        String readfilename = fname.read.getText().toString();
        FileOperations fop = new FileOperations();
        String text = fop.read(readfilename);
        if(text != null){
            filecon.setText(text);
        }
        else {
            Toast.makeText(getApplicationContext(), "File
not Found", Toast.LENGTH_SHORT).show();
            filecon.setText(null);
        }
    }
});
}
}
}

```





## 26. Android External Storage Example Code

Manifest.xml file

```
<uses-permission  
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>  
<uses-permission  
android:name="android.permission.READ_EXTERNAL_STORAGE"/>
```

Main\_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="fill_parent" android:layout_height="fill_parent"  
    android:orientation="vertical">  
  
    <TextView android:layout_width="fill_parent"  
        android:layout_height="wrap_content"  
        android:text="Reading and Writing to External Storage"  
        android:textSize="24sp"/>  
  
    <EditText android:id="@+id/myInputText"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:ems="10" android:lines="5"  
        android:minLines="3" android:gravity="top|left"  
        android:inputType="textMultiLine">  
  
        <requestFocus />  
    </EditText>  
  
    <LinearLayout  
        android:layout_width="match_parent" android:layout_height="wrap_content"  
        android:orientation="horizontal"  
        android:weightSum="1.0"  
        android:layout_marginTop="20dp">  
  
        <Button android:id="@+id/saveExternalStorage"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:text="SAVE"  
            android:layout_weight="0.5"/>  
  
        <Button android:id="@+id/getExternalStorage"  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:layout_weight="0.5"
```

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

```
</LinearLayout>
```

```
<TextView android:id="@+id/response"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content" android:padding="5dp"  
    android:text=""  
    android:textAppearance="?android:attr/textAppearanceMedium" />
```

```
</LinearLayout>
```

MainActivity.java

```
import java.io.BufferedReader;  
import java.io.DataInputStream;  
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import android.os.Bundle;  
import android.app.Activity;  
import android.os.Environment;  
import android.view.View;  
import android.view.View.OnClickListener;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
public class MainActivity extends Activity {  
    EditText inputText;  
    TextView response;  
    Button saveButton, readButton;  
  
    private String filename = "SampleFile.txt";  
    private String filepath = "MyFileStorage";  
    File myExternalFile;  
    String myData = "";  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        inputText = (EditText) findViewById(R.id.myInputText);  
        response = (TextView) findViewById(R.id.response);  
    }  
}
```

```

saveButton =
    (Button) findViewById(R.id.saveExternalStorage);
saveButton.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            FileOutputStream fos = new FileOutputStream(myExternalFile);
            fos.write(inputText.getText().toString().getBytes());
            fos.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
        inputText.setText("");
        response.setText("SampleFile.txt saved to External Storage...");
    }
});

readButton = (Button) findViewById(R.id.getExternalStorage);
readButton.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        try {
            FileInputStream fis = new FileInputStream(myExternalFile);
            DataInputStream in = new DataInputStream(fis);
            BufferedReader br =
                new BufferedReader(new InputStreamReader(in));
            String strLine;
            while ((strLine = br.readLine()) != null) {
                myData = myData + strLine;
            }
            in.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
        inputText.setText(myData);
        response.setText("SampleFile.txt data retrieved from Internal Storage...");
    }
});

if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
    saveButton.setEnabled(false);
}
else {
    myExternalFile = new File(getExternalFilesDir(filepath), filename);
}

}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    if (Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState)) {

```

```

        return true;
    }
    return false;
}

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    if (Environment.MEDIA_MOUNTED.equals(extStorageState)) {
        return true;
    }
    return false;
}
}

```

---

## 27. Working with Content Provider

- *Content Provider*

Program No -1

```

import android.content.ContentProvider;
import android.content.ContentValues;
import android.content.UriMatcher;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.net.Uri;
import android.support.annotation.Nullable;

public class CouponsContentProvider extends ContentProvider {

    private CouponSQLiteOpenHelper sqLiteOpenHelper;

    private static final String COUPONS_DBNAME = "zoftino";

    private static final String COUPON_TABLE = "coupon";

    private SQLiteDatabase cpnDB;

    private static final String SQL_CREATE_COUPON = "CREATE TABLE " +
        COUPON_TABLE +
        "(" +
        "_id INTEGER PRIMARY KEY, " +
        "STORE TEXT, " +
        "COUPON TEXT, " +
        "EXPIRES TEXT)";

```

```

private static final UriMatcher uriMatcher = new
UriMatcher(UriMatcher.NO_MATCH);
static {
    uriMatcher.addURI("com.zoftino.coupon.provider", COUPON_TABLE, 1);
}
@Override
public boolean onCreate() {
    //this way db create or open is delayed till getWritableDatabase() is called frist
    time
    sqLiteOpenHelper = new CouponSQLiteOpenHelper( getContext(),
COUPONS_DBNAME, SQL_CREATE_COUPON );
    return true;
}

```

```

@Nullable
@Override
public Cursor query(Uri uri,String[] projection,String selection,String[]
selectionArgs,
                    String sortOrder) {

```

```

    String tableNme = "";
    switch(uriMatcher.match(uri)){
        case 1 :
            tableNme = COUPON_TABLE;
            break;
        default:
            return null;
    }

```

```

    cpnDB = sqLiteOpenHelper.getWritableDatabase();

```

```

    Cursor cursor = (SQLiteCursor)cpnDB.query(tableNme, projection, selection,
selectionArgs,
        null, null, sortOrder);
    return cursor;
}

```

```

@Nullable
@Override
public String getType(Uri uri) {
    return null;
}

```

```

@Nullable
@Override
public Uri insert(Uri uri, ContentValues contentValues) {

```

```

    String tableNme = "";
    switch(uriMatcher.match(uri)){
        case 1 :
            tableNme = COUPON_TABLE;

```

```

        break;
    default:
        return null;
    }

```

```

    cpnDB = sqLiteOpenHelper.getWritableDatabase();
    long rowid = cpnDB.insert(tableNme, null, contentValues);
    return getContentUriRow(rowid);
}

```

```

@Override
public int delete(Uri uri, String where, String[] selectionArgs) {
    String tableNme = "";
    switch(uriMatcher.match(uri)){
        case 1 :
            tableNme = COUPON_TABLE;
            break;
        default:
            return 0;
    }
}

```

```

    cpnDB = sqLiteOpenHelper.getWritableDatabase();

```

```

    return cpnDB.delete(tableNme, where, selectionArgs);
}

```

```

@Override
public int update(Uri uri, ContentValues contentValues, String where, String[]
selectionArgs) {
    String tableNme = "";
    switch(uriMatcher.match(uri)){
        case 1 :
            tableNme = COUPON_TABLE;
            break;
        default:
            return 0;
    }
    cpnDB = sqLiteOpenHelper.getWritableDatabase();
    return cpnDB.update(tableNme, contentValues, where, selectionArgs );
}
private Uri getContentUriRow(long rowid){
    return Uri.fromParts("com.zoftino.coupon.provider", COUPON_TABLE,
Long.toString(rowid));
}
}

```

Program NO. 2  
*SQLiteOpenHelper*

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class CouponSQLiteOpenHelper extends SQLiteOpenHelper {

    private String sql;
    CouponSQLiteOpenHelper(Context context, String dbName, String msql) {
        super(context, dbName, null, 1);
        sql = msql;
    }

    public void onCreate(SQLiteDatabase db) {
        db.execSQL(sql);
    }

    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

    }
}
```

*Contract Class*

```
import android.net.Uri;

public class CouponsContract {

    public static final String Table_COUPON = "coupon";

    public static final String Column_ID = "_id";
    public static final String Column_STORE = "STORE";
    public static final String Column_COUPON = "COUPON";
    public static final String Column_EXPIRY = "EXPIRES";

    public static final String AUTHORITY = "com.zoftino.coupon.provider";

    public static final Uri AUTHORITY_URI = Uri.parse("content://" +
AUTHORITY);

    public static final Uri CONTENT_URI =
```

```
        Uri.withAppendedPath(AUTHORITY_URI, Table_COUPON);
    }
```

Prog No. 3

*Manifest.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.zoftino.content">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".CouponsContentResolverActivity"></activity>
        <provider
            android:authorities="com.zoftino.coupon.provider"
            android:name=".CouponsContentProvider"></provider>
    </application>

</manifest>
```

Prog No. 4

*Content Resolver Activity*

```
import android.content.ContentValues;
import android.database.Cursor;
import android.os.Bundle;
import android.support.v4.widget.SimpleCursorAdapter;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.ListView;

public class CouponsContentResolverActivity extends
AppCompatActivity {

    ListView cpnLst;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_resolver);
    }
}
```



```
cpnLst = (ListView) findViewById(R.id.couponsList);
}
```

```
public void viewCouponsFromCouponsContentProvider(View view){
```

```
    Cursor cursor = getCouponsFromProvider();
```

```
    String[] cursorColumns =
    {
        CouponsContract.Column_STORE,
        CouponsContract.Column_COUPON,
        CouponsContract.Column_EXPIRY
    };
```

```
    int[] viewIds = {R.id.storeName, R.id.coupon, R.id.expirationDt};
```

```
    SimpleCursorAdapter simpleCursorAdapter = new SimpleCursorAdapter(
        getApplicationContext(),
        R.layout.coupon_row,
        cursor,
        cursorColumns,
        viewIds,
        0);
```

```
    cpnLst.setAdapter(simpleCursorAdapter);
}
private Cursor getCouponsFromProvider() {
    String[] mProjection =
    {
        CouponsContract.Column_ID,
        CouponsContract.Column_STORE,
        CouponsContract.Column_COUPON,
        CouponsContract.Column_EXPIRY
    };
}
```

```
String mSelectionClause = CouponsContract.Column_STORE + " = ?";
```

```
String[] mSelectionArgs = {"amazon"};
```

```
String orderBy = CouponsContract.Column_EXPIRY + " ASC";
```

```
    return
    getContentResolver().query(CouponsContract.CONTENT_URI, mProjection, mSelectionClause, mSelectionArgs, orderBy);
}
```

```
public void addCouponsToCouponsContentProvider(View view){
```

```
    ContentValues contentValues = new ContentValues();
    contentValues.put(CouponsContract.Column_ID, 2);
```

```

        contentValues.put(CouponsContract.Column_STORE , "amazon");
        contentValues.put(CouponsContract.Column_COUPON , "Get Upto 40% Off on
Shoes");
        contentValues.put(CouponsContract.Column_EXPIRY , "2017/02/21");

        getContentResolver().insert(CouponsContract.CONTENT_URI, contentValues);
    }
}

```

#### Prog. No. 5

*Content Resolver Activity Layout (activity\_resolver.xml)*

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    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"
    android:orientation="vertical"
    tools:context="com.zoftino.content.MainActivity">
    <Button
        android:id="@+id/button2"
        android:text="Add Coupon To Provider"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="addCouponsToCouponsContentProvider"></Button>

    <Button
        android:id="@+id/button1"
        android:text="View Coupons From Provider"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="viewCouponsFromCouponsContentProvider"></Button>
    <ListView android:id="@+id/couponsList"
        android:layout_width="match_parent"
        android:layout_height="match_parent"></ListView>
</LinearLayout>

```

#### Prog. No. 6

*coupon\_row.xml*

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/coupon_row"
    android:orientation="horizontal"
    android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:paddingTop="@dimen/activity_vertical_margin">
        <TextView android:id="@+id/storeName" android:layout_width="100dp"
        android:textSize="20dp"
        android:textColor="@color/colorPrimary" android:textAlignment="center"
        android:layout_height="match_parent" android:text=""></TextView>

        <LinearLayout
        android:orientation="vertical"
        android:layout_weight="1"
        android:layout_width="odp"
        android:layout_marginLeft="@dimen/activity_horizontal_margin"
        android:layout_height="match_parent">
        <TextView android:id="@+id/coupon" android:layout_width="match_parent"
        android:textSize="15dp"
        android:layout_height="wrap_content" android:text=""></TextView>
        <TextView android:id="@+id/expirationDt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" android:text=""></TextView>
        </LinearLayout>
    </LinearLayout>

```

---

28.

## Android Animation Examples XML Code

### Fade In Animation

**fade\_in.xml**

```

<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true" >

    <alpha
        android:duration="1000"
        android:fromAlpha="0.0"
        android:interpolator="@android:anim/accelerate_interpolator"
        android:toAlpha="1.0" />

</set>

```

### Fade Out Animation

**fade\_out.xml**

```

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

```

```
android:fillAfter="true" >
```

```
<alpha  
  android:duration="1000"  
  android:fromAlpha="1.0"  
  android:interpolator="@android:anim/accelerate_interpolator"  
  android:toAlpha="0.0" />
```

```
</set>
```

## **Blink Animation**

**blink.xml**

```
<set xmlns:android="http://schemas.android.com/apk/res/android">  
  <alpha android:fromAlpha="0.0"  
    android:toAlpha="1.0"  
    android:interpolator="@android:anim/accelerate_interpolator"  
    android:duration="600"  
    android:repeatMode="reverse"  
    android:repeatCount="infinite"/>  
</set>
```

## **Zoom In Animation**

**zoom\_in.xml**

```
<set xmlns:android="http://schemas.android.com/apk/res/android"  
  android:fillAfter="true" >  
  
  <scale  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:duration="1000"  
    android:fromXScale="1"  
    android:fromYScale="1"  
    android:pivotX="50%"  
    android:pivotY="50%"  
    android:toXScale="3"  
    android:toYScale="3" >  
  </scale>  
  
</set>
```

## **Zoom Out Animation**

**zoom\_out.xml**

```
<set xmlns:android="http://schemas.android.com/apk/res/android"  
  android:fillAfter="true" >
```

```
<scale
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:duration="1000"
  android:fromXScale="1.0"
  android:fromYScale="1.0"
  android:pivotX="50%"
  android:pivotY="50%"
  android:toXScale="0.5"
  android:toYScale="0.5" >
</scale>
```

```
</set>
```

.

## Rotate Animation

**rotate.xml**

```
<set xmlns:android="http://schemas.android.com/apk/res/android">
  <rotate android:fromDegrees="0"
    android:toDegrees="360"
    android:pivotX="50%"
    android:pivotY="50%"
    android:duration="600"
    android:repeatMode="restart"
    android:repeatCount="infinite"
    android:interpolator="@android:anim/cycle_interpolator"/>
</set>
```

```
</set>
```

## Move Animation

**move.xml**

```
<set
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:interpolator="@android:anim/linear_interpolator"
  android:fillAfter="true">

  <translate
    android:fromXDelta="0%p"
    android:toXDelta="75%p"
    android:duration="800" />
</set>
```

## Slide Up Animation

**slide\_up.xml**

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

```
android:fillAfter="true" >
```

```
<scale
  android:duration="500"
  android:fromXScale="1.0"
  android:fromYScale="1.0"
  android:interpolator="@android:anim/linear_interpolator"
  android:toXScale="1.0"
  android:toYScale="0.0" />
```

```
</set>
```

## Slide Down Animation

`slide_down.xml`

```
<set xmlns:android="http://schemas.android.com/apk/res/android"
  android:fillAfter="true">
```

```
<scale
  android:duration="500"
  android:fromXScale="1.0"
  android:fromYScale="0.0"
  android:toXScale="1.0"
  android:toYScale="1.0" />
```

```
</set>
```

## Bounce Animation

`bounce.xml`

```
<set xmlns:android="http://schemas.android.com/apk/res/android"
  android:fillAfter="true"
  android:interpolator="@android:anim/bounce_interpolator">
```

```
<scale
  android:duration="500"
  android:fromXScale="1.0"
  android:fromYScale="0.0"
  android:toXScale="1.0"
  android:toYScale="1.0" />
```

```
</set>
```

## Sequential Animation

`sequential.xml`

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

```
android:fillAfter="true"
android:interpolator="@android:anim/linear_interpolator" >
```

```
<!-- Move -->
<translate
  android:duration="800"
  android:fillAfter="true"
  android:fromXDelta="0%p"
  android:startOffset="300"
  android:toXDelta="75%p" />
<translate
  android:duration="800"
  android:fillAfter="true"
  android:fromYDelta="0%p"
  android:startOffset="1100"
  android:toYDelta="70%p" />
<translate
  android:duration="800"
  android:fillAfter="true"
  android:fromXDelta="0%p"
  android:startOffset="1900"
  android:toXDelta="-75%p" />
<translate
  android:duration="800"
  android:fillAfter="true"
  android:fromYDelta="0%p"
  android:startOffset="2700"
  android:toYDelta="-70%p" />

<!-- Rotate 360 degrees -->
<rotate
  android:duration="1000"
  android:fromDegrees="0"
  android:interpolator="@android:anim/cycle_interpolator"
  android:pivotX="50%"
  android:pivotY="50%"
  android:startOffset="3800"
  android:repeatCount="infinite"
  android:repeatMode="restart"
  android:toDegrees="360" />

</set>
```

## Together Animation

**together.xml**

```
<set xmlns:android="http://schemas.android.com/apk/res/android"
  android:fillAfter="true"
  android:interpolator="@android:anim/linear_interpolator" >
```

```
<!-- Move -->
<scale
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:duration="4000"
  android:fromXScale="1"
  android:fromYScale="1"
  android:pivotX="50%"
  android:pivotY="50%"
  android:toXScale="4"
  android:toYScale="4" >
</scale>
```

```
<!-- Rotate 180 degrees -->
<rotate
  android:duration="500"
  android:fromDegrees="0"
  android:pivotX="50%"
  android:pivotY="50%"
  android:repeatCount="infinite"
  android:repeatMode="restart"
  android:toDegrees="360" />
```

```
</set>
```

```
activity_main.xml
```

```
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent" >
```

```
<RelativeLayout
  android:layout_width="match_parent"
  android:layout_height="match_parent">
```

```
<Button
  android:id="@+id/btnFadeIn"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:layout_margin="5dp"
  android:text="Fade In" />
```

```
<TextView
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:textAppearance="?android:attr/textAppearanceMedium"
  android:text="Fade In"
  android:id="@+id/txt_fade_in"
  android:layout_alignBottom="@+id/btnFadeIn"
  android:layout_alignLeft="@+id/txt_fade_out"
  android:layout_alignStart="@+id/txt_fade_out" />
```



```
<Button
    android:id="@+id/btnFadeOut"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnFadeIn"
    android:text="Fade Out" />
```

```
<Button
    android:id="@+id/btnCrossFade"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnFadeOut"
    android:text="Cross Fade" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Cross Fade In"
    android:id="@+id/txt_out"
    android:visibility="gone"
    android:layout_gravity="center_horizontal"
    android:layout_alignTop="@+id/txt_in"
    android:layout_alignLeft="@+id/txt_in"
    android:layout_alignStart="@+id/txt_in" />
```

```
<Button
    android:id="@+id/btnBlink"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnCrossFade"
    android:text="Blink" />
```

```
<Button
    android:id="@+id/btnZoomIn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnBlink"
    android:text="Zoom In" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
```

```
        android:text="Blink"
        android:id="@+id/txt_blink"
        android:layout_gravity="center_horizontal"
        android:layout_alignBottom="@+id/btnBlink"
        android:layout_alignLeft="@+id/txt_zoom_in"
        android:layout_alignStart="@+id/txt_zoom_in" />
```

```
<Button
    android:id="@+id/btnZoomOut"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnZoomIn"
    android:text="Zoom Out" />
```

```
<Button
    android:id="@+id/btnRotate"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnZoomOut"
    android:text="Rotate" />
```

```
<Button
    android:id="@+id/btnMove"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnRotate"
    android:text="Move" />
```

```
<Button
    android:id="@+id/btnSlideUp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="5dp"
    android:layout_below="@id/btnMove"
    android:text="Slide Up" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Fade Out"
    android:id="@+id/txt_fade_out"
    android:layout_gravity="center_horizontal"
    android:layout_alignBottom="@+id/btnFadeOut"
    android:layout_alignLeft="@+id/txt_in"
    android:layout_alignStart="@+id/txt_in" />
```

```
<Button
```

```
android:id="@+id/btnSlideDown"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="5dp"
android:layout_below="@id/btnSlideUp"
android:text="Slide Down" />
```

<Button

```
android:id="@+id/btnBounce"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="5dp"
android:layout_below="@id/btnSlideDown"
android:text="Bounce" />
```

<Button

```
android:id="@+id/btnSequential"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="5dp"
android:layout_below="@id/btnBounce"
android:text="Sequential Animation" />
```

<Button

```
android:id="@+id/btnTogether"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_below="@id/btnSequential"
android:layout_margin="5dp"
android:text="Together Animation" />
```

<TextView

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAppearance="?android:attr/textAppearanceMedium"
android:text="Cross Fade Out"
android:id="@+id/txt_in"
android:layout_gravity="center_horizontal"
android:layout_alignBottom="@+id/btnCrossFade"
android:layout_alignLeft="@+id/txt_blink"
android:layout_alignStart="@+id/txt_blink" />
```

<TextView

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAppearance="?android:attr/textAppearanceMedium"
android:text="Zoom In"
android:id="@+id/txt_zoom_in"
android:layout_alignBottom="@+id/btnZoomIn"
android:layout_alignLeft="@+id/txt_zoom_out"
android:layout_alignStart="@+id/txt_zoom_out" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Zoom Out"
    android:id="@+id/txt_zoom_out"
    android:layout_alignBottom="@+id/btnZoomOut"
    android:layout_toRightOf="@+id/btnSequential"
    android:layout_toEndOf="@+id/btnSequential" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Rotate"
    android:id="@+id/txt_rotate"
    android:layout_above="@+id/btnMove"
    android:layout_toRightOf="@+id/btnSequential"
    android:layout_toEndOf="@+id/btnSequential" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Move"
    android:id="@+id/txt_move"
    android:layout_alignBottom="@+id/btnMove"
    android:layout_alignLeft="@+id/txt_slide_up"
    android:layout_alignStart="@+id/txt_slide_up" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Slide Up"
    android:id="@+id/txt_slide_up"
    android:layout_alignBottom="@+id/btnSlideUp"
    android:layout_toRightOf="@+id/btnSequential"
    android:layout_toEndOf="@+id/btnSequential" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Slide Down"
    android:id="@+id/txt_slide_down"
    android:layout_alignBottom="@+id/btnSlideDown"
    android:layout_alignLeft="@+id/txt_slide_up"
    android:layout_alignStart="@+id/txt_slide_up" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Bounce"
    android:id="@+id/txt_bounce"
    android:layout_alignBottom="@+id/btnBounce"
    android:layout_alignLeft="@+id/txt_slide_down"
    android:layout_alignStart="@+id/txt_slide_down" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Sequential"
    android:id="@+id/txt_seq"
    android:layout_alignBottom="@+id/btnSequential"
    android:layout_alignLeft="@+id/txt_bounce"
    android:layout_alignStart="@+id/txt_bounce" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:text="Together"
    android:id="@+id/txt_tog"
    android:layout_alignBottom="@+id/btnTogether"
    android:layout_toRightOf="@+id/btnSequential"
    android:layout_toEndOf="@+id/btnSequential" />
```

```
</RelativeLayout>
```

```
</ScrollView>
```

**MainActivity.java** .

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.TextView;
```

```
public class MainActivity extends Activity {
```

```

        Button btnFadeIn, btnFadeOut, btnCrossFade, btnBlink, btnZoomIn,
            btnZoomOut, btnRotate, btnMove, btnSlideUp, btnSlideDown,
            btnBounce, btnSequential, btnTogether;
    Animation
    animFadeIn, animFadeOut, animBlink, animZoomIn, animZoomOut, animRotate

    , animMove, animSlideUp, animSlideDown, animBounce, animSequential, animTogether,
    animCrossFadeIn, animCrossFadeOut;
    TextView
    txtFadeIn, txtFadeOut, txtBlink, txtZoomIn, txtZoomOut, txtRotate, txtMove, txtSlideUp
    ,
        txtSlideDown, txtBounce, txtSeq, txtTog, txtIn, txtOut;

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

        btnFadeIn = (Button) findViewById(R.id.btnFadeIn);
        btnFadeOut = (Button) findViewById(R.id.btnFadeOut);
        btnCrossFade = (Button) findViewById(R.id.btnCrossFade);
        btnBlink = (Button) findViewById(R.id.btnBlink);
        btnZoomIn = (Button) findViewById(R.id.btnZoomIn);
        btnZoomOut = (Button) findViewById(R.id.btnZoomOut);
        btnRotate = (Button) findViewById(R.id.btnRotate);
        btnMove = (Button) findViewById(R.id.btnMove);
        btnSlideUp = (Button) findViewById(R.id.btnSlideUp);
        btnSlideDown = (Button) findViewById(R.id.btnSlideDown);
        btnBounce = (Button) findViewById(R.id.btnBounce);
        btnSequential = (Button) findViewById(R.id.btnSequential);
        btnTogether = (Button) findViewById(R.id.btnTogether);
        txtFadeIn = (TextView) findViewById(R.id.txt_fade_in);
        txtFadeOut = (TextView) findViewById(R.id.txt_fade_out);
        txtBlink = (TextView) findViewById(R.id.txt_blink);
        txtZoomIn = (TextView) findViewById(R.id.txt_zoom_in);
        txtZoomOut = (TextView) findViewById(R.id.txt_zoom_out);
        txtRotate = (TextView) findViewById(R.id.txt_rotate);
        txtMove = (TextView) findViewById(R.id.txt_move);
        txtSlideUp = (TextView) findViewById(R.id.txt_slide_up);
        txtSlideDown = (TextView) findViewById(R.id.txt_slide_down);
        txtBounce = (TextView) findViewById(R.id.txt_bounce);
        txtSeq = (TextView) findViewById(R.id.txt_seq);
        txtTog = (TextView) findViewById(R.id.txt_tog);
        txtIn = (TextView) findViewById(R.id.txt_in);
        txtOut = (TextView) findViewById(R.id.txt_out);
        animFadeIn = AnimationUtils.loadAnimation(getApplicationContext(),
            R.anim.fade_in);

        animFadeIn = AnimationUtils.loadAnimation(getApplicationContext(),
            R.anim.fade_in);
        // fade in

```

```

btnFadeIn.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
        txtFadeIn.setVisibility(View.VISIBLE);
        txtFadeIn.startAnimation(animFadeIn);
    }
});

animFadeOut = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.fade_out);

// fade out
btnFadeOut.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtFadeOut.setVisibility(View.VISIBLE);
        txtFadeOut.startAnimation(animFadeOut);
    }
});
animCrossFadeIn = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.fade_in);
animCrossFadeOut = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.fade_out);
// cross fade
btnCrossFade.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtOut.setVisibility(View.VISIBLE);
        // start fade in animation
        txtOut.startAnimation(animCrossFadeIn);

        // start fade out animation
        txtIn.startAnimation(animCrossFadeOut);
    }
});
animBlink = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.blink);
// blink
btnBlink.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtBlink.setVisibility(View.VISIBLE);
        txtBlink.startAnimation(animBlink);
    }
});

animZoomIn = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.zoom_in);
// Zoom In
btnZoomIn.setOnClickListener(new View.OnClickListener() {

```

```

@Override
public void onClick(View v) {
    txtZoomIn.setVisibility(View.VISIBLE);
    txtZoomIn.startAnimation(animZoomIn);
}
});
animZoomOut = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.zoom_out);
// Zoom Out
btnZoomOut.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtZoomOut.setVisibility(View.VISIBLE);
        txtZoomOut.startAnimation(animZoomOut);
    }
});
animRotate = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.rotate);

// Rotate
btnRotate.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtRotate.startAnimation(animRotate);
    }
});
animMove = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.move);
// Move
btnMove.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtMove.startAnimation(animMove);
    }
});
animSlideUp = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.slide_up);
// Slide Up
btnSlideUp.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtSlideUp.startAnimation(animSlideUp);
    }
});
animSlideDown = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.slide_down);
// Slide Down
btnSlideDown.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtSlideDown.startAnimation(animSlideDown);
    }
});

```



```

    }
});
animBounce = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.bounce);
// Slide Down
btnBounce.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtBounce.startAnimation(animBounce);
    }
});
animSequential = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.sequential);
// Sequential
btnSequential.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        txtSeq.startAnimation(animSequential);
    }
});
animTogether = AnimationUtils.loadAnimation(getApplicationContext(),
    R.anim.together);

// Together
btnTogether.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        txtTog.startAnimation(animTogether);
    }
});
}
}

```

---

29.

## **Android Capture Image Camera Gallery Using FileProvider**

### **FileProvider**

FileProvider is a special subclass of ContentProvider which allows sharing of files between application through content URI instead of `file://` URI.

Using `file://` URI is not the best idea. It gives all apps the permission to access the files once the Storage Permissions are granted.

We somehow need to restrict this such that the user knows the applications with which it would be sharing the files.

For this, we use **FileProviders** which allow temporary access permissions to the files. Otherwise, we were able to access files from other apps by simply getting their URI from `Uri.parse()`

By using FileProvider in your app, you do not need to ask user to grant **WRITE\_EXTERNAL\_STORAGE** permission everytime.

## Defining FileProvider

To define a FileProvider in our android application, we need to do the following things:

- Define the FileProvider in your AndroidManifest file
- Create an XML file that contains all paths that the FileProvider will share with other applications

```
<provider
    android:name="android.support.v4.content.FileProvider"
    android:authorities="${applicationId}.provider"
    android:exported="false"
    android:grantUriPermissions="true">
    <meta-data
        android:name="android.support.FILE_PROVIDER_PATHS"
        android:resource="@xml/provider_paths"/>
</provider>
```

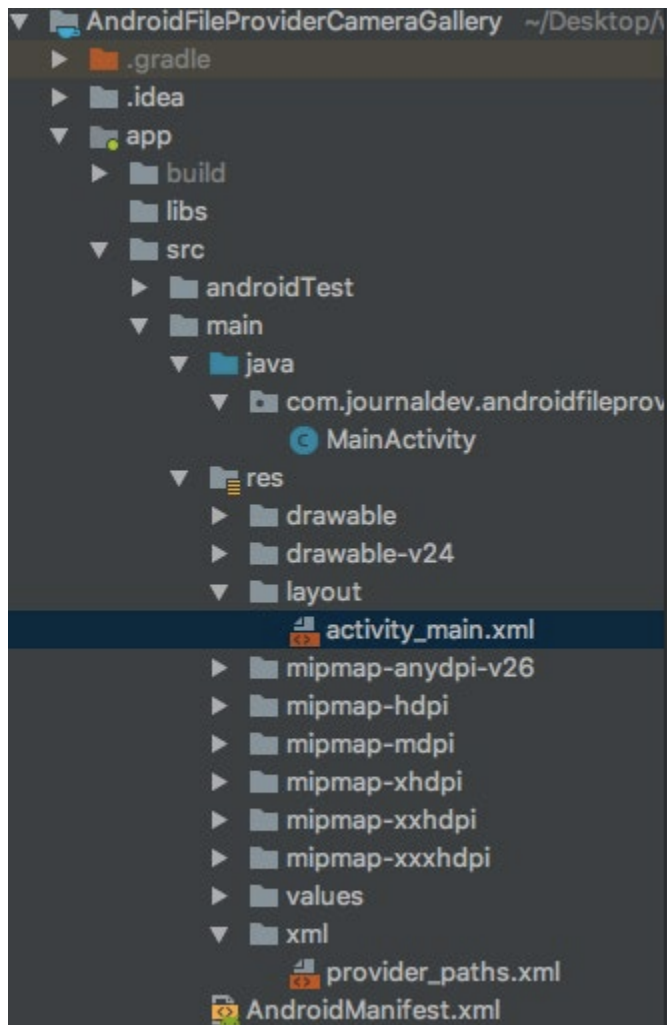
Create an xml folder inside the res directory.  
Add the `provider_paths.xml` file in it:

```
<?xml version="1.0" encoding="utf-8"?>
<paths xmlns:android="http://schemas.android.com/apk/res/android">
    <external-path name="external_files" path="."/>
</paths>
```

Depending on the storage we need to access, we pass the value in the external-path.  
Example of other values that can be passed – `sdcard`

Now let's write our Version 2.0 Application of Capturing Image from Camera And Gallery that works on Android Nougat and above.

## Project Structure



The AndroidManifest.xml with all the permissions looks like:

```

<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.journaldev.imagepicker">

    <uses-feature android:name="android.hardware.camera" android:required="false" />
    <uses-feature android:name="android.hardware.camera.autofocus" android:required="false" />
    <uses-feature android:name="android.hardware.camera.flash" android:required="false" />
    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="ImagePickerMine"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <provider
            android:name="android.support.v4.content.FileProvider"
            android:authorities="${applicationId}.provider"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/provider_paths"/>
            </meta-data>
        </provider>

        <activity
            android:name=".MainActivity"
            android:label="ImagePickerMine">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

## Code

The code for the activity\_main.xml layout is:

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout
    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:fitsSystemWindows="true">

```

```
<RelativeLayout  
    android:id="@+id/content_main"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:padding="16dp"  
    app:layout_behavior="@string/appbar_scrolling_view_behavior">
```

```
<ImageView  
    android:id="@+id/imageView"  
    android:layout_width="250dp"  
    android:layout_height="250dp"  
    android:layout_centerInParent="true"  
    android:adjustViewBounds="true"  
    android:scaleType="centerCrop" />
```

```
</RelativeLayout>
```

```
<android.support.design.widget.FloatingActionButton  
    android:id="@+id/fab"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="bottom|end"  
    android:layout_margin="16dp"
```

```
app:srcCompat="@android:drawable/ic_menu_camera" />
```

```
</android.support.design.widget.CoordinatorLayout>
```

The code for the MainActivity.java is given below:

```
package com.journaldev.androidfileprovidercameragallery;
```

```
import android.annotation.TargetApi;
import android.app.Activity;
import android.content.ComponentName;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.content.pm.ResolveInfo;
import android.database.Cursor;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.net.Uri;
import android.os.Build;
import android.os.Bundle;
import android.os.Parcelable;
import android.provider.MediaStore;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
```

```
import android.widget.ImageView;
```

```
import java.io.File;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
import static android.Manifest.permission.CAMERA;
```

```
import static android.Manifest.permission.READ_EXTERNAL_STORAGE;
```

```
import static android.Manifest.permission.WRITE_EXTERNAL_STORAGE;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    Uri picUri;
```

```
    private ArrayList<String> permissionsToRequest;
```

```
    private ArrayList<String> permissionsRejected = new ArrayList<>();
```

```
    private ArrayList<String> permissions = new ArrayList<>();
```

```
    private final static int ALL_PERMISSIONS_RESULT = 107;
```

```
    private final static int IMAGE_RESULT = 200;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
FloatingActionButton fab = findViewById(R.id.fab);  
fab.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        startActivityForResult(getPickImageChooserIntent(), IMAGE_RESULT);  
    }  
});
```

```
permissions.add(CAMERA);  
permissions.add(WRITE_EXTERNAL_STORAGE);  
permissions.add(READ_EXTERNAL_STORAGE);  
permissionsToRequest = findUnAskedPermissions(permissions);
```

```
if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
```

```
    if (permissionsToRequest.size() > 0)  
        requestPermissions(permissionsToRequest.toArray(new  
String[permissionsToRequest.size()]), ALL_PERMISSIONS_RESULT);  
    }  
}
```

```
public Intent getPickImageChooserIntent() {
```



```
Uri outputFileUri = getCaptureImageOutputUri();
```

```
List<Intent> allIntents = new ArrayList<>();
```

```
PackageManager packageManager = getPackageManager();
```

```
Intent captureIntent = new  
Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
```

```
List<ResolveInfo> listCam =  
packageManager.queryIntentActivities(captureIntent, 0);
```

```
for (ResolveInfo res : listCam) {
```

```
    Intent intent = new Intent(captureIntent);
```

```
    intent.setComponent(new ComponentName(res.activityInfo.packageName,  
res.activityInfo.name));
```

```
    intent.setPackage(res.activityInfo.packageName);
```

```
    if (outputFileUri != null) {
```

```
        intent.putExtra(MediaStore.EXTRA_OUTPUT, outputFileUri);
```

```
    }
```

```
    allIntents.add(intent);
```

```
}
```

```
Intent galleryIntent = new Intent(Intent.ACTION_GET_CONTENT);
```

```
galleryIntent.setType("image/*");
```

```
List<ResolveInfo> listGallery =  
packageManager.queryIntentActivities(galleryIntent, 0);
```

```
for (ResolveInfo res : listGallery) {
```

```
    Intent intent = new Intent(galleryIntent);
```

```
    intent.setComponent(new ComponentName(res.activityInfo.packageName,  
res.activityInfo.name));
```

```

        intent.setPackage(res.activityInfo.packageName);
        allIntents.add(intent);
    }

    Intent mainIntent = allIntents.get(allIntents.size() - 1);
    for (Intent intent : allIntents) {
        if
(intent.getComponent().getClassName().equals("com.android.documentsui.Docume
ntsActivity")) {
            mainIntent = intent;
            break;
        }
    }
    allIntents.remove(mainIntent);

    Intent chooserIntent = Intent.createChooser(mainIntent, "Select source");

    chooserIntent.putExtra(Intent.EXTRA_INITIAL_INTENTS,
allIntents.toArray(new Parcelable[allIntents.size()]));

    return chooserIntent;
}

private Uri getCaptureImageOutputUri() {
    Uri outputFileUri = null;
    File getImage = getExternalFilesDir("");
    if (getImage != null) {
        outputFileUri = Uri.fromFile(new File(getImage.getPath(), "profile.png"));
    }
}

```

```
    }

    return outputFileUri;
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    if (resultCode == Activity.RESULT_OK) {

        ImageView imageView = findViewById(R.id.imageView);

        if (requestCode == IMAGE_RESULT) {

            String filePath = getImageFilePath(data);
            if (filePath != null) {
                Bitmap selectedImage = BitmapFactory.decodeFile(filePath);
                imageView.setImageBitmap(selectedImage);
            }
        }

    }

}

private String getImageFromFilePath(Intent data) {
```

```

        boolean isCamera = data == null || data.getData() == null;

        if (isCamera) return getCaptureImageOutputUri().getPath();
        else return getPathFromURI(data.getData());

    }

    public String getImageFilePath(Intent data) {
        return getImageFromFilePath(data);
    }

    private String getPathFromURI(Uri contentUri) {
        String[] proj = {MediaStore.Audio.Media.DATA};
        Cursor cursor = getContentResolver().query(contentUri, proj, null, null, null);
        int column_index =
cursor.getColumnIndexOrThrow(MediaStore.Audio.Media.DATA);
        cursor.moveToFirst();
        return cursor.getString(column_index);
    }

    @Override
    protected void onSaveInstanceState(Bundle outState) {
        super.onSaveInstanceState(outState);

        outState.putParcelable("pic_uri", picUri);
    }

    @Override

```

```

protected void onRestoreInstanceState(Bundle savedInstanceState) {
    super.onRestoreInstanceState(savedInstanceState);

    // get the file url
    picUri = savedInstanceState.getParcelable("pic_uri");
}

private ArrayList<String> findUnAskedPermissions(ArrayList<String> wanted) {
    ArrayList<String> result = new ArrayList<String>();

    for (String perm : wanted) {
        if (!hasPermission(perm)) {
            result.add(perm);
        }
    }

    return result;
}

private boolean hasPermission(String permission) {
    if (canMakeSmoes()) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            return (checkSelfPermission(permission) ==
PackageManager.PERMISSION_GRANTED);
        }
    }

    return true;
}

```

```
private void showMessageOKCancel(String message,
DialogInterface.OnClickListener okListener) {
```

```
    new AlertDialog.Builder(this)

        .setMessage(message)

        .setPositiveButton("OK", okListener)

        .setNegativeButton("Cancel", null)

        .create()

        .show();
}
```

```
private boolean canMakeSmores() {

    return (Build.VERSION.SDK_INT >
Build.VERSION_CODES.LOLLIPOP_MR1);
}
```

```
@TargetApi(Build.VERSION_CODES.M)
```

```
@Override
```

```
public void onRequestPermissionsResult(int requestCode, String[] permissions,
int[] grantResults) {
```

```
    switch (requestCode) {
```

```
        case ALL_PERMISSIONS_RESULT:
```

```
            for (String perms : permissionsToRequest) {
```

```
                if (!hasPermission(perms)) {
```

```
                    permissionsRejected.add(perms);
```

```
                }
```

```

    }

    if (permissionsRejected.size() > 0) {

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            if
(shouldShowRequestPermissionRationale(permissionsRejected.get(0))) {
                showMessageOKCancel("These permissions are mandatory for the
application. Please allow access.",
                    new DialogInterface.OnClickListener() {
                        @Override
                        public void onClick(DialogInterface dialog, int which) {
                            if (Build.VERSION.SDK_INT >=
Build.VERSION_CODES.M) {

                                requestPermissions(permissionsRejected.toArray(new
String[permissionsRejected.size()]), ALL_PERMISSIONS_RESULT);
                            }
                        }
                    });
                return;
            }
        }

    }

    break;

```

```
    }  
  
    }  
}
```

Following are the methods which get the content uri from the filepath provided that the FileProvider has granted the appropriate permissions:

26.

### Android Capture Image from Camera and Gallery

Add the following permissions in the Android Manifest.xml file, above the application tag.

```
<uses-feature  
    android:name="android.hardware.camera"  
    android:required="false" />  
<uses-feature  
    android:name="android.hardware.camera.autofocus"  
    android:required="false" />  
<uses-feature  
    android:name="android.hardware.camera.flash"  
    android:required="false" />  
  
<uses-permission android:name="android.permission.CAMERA" />  
  
<uses-permission  
    android:name="android.permission.READ_EXTERNAL_STORAGE" />
```



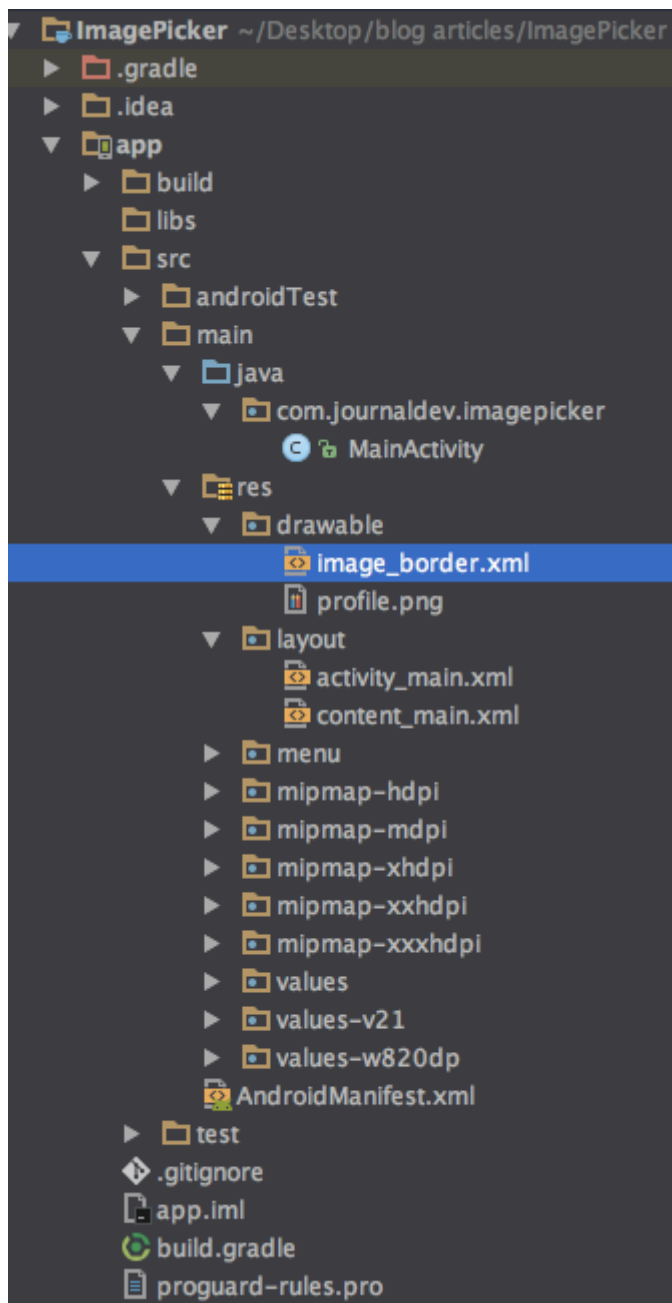
By adding **android.hardware.camera**, Play Store detects and prevents installing the application on devices with no camera.

Intent is the standard way to delegate actions to another application. To start the native camera the Intent requires **android.provider.MediaStore.ACTION\_IMAGE\_CAPTURE**.

To choose an image from gallery, the Intent requires the following argument : **Intent.ACTION\_GET\_CONTENT**.

In this tutorial we'll be invoking an image picker, that lets us select an image from camera or gallery and displays the image in a circular image view and a normal image view. Add the following dependency inside the build.gradle file.  
**compile 'de.hdodenhof:circleimageview:2.1.0'**

## **Android Image Capture Project Structure**



## Android Capture Image Code

The layout for the activity\_main.xml stays the same barring the icon change for the FAB button to `@android:drawable/ic_menu_camera`.

The `content_main.xml` is given below:

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

```
<RelativeLayout 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:id="@+id/content_main"
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"
android:background="#000000"
app:layout_behavior="@string/appbar_scrolling_view_behavior"
tools:context="com.journaldev.imagepicker.MainActivity"
tools:showIn="@layout/activity_main">
```

<RelativeLayout

```
    android:layout_width="250dp"
    android:layout_height="250dp"
    android:layout_centerHorizontal="true"
    android:layout_centerVertical="true"
    android:background="@drawable/image_border"
    android:clickable="true"
    android:orientation="vertical">
```

<ImageView

```
    android:id="@+id/imageView"
```

```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:adjustViewBounds="true"
        android:scaleType="centerCrop" />
```

```
</RelativeLayout>
```

```
<de.hdodenhof.circleimageview.CircleImageView
    android:id="@+id/img_profile"
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:layout_gravity="center_horizontal"
    android:src="@drawable/profile"
    app:civ_border_width="5dp"
    app:civ_border_color="#FFFFFF"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true" />
```

```
</RelativeLayout>
```

The code for the **MainActivity.java** is given below

```
public class MainActivity extends AppCompatActivity {
```

```
    Bitmap myBitmap;
```

```
    Uri picUri;
```

```
private ArrayList permissionsToRequest;

private ArrayList permissionsRejected = new ArrayList();

private ArrayList permissions = new ArrayList();

private final static int ALL_PERMISSIONS_RESULT = 107;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);

    setSupportActionBar(toolbar);

    FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);

    fab.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View view) {

            startActivityForResult(getPickImageChooserIntent(), 200);

        }

    });

    permissions.add(CAMERA);

    permissionsToRequest = findUnAskedPermissions(permissions);

    //get the permissions we have asked for before but are not granted..

    //we will store this in a global list to access later.
```

```

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

            if (permissionsToRequest.size() > 0)

                requestPermissions(permissionsToRequest.toArray(new
String[permissionsToRequest.size()]), ALL_PERMISSIONS_RESULT);
            }

        }

@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(MenuItem item) {

    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {

        return true;
    }
}

```

```

    }

    return super.onOptionsItemSelected(item);
}

/**
 * Create a chooser intent to select the source to get image from.<br />
 * The source can be camera's (ACTION_IMAGE_CAPTURE) or gallery's
 (ACTION_GET_CONTENT).<br />
 * All possible sources are added to the intent chooser.
 */
public Intent getPickImageChooserIntent() {

    // Determine Uri of camera image to save.
    Uri outputFileUri = getCaptureImageOutputUri();

    List allIntents = new ArrayList();

    PackageManager packageManager = getPackageManager();

    // collect all camera intents

    Intent captureIntent = new
Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);

    List listCam = packageManager.queryIntentActivities(captureIntent, 0);

    for (ResolveInfo res : listCam) {

        Intent intent = new Intent(captureIntent);

        intent.setComponent(new ComponentName(res.activityInfo.packageName,
res.activityInfo.name));

```

```

        intent.setPackage(res.activityInfo.packageName);
        if (outputFileUri != null) {
            intent.putExtra(MediaStore.EXTRA_OUTPUT, outputFileUri);
        }
        allIntents.add(intent);
    }

    // collect all gallery intents
    Intent galleryIntent = new Intent(Intent.ACTION_GET_CONTENT);
    galleryIntent.setType("image/*");
    List listGallery = packageManager.queryIntentActivities(galleryIntent, 0);
    for (ResolveInfo res : listGallery) {
        Intent intent = new Intent(galleryIntent);
        intent.setComponent(new ComponentName(res.activityInfo.packageName,
res.activityInfo.name));
        intent.setPackage(res.activityInfo.packageName);
        allIntents.add(intent);
    }

    // the main intent is the last in the list (fucking android) so pickup the useless
    one
    Intent mainIntent = allIntents.get(allIntents.size() - 1);
    for (Intent intent : allIntents) {
        if
(intent.getComponent().getClassName().equals("com.android.documentsui.Docume
ntsActivity")) {
            mainIntent = intent;
            break;
        }
    }

```



```

    }

    allIntents.remove(mainIntent);

    // Create a chooser from the main intent
    Intent chooserIntent = Intent.createChooser(mainIntent, "Select source");

    // Add all other intents
    chooserIntent.putExtra(Intent.EXTRA_INITIAL_INTENTS,
allIntents.toArray(new Parcelable[allIntents.size()]));

    return chooserIntent;
}

/**
 * Get URI to image received from capture by camera.
 */
private Uri getCaptureImageOutputUri() {
    Uri outputFileUri = null;
    File getImage = getExternalCacheDir();
    if (getImage != null) {
        outputFileUri = Uri.fromFile(new File(getImage.getPath(), "profile.png"));
    }
    return outputFileUri;
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {

```

```
Bitmap bitmap;

if (resultCode == Activity.RESULT_OK) {

    ImageView imageView = (ImageView) findViewById(R.id.imageView);

    if (getPickImageResultUri(data) != null) {

        picUri = getPickImageResultUri(data);

        try {

            myBitmap =
MediaStore.Images.Media.getBitmap(this.getContentResolver(), picUri);

            myBitmap = rotateImageIfRequired(myBitmap, picUri);

            myBitmap = getResizedBitmap(myBitmap, 500);

            CircleImageView croppedImageView = (CircleImageView)
findViewById(R.id.img_profile);

            croppedImageView.setImageBitmap(myBitmap);

            imageView.setImageBitmap(myBitmap);

        } catch (IOException e) {

            e.printStackTrace();

        }

    } else {
```

```

        bitmap = (Bitmap) data.getExtras().get("data");

        myBitmap = bitmap;

        CircleImageView croppedImageView = (CircleImageView)
findViewById(R.id.img_profile);

        if (croppedImageView != null) {

            croppedImageView.setImageBitmap(myBitmap);

        }

        imageView.setImageBitmap(myBitmap);

    }

}

}

private static Bitmap rotateImageIfRequired(Bitmap img, Uri selectedImage)
throws IOException {

    ExifInterface ei = new ExifInterface(selectedImage.getPath());

    int orientation = ei.getAttributeInt(ExifInterface.TAG_ORIENTATION,
ExifInterface.ORIENTATION_NORMAL);

    switch (orientation) {

        case ExifInterface.ORIENTATION_ROTATE_90:

            return rotateImage(img, 90);

        case ExifInterface.ORIENTATION_ROTATE_180:

```

```

        return rotateImage(img, 180);
    case ExifInterface.ORIENTATION_ROTATE_270:
        return rotateImage(img, 270);
    default:
        return img;
    }
}

```

```

private static Bitmap rotateImage(Bitmap img, int degree) {
    Matrix matrix = new Matrix();
    matrix.postRotate(degree);

    Bitmap rotatedImg = Bitmap.createBitmap(img, 0, 0, img.getWidth(),
img.getHeight(), matrix, true);

    img.recycle();

    return rotatedImg;
}

```

```

public Bitmap getResizedBitmap(Bitmap image, int maxSize) {
    int width = image.getWidth();
    int height = image.getHeight();

    float bitmapRatio = (float) width / (float) height;
    if (bitmapRatio > 0) {
        width = maxSize;
        height = (int) (width / bitmapRatio);
    } else {
        height = maxSize;
        width = (int) (height * bitmapRatio);
    }
}

```

```

    }

    return Bitmap.createScaledBitmap(image, width, height, true);
}

/**
 * Get the URI of the selected image from {@link
 #getPickImageChooserIntent()}.<br />
 * Will return the correct URI for camera and gallery image.
 *
 * @param data the returned data of the activity result
 */
public Uri getPickImageResultUri(Intent data) {

    boolean isCamera = true;

    if (data != null) {

        String action = data.getAction();

        isCamera = action != null &&
action.equals(MediaStore.ACTION_IMAGE_CAPTURE);
    }

    return isCamera ? getCaptureImageOutputUri() : data.getData();
}

@Override
protected void onSaveInstanceState(Bundle outState) {

    super.onSaveInstanceState(outState);

```

```

        // save file url in bundle as it will be null on screen orientation
        // changes
        outState.putParcelable("pic_uri", picUri);
    }

    @Override
    protected void onRestoreInstanceState(Bundle savedInstanceState) {
        super.onRestoreInstanceState(savedInstanceState);

        // get the file url
        picUri = savedInstanceState.getParcelable("pic_uri");
    }

    private ArrayList findUnAskedPermissions(ArrayList wanted) {
        ArrayList result = new ArrayList();

        for (String perm : wanted) {
            if (!hasPermission(perm)) {
                result.add(perm);
            }
        }

        return result;
    }

    private boolean hasPermission(String permission) {
        if (canMakeSmores()) {

```

```

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

            return (checkSelfPermission(permission) ==
PackageManager.PERMISSION_GRANTED);

        }

    }

    return true;

}

private void showMessageOKCancel(String message,
DialogInterface.OnClickListener okListener) {

    new AlertDialog.Builder(this)

        .setMessage(message)

        .setPositiveButton("OK", okListener)

        .setNegativeButton("Cancel", null)

        .create()

        .show();

}

private boolean canMakeSmores() {

    return (Build.VERSION.SDK_INT >
Build.VERSION_CODES.LOLLIPOP_MR1);

}

@TargetApi(Build.VERSION_CODES.M)

@Override

public void onRequestPermissionsResult(int requestCode, String[] permissions,
int[] grantResults) {

    switch (requestCode) {

```

```
case ALL_PERMISSIONS_RESULT:

    for (String perms : permissionsToRequest) {

        if (hasPermission(perms)) {


            } else {


                permissionsRejected.add(perms);
            }
        }

        if (permissionsRejected.size() > 0) {


            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

                if
                (shouldShowRequestPermissionRationale(permissionsRejected.get(0))) {

                    showMessageOKCancel("These permissions are mandatory for the
application. Please allow access.",

                        new DialogInterface.OnClickListener() {

                            @Override

                            public void onClick(DialogInterface dialog, int which) {

                                if (Build.VERSION.SDK_INT >=
Build.VERSION_CODES.M) {

                                    //Log.d("API123", "permissionrejected " +
permissionsRejected.size());
```



```

        requestPermissions(permissionsRejected.toArray(new
String[permissionsRejected.size()]), ALL_PERMISSIONS_RESULT);

    }

}

});

return;

}

}

}

break;

}

}

}

```

---

27.

## Android Google Maps Example Code

The MainActivity.java is defined as below:

```

package com.journaldev.MapsInAction;

import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;

```

```

import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MainActivity extends AppCompatActivity implements
OnMapReadyCallback {

    SupportMapFragment mapFragment;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);

        FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                mapFragment.getMapAsync(new OnMapReadyCallback() {
                    @Override
                    public void onMapReady(GoogleMap googleMap) {
                        googleMap.setMapType(GoogleMap.MAP_TYPE_TERRAIN);

                        googleMap.addMarker(new MarkerOptions()
                            .position(new LatLng(37.4233438, -122.0728817))
                            .title("LinkedIn"))

                        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN));

                        googleMap.addMarker(new MarkerOptions()
                            .position(new LatLng(37.4629101, -122.2449094))
                            .title("Facebook")
                            .snippet("Facebook HQ: Menlo Park"));

                        googleMap.addMarker(new MarkerOptions()
                            .position(new LatLng(37.3092293, -122.1136845))
                            .title("Apple"));

                        googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(new
                            LatLng(37.4233438, -122.0728817), 10));
                    }
                });
            }
        });
    }
}

```

```

    }
});

}

@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(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}

@Override
public void onMapReady(GoogleMap googleMap) {

    googleMap.addMarker(new MarkerOptions()
        .position(new LatLng(37.4233438, -122.0728817))
        .title("LinkedIn")

        .icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN)));

    googleMap.addMarker(new MarkerOptions()
        .position(new LatLng(37.4629101, -122.2449094))
        .title("Facebook")
        .snippet("Facebook HQ: Menlo Park"));

    googleMap.addMarker(new MarkerOptions()
        .position(new LatLng(37.3092293, -122.1136845))
        .title("Apple"));

    googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(new
    LatLng(37.4233438, -122.0728817), 10));
}
}

```

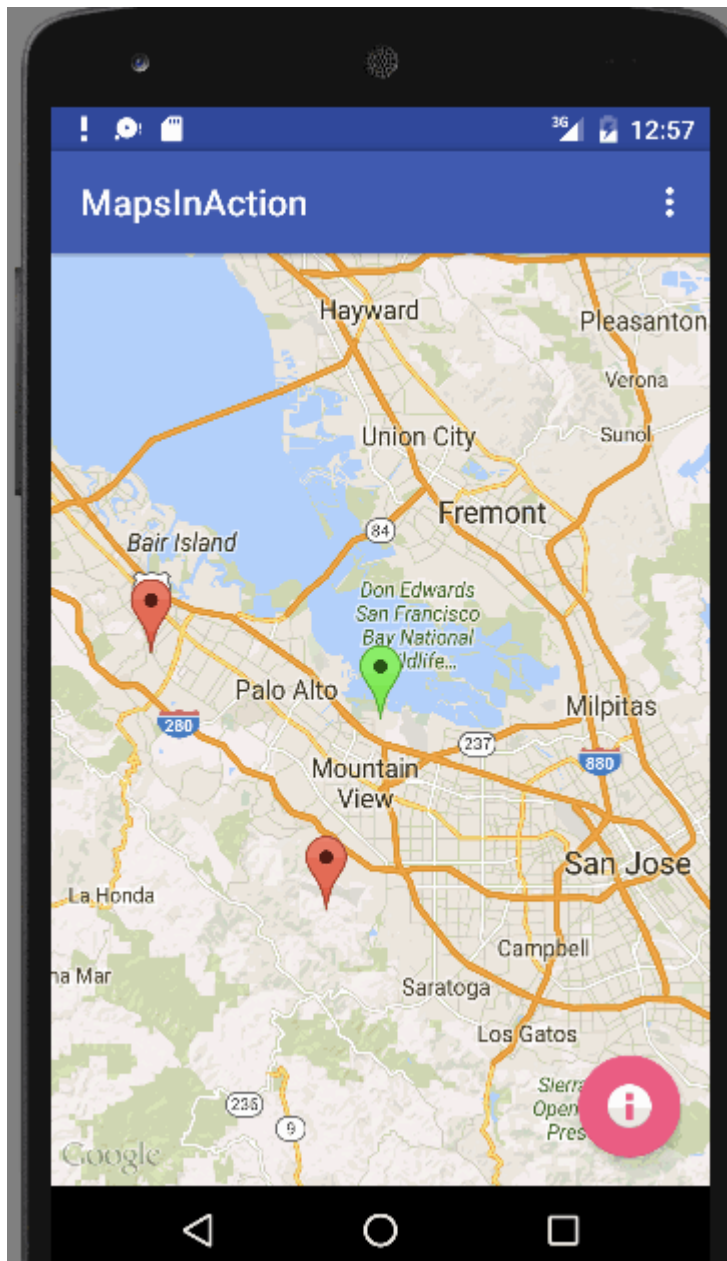
We call `getMapAsync()` on the `SupportMapFragment` object to register the callback. The `FloatingActionButton` invokes a new `OnMapReadyCallback` method with a different map type.

The `content_main.xml` contains the `MapFragment` as shown below:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context="com.journaldev.MapsInAction.MainActivity"
    tools:showIn="@layout/activity_main">

    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_gravity="center"
        android:layout_height="match_parent"
        />

</RelativeLayout>
```



## 28. Google Static Maps Android Code

```
compile 'com.pkmmte.view:circularimageview:1.1'
```

To send and receive the http url and response you need to add a few jar files in the project.

- httpclient-4.3.3.jar
- httpcore-4.3.3.jar
- httpmime-4.3.3.jar

Sync the gradle dependencies to add the libraries in the project. On running this project now a `DuplicateFileException` might arise in the `build.gradle`. It's due to conflicting package files of the libraries. A workaround is adding the following in the android tag of the build.gradle file.

```

packagingOptions {
    exclude 'META-INF/DEPENDENCIES.txt'
    exclude 'META-INF/DEPENDENCIES'
    exclude 'META-INF/dependencies.txt'
    exclude 'META-INF/LICENSE.txt'
    exclude 'META-INF/LICENSE'
    exclude 'META-INF/license.txt'
    exclude 'META-INF/LGPL2.1'
    exclude 'META-INF/NOTICE.txt'
    exclude 'META-INF/NOTICE'
    exclude 'META-INF/notice.txt'
}

```

The `content_main.xml` is defined as below.

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:background="@android:color/black"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context="com.journaldev.staticmaps.MainActivity"
    tools:showIn="@layout/activity_main">

    <com.pkmmte.view.CircularImageView
        android:layout_width="250dp"
        android:layout_height="250dp"
        android:clickable="true"
        android:id="@+id/img_map_route"
        android:layout_gravity="center"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true" />

</RelativeLayout>

```

The `MainActivity.java` looks like below.

```

package com.journaldev.staticmaps;

import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;

```

```

import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;

import com.pkmmte.view.CircularImageView;

import org.apache.http.HttpResponse;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.DefaultHttpClient;

import java.io.InputStream;
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;

public class MainActivity extends AppCompatActivity {

    private String STATIC_MAP_API_ENDPOINT =
"http://maps.googleapis.com/maps/api/staticmap?size=230x200&path=";
    String path;

    CircularImageView iv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        iv=(CircularImageView)findViewById(R.id.img_map_route);

        FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)
                    .setAction("Action", null).show();
            }
        });

        try {

            String marker_me = "color:orange|label:1|Brisbane";
            String marker_dest = "color:orange|label:7|San Francisco,USA";

```

```

        marker_me = URLEncoder.encode(marker_me, "UTF-8");

        marker_dest = URLEncoder.encode(marker_dest, "UTF-8");
        path = "weight:3|color:blue|geodesic:true|Brisbane,Australia|Hong
Kong|Moscow,Russia|London,UK|Reyjavik,Iceland|New York,USA|San
Francisco,USA";
        path = URLEncoder.encode(path, "UTF-8");

        STATIC_MAP_API_ENDPOINT = STATIC_MAP_API_ENDPOINT + path +
"&markers=" + marker_me + "&markers=" + marker_dest;

        Log.d("STATICMAPS", STATIC_MAP_API_ENDPOINT);

        AsyncTask<Void, Void, Bitmap> setImageFromUrl = new AsyncTask<Void,
Void, Bitmap>(){
            @Override
            protected Bitmap doInBackground(Void... params) {
                Bitmap bmp = null;
                HttpClient httpclient = new DefaultHttpClient();
                HttpGet request = new HttpGet(STATIC_MAP_API_ENDPOINT);

                InputStream in = null;
                try {
                    HttpResponse response = httpclient.execute(request);
                    in = response.getEntity().getContent();
                    bmp = BitmapFactory.decodeStream(in);
                    in.close();
                } catch (Exception e) {
                    e.printStackTrace();
                }
                return bmp;
            }
            protected void onPostExecute(Bitmap bmp) {
                if (bmp!=null) {

                    iv.setImageBitmap(bmp);

                }

            }
        };

        setImageFromUrl.execute();

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

@Override
public boolean onCreateOptionsMenu(Menu menu) {

```

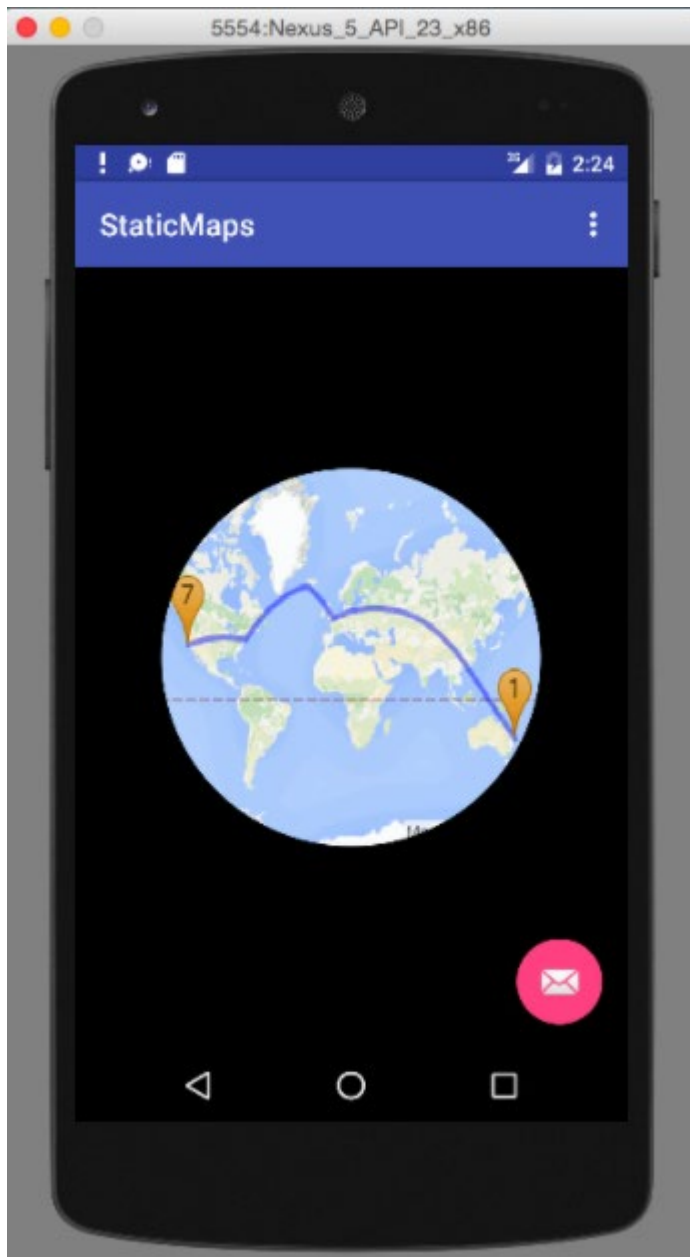


```
// Inflate the menu; this adds items to the action bar if it is present.
getMenuInflater().inflate(R.menu.menu_main, menu);
return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}
}
```



---

30.

### **Android Google Maps Current Location, Night Mode Features**

#### **Android Google Maps Current Location**

Before we start implementing some cool android google maps features in our application, add the Google Maps v2 API key value in the `meta-data` tag in the `AndroidManifest.xml` file.

Create a new project in Android Studio and select the template as Google Maps Activity.

**Note:** Google Play Services dependency will be added by default for this template.

Implement Google Play Location Services in your `MapsActivity.java` class as shown below.

```
public class MapsActivity extends FragmentActivity implements
    OnMapReadyCallback, GoogleApiClient.ConnectionCallbacks,
    GoogleApiClient.OnConnectionFailedListener, LocationListener {

    private GoogleMap mMap;

    Location mLocation;
    GoogleApiClient mGoogleApiClient;
    private static final int PLAY_SERVICES_RESOLUTION_REQUEST = 9000;

    private LocationRequest mLocationRequest;
    private long UPDATE_INTERVAL = 15000; /* 15 secs */
    private long FASTEST_INTERVAL = 5000; /* 5 secs */

    private ArrayList permissionsToRequest;
    private ArrayList permissionsRejected = new ArrayList();
    private ArrayList permissions = new ArrayList();

    private final static int ALL_PERMISSIONS_RESULT = 101;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when the map is ready to
        be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
            getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);

        permissions.add(ACCESS_FINE_LOCATION);
        permissions.add(ACCESS_COARSE_LOCATION);

        permissionsToRequest = findUnAskedPermissions(permissions);
        //get the permissions we have asked for before but are not granted..
        //we will store this in a global list to access later.

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

            if (permissionsToRequest.size() > 0)
                requestPermissions(permissionsToRequest.toArray(new
                    String[permissionsToRequest.size()]), ALL_PERMISSIONS_RESULT);
        }
    }
}
```

```

    mGoogleApiClient = new GoogleApiClient.Builder(this)
        .addApi(LocationServices.API)
        .addConnectionCallbacks(this)
        .addOnConnectionFailedListener(this)
        .build();

    connectClient();
}

/**
 * Manipulates the map once available.
 * This callback is triggered when the map is ready to be used.
 * This is where we can add markers or lines, add listeners or move the camera. In
this case,
 * we just add a marker near Sydney, Australia.
 * If Google Play services is not installed on the device, the user will be prompted to
install
 * it inside the SupportMapFragment. This method will only be triggered once the
user has
 * installed Google Play services and returned to the app.
 */
@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;

    if (ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_FINE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED) {
        // TODO: Consider calling
        //    ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        //    public void onRequestPermissionsResult(int requestCode, String[]
permissions,
        //                                     int[] grantResults)
        // to handle the case where the user grants the permission. See the
documentation
        // for ActivityCompat#requestPermissions for more details.
        return;
    }
    mMap.setMyLocationEnabled(true);
}

public void connectClient()
{
    mGoogleApiClient = new GoogleApiClient.Builder(this)

```

```

        .addApi(LocationServices.API)
        .addConnectionCallbacks(this)
        .addOnConnectionFailedListener(this)
        .build();
    }

    private ArrayList findUnAskedPermissions(ArrayList wanted) {
        ArrayList result = new ArrayList();

        for (String perm : wanted) {
            if (!hasPermission(perm)) {
                result.add(perm);
            }
        }

        return result;
    }

    @Override
    protected void onStart() {
        super.onStart();
        if (mGoogleApiClient != null) {
            mGoogleApiClient.connect();
        }
    }

    @Override
    protected void onResume() {
        super.onResume();

        if (!checkPlayServices()) {
            Toast.makeText(getApplicationContext(), "Please install google play
services", Toast.LENGTH_LONG).show();
        }
    }

    @Override
    public void onConnected(@Nullable Bundle bundle) {

        if (ActivityCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            // TODO: Consider calling
            //    ActivityCompat#requestPermissions
            // here to request the missing permissions, and then overriding
            //    public void onRequestPermissionsResult(int requestCode, String[]
permissions,

```

```

        //                                int[] grantResults)
        // to handle the case where the user grants the permission. See the
documentation
        // for ActivityCompat#requestPermissions for more details.
        return;
    }
    mLocation =
LocationServices.FusedLocationApi.getLastLocation(mGoogleApiClient);
    startLocationUpdates();

}

@Override
public void onConnectionSuspended(int i) {

}

@Override
public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

}

@Override
public void onLocationChanged(Location location) {

}

private boolean checkPlayServices() {
    GoogleApiAvailability apiAvailability = GoogleApiAvailability.getInstance();
    int resultCode = apiAvailability.isGooglePlayServicesAvailable(this);
    if (resultCode != ConnectionResult.SUCCESS) {
        if (apiAvailability.isUserResolvableError(resultCode)) {
            apiAvailability.getErrorDialog(this, resultCode,
PLAY_SERVICES_RESOLUTION_REQUEST)
                .show();
        } else {
            finish();
        }
        return false;
    }
    return true;
}

protected void startLocationUpdates() {
    mLocationRequest = new LocationRequest();

mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    mLocationRequest.setInterval(UPDATE_INTERVAL);
    mLocationRequest.setFastestInterval(FATEST_INTERVAL);
}

```

```

        if (ActivityCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            Toast.makeText(getApplicationContext(), "Enable Permissions",
Toast.LENGTH_LONG).show();
        }

        LocationServices.FusedLocationApi.requestLocationUpdates(
            mGoogleApiClient, mLocationRequest, this);

    }

    private boolean hasPermission(String permission) {
        if (canMakeSmoers()) {
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
                return (checkSelfPermission(permission) ==
PackageManager.PERMISSION_GRANTED);
            }
        }
        return true;
    }

    private boolean canMakeSmoers() {
        return (Build.VERSION.SDK_INT >
Build.VERSION_CODES.LOLLIPOP_MR1);
    }

    @TargetApi(Build.VERSION_CODES.M)
    @Override
    public void onRequestPermissionsResult(int requestCode, String[] permissions,
int[] grantResults) {

        switch (requestCode) {

            case ALL_PERMISSIONS_RESULT:
                for (String perms : permissionsToRequest) {
                    if (!hasPermission(perms)) {
                        permissionsRejected.add(perms);
                    }
                }

                if (permissionsRejected.size() > 0) {

                    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

```

```

        if
        (shouldShowRequestPermissionRationale(permissionsRejected.get(o))) {
            showMessageOKCancel("These permissions are mandatory for the
            application. Please allow access.",
            new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    if (Build.VERSION.SDK_INT >=
                    Build.VERSION_CODES.M) {
                        requestPermissions(permissionsRejected.toArray(new
                        String[permissionsRejected.size()]), ALL_PERMISSIONS_RESULT);
                    }
                }
            });
        }
        return;
    }
}

}

break;
}

}

private void showMessageOKCancel(String message,
DialogInterface.OnClickListener okListener) {
    new AlertDialog.Builder(MapsActivity.this)
        .setMessage(message)
        .setPositiveButton("OK", okListener)
        .setNegativeButton("Cancel", null)
        .create()
        .show();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    stopLocationUpdates();
}

public void stopLocationUpdates()
{
    if (mGoogleApiClient.isConnected()) {
        LocationServices.FusedLocationApi
            .removeLocationUpdates(mGoogleApiClient, this);
        mGoogleApiClient.disconnect();
    }
}
}

```



In the above code `map.setMyLocationEnabled(true);` is used to show the user's current location.

The below image is the output of the application when the above code is run.



The blue dot is our current location. We need to focus the camera on the current location in the map to prevent zooming and scrolling manually.

Change the `onConnected()` method as;

```
@Override
public void onConnected(@Nullable Bundle bundle) {

    if (ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_FINE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
        android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED) {
        // TODO: Consider calling
        //    ActivityCompat#requestPermissions
        // here to request the missing permissions, and then overriding
        //    public void onRequestPermissionsResult(int requestCode, String[]
permissions,
        //                                     int[] grantResults)
        // to handle the case where the user grants the permission. See the
documentation
        // for ActivityCompat#requestPermissions for more details.
```

```
        return;
    }
    mLocation =
    LocationServices.FusedLocationApi.getLastLocation(mGoogleApiClient);

    LatLng latLng = new LatLng(mLocation.getLatitude(),
    mLocation.getLongitude());
    CameraUpdate cameraUpdate =
    CameraUpdateFactory.newLatLngZoom(latLng, 12);
    mMap.animateCamera(cameraUpdate);

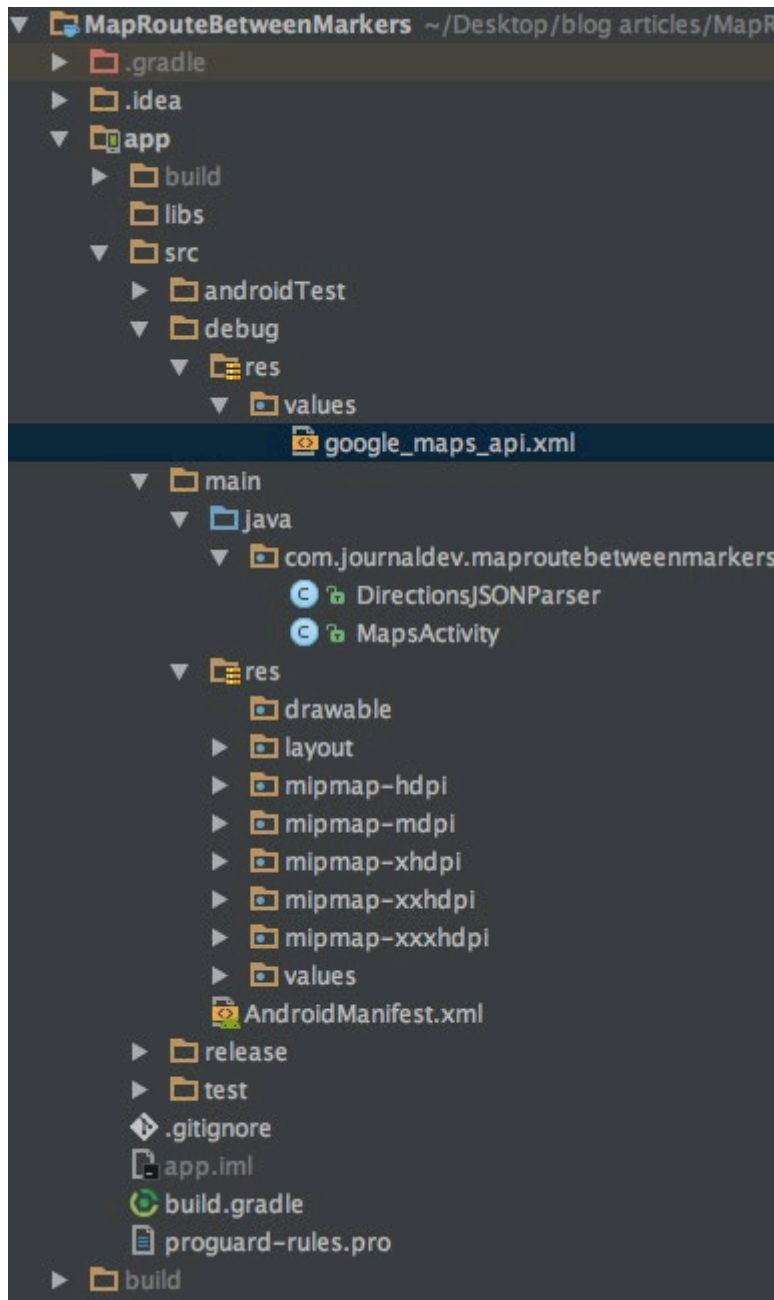
    startLocationUpdates();
}
```

In the above code **12** is the zoom level set. We can set the minimum and maximum zoom level using `mMap.setMinZoomPreference(float v);` and `mMap.setMaxZoomPreference(float v);`.

---

32.

## Android Google Maps Drawing Path Project Structure



The **DirectionsJSONParser.java** file is the one that parses the locations and returns the route. **decodePoly()** method is then invoked to get the polyline data that's later drawn on the map.

## Android Google Maps Drawing Route Code

The **MainActivity.java** code is given below.

```
public class MapsActivity extends FragmentActivity implements  
OnMapReadyCallback {
```

```

private GoogleMap mMap;
ArrayList markerPoints= new ArrayList();

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);
    // Obtain the SupportMapFragment and get notified when the map is ready to
    be used.
    SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
    .findFragmentById(R.id.map);
    mapFragment.getMapAsync(this);
}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    LatLng sydney = new LatLng(-34, 151);
    //mMap.addMarker(new MarkerOptions().position(sydney).title("Marker in
Sydney"));
    mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(sydney, 16));

    mMap.setOnMapClickListener(new GoogleMap.OnMapClickListener() {
        @Override
        public void onMapClick(LatLng latLng) {

            if (markerPoints.size() > 1) {
                markerPoints.clear();
                mMap.clear();
            }

            // Adding new item to the ArrayList
            markerPoints.add(latLng);

            // Creating MarkerOptions
            MarkerOptions options = new MarkerOptions();

            // Setting the position of the marker
            options.position(latLng);

            if (markerPoints.size() == 1) {

options.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HU
E_GREEN));
            } else if (markerPoints.size() == 2) {

options.icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HU
E_RED));
            }

```

```

        // Add new marker to the Google Map Android API V2
        mMap.addMarker(options);

        // Checks, whether start and end locations are captured
        if (markerPoints.size() >= 2) {
            LatLng origin = (LatLng) markerPoints.get(0);
            LatLng dest = (LatLng) markerPoints.get(1);

            // Getting URL to the Google Directions API
            String url = getDirectionsUrl(origin, dest);

            DownloadTask downloadTask = new DownloadTask();

            // Start downloading json data from Google Directions API
            downloadTask.execute(url);
        }
    }
});
}

private class DownloadTask extends AsyncTask {

    @Override
    protected String doInBackground(String... url) {

        String data = "";

        try {
            data = downloadUrl(url[0]);
        } catch (Exception e) {
            Log.d("Background Task", e.toString());
        }
        return data;
    }

    @Override
    protected void onPostExecute(String result) {
        super.onPostExecute(result);

        ParserTask parserTask = new ParserTask();

        parserTask.execute(result);
    }
}

```

```

private class ParserTask extends AsyncTask<String, Integer,
List<List<HashMap>>> {

    // Parsing the data in non-ui thread
    @Override
    protected List<List<HashMap>> doInBackground(String... jsonData) {

        JSONObject jObject;
        List<List<HashMap>> routes = null;

        try {
            jObject = new JSONObject(jsonData[0]);
            DirectionsJSONParser parser = new DirectionsJSONParser();

            routes = parser.parse(jObject);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return routes;
    }

    @Override
    protected void onPostExecute(List<List<HashMap>> result) {
        ArrayList points = null;
        PolylineOptions lineOptions = null;
        MarkerOptions markerOptions = new MarkerOptions();

        for (int i = 0; i < result.size(); i++) {
            points = new ArrayList();
            lineOptions = new PolylineOptions();

            List<HashMap> path = result.get(i);

            for (int j = 0; j < path.size(); j++) {
                HashMap point = path.get(j);

                double lat = Double.parseDouble(point.get("lat"));
                double lng = Double.parseDouble(point.get("lng"));
                LatLng position = new LatLng(lat, lng);

                points.add(position);
            }

            lineOptions.addAll(points);
            lineOptions.width(12);
            lineOptions.color(Color.RED);
            lineOptions.geodesic(true);
        }

        // Drawing polyline in the Google Map for the i-th route
    }
}

```

```

        mMap.addPolyline(lineOptions);
    }
}

private String getDirectionsUrl(LatLng origin, LatLng dest) {

    // Origin of route
    String str_origin = "origin=" + origin.latitude + "," + origin.longitude;

    // Destination of route
    String str_dest = "destination=" + dest.latitude + "," + dest.longitude;

    // Sensor enabled
    String sensor = "sensor=false";
    String mode = "mode=driving";

    // Building the parameters to the web service
    String parameters = str_origin + "&" + str_dest + "&" + sensor + "&" + mode;

    // Output format
    String output = "json";

    // Building the url to the web service
    String url = "https://maps.googleapis.com/maps/api/directions/" + output +
    "?" + parameters;

    return url;
}

private String downloadUrl(String strUrl) throws IOException {
    String data = "";
    InputStream iStream = null;
    HttpURLConnection urlConnection = null;
    try {
        URL url = new URL(strUrl);

        urlConnection = (HttpURLConnection) url.openConnection();

        urlConnection.connect();

        iStream = urlConnection.getInputStream();

        BufferedReader br = new BufferedReader(new InputStreamReader(iStream));

        StringBuffer sb = new StringBuffer();

        String line = "";
        while ((line = br.readLine()) != null) {
            sb.append(line);
        }
    }
}

```

```

        data = sb.toString();

        br.close();

    } catch (Exception e) {
        Log.d("Exception", e.toString());
    } finally {
        iStream.close();
        urlConnection.disconnect();
    }
    return data;
}
}

```

We've called an **onMapClickListener** on the google map object. It's used to set a marker on the clicked location and store that location in an ArrayList. The ArrayList is used to store the source and destination markers only.

The **getDirectionsUrl()** is called the Directions API URL with the output and parameters as shown below.

```

"https://maps.googleapis.com/maps/api/directions/" + output + "?" +
parameters;

```

The output variable holds a "json" string and the parameter string is created as:

```

String parameters = str_origin + "&" + str_dest + "&" + sensor + "&" +
mode;

```

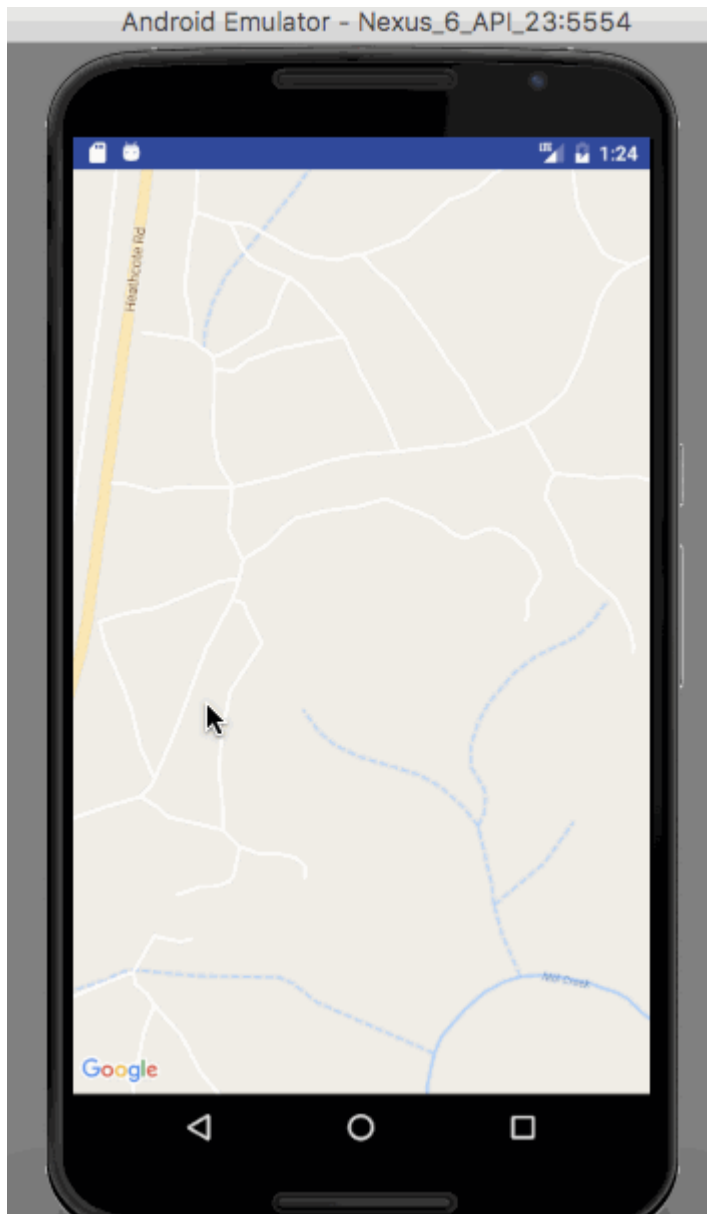
We've set the **mode=driving** in the current application.

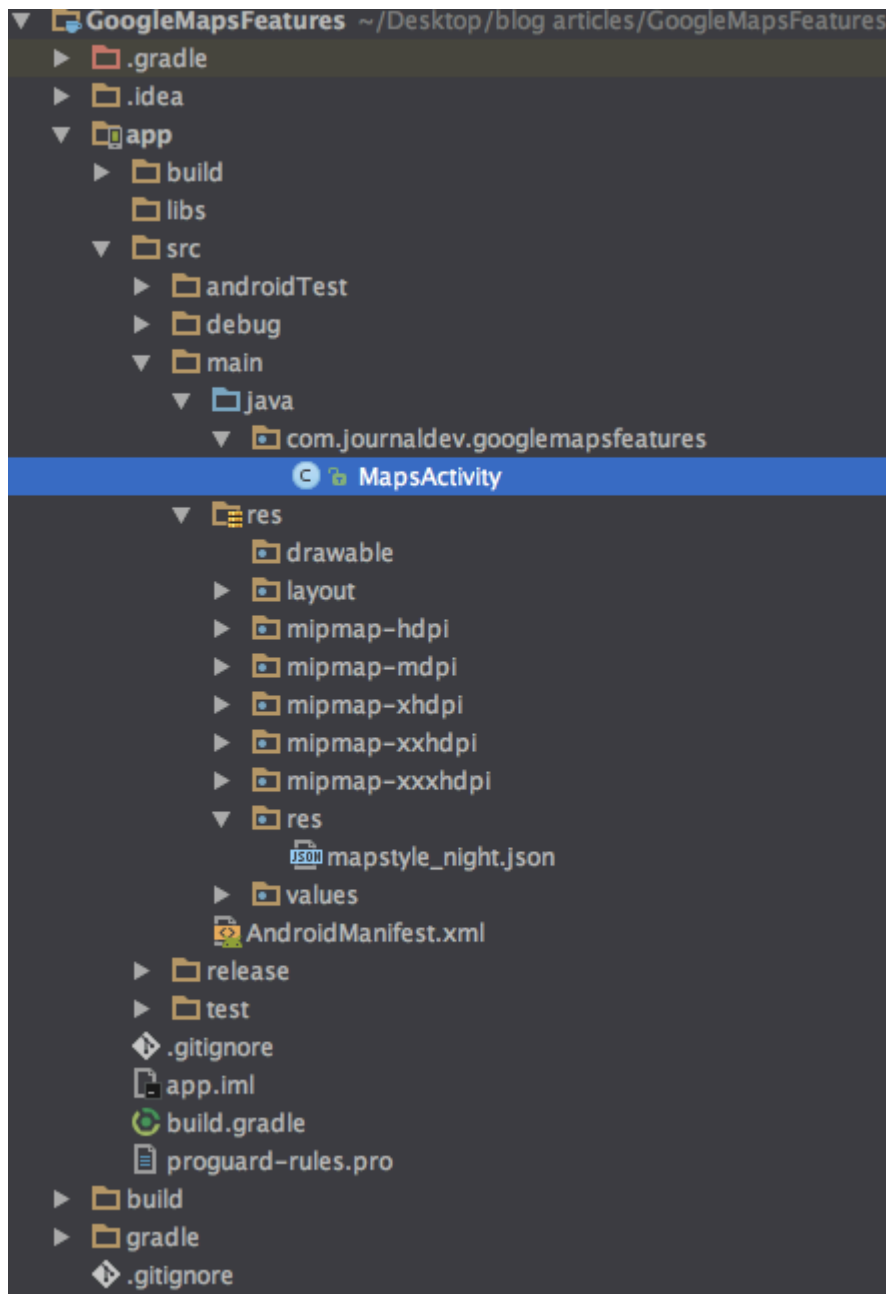
The other modes of transport are:

- driving (default)
- walking
- bicycling
- transit

The output of the application is given below:







To enable night mode in the apps. We need to set the map style in the **onMapReady** method as;  
`mMap.setMapStyle(MapStyleOptions.loadRawResourceStyle(this, R.raw.mapstyle_night));`  
The **mapstyle\_night.json** code is shown below.

```
[
  {
    "featureType": "all",
    "elementType": "geometry",
    "stylers": [
      {
        "color": "#242f3e"
      }
    ]
  }
]
```

```
]
},
{
  "featureType": "all",
  "elementType": "labels.text.stroke",
  "stylers": [
    {
      "lightness": -80
    }
  ]
},
{
  "featureType": "administrative",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#746855"
    }
  ]
},
{
  "featureType": "administrative.locality",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#d59563"
    }
  ]
},
{
  "featureType": "poi",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#d59563"
    }
  ]
},
{
  "featureType": "poi.park",
  "elementType": "geometry",
  "stylers": [
    {
      "color": "#263c3f"
    }
  ]
},
{
  "featureType": "poi.park",
  "elementType": "labels.text.fill",
  "stylers": [
```

```
{
  "color": "#6b9a76"
}
],
{
  "featureType": "road",
  "elementType": "geometry.fill",
  "stylers": [
    {
      "color": "#2b3544"
    }
  ]
},
{
  "featureType": "road",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#9ca5b3"
    }
  ]
},
{
  "featureType": "road.arterial",
  "elementType": "geometry.fill",
  "stylers": [
    {
      "color": "#38414e"
    }
  ]
},
{
  "featureType": "road.arterial",
  "elementType": "geometry.stroke",
  "stylers": [
    {
      "color": "#212a37"
    }
  ]
},
{
  "featureType": "road.highway",
  "elementType": "geometry.fill",
  "stylers": [
    {
      "color": "#746855"
    }
  ]
},
{
  }
```

```
"featureType": "road.highway",
"elementType": "geometry.stroke",
"stylers": [
  {
    "color": "#1f2835"
  }
],
},
{
  "featureType": "road.highway",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#f3d19c"
    }
  ]
},
{
  "featureType": "road.local",
  "elementType": "geometry.fill",
  "stylers": [
    {
      "color": "#38414e"
    }
  ]
},
{
  "featureType": "road.local",
  "elementType": "geometry.stroke",
  "stylers": [
    {
      "color": "#212a37"
    }
  ]
},
{
  "featureType": "transit",
  "elementType": "geometry",
  "stylers": [
    {
      "color": "#2f3948"
    }
  ]
},
{
  "featureType": "transit.station",
  "elementType": "labels.text.fill",
  "stylers": [
    {
      "color": "#d59563"
    }
  ]
}
```

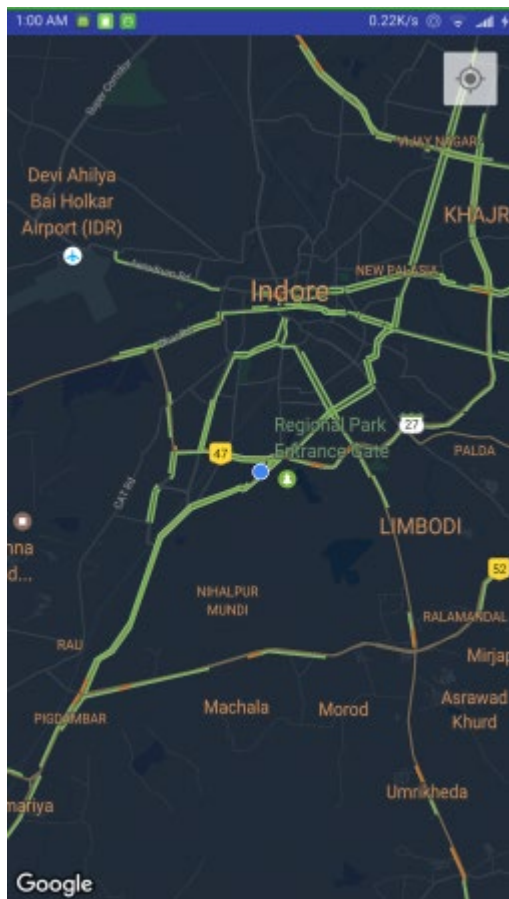
```

    ]
  },
  {
    "featureType": "water",
    "elementType": "geometry",
    "stylers": [
      {
        "color": "#17263c"
      }
    ]
  },
  {
    "featureType": "water",
    "elementType": "labels.text.fill",
    "stylers": [
      {
        "color": "#515c6d"
      }
    ]
  },
  {
    "featureType": "water",
    "elementType": "labels.text.stroke",
    "stylers": [
      {
        "lightness": -20
      }
    ]
  }
]

```

Enable traffics in the map by the following code:

```
mMap.setTrafficEnabled(true);
```



**mMap.setLatLngBoundsForCameraTarget();** is used to constrain the lat/lng center bounds of the focal point of the map (the camera target) so that users can only scroll and pan within these bounds.

To implement the above. Let's take LatLngBounds for a part of city Adelaide for example.

Following is a snippet that's put inside **onMapReady** method

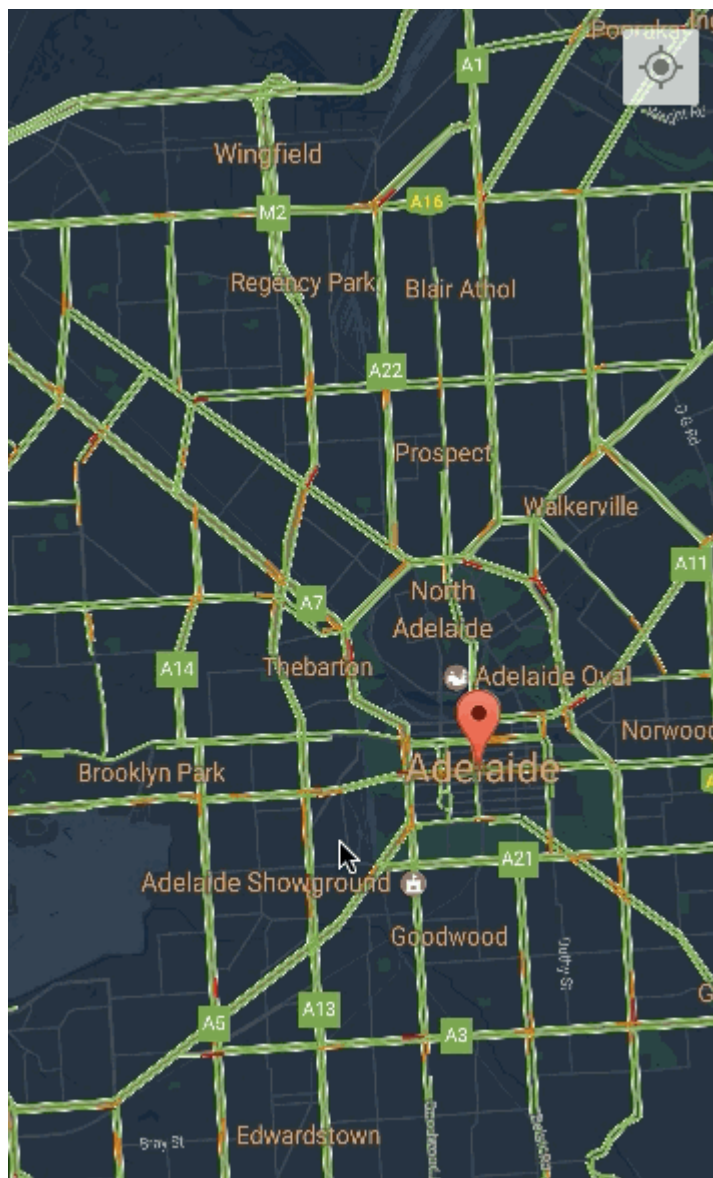
```
final LatLngBounds ADELAIDE = new LatLngBounds(
    new LatLng(-35.0, 138.58), new LatLng(-34.9, 138.61));
final CameraPosition ADELAIDE_CAMERA = new CameraPosition.Builder()
    .target(new LatLng(-34.92873,
138.59995)).zoom(20.0f).bearing(0).tilt(0).build();

mMap.setLatLngBoundsForCameraTarget(ADELAIDE);

mMap.addMarker(new MarkerOptions()
    .position(new LatLng(-34.92873, 138.59995))
    .title("My Marker"));

mMap.animateCamera(CameraUpdateFactory.newCameraPosition(ADELAIDE_CAMERA));
```

Following is the output of the application.

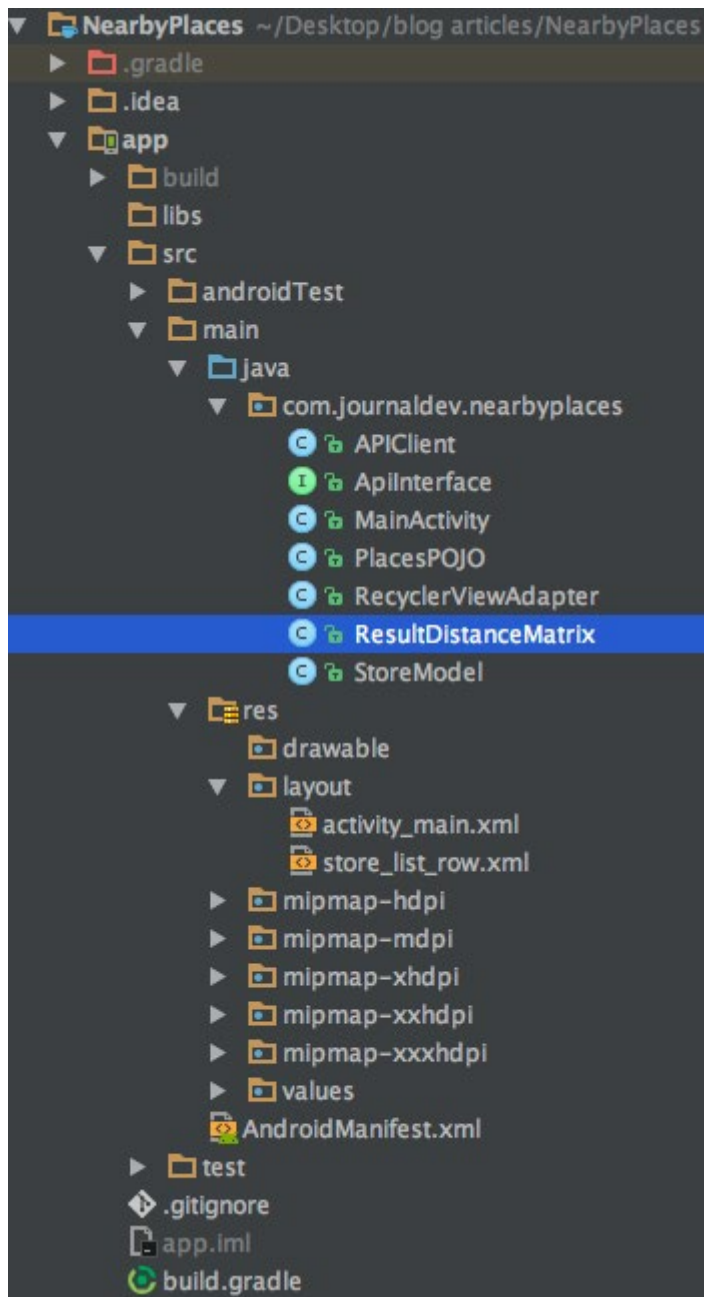


0:03 / 0:14



34.

## Google Places API Example Project Structure



The Project consists of a single Activity. An adapter class for the RecyclerView. A Model class that holds the data for each RecyclerView row. Two POJO classes for converting the JSON responses to [Gson](#) from the Google places API and Distance Matrix API. APIClient and ApiInterface for using [Retrofit](#) and the endpoints.

## Google Places API Example Code

Add the following dependencies inside the `build.gradle` file

```
compile 'com.google.android.gms:play-services-location:10.2.1'
```

```
compile 'com.google.android.gms:play-services-places:10.2.1'
compile 'com.google.code.gson:gson:2.7'
compile 'com.squareup.retrofit2:retrofit:2.1.0'
compile 'com.squareup.retrofit2:converter-gson:2.1.0'
compile 'com.squareup.okhttp3:logging-interceptor:3.4.1'
compile 'com.squareup.okhttp3:okhttp:3.4.1'
compile 'io.nlopez.smartlocation:library:3.3.1'
compile 'com.android.support:cardview-v7:25.3.0'
compile 'com.android.support:recyclerview-v7:25.3.0'
```

`compile 'io.nlopez.smartlocation:library:3.3.1'` is a LocationTracking third party **library** that reduces the boilerplate code.  
The **APIClient.java** code is given below:

```
package com.journaldev.nearbyplaces;

import java.util.concurrent.TimeUnit;
import okhttp3.OkHttpClient;
import okhttp3.logging.HttpLoggingInterceptor;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class APIClient {

    private static Retrofit retrofit = null;

    public static final String GOOGLE_PLACE_API_KEY =
"ADD_YOUR_API_KEY_HERE";

    public static String base_url = "https://maps.googleapis.com/maps/api/";

    public static Retrofit getClient() {

        HttpLoggingInterceptor interceptor = new HttpLoggingInterceptor();
        interceptor.setLevel(HttpLoggingInterceptor.Level.BODY);
        OkHttpClient client = new OkHttpClient.Builder().readTimeout(30,
TimeUnit.SECONDS).writeTimeout(30,
TimeUnit.SECONDS).addInterceptor(interceptor).build();

        retrofit = null;

        retrofit = new Retrofit.Builder()
            .baseUrl(base_url)
            .addConverterFactory(GsonConverterFactory.create())
            .client(client)
            .build();

        return retrofit;
    }
}
```

```
}
```

The **ApiInterface.java** code is given below

```
package com.journaldev.nearbyplaces;
```

```
import retrofit2.Call;  
import retrofit2.http.GET;  
import retrofit2.http.Query;
```

```
public interface ApiInterface {
```

```
    @GET("place/nearbysearch/json?")  
    Call<PlacesPOJO.Root> doPlaces(@Query(value = "type", encoded = true) String  
type, @Query(value = "location", encoded = true) String location, @Query(value =  
"name", encoded = true) String name, @Query(value = "opennow", encoded = true)  
boolean opennow, @Query(value = "rankby", encoded = true) String rankby,  
@Query(value = "key", encoded = true) String key);
```

```
    @GET("distancematrix/json") // origins/destinations: LatLng as string  
    Call<ResultDistanceMatrix> getDistance(@Query("key") String key,  
@Query("origins") String origins, @Query("destinations") String destinations);  
}
```

**PlacesPOJO.java** is the file which holds the response from Places API. Its code is given below

```
package com.journaldev.nearbyplaces;
```

```
import com.google.gson.annotations.SerializedName;  
import java.io.Serializable;  
import java.util.ArrayList;  
import java.util.List;
```

```
public class PlacesPOJO {
```

```
    public class Root implements Serializable {
```

```
        @SerializedName("results")  
        public List<CustomA> customA = new ArrayList<>();  
        @SerializedName("status")  
        public String status;  
    }
```

```
    public class CustomA implements Serializable {
```

```
        @SerializedName("geometry")  
        public Geometry geometry;  
        @SerializedName("vicinity")  
        public String vicinity;  
        @SerializedName("name")
```

```

        public String name;
    }

    public class Geometry implements Serializable{

        @SerializedName("location")
        public LocationA locationA;
    }

    public class LocationA implements Serializable {

        @SerializedName("lat")
        public String lat;
        @SerializedName("lng")
        public String lng;

    }

}

```

**ResultDistanceMatrix.java** class holds the response from Distance Matrix API. It's code is given below:

```

package com.journaldev.nearbyplaces;

import com.google.gson.annotations.SerializedName;

import java.util.List;

public class ResultDistanceMatrix {
    @SerializedName("status")
    public String status;

    @SerializedName("rows")
    public List rows;

    public class InfoDistanceMatrix {
        @SerializedName("elements")
        public List elements;

        public class DistanceElement {
            @SerializedName("status")
            public String status;
            @SerializedName("duration")
            public ValueItem duration;
            @SerializedName("distance")
            public ValueItem distance;
        }
    }
}

```

```

    }

    public class ValueItem {
        @SerializedName("value")
        public long value;
        @SerializedName("text")
        public String text;
    }
}

```

The **activity\_main.xml** file is given below

```

<?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"
    android:background="#212121"
    tools:context="com.journaldev.nearbyplaces.MainActivity">

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:textColor="@android:color/white"
        android:textColorHint="@android:color/white"
        android:text="restaurant mcdonalds"
        android:hint="type name"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_toLeftOf="@+id/button"
        android:layout_toStartOf="@+id/button" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:text="Search" />

    <android.support.v7.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/editText"
        android:scrollbars="vertical" />

</RelativeLayout>

```

The `MainActivity.java` class code is given below.

```
package com.journaldev.nearbyplaces;

import android.annotation.TargetApi;
import android.content.DialogInterface;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Build;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import com.google.android.gms.maps.model.LatLng;
import java.util.ArrayList;
import java.util.List;

import io.nlopez.smartlocation.OnLocationUpdatedListener;
import io.nlopez.smartlocation.SmartLocation;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;

import static android.Manifest.permission.ACCESS_COARSE_LOCATION;
import static android.Manifest.permission.ACCESS_FINE_LOCATION;

public class MainActivity extends AppCompatActivity {

    private ArrayList<String> permissionsToRequest;
    private ArrayList<String> permissionsRejected = new ArrayList<>();
    private ArrayList<String> permissions = new ArrayList<>();
    private final static int ALL_PERMISSIONS_RESULT = 101;
    List<StoreModel> storeModels;
    ApiInterface apiService;

    String latLngString;
    LatLng latLng;

    RecyclerView recyclerView;
    EditText editText;
    Button button;
    List<PlacesPOJO.CustomA> results;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

permissions.add(ACCESS_FINE_LOCATION);
permissions.add(ACCESS_COARSE_LOCATION);

permissionsToRequest = findUnAskedPermissions(permissions);

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {

    if (permissionsToRequest.size() > 0)
        requestPermissions(permissionsToRequest.toArray(new
String[permissionsToRequest.size()]), ALL_PERMISSIONS_RESULT);
    else {
        fetchLocation();
    }
} else {
    fetchLocation();
}

apiService = APIClient.getClient().create(ApiInterface.class);

recyclerView = (RecyclerView) findViewById(R.id.recyclerView);

recyclerView.setNestedScrollingEnabled(false);
recyclerView.setHasFixedSize(true);

LinearLayoutManager layoutManager = new LinearLayoutManager(this);
recyclerView.setLayoutManager(layoutManager);

editText = (EditText) findViewById(R.id.editText);
button = (Button) findViewById(R.id.button);

button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String s = editText.getText().toString().trim();
        String[] split = s.split("\\s+");

        if (split.length != 2) {
            Toast.makeText(getApplicationContext(), "Please enter text in the
required format", Toast.LENGTH_SHORT).show();
        } else
            fetchStores(split[0], split[1]);
    }
});

```

```

    }

    private void fetchStores(String placeType, String businessName) {

        /**
         * For Locations In India McDonalds stores aren't returned accurately
         */

        //Call<PlacesPOJO.Root> call = apiService.doPlaces(placeType,
        latLngString,"\""+ businessName + "\"", true, "distance",
        APIClient.GOOGLE_PLACE_API_KEY);

        Call<PlacesPOJO.Root> call = apiService.doPlaces(placeType, latLngString,
        businessName, true, "distance", APIClient.GOOGLE_PLACE_API_KEY);
        call.enqueue(new Callback<PlacesPOJO.Root>() {
            @Override
            public void onResponse(Call<PlacesPOJO.Root> call,
            Response<PlacesPOJO.Root> response) {
                PlacesPOJO.Root root = response.body();

                if (response.isSuccessful()) {

                    if (root.status.equals("OK")) {

                        results = root.customA;
                        storeModels = new ArrayList<>();
                        for (int i = 0; i < results.size(); i++) {

                            if (i == 10)
                                break;
                            PlacesPOJO.CustomA info = results.get(i);

                            fetchDistance(info);

                        }

                    } else {
                        Toast.makeText(getApplicationContext(), "No matches found near
                        you", Toast.LENGTH_SHORT).show();
                    }

                } else if (response.code() != 200) {
                    Toast.makeText(getApplicationContext(), "Error " + response.code() + "
                    found.", Toast.LENGTH_SHORT).show();
                }

            }

        })
    }

```



```

        @Override
        public void onFailure(Call<PlacesPOJO.Root> call, Throwable t) {
            // Log error here since request failed
            call.cancel();
        }
    });

}

private ArrayList<String> findUnAskedPermissions(ArrayList<String> wanted) {
    ArrayList<String> result = new ArrayList<>();

    for (String perm : wanted) {
        if (!hasPermission(perm)) {
            result.add(perm);
        }
    }

    return result;
}

private boolean hasPermission(String permission) {
    if (canMakeSmores()) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            return (checkSelfPermission(permission) ==
PackageManager.PERMISSION_GRANTED);
        }
    }
    return true;
}

private boolean canMakeSmores() {
    return (Build.VERSION.SDK_INT >
Build.VERSION_CODES.LOLLIPOP_MR1);
}

@TargetApi(Build.VERSION_CODES.M)
@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions,
int[] grantResults) {

    switch (requestCode) {

        case ALL_PERMISSIONS_RESULT:
            for (String perms : permissionsToRequest) {
                if (!hasPermission(perms)) {
                    permissionsRejected.add(perms);
                }
            }

```

```

    }

    if (permissionsRejected.size() > 0) {

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
            if
            (shouldShowRequestPermissionRationale(permissionsRejected.get(0))) {
                showMessageOKCancel("These permissions are mandatory for the
                application. Please allow access.",
                new DialogInterface.OnClickListener() {
                    @Override
                    public void onClick(DialogInterface dialog, int which) {
                        if (Build.VERSION.SDK_INT >=
                        Build.VERSION_CODES.M) {
                            requestPermissions(permissionsRejected.toArray(new
                            String[permissionsRejected.size()]), ALL_PERMISSIONS_RESULT);
                        }
                    }
                });
            }
            return;
        }
    }

    } else {
        fetchLocation();
    }

    break;
}

}

private void showMessageOKCancel(String message,
DialogInterface.OnClickListener okListener) {
    new AlertDialog.Builder(MainActivity.this)
        .setMessage(message)
        .setPositiveButton("OK", okListener)
        .setNegativeButton("Cancel", null)
        .create()
        .show();
}

private void fetchLocation() {

    SmartLocation.with(this).location()
        .oneFix()
        .start(new OnLocationUpdatedListener() {
            @Override
            public void onLocationUpdated(Location location) {
                latLngString = location.getLatitude() + "," + location.getLongitude();
            }
        });
}

```

```

        latLng = new LatLng(location.getLatitude(), location.getLongitude());
    }
});
}

private void fetchDistance(final PlacesPOJO.CustomA info) {

    Call<ResultDistanceMatrix> call =
    apiService.getDistance(APIClient.GOOGLE_PLACE_API_KEY, latLngString,
    info.geometry.locationA.lat + "," + info.geometry.locationA.lng);
    call.enqueue(new Callback<ResultDistanceMatrix>() {
        @Override
        public void onResponse(Call<ResultDistanceMatrix> call,
        Response<ResultDistanceMatrix> response) {

            ResultDistanceMatrix resultDistance = response.body();
            if ("OK".equalsIgnoreCase(resultDistance.status)) {

                ResultDistanceMatrix.InfoDistanceMatrix infoDistanceMatrix =
                resultDistance.rows.get(0);
                ResultDistanceMatrix.InfoDistanceMatrix.DistanceElement
                distanceElement = infoDistanceMatrix.elements.get(0);
                if ("OK".equalsIgnoreCase(distanceElement.status)) {
                    ResultDistanceMatrix.InfoDistanceMatrix.ValueItem itemDuration =
                    distanceElement.duration;
                    ResultDistanceMatrix.InfoDistanceMatrix.ValueItem itemDistance =
                    distanceElement.distance;
                    String totalDistance = String.valueOf(itemDistance.text);
                    String totalDuration = String.valueOf(itemDuration.text);

                    storeModels.add(new StoreModel(info.name, info.vicinity,
                    totalDistance, totalDuration));

                    if (storeModels.size() == 10 || storeModels.size() == results.size()) {
                        RecyclerViewAdapter adapterStores = new
                        RecyclerViewAdapter(results, storeModels);
                        recyclerView.setAdapter(adapterStores);
                    }
                }
            }
        }
    });

    @Override
    public void onFailure(Call<ResultDistanceMatrix> call, Throwable t) {
        call.cancel();
    }
});

```

```
}  
}
```

In the above code, we start by asking for runtime permissions followed by fetching the current location using the SmartLocation Library.

Once we have that in place, we pass the first word from the EditText in the type and the second word in the name parameter of the `fetchStores()` method that eventually calls the Google Places API web service. We limit the search results to 10.

For each result, we calculate the distance and time from the store inside the method `fetchDistance()`. Once it's done for all the stores, we populate the data inside the `RecyclerViewAdapter.java` class using a `StoreModel.java` data class.

**StoreModel.java** code is given below:

```
package com.journaldev.nearbyplaces;  
  
public class StoreModel {  
  
    public String name, address, distance, duration;  
  
    public StoreModel(String name, String address, String distance, String duration) {  
  
        this.name = name;  
        this.address = address;  
        this.distance = distance;  
        this.duration = duration;  
    }  
  
}
```

The layout for each row of the RecyclerView is given in the xml below:

**store\_list\_row.xml**

```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_marginBottom="@dimen/activity_horizontal_margin"  
    android:orientation="vertical">  
  
    <android.support.v7.widget.CardView  
        xmlns:card_view="http://schemas.android.com/apk/res-auto"  
        android:id="@+id/card_view"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        card_view:cardCornerRadius="odp"  
        card_view:cardElevation="5dp">  
  
        <LinearLayout  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"
```

```

        android:orientation="vertical"
        android:padding="5dp">

        <TextView
            android:id="@+id/txtStoreName"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:paddingBottom="5dp"
            android:textColor="#212121" />

        <TextView
            android:id="@+id/txtStoreAddr"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:paddingBottom="5dp"
            android:textColor="#212121" />

        <TextView
            android:id="@+id/txtStoreDist"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:paddingBottom="5dp" />

```

```

</LinearLayout>

```

```

</android.support.v7.widget.CardView>

```

```

</LinearLayout>

```

The **RecyclerViewAdapter.java** code is given below.

```

public class RecyclerViewAdapter extends
RecyclerView.Adapter<RecyclerViewAdapter.MyViewHolder> {

    private List<PlacesPOJO.CustomA> stLstStores;
    private List<StoreModel> models;

    public RecyclerViewAdapter(List<PlacesPOJO.CustomA> stores,
List<StoreModel> storeModels) {

        stLstStores = stores;
        models = storeModels;
    }

    @Override
    public MyViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {
        final View view = LayoutInflater.from(parent.getContext())
            .inflate(R.layout.store_list_row, parent, false);
    }

```

```

        return new MyViewHolder(view);
    }

    @Override
    public void onBindViewHolder(MyViewHolder holder, int position) {

        holder.setData(stLstStores.get(holder.getAdapterPosition()), holder,
models.get(holder.getAdapterPosition()));
    }

    @Override
    public int getItemCount() {
        return Math.min(5, stLstStores.size());
    }

    public class MyViewHolder extends RecyclerView.ViewHolder {

        TextView txtStoreName;
        TextView txtStoreAddr;
        TextView txtStoreDist;
        StoreModel model;

        public MyViewHolder(View itemView) {
            super(itemView);

            this.txtStoreDist = (TextView) itemView.findViewById(R.id.txtStoreDist);
            this.txtStoreName = (TextView) itemView.findViewById(R.id.txtStoreName);
            this.txtStoreAddr = (TextView) itemView.findViewById(R.id.txtStoreAddr);

        }

        public void setData(PlacesPOJO.CustomA info, MyViewHolder holder,
StoreModel storeModel) {

            this.model = storeModel;

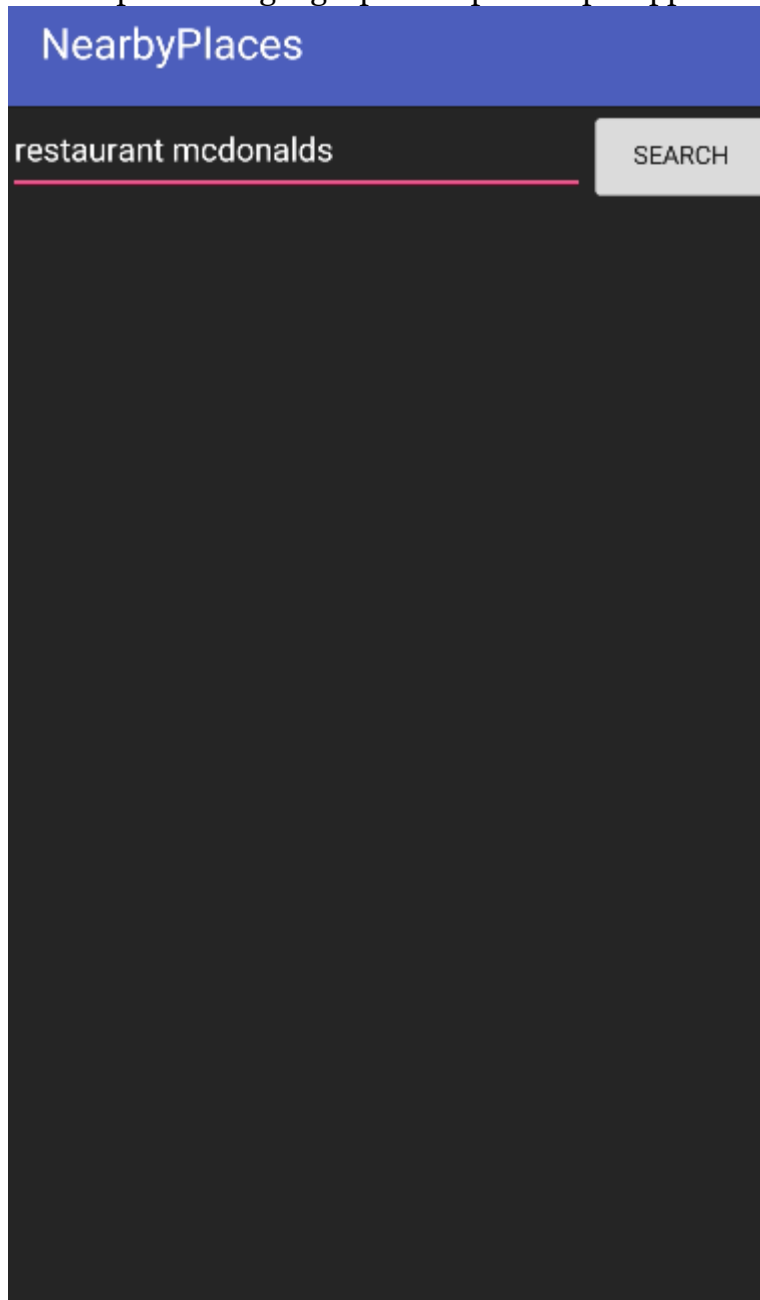
            holder.txtStoreDist.setText(model.distance + "\n" + model.duration);
            holder.txtStoreName.setText(info.name);
            holder.txtStoreAddr.setText(info.vicinity);

        }
    }

```

```
}  
}
```

The output of the google places api example application in action is given below:



---

---

### 37. Android Countdown Timer Code

`activity_main.xml`

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">
```

```
<ProgressBar
    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyleHorizontal"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:indeterminate="false"
    android:max="10"
    android:minHeight="50dp"
    android:minWidth="200dp"
    android:progress="0"
    android:layout_centerVertical="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
```



```
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
```

```
<Button
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Start Timer"
    android:id="@+id/button"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="61dp" />
```

```
<Button
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Stop Timer"
    android:id="@+id/button2"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="46dp"
    android:layout_below="@+id/progressBar" />
```

```
</RelativeLayout>
```

The MainActivity.java is given below :

```
package com.journaldev.countdowntimer;
```

```
import android.os.CountDownTimer;
```

```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.ProgressBar;
import android.widget.TextView;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    ProgressBar progressBar;
    Button start_timer, stop_timer;
    MyCountDownTimer myCountDownTimer;
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
        progressBar=(ProgressBar)findViewById(R.id.progressBar);
        start_timer=(Button)findViewById(R.id.button);
        stop_timer=(Button)findViewById(R.id.button2);
```

```
        start_timer.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
```

```
            public void onClick(View v) {
```

```

        myCountDownTimer = new MyCountDownTimer(10000, 1000);
        myCountDownTimer.start();

    }

});

stop_timer.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {

        myCountDownTimer.cancel();

    }

});

}

public class MyCountDownTimer extends CountDownTimer {

    public MyCountDownTimer(long millisInFuture, long countDownInterval) {
        super(millisInFuture, countDownInterval);
    }

    @Override
    public void onTick(long millisUntilFinished) {

```

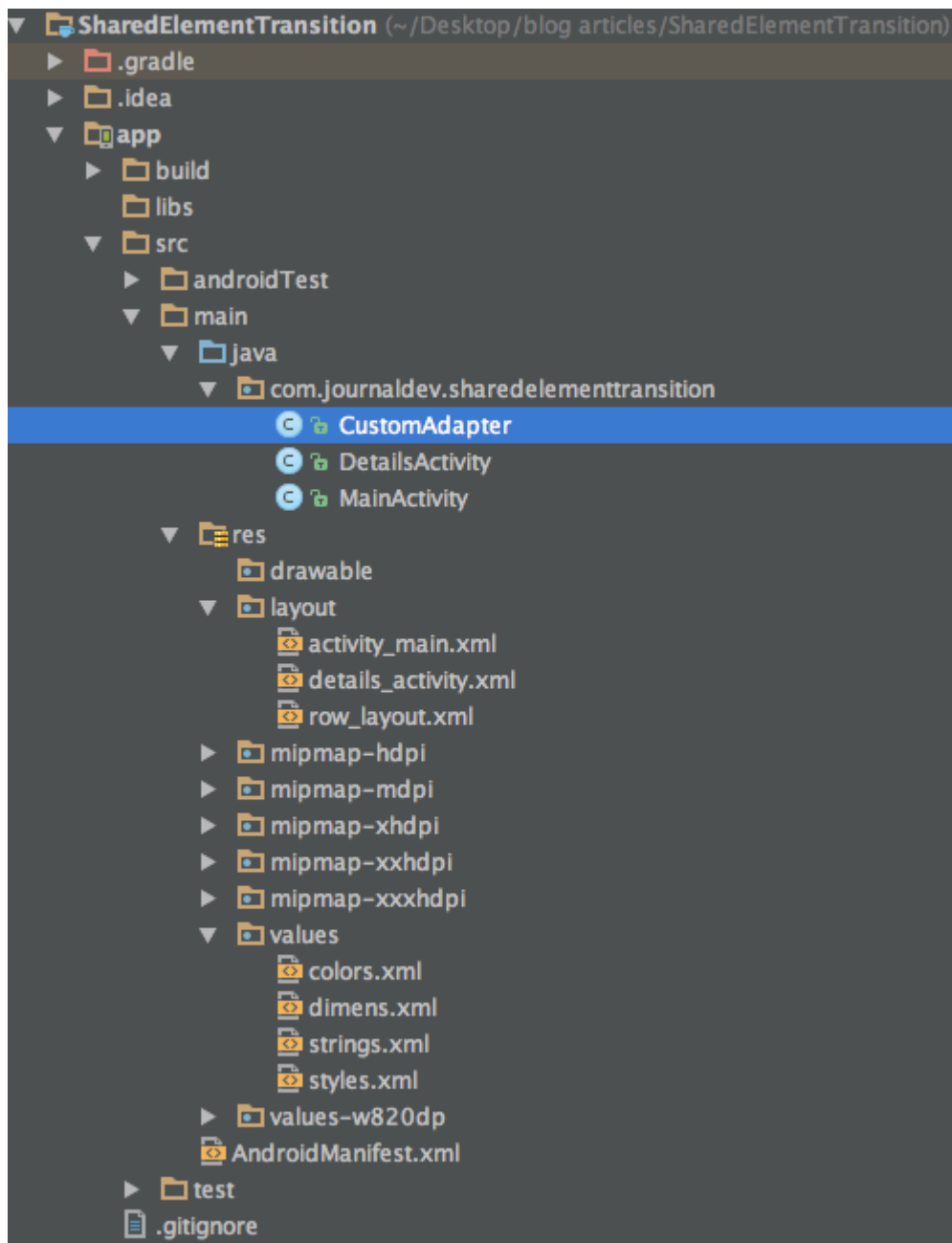
```
int progress = (int) (millisUntilFinished/1000);

progressBar.setProgress(progressBar.getMax()-progress);
}

@Override
public void onFinish() {
    finish();
}
}
}
```

---

### **38. Android Shared Element Transition Animation Project Structure**



This project consists of 2 activities and a CustomAdapter for the ListView.

### Android Transition Animation – Shared Element Transition Code

To enable this transitions add the following snippet inside the AppTheme tag in `styles.xml`.

```
<item name="android:windowContentTransitions">true</item>
```

For both the layouts with this transition we need to assign a `android:transitionName` attribute.

The `activity_main.xml` populates a ListView and the `details_activity.xml` is for the details screen. Both are shown below

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

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:transitionName="@string/transition"

    android:orientation="vertical">
```

```
<ListView

    android:layout_width="wrap_content"

    android:id="@+id/list_view"

    android:layout_height="wrap_content"/>
```

```
</LinearLayout>
```

---

```
<?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"

android:padding="@dimen/activity_horizontal_margin"

android:id="@+id/layout"

android:transitionName="@string/transition"

tools:context="com.journaldev.sharedelementtransition.MainActivity">
```

```
<TextView

    android:gravity="center"

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

    android:id="@+id/heading"

    android:layout_width="match_parent"

    android:textAppearance="?android:attr/textAppearanceLarge"

    android:layout_height="wrap_content" />
```

```
<TextView

    android:gravity="center"

    android:id="@+id/language"

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

    android:layout_width="match_parent"
```

```
        android:textAppearance="?android:attr/textAppearanceMedium"

        android:layout_height="wrap_content"

        android:layout_below="@+id/heading"

        android:layout_alignParentLeft="true"

        android:layout_alignParentStart="true" />
```

```
<TextView
```

```
    android:gravity="center"

    android:id="@+id/desc"

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

    android:layout_width="match_parent"

    android:textAppearance="?android:attr/textAppearanceMedium"

    android:layout_height="wrap_content"

    android:layout_centerInParent="true"

    />
```

```
</RelativeLayout>
```

As you can see a `android:transitionName` attribute is declared as a string in the root view of both the layouts.

We've created a custom `ListView` which populates its layout from a `ArrayList` of `String` arrays. The layout and adapter of the `ListView` are given below.



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

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

    android:orientation="vertical" android:layout_width="match_parent"

    android:padding="@dimen/activity_horizontal_margin"

    android:background="@color/md_black_1000"

    android:layout_margin="5dp"

    android:id="@+id/rl"

    android:layout_height="wrap_content">
```

```
<TextView

    android:layout_width="match_parent"

    android:layout_height="wrap_content"

    android:textAppearance="?android:attr/textAppearanceLarge"

    android:id="@+id/primary_textview"

    android:gravity="center"

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

/>
```

```
<TextView

    android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"

        android:textAppearance="?android:attr/textAppearanceMedium"

        android:id="@+id/textView"

        android:layout_below="@+id/primary_textview"

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

        android:gravity="center"

    />
```

```
</RelativeLayout>
```

---

```
public class CustomAdapter extends BaseAdapter {

    ArrayList<String[]> arrayList;

    Context c;

    public CustomAdapter(Context c, ArrayList<String[]> list) {

        arrayList = list;

        this.c = c;

    }

    @Override
```

```
public int getCount() {  
  
    // TODO Auto-generated method stub  
  
    return arrayList.size();  
  
}
```

```
@Override  
  
public Object getItem(int position) {  
  
    // TODO Auto-generated method stub  
  
    return arrayList.get(position);  
  
}
```

```
@Override  
  
public long getItemId(int position) {  
  
    // TODO Auto-generated method stub  
  
    return position;  
  
}
```

```
@Override  
  
public View getView(int position, View convertView, ViewGroup parent) {  
  
    // TODO Auto-generated method stub  
  
    View row = null;
```

```
LayoutInflater inflater = (LayoutInflater) c

    .getSystemService(Context.LAYOUT_INFLATER_SERVICE);

if (convertView == null) {

    row = inflater.inflate(R.layout.row_layout, parent,

        false);

} else {

    row = convertView;

}

String[] detail = arrayList.get(position);


RelativeLayout rl= (RelativeLayout)row.findViewById(R.id.rl);

rl.setBackgroundColor(Color.parseColor(detail[3]));

TextView name = (TextView) row.findViewById(R.id.primary_textview);

name.setText(detail[0]);

TextView email = (TextView) row.findViewById(R.id.textview);

email.setText(detail[1]);


return row;

}
```

```
}
```

The `MainActivity.java` and `DetailsActivity.java` are given below.

```
package com.journaldev.sharedelementtransition;
```

```
import android.content.Intent;
```

```
import android.support.v4.app.ActivityCompat;
```

```
import android.support.v4.app.ActivityOptionsCompat;
```

```
import android.support.v7.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.view.View;
```

```
import android.widget.AdapterView;
```

```
import android.widget.ListView;
```

```
import java.util.ArrayList;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.activity_main);
```

```
final ArrayList<String[]> values = new ArrayList<String[]>();
```

```
values.add(new String[]{"Android", "Java", getString(R.string.android),'#' +  
Integer.toHexString(getResources().getColor(R.color.md_light_green_900))});
```

```
values.add(new String[]{"iOS", "Swift", getString(R.string.ios),'#' +  
Integer.toHexString(getResources().getColor(R.color.md_amber_A700))});
```

```
values.add(new String[]{"Xamarin", "C#",getString(R.string.xamarin),'#' +  
Integer.toHexString(getResources().getColor(R.color.md_pink_A700))});
```

```
values.add(new String[]{"PhoneGap", "HTML CSS and  
JScript",getString(R.string.phonegap),'#' +  
Integer.toHexString(getResources().getColor(R.color.md_brown_800))});
```

```
ListView listView = (ListView) findViewById(R.id.list_view);
```

```
CustomAdapter adapter = new CustomAdapter(this, values);
```

```
listView.setAdapter(adapter);
```

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
```

```
    @Override
```

```
    public void onItemClick(AdapterView<?> parent, View view, int position, long  
id) {
```

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

```

        intent.putExtra("array",values.get(position));

        // Get the transition name from the string

        String transitionName = getString(R.string.transition);


        ActivityOptionsCompat options =

        ActivityOptionsCompat.makeSceneTransitionAnimation(MainActivity.this,

                view, // Starting view

                transitionName // The String

        );


        ActivityCompat.startActivity(MainActivity.this, intent, options.toBundle());

    }

});

}

}

```

When an activity is finished, instead of finish() we invoke `ActivityCompat.finishAfterTransition(this)` as shown in the code below.

```

public class DetailsActivity extends AppCompatActivity {

```

@Override

```
protected void onCreate(@Nullable Bundle savedInstanceState) {
```

```
    super.onCreate(savedInstanceState);
```

```
    setContentView(R.layout.details_activity);
```

```
    String[] array= getIntent().getStringArrayExtra("array");
```

```
    RelativeLayout rl= (RelativeLayout)findViewById(R.id.layout);
```

```
    rl.setBackgroundColor(Color.parseColor(array[3]));
```

```
    TextView textView= (TextView)findViewById(R.id.heading);
```

```
    textView.setText(array[0]);
```

```
    TextView type= (TextView)findViewById(R.id.language);
```

```
    type.setText(array[1]);
```

```
    TextView desc=(TextView)findViewById(R.id.desc);
```

```
    desc.setText(array[2]);
```

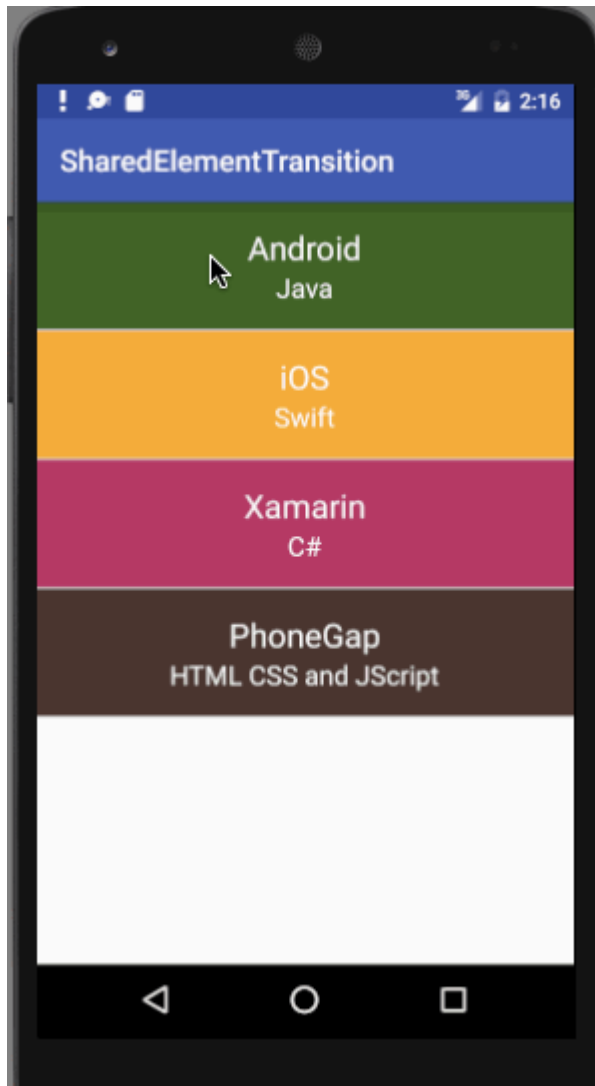
```
}
```

@Override

```
public void onBackPressed() {
```



```
        ActivityCompat.finishAfterTransition(this);  
    }  
}
```



40.

### How To Publish Android App On PlayStore [Step By Step]

Haven't you publish any Android App on Play store? Don't worry if you haven't because it's very easy and this tutorial is going to teach step by step how to publish your first Android App on Play store.


---

## How To Publish Android App On PlayStore:

Follow the below steps:

**Step 1:** First [generate signed apk of your Android App](#) to publish it on Play Store.

**Step 2:** Now you will need to sign up for Google Play Console to publish and manage your Android App.






**Jatin singh**  
singhjatin775@gmail.com

This is the Google account that will be associated with your Developer Console.  
If you would like to use a different account, you can choose from the following options below. If you are an organization, consider registering a new Google account rather than using a personal account.

[SIGN IN WITH A DIFFERENT ACCOUNT](#) [CREATE A NEW GOOGLE ACCOUNT](#)

Before you continue...

		
<b>Accept developer agreement</b> Read and agree to the <a href="#">Google Play Developer distribution agreement</a> . <input type="checkbox"/> I agree and I am willing to associate my account registration with the Google Play Developer distribution agreement.	<b>Review distribution countries</b> Review the distribution countries where you can distribute and sell applications. If you are planning to sell apps or in-app products, check if you can have a merchant account in your country.	<b>Credit card</b> Make sure you have your credit card handy to pay the \$25 registration fee in the next step.

**Important Note:** You can signup with this link <https://play.google.com/apps/publish/>

**Step 3:** Login with your Gmail account that you want to use for publishing App on Play Store.

1

2

3

4


Sign-in with your Google account

Accept Developer Agreement

Pay Registration Fee

Complete your Account details

You are signed in as...



**Jatin singh**  
singhjatin775@gmail.com

This is the Google account that will be associated with your Developer Console.  
If you would like to use a different account, you can choose from the following options below. If you are an organization, consider registering a new Google account rather than using a personal account.

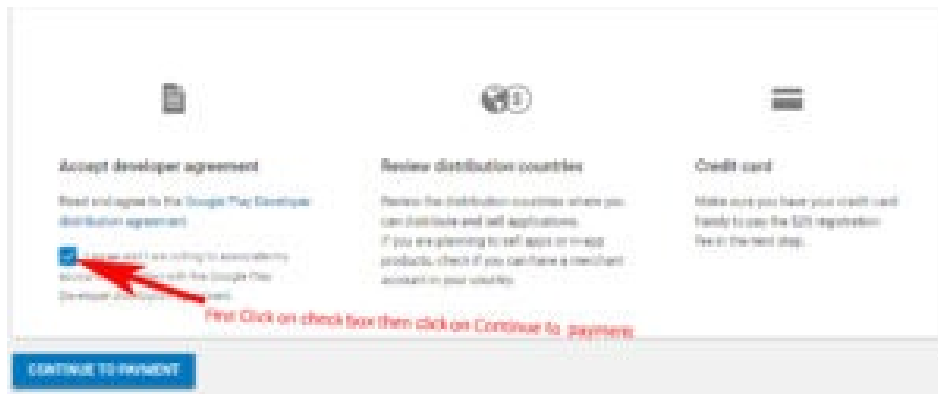
[SIGN IN WITH A DIFFERENT ACCOUNT](#) [CREATE A NEW GOOGLE ACCOUNT](#)

For use a different e-mail account.

Create new account here

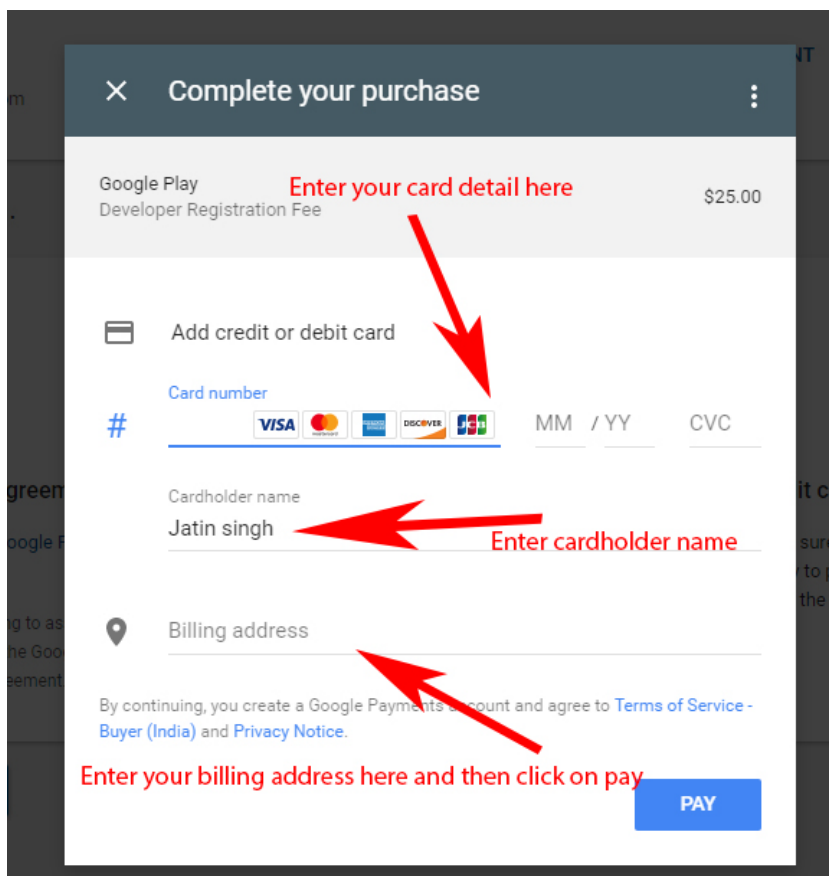
**Step 4:** Now there are 4 steps to complete the registration for Google play store console. You have already completed two.

**Step 5:** After reading the Google play store developer distribution agreement agree to their terms by clicking on check box



**Step 6:** Now you will need to pay one time 'Developer Registration Fee' of \$25 to Google. Please fill your credit card details to make the payment.

**Important Note:** You can upload unlimited number of Android App on Play store from single account with a limit of uploading 15 apk/day.



**Step 7:** Complete your account details for Google developer account. For example see the below image:

Sign-In with your Google account

Accept Developer Agreement

Pay Registration Fee

Complete your Account details

YOU ARE ALMOST DONE...

Just complete the following details. You can change this information later in your account settings if you need to.

DEVELOPER PROFILE Fields marked with \* need to be filled before saving.

Developer name \*   
0 of 50 characters  
The developer name will appear to users under the name of your application.

Email address \*

Website

Phone Number \*   
Include plus sign, country code and area code. For example, +1-800-555-0199.  
[Why do we ask for your phone number?](#)

Email updates ☐ I'd like to get occasional emails about development and Google Play opportunities.  
[Check it for email updates](#)

[Click here for complete your registration](#)

## Step 8: Now click on Create Application



### Playtime 2017: New features now available

At our annual Playtime event, we shared our latest improvements to app discovery and engagement on Google Play as well as new features in the Play Console to help you improve your app quality and grow your business.

[READ THE ANNOUNCEMENT](#)

er ▼

## Step 9: Enter the name of your App.

Create application

Default language \*  
English (United States) – en-US

Title \*  
 0/50

Now Click on Create

**Step 10:** Now fill store listing details of your App which include Title, Short description, and Full description.

**Product details** ENGLISH (UNITED STATES) - en-US Manage translations

Fields marked with \* need to be filled before publishing.

**Title \***  
English (United States) - en-US  
App Demo

**Short description \***  
English (United States) - en-US  
This is a app publish Demo

**Full description \***  
English (United States) - en-US  
This is a demo of how To Publish Android app On Play Store. It is very easy to publish and teach you step by step

**Step 11:** After this you need to put some App screenshots here. The minimum required are 2 screenshots and maximum limit is 8.

**Product details** ENGLISH (UNITED STATES) - en-US Manage translations

**Screenshots \***  
Default - English (United States) - en-US  
JPEG or 24-bit PNG (no alpha). Min length for any side: 320px. Max length for any side: 3840px.  
At least 2 screenshots are required overall. Max 8 screenshots per type. Drag to reorder or to move between types.

For your app to be showcased in the 'Designed for tablets' list in the Play Store, you need to upload at least one 7-inch and one 10-inch screenshot. If you previously uploaded screenshots, make sure to move them into the right area below.  
[Learn how tablet screenshots will be displayed in the store listing.](#)

Please check out our [Impersonation and Intellectual Property policy](#) to avoid common violations.

**PHONE** **TABLET** **ANDROID TV** **ANDROID WEAR**

Add at least one phone screenshot here to help phone users see how your app will look on their device.

**BROWSE FILES**

**Step 12:** After screenshot now you need to put a high Resolution icon or logo with a size of 512 \* 512 pixel. This will be displayed on Play Store.

After that another mandatory thing is you need to put a feature graphic of 1024 \* 500 pixel dimension. See below image for more detail.

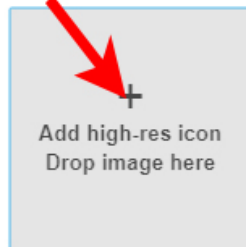
**Hi-res icon \***

Default – English (United States) – en-US

512 x 512

32-bit PNG (with alpha)

Click here to upload logo

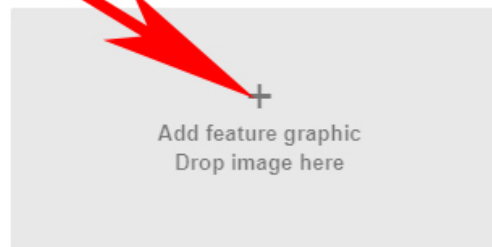
**Feature Graphic \***

Default – English (United States) – en-US

1024 w x 500 h

JPG or 24-bit PNG (no alpha)

Click here to upload feature graphic or banner



**Step 13:** Now scroll down and fill other details which include application type, category, website, email and phone no.

After this check privacy policy because now we are not submitting and then click on save draft. If your App require user permission then it is mandatory to put privacy url.

Click on Save Draft to save your work so far.

A screenshot of the "Store listing" form. At the top right, a red arrow points to a "SAVE DRAFT" button with the text "At last save it in draft". Below this, a red arrow points to the "Application type" dropdown menu with the text "Choose application type". The dropdown is open, showing "Application" and "Games". Another red arrow points to the "Category" dropdown with the text "Select your category". Below that, a red arrow points to the "Privacy Policy" section with the text "Check it". The "Privacy Policy" section has a checkbox that is checked, with the text "Not submitting a privacy policy URL at this time. Learn more". The "Contact details" section shows a "Website" field with the value "http://abhiandroid.com/".

At last save it in draft

SAVE DRAFT

Store listing

Search for apps

Categorization

Application type \*

Application

Games

Category \*

Select a category

Select your category

Content rating \*

You need to fill a rating questionnaire and apply a content rating.

Contact details

Website

http://abhiandroid.com/

Privacy Policy \*

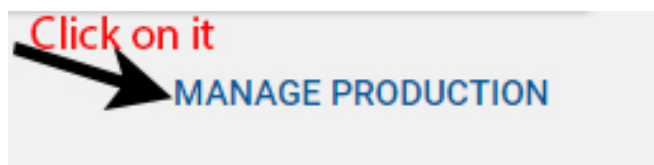
If you wish to provide a privacy policy URL for this application, please enter it below. Also, please check out our [User Data policy](#) to avoid con

Privacy Policy

http://

Not submitting a privacy policy URL at this time. [Learn more](#)

**Step 14:** After saving data on draft now go to **app release** and click on **manage production**.



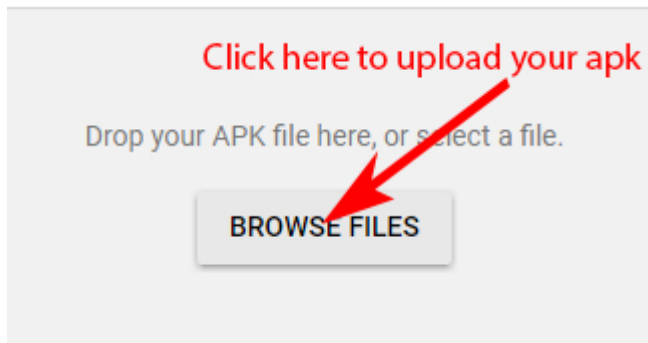
**Step 15:** Now you will see create release now click on it.

## Create release

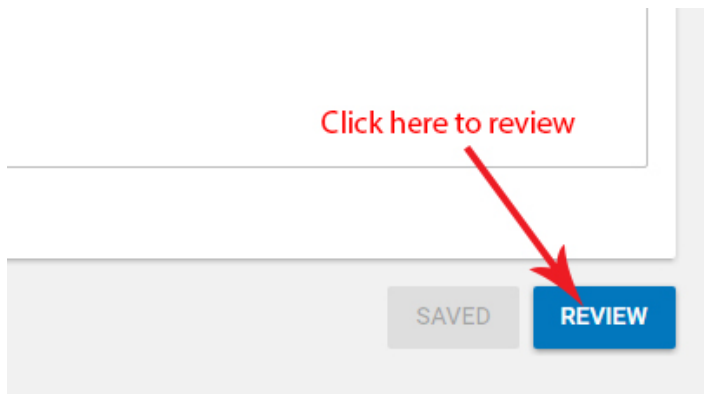
You can prepare, review, and then publish the version of your app you want to make available to users of the Play



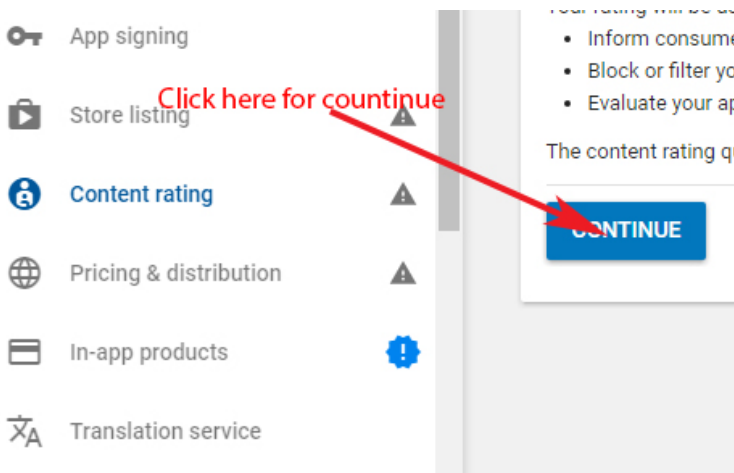
**Step 16:** After click on create release you will see browse files click on it and upload your signed APK.



**Step 17:** Once the upload is successful then scroll down and click on review to check.



**Step 18:** Now go to Content Rating and click on continue.




**Step 19:** Fill details which include email address and select your categories.

Email address \*


Confirm email address \*

Select your category

Select your app category



**REFERENCE, NEWS, OR EDUCATIONAL**  
The primary purpose of the app is to present factual information in a neutral way, alert users to current events, or educate users. Examples include: Wikipedia, BBC News, Dictionary.com, and Medscape. Apps that mainly focus on sexual advice or instruction (such as "IK Sex Positions" or "Best Sex Tips") should be categorized as "Entertainment" apps and not listed here. [Learn more](#)



**SOCIAL NETWORKING, FORUMS, BLOGS, AND UGC SHARING**  
The primary purpose of the app is to enable users to share content or communicate with large groups of people. Examples include Facebook, Chat Roulette, 9Gag, Yelp, Google Plus, YouTube, Twitter. Apps that only facilitate communication between a limited number of people (such as SMS, WhatsApp, or Skype) should be categorized as "Communication" apps and not listed here. [Learn more](#)

**Step 20:** Now select Violence, Sexuality, Language, Controlled Substance and Miscellaneous based on your App. First click on save questionnaire for save and then click on calculate rating.

Content rating - App Developer

Can the app contain violent material? \* [Learn more](#)  
Please note that this question does not refer to user-generated content.

☐ Yes ☐ No

SEXUALITY

LANGUAGE

CONTROLLED SUBSTANCE

MISCELLANEOUS

[CALCULATE RATING](#) [SAVE QUESTIONNAIRE](#)

© 2017 Google - Mobile App - Help - Site Terms - Privacy - Developer Distribution Agreement

**Step 21:** Now click on apply rating.

Google may use your questionnaire data to improve its services.

- Rating authorities participating in IARC
- Google and IARC will share your contact information with participating rating authorities.
- Please review the [Help center](#) for more information.

[APPLY RATING](#) [GO BACK](#)



**Step 22:** Click on pricing and distribution and select free/paid based on how you want user to access your App.

This application is **PAID** **FREE** This app is free

To publish paid applications, you need to [set up a merchant account](#). [Learn more](#)

---

App Availability Your app is currently in a draft state, it will become available on the Play Store wh

---

Countries ★

Unavailable countries <b>0</b>	Available countries <b>142</b> + rest of world Alpha and Beta synced with production	Unsaved changes
-----------------------------------	-----------------------------------------------------------------------------------------------	-----------------

Select Available

Status ?☐ Unavailable☒ Available

**Step 23:** Now scroll down and see mandatory things with \* you need to select After this click on save draft .

Search for apps 🔔 ? 🤖

at last click on save draft

**SAVE DRAFT**

**Step 24:** Now Click on ready on publish along with save draft and click on Manage release.

**Step 25:** Click on Manage Production.

**MANAGE PRODUCTION**

Click here

**EDIT RELEASE**

**Step 26:** After Manage production click on edit release.

Click on edit release



EDIT RELEASE

**Step 27:** Now click on review.

Click on review



SAVED

REVIEW

**Step 28:** After review click on Start Rollout to production. Now you need to confirm. After confirm you will need to wait for one or six hour for approval.

Now click on it



START ROLLOUT TO PRODUCTION

## Flutter Section

### Prog. 1

#### A. Simple Flutter Program on Material Design

```
import 'package:flutter/material.dart';

class LMaterialWidget extends StatelessWidget {
  const LMaterialWidget({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(16.0),
      child: MaterialApp(
        home: Scaffold(
          backgroundColor:
            Theme.of(context).colorScheme.secondary.withOpacity(0.5),
          appBar: AppBar(
            backgroundColor: Theme.of(context).primaryColor,
            title: const Text('MaterialApp'),
          ),
          body: const Center(
            child: Padding(
              padding: EdgeInsets.all(16.0),
              child: Text(
                'A convenience widget that wraps '
                'a number of widgets that are commonly '
                'required for applications implementing '
                'Material Design.',
                style: TextStyle(fontSize: 14, height: 1.5),
                textAlign: TextAlign.center,
              ),
            ),
          ),
          debugShowCheckedModeBanner: false,
        ),
      );
  }
}
```

---

#### B.Simple Flutter Program on Layout Design

```
import 'package:flutter/material.dart';

class LCWidget extends StatelessWidget {
  const LCWidget({Key? key}) : super(key: key);
```

```

@override
Widget build(BuildContext context) {
  return const MaterialApp(
    home: LContainerWidget(),
  );
}
}

```

```

class LContainerWidget extends StatelessWidget {
  const LContainerWidget({Key? key}) : super(key: key);

```

```

@override
Widget build(BuildContext context) {
  return SingleChildScrollView(
    child: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: <Widget>[
          Container(
            margin: const EdgeInsets.all(5.0),
            color: Theme.of(context).primaryColor,
            child: const Center(
              child: Text(
                'I am container',
                style: TextStyle(fontSize: 16, color: Colors.white),
              )),
            width: MediaQuery.of(context).size.width,
            height: 96.0,
          ),
          Container(
            padding: const EdgeInsets.all(8.0),
            color: Theme.of(context).primaryColor,
            alignment: Alignment.center,
            child: Text('Hai, I am Slanting',
              style: Theme.of(context)
                .textTheme
                .headline4!
                .copyWith(color: Colors.white)),
            transform: Matrix4.skewY(0.2),
          ),
          const SizedBox(
            height: 48,
          ),
          Container(
            constraints: BoxConstraints.expand(
              height: Theme.of(context).textTheme.headline4!.fontSize! * 1.0 +
                50.0,
            ),
            padding: const EdgeInsets.all(8.0),
            color: Theme.of(context).primaryColor,

```

```

        alignment: Alignment.center,
        child: const Text(
          'I am also Slanting,but see my edges',
          style: TextStyle(fontSize: 16, color: Colors.white),
        ),
        transform: Matrix4.rotationZ(0.2),
      ),
      const SizedBox(
        height: 64,
      ),
      Container(
        height: 200,
        width: 200,
        color: Colors.blue,
        child: Center(
          child: Container(
            height: 100,
            width: 100,
            color: Colors.yellow,
            child: Center(
              child: Container(
                height: 50,
                width: 50,
                color: Colors.green,
                child: Center(
                  child: Container(
                    height: 25,
                    width: 25,
                    color: Colors.red,
                    child: Center(
                      child: Container(
                        height: 15,
                        width: 15,
                        color: Colors.blue,
                      ),
                    ),
                  ),
                ),
              ),
            ),
          ),
        ),
      ),
    ),
  ),
),
);
}
}

```

---

### C. Simple Flutter Program on Button Design

```
import 'package:flutter/material.dart';

class LBWidget extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: LButtonWidget(),
    );
  }
}

class LButtonWidget extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: <Widget>[
          TextButton(
            child: Text("Text Button"),
            onPressed: () {},
          ),

          ElevatedButton(
            child: Text("Elevated Button"),
            onPressed: () {},
          ),
          IconButton(
            icon: Icon(
              Icons.sms_failed,
              color: Colors.blue,
              size: 36,
            ),
            tooltip: 'Image Button',
            onPressed: () {},
          ),

          //For Ripple Effect Ink Well is used
          Material(
            // needed
            color: Colors.orange,
            child: InkWell(
              onTap: () {},
              child: Container(width: 80.0, height: 40.0),
            ),
          ),
          RawMaterialButton(
```

```

        child: Icon(
          Icons.play_arrow,
          color: Colors.limeAccent,
        ),
        onPressed: () {},
        shape: CircleBorder(),
        elevation: 2,
        splashColor: Colors.transparent,
        fillColor: Colors.deepPurpleAccent,
        highlightColor: Colors.transparent,
      ),
      BackButton(
        onPressed: () {},
      ),
      CloseButton(
        onPressed: () {},
      )
    ],
  ),
);
}
}

```

---

Prog. 2

### **A.Flutter program on Animated Icon**

```

import 'package:flutter/material.dart';

class LAnimateIcon extends StatefulWidget {
  @override
  _LAnimateIconState createState() => _LAnimateIconState();
}

class _LAnimateIconState extends State<LAnimateIcon>
  with SingleTickerProviderStateMixin {
  late AnimationController _animationController;
  bool isPlaying = false;

  @override
  Widget build(BuildContext context) {
    return SingleChildScrollView(
      child: Container(
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: <Widget>[
              Text("Click below buttons"),
              IconButton(

```

```

        iconSize: 50,
        tooltip: "play_pause",
        icon: AnimatedIcon(
          icon: AnimatedIcons.play_pause,
          progress: _animationController,
          color: Colors.lightBlue,
        ),
        onPressed: () => _handleOnPressed(),
      ),
      IconButton(
        iconSize: 50,
        tooltip: "add_event",
        icon: AnimatedIcon(
          icon: AnimatedIcons.add_event,
          progress: _animationController,
          color: Colors.green,
        ),
        onPressed: () => _handleOnPressed(),
      ),
      IconButton(
        iconSize: 50,
        tooltip: "arrow_menu",
        icon: AnimatedIcon(
          icon: AnimatedIcons.arrow_menu,
          progress: _animationController,
          color: Colors.lightBlue,
        ),
        onPressed: () => _handleOnPressed(),
      ),
      IconButton(
        iconSize: 50,
        tooltip: 'ellipsis_search',
        icon: AnimatedIcon(
          icon: AnimatedIcons.ellipsis_search,
          progress: _animationController,
          color: Colors.red,
        ),
        onPressed: () => _handleOnPressed(),
      ),
      IconButton(
        iconSize: 50,
        tooltip: "list_view",
        icon: AnimatedIcon(
          icon: AnimatedIcons.list_view,
          progress: _animationController,
        ),
        onPressed: () => _handleOnPressed(),
      ),
      IconButton(
        iconSize: 50,
        tooltip: "home_menu",

```



```

        icon: AnimatedIcon(
          icon: AnimatedIcons.home_menu,
          progress: _animationController,
          color: Colors.pink,
        ),
        onPressed: () => _handleOnPressed(),
      ),
    ],
  ),
),
);
}

@override
void initState() {
  super.initState();
  _animationController =
    AnimationController(vsync: this, duration: Duration(milliseconds: 300));
}

@override
void dispose() {
  _animationController.dispose();
  super.dispose();
}

void _handleOnPressed() {
  setState() {
    isPlaying = !isPlaying;
    isPlaying
      ? _animationController.forward()
      : _animationController.reverse();
  });
}
}

```

---

## **B. Flutter program on Simple List**

```

import 'package:flutter/material.dart';

class LListViewWidget extends StatelessWidget {
  final List<int> colorCodes = <int>[
    50,
    100,
    200,
    300,
    400,
    500,
    600,
    700,

```

```

      800,
      900
    ];

    @override
    Widget build(BuildContext context) {
      return Container(
        child: ConstrainedBox(
          constraints: new BoxConstraints(
            minHeight: 35.0,
            maxHeight: 160.0,
          ),
          child: Row(
            children: <Widget>[
              Expanded(child: _listBuilder()),
            ],
          )),
    );
  }

```

// The ListView.builder constructor takes an IndexedWidgetBuilder, which builds the children on demand.

```

Widget _listBuilder() {
  return ListView.builder(
    padding: const EdgeInsets.all(8),
    itemCount: 500,
    itemBuilder: (BuildContext context, int index) {
      return Container(
        height: 50,
        color: Colors.amber[colorCodes[index % 10]],
        child: Center(child: Text('Item $index')),
      );
    });
}

```

---

### C. Flutter Program on Animated List

```

import 'package:flutter/material.dart';

class LAnimatedList extends StatefulWidget {
  @override
  LAnimatedListState createState() {
    return new LAnimatedListState();
  }
}

class LAnimatedListState extends State<LAnimatedList> {
  final GlobalKey<AnimatedListState> _listKey = GlobalKey();

```

```

List<String> _data = [
  "Item 1",
  "Item 2",
  "Item 3",
  "Item 4",
  "Item 5",
  "Item 6",
  "Item 7",
  "Item 8",
];

@override
Widget build(BuildContext context) {
  return Column(
    children: <Widget>[
      Expanded(
        child: AnimatedList(
          key: _listKey,
          initialItemCount: _data.length,
          itemBuilder: (context, index, animation) =>
            _buildItem(context, _data[index], animation),
        ),
      ),
      Row(
        mainAxisAlignment: MainAxisAlignment.max,
        crossAxisAlignment: CrossAxisAlignment.center,
        mainAxisSize: MainAxisSize.spaceAround,
        children: <Widget>[
          ElevatedButton(
            child: Text(
              'Add',
              style: TextStyle(fontSize: 20, color: Colors.white),
            ),
            onPressed: () {
              _addAnItem();
            },
          ),
          ElevatedButton(
            child: Text(
              'Remove',
              style: TextStyle(fontSize: 20, color: Colors.white),
            ),
            onPressed: () {
              _removeLastItem();
            },
          ),
        ],
      ),
    ],
  );
}

```

```

Widget _buildItem(
  BuildContext context, String item, Animation<double> animation) {
  TextStyle textStyle = new TextStyle(fontSize: 20);

  return Padding(
    padding: const EdgeInsets.all(2.0),
    child: SizeTransition(
      sizeFactor: animation,
      axis: Axis.vertical,
      child: SizedBox(
        height: 50.0,
        child: Card(
          child: Center(
            child: Text(item, style: textStyle),
          ),
        ),
      ),
    ),
  );
}

void _addAnItem() {
  _data.insert(0, "Inserted Item");
  _listKey.currentState!.insertItem(0);
}

void _removeLastItem() {
  String itemToRemove = _data[0];

  _listKey.currentState!.removeItem(
    0,
    (BuildContext context, Animation<double> animation) =>
      _buildItem(context, itemToRemove, animation),
    duration: const Duration(milliseconds: 250),
  );

  _data.removeAt(0);
}

```

---

Prog. 3

### 1. Scaffold

**Scaffold** class is an expandable widget that fills the available space or the screen. It provides an **API** to display the main widgets of the application such as **Drawer**, **SnackBar**, **Bottom-Sheet**, **FloatingActionButton**, **AppBar**, and **BottomNavigationBar**, etc.

## Scaffold Constructor:

Scaffold constructor

```
const Scaffold({  
  Key key,  
  PreferredSizeWidget appBar,  
  Widget body,  
  Widget floatingActionButton,  
  FloatingActionButtonLocation floatingActionButtonLocation,  
  FloatingActionButtonAnimator floatingActionButtonAnimator,  
  List<Widget> persistentFooterButtons,  
  Widget drawer,  
  Widget endDrawer,  
  Widget bottomNavigationBar,  
  Widget bottomSheet,  
  Color backgroundColor,  
  bool resizeToAvoidBottomPadding,  
  bool resizeToAvoidBottomInset,  
  bool primary: true,  
  DragStartBehavior drawerDragStartBehavior: DragStartBehavior.down  
})
```

Example 1

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  // This widget is the root of your application.  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Mumbai',  
      debugShowCheckedModeBanner: false,  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
        visualDensity: VisualDensity.adaptivePlatformDensity,  
      ),  
      home: MyHomePage(title: 'Flutter Scaffold Example'),  
    );  
  }  
}  
  
class MyHomePage extends StatelessWidget {  
  MyHomePage({Key key, this.title}) : super(key: key);  
  final String title;  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  

```

```
    appBar: AppBar(  
      title: Text(this.title),  
    ),  
    body: Center(  
      child:  
        Text (  
          'Hello World',  
        )  
      ),  
    );  
  }  
}
```

---

Example 2

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Mumbai',  
      debugShowCheckedModeBanner: false,  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
        visualDensity: VisualDensity.adaptivePlatformDensity,  
      ),  
      home: MyHomePage(title: 'Flutter Scaffold Example'),  
    );  
  }  
}  
  
class MyHomePage extends StatelessWidget {  
  MyHomePage({Key key, this.title}) : super(key: key);  
  final String title;  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold (  
      appBar: AppBar(  
        title: Text(this.title),  
      ),  
      body: Center(  
        child:  
          Text(  
            'Hello World',  
          )  
        )  
      )  
    );  
  }  
}
```

```

    ),
    endDrawer: Drawer(
      child: ListView(
        children: const <Widget> [
          DrawerHeader(
            decoration: BoxDecoration(
              color: Colors.green,
            ),
            child: Text(
              'Hello World',
              style: TextStyle(
                color: Colors.green,
                fontSize: 24,
              ),
            ),
          ),
        ],
      ),
    ),
    ListTile(
      title: Text('Gallery'),
    ),
    ListTile(
      title: Text('Slideshow'),
    ),
  ],
),
);
}
}

```

---

### Example 3

```

import 'package:flutter/material.dart';

void main() => runApp(MyApp());

// This Widget is the main application widget.
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: "Mumbai",
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.blue,
        visualDensity: VisualDensity.adaptivePlatformDensity,
      ),
      home: MyHomePage(),
    );
  }
}

```

```
class MyHomePage extends StatefulWidget {
```

```
  MyHomePage({Key key}) : super(key: key);
```

```
  @override
```

```
  MyHomePageState createState() => MyHomePageState();
```

```
}
```

```
class MyHomePageState extends State<MyHomePage> {
```

```
Widget build(BuildContext context) {
```

```
  return Scaffold(
```

```
    appBar: AppBar(
```

```
      title: Text('Flutter Scaffold Example'),
```

```
    ),
```

```
    body: Center(
```

```
      child: Center(
```

```
        child: Builder ( // The Builder return a ElevatedButton
```

```
          // We need a context of Scaffold
```

```
          builder: (BuildContext ctxOfScaffold) {
```

```
            return ElevatedButton(
```

```
              onPressed: () {
```

```
                this._showMyBottomSheet(ctxOfScaffold);
```

```
              },
```

```
              child: Icon(Icons.add),
```

```
            );
```

```
          },
```

```
        )
```

```
      ),
```

```
    ),
```

```
    floatingActionButton: Builder ( // The Builder return a FloatingActionButton
```

```
      // We need a context of Scaffold
```

```
      builder: (BuildContext ctxOfScaffold) {
```

```
        return FloatingActionButton(
```

```
          onPressed: () {
```

```
            this._showMyBottomSheet(ctxOfScaffold);
```

```
          },
```

```
          tooltip: 'Increment Counter',
```

```
          child: Icon(Icons.add),
```

```
        );
```

```
      },
```

```
    ),
```

```
    bottomNavigationBar: BottomAppBar(
```

```
      color: Colors.lightGreen[200],
```

```
      child: new Row(
```

```
        mainAxisAlignment: MainAxisAlignment.max,
```

```
        mainAxisAlignment: MainAxisAlignment.start,
```

```
        children: <Widget>[
```

```
          TextButton.icon(icon: Icon(Icons.home), label: Text(''), onPressed: () {}),
```

```
          TextButton.icon(icon: Icon(Icons.email), label: Text(''), onPressed: () {}),
```

```
        ],
```



```

    ),
  )
);
}

// We need a context object of Scaffold to draw Scaffold.bottomSheet.
void _showMyBottomSheet(BuildContext ctxOfScaffold) { // context of Scaffold.

  // ScaffoldState.showBottomSheet:
  Scaffold.of(ctxOfScaffold).showBottomSheet<void>(
    (BuildContext context) {
      return Container(
        height: 200,
        color: Colors.amber,
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            mainAxisAlignment: MainAxisAlignment.min,
            children: <Widget>[
              const Text('BottomSheet'),
              ElevatedButton(
                child: Text('Close BottomSheet'),
                onPressed: () => Navigator.pop(context),
              )
            ],
          ),
        ),
      );
    },
  );
}
}

```

---

Prog. 4

**Flutter BottomAppBar.** Example

```

import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(

```

```

    title: 'Title of Application',
    theme: ThemeData(
      primarySwatch: Colors.blue,
      visualDensity: VisualDensity.adaptivePlatformDensity,
    ),
    home: MyHomePage(),
  );
}
}

```

```

class MyHomePage extends StatelessWidget {
  MyHomePage({Key key}) : super(key: key);

```

**@override**

```

Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("BottomAppBar Example"),
    ),
    body: Center(
      child: Text(
        'Flutter BottomAppBar Example',
      )
    ),
    floatingActionButton: FloatingActionButton.extended (
      elevation: 4.0,
      icon: const Icon(Icons.add),
      label: const Text('Add a task'),
      onPressed: () {},
    ),
    floatingActionButtonLocation: FloatingActionButtonLocation.endDocked,
    bottomNavigationBar: BottomAppBar(
      child: new Row(
        mainAxisAlignment: MainAxisAlignment.max,
        mainAxisAlignment: MainAxisAlignment.start,
        children: <Widget>[
          IconButton(icon: Icon(Icons.home), onPressed: () {}),
          PopupMenuButton(
            icon: Icon(Icons.share),
            itemBuilder: (context) => [
              PopupMenuItem(
                value: 1,
                child: Text("Facebook"),
              ),
              PopupMenuItem(
                value: 2,
                child: Text("Instagram"),
              ),
            ],
          ),
          IconButton(icon: Icon(Icons.email), onPressed: () {}),

```

```
    ],  
  ),  
)  
);  
}  
}
```

---

Prog. 5

### Flutter Navigation Example

**Flutter** introduces the concept of "**Route Transition**" to describe the action of jumping from the first screen to the second. This transition may include an animation effect to offer a favorable feeling to the user. In this article, I'm going to cover a few ways to get there.

```
import 'package:flutter/material.dart';
```

```
main() {  
  runApp(MyApp());  
}
```

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Mumbai',  
      debugShowCheckedModeBanner: false,  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
        visualDensity: VisualDensity.adaptivePlatformDensity,  
      ),  
      home: Page1(),  
    );  
  }  
}
```

```
class Page1 extends StatelessWidget {  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Title of Page 1"),  
      ),  
      body: Center(  
        child: ElevatedButton(  
          child: Text('Go!'),  
          onPressed: () {  
            Navigator.of(context).push(_createRoute());  
          },  
        ),  
      ),  
    );  
  }  
}
```

```

    }
  }

Route _createRoute() {
  return PageRouteBuilder(
    pageBuilder: (BuildContext context, Animation<double> animation, //
      Animation<double> secondaryAnimation) {
      return Page2();
    },
    transitionsBuilder: (BuildContext context, Animation<double> animation, //
      Animation<double> secondaryAnimation, Widget child) {
      return child;
    },
  );
}

class Page2 extends StatelessWidget {
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Title of Page 2"),
      ),
      body: Center(
        child: Text('Page 2'),
      ),
      backgroundColor: Colors.lightGreen[100],
    );
  }
}

```

---

## Example 2

```

import 'package:flutter/material.dart';

main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'o7planning.org',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        primarySwatch: Colors.blue,
        visualDensity: VisualDensity.adaptivePlatformDensity,
      ),
      initialRoute: '/home',
      routes: <String, WidgetBuilder>{

```

```

    '/home': (BuildContext context) => Home(),
    '/details': (BuildContext context) => Details(),
    '/about': (BuildContext context) => About(),
  },
);
}
}

```

```

class Home extends StatelessWidget {
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Title of Home Page"),
      ),
      body: Center(

        child: Row (
          mainAxisAlignment: MainAxisAlignment.spaceAround,
          children: [
            ElevatedButton(
              child: Text('Go to Details Page'),
              onPressed: () {
                Navigator.of(context).pushNamed('/details');
              },
            ),
            ElevatedButton(
              child: Text('Go to About Page'),
              onPressed: () {
                Navigator.of(context).pushNamed('/about');
              },
            ),
          ],
        ),
      ),
    );
  }
}

```

```

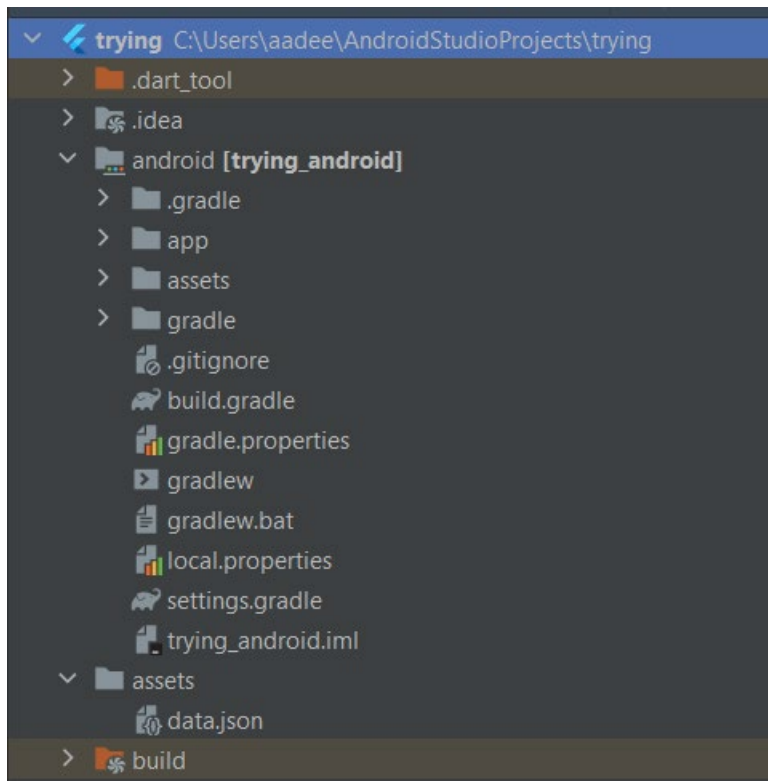
class Details extends StatelessWidget {
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Title of Details Page"),
      ),
      body: Center(
        child: ElevatedButton(
          child: Text('Close'),
          onPressed: () {
            // Close page and pass a value back to previous page
            Navigator.of(context).pop();
          },
        ),
      ),
    );
  }
}

```

```
    ),  
    ),  
    backgroundColor: Colors.lightGreen[100],  
  );  
}  
}
```

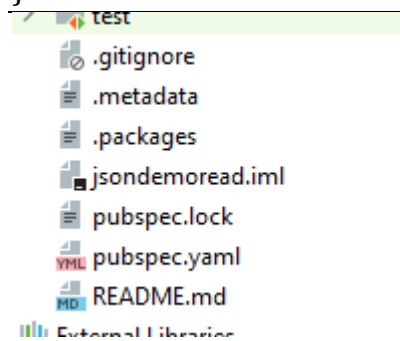
```
class About extends StatelessWidget {  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Title of About Page"),  
      ),  
      body: Center(  
        child: ElevatedButton(  
          child: Text('Close'),  
          onPressed: () {  
            // Close page  
            Navigator.of(context).pop();  
          },  
        ),  
      ),  
      backgroundColor: Colors.cyan[100],  
    );  
  }  
}
```

---



```
{
  "items": [
    {
      "id": "1",
      "name": "Apple",
      "description": "An apple is an edible fruit produced by an apple tree (Malus domestica). Apple trees are cultivated worldwide and are the most widely grown species in the"
    },
    {
      "id": "2",
      "name": "Banana",
      "description": "banana is an elongated, edible fruit - botanically a berry - produced by several kinds of large herbaceous flowering plants in the genus Musa."
    },
    {
      "id": "3",
      "name": "Grape",
      "description": "The grapefruit (Citrus × paradisi) is a subtropical citrus tree known for its relatively large, sour to semi-sweet, somewhat bitter fruit."
    },
    {
      "id": "4",
      "name": "Mango",
      "description": "A mango is an edible stone fruit produced by the tropical tree Mangifera indica which is believed to have originated from the region between northwestern "
```

```
{
  "id": "5",
  "name": "Orange",
  "description": "An orange is a fruit of various citrus species in the family Rutaceae
); it primarily refers to Citrus × sinensis, which is also called sweet orange"
}
```



```
23 dependencies:
24   flutter:
25     sdk: flutter
26   cupertino_icons: ^1.0.0
```

```
50 # To add assets to your application
51 assets:
52   - assets/data.json
53   # - images/a_dot_ham.jpeg
```

## Main.dart

```
import 'package:flutter/material.dart';
import 'dart:async';
import 'dart:convert';
import 'package:flutter/services.dart';
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return new MaterialApp(
      title: 'Flutter JSON Demo',
      theme: new ThemeData(
        // This is the theme of your application.
```



```

//
// Try running your application with "flutter run". You'll see the
// application has a blue toolbar. Then, without quitting the app, try
// changing the primarySwatch below to Colors.green and then invoke
// "hot reload" (press "r" in the console where you ran "flutter run",
// or simply save your changes to "hot reload" in a Flutter IDE).
// Notice that the counter didn't reset back to zero; the application
// is not restarted.
primarySwatch: Colors.blue,
// This makes the visual density adapt to the platform that you run
// the app on. For desktop platforms, the controls will be smaller and
// closer together (more dense) than on mobile platforms.
visualDensity: VisualDensity.adaptivePlatformDensity,
),
home: LocalJsonScreen(apiRoot: 'http://api.flutter.institute/'),
);
}
}

class LocalJsonScreen extends StatefulWidget {
  const
  LocalJsonScreen({Key key, String apiRoot, this.title}) : super(key: key);

  // This widget is the home page of your application. It is stateful, meaning
  // that it has a State object (defined below) that contains fields that affect
  // how it looks.

  // This class is the configuration for the state. It holds the values (in this
  // case the title) provided by the parent (in this case the App widget) and
  // used by the build method of the State. Fields in a Widget subclass are
  // always marked "final".

  final String title;

  @override
  State<LocalJsonScreen> createState() => _LocalJsonScreenState();
}

class _LocalJsonScreenState extends State<LocalJsonScreen> {
  List jsonData = [];

  // Fetch content from the json file
  Future<void> readJson() async {
    final String response = await rootBundle.loadString('assets/data.json');
    final data = await json.decode(response);
    setState(() {
      jsonData = data["items"];
    });
  }
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      centerTitle: true,
      title: const Text(
        'Read Data From Local Json',
      ),
    ),
    body: Padding(
      padding: const EdgeInsets.all(25),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Center(
            child: ElevatedButton(
              child: const Text('Load Fruit Data'),
              onPressed: readJson,
            ),
          ),

          // Display the data loaded from data.json
          jsonData.isNotEmpty
            ? Expanded(
              child: ListView.builder(
                itemCount: jsonData.length,
                itemBuilder: (context, index) {
                  return Card(
                    margin: const EdgeInsets.all(10),
                    child: ListTile(
                      leading: CircleAvatar(
                        child: Text(jsonData[index]["id"]),
                      ),
                      title: Text(jsonData[index]["name"]),
                      subtitle: Text(jsonData[index]["description"]),
                    ),
                  );
                },
              ),
            )
            : const SizedBox()
        ],
      ),
    ),
  );
}

```

## What Is SQLite

SQLite is an open source relational database, it is used to create a database, perform different operation like add, delete, and remove data.

SQLite does not require a server or backend code, all the data is saved to a text file in the device. You can learn more about it [here](#).

## Adding Sqlite Plugin to Flutter

To be able to use SQLite in Flutter, you need to add the plugin `sqflite`. So to add it you need to navigate to the `pubspec.yaml` file, and write the following:

```
1 dependencies:
2   cupertino_icons: ^1.0.2
3 flutter:
4   sdk: flutter
5 path: ^1.8.0
6 sqflite: ^2.0.0+4
```

Now you can start using SQLite in the Flutter project! In the following sections we will create a list of users and add those users to the database.

## Creating the Model Class

Since we need to add multiple users to the database, then we have to create a class called `User` which will contain different fields related to a user, for example:

```
1 class User {
2   final int? id;
3   final String name;
4   final int age;
5   final String country;
6   final String? email;
7
8   User(
9     { this.id,
10      required this.name,
11      required this.age,
12      required this.country,
13      this.email});
14
15
16   User.fromMap(Map<String, dynamic> res)
17     : id = res["id"],
18       name = res["name"],
19       age = res["age"],
20       country = res["country"],
21       email = res["email"];
```

```

22
23 Map<String, Object?> toMap() {
24   return {'id':id, 'name': name, 'age': age, 'country': country, 'email': email};
25 }
  }

```

Here we create a named constructor called `User.fromMap()` and a method called `toMap()` since to save data to the SQLite database we need to convert it to a map. Now, we will create the `DatabaseHandler` class.

### Create a table in SQLite

Now, under the `lib` folder, create another folder called `services` and inside of it create a class called `DatabaseHandler`. This class will take care of all the operations regarding the SQLite database.

---

Before initializing the database, we need to specify the location of the file that will be created which will contain the database, to do that we need to add another plugin called `path`, therefore navigate to the `pubspec.yaml` and add the following:

```

1 dependencies:
2   flutter:
3     sdk: flutter
4   cupertino_icons: ^1.0.2
5   sqflite: ^2.0.0+3
6   path: ^1.8.0

```

---

Now inside the `DatabaseHandler` class we can initialize the database, for example:

```

1 import 'package:sqflite/sqflite.dart';
2 import 'package:path/path.dart';
3
4 class DatabaseHandler {
5   Future<Database> initializeDB() async {
6     String path = await getDatabasesPath();
7     return openDatabase(
8       join(path, 'example.db'),
9       onCreate: (database, version) async {
10        await database.execute(
11          "CREATE TABLE users(id INTEGER PRIMARY KEY AUTOINCREMENT,
12 name TEXT NOT NULL,age INTEGER NOT NULL, country TEXT NOT NULL,
13 email TEXT)",
14    );

```

```

15     },
16     version: 1,
17 );
    }
}

```

So here the method `getDatabasePath()` is inside the `sqflite` package and it will get the default database location. The `openDatabase()` method is also inside the package `sqflite` and it accepts a mandatory `String` as an argument which will be the path of the database.

As you can see above, we use the method `join()` which is inside the package `path`, it will join the given path into a single path, so for example we would get `databasepath/example.db`.

The `onCreate()` callback will be called after the database was created, and it will execute the above sql query that will create the table `users`.

### Saving Data in SQLite

Now inside the class `DatabaseHandler`, we can create another method to insert users to the database, for example:

```

1 Future<int> insertUser(List<User> users) async {
2     int result = 0;
3     final Database db = await initializeDB();
4     for(var user in users){
5         result = await db.insert('users', user.toMap());
6     }
7     return result;
8 }

```

Here `insertUser()` will take a list of users, then we loop inside the collection and insert each user to the table `users`. The `insert()` method takes the following parameters `String table`, `Map<String, Object?> values`, and that's why we create a `toMap()` method in the model class.

### Retrieve Data From SQLite

Then to retrieve data, we can create another method in the `DatabaseHandler` class, for example:

```

1 Future<List<User>> retrieveUsers() async {
2     final Database db = await initializeDB();
3     final List<Map<String, Object?>> queryResult = await db.query('users');
4     return queryResult.map((e) => User.fromMap(e)).toList();
5 }

```

So here we use the `query()` method and give it the string `users` which is the table name. So this will select all columns from the table `users`. Then

since `queryResult` returns a `List`, therefore we use the `map()` method to transform the `List<Map<String, Object?>>` into a `List<User>`.

### Delete Data From SQLite

To delete data, we create the following method:

```
1 Future<void> deleteUser(int id) async {
2   final db = await initializeDB();
3   await db.delete(
4     'users',
5     where: "id = ?",
6     whereArgs: [id],
7   );
8 }
```

Here using the `delete()` method we pass the table name and then specify according to which column we need to delete the row in the database table.

### Display Data From The Database

Navigate to the `main.dart` file, and remove all the code related to the counter Flutter application, and inside the `_MyHomePageState` add the following:

```
1 class _MyHomePageState extends State<MyHomePage> {
2
3   late DatabaseHandler handler;
4
5   @override
6   void initState() {
7     super.initState();
8     this.handler = DatabaseHandler();
9     this.handler.initializeDB().whenComplete(() async {
10       await this.addUsers();
11       setState(() {});
12     });
13 }
```

So here we create an instance of the class `DatabaseHandler()` and then call `inititalizeDb()` to create the database which will contain the `users` table. When the `Future` is completed we call `addUsers()`. For simplicity, I'm creating two users manually instead of creating a form, therefore inside the `addUsers()` we would have the following:

```
1 Future<int> addUsers() async {
2   User firstUser = User(name: "peter", age: 24, country: "Lebanon");
```

```

3  User secondUser = User(name: "john", age: 31, country: "United Kingdom");
4  List<User> listOfUsers = [firstUser, secondUser];
5  return await this.handler.insertUser(listOfUsers);
6  }

```

Here, we create two users and add them to the list, then call `insertUser()` method. After this is done, `setState()` will be called which will rebuild the widget tree. Inside the `build()` method we will use a `FutureBuilder` widget to call `retrieveUsers()` method:

```

1  body: FutureBuilder(
2    future: this.handler.retrieveUsers(),
3    builder: (BuildContext context, AsyncSnapshot<List<User>> snapshot) {
4      if (snapshot.hasData) {
5        return ListView.builder(
6          itemCount: snapshot.data?.length,
7          itemBuilder: (BuildContext context, int index) {
8            return Dismissible(
9              direction: DismissDirection.endToStart,
10             background: Container(
11               color: Colors.red,
12               alignment: Alignment.centerRight,
13               padding: EdgeInsets.symmetric(horizontal: 10.0),
14               child: Icon(Icons.delete_forever),
15             ),
16             key: ValueKey<int>(snapshot.data![index].id!),
17             onDismissed: (DismissDirection direction) async {
18               await this.handler.deleteUser(snapshot.data![index].id!);
19               setState(() {
20                 snapshot.data!..remove(snapshot.data![index]);
21               });
22             },
23             child: Card(
24               child: ListTile(
25                 contentPadding: EdgeInsets.all(8.0),
26                 title: Text(snapshot.data![index].name),
27                 subtitle: Text(snapshot.data![index].age.toString()),
28               )),
29             );
30           },
31         );

```

```
32     } else {  
33         return Center(child: CircularProgressIndicator());  
34     }  
35 },  
36 ),
```

The above `build()` method will give us the following:

---

To delete a user, we can use the `Dismissible` widget to swipe to the right, which will remove the item from the list and delete the user from the database by calling the method `deleteUser()`:

---