



Dr. Babasaheb Ambedkar Open University

(Established by Government of Gujarat)

PGDMAD-105

Software Lab For Android Mobile Application Development

This section features a dark-themed background with a code editor on the left showing Java code for an Android application. The code includes package declarations, imports for Android resources and Java locales, and a class named 'LanguageHelper'. A laptop in the foreground displays a mobile application interface with several clock widgets. To the right is a large green Android robot logo with a white hexagonal pattern on its chest. At the bottom right, there is a logo for 'Android Mobile Application Development' consisting of a green circle with a white 'A' and the text 'Android Mobile Application Development' in white.

Post Graduate Diploma in Mobile Application Development

2019

Software Lab for Android Mobile Application Development

Dr. Babasaheb Ambedkar Open University



Software Lab for Android Mobile Application Development

Course Writer

Dr. Himanshu Patel Assistant Professor, School of Computer Science,
Dr. Babasaheb Ambedkar Open University, Ahmedabad

Content Reviewer

Mr. Nikhil Patel Manager, CPP Software, Bhavnagar

Editors

Prof. (Dr.) Nilesh Modi Professor and Director, School of Computer Science,
Dr. Babasaheb Ambedkar Open University, Ahmedabad
Dr. Himanshu Patel Assistant Professor, School of Computer Science,
Dr. Babasaheb Ambedkar Open University, Ahmedabad

Copyright © Dr. Babasaheb Ambedkar Open University – Ahmedabad. June 2019



This publication is made available under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

ISBN: 978-81-940577-4-7

Printed and published by: Dr. Babasaheb Ambedkar Open University, Ahmedabad

While all efforts have been made by editors to check accuracy of the content, the representation of facts, principles, descriptions and methods are that of the respective module writers. Views expressed in the publication are that of the authors, and do not necessarily reflect the views of Dr. Babasaheb Ambedkar Open University. All products and services mentioned are owned by their respective copyrights holders, and mere presentation in the publication does not mean endorsement by Dr. Babasaheb Ambedkar Open University. Every effort has been made to acknowledge and attribute all sources of information used in preparation of this learning material. Readers are requested to kindly notify missing attribution, if any.



BAOU
Education
for All

**Dr. Babasaheb
Ambedkar Open
University**

PGMAD-105

**Software Lab for
Android Mobile Application Development**

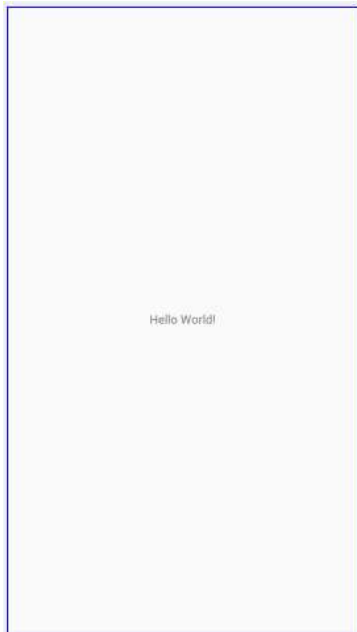
List of Programs

Practical-1: Activity Life Cycle	01
Practical-2: TextView, EditText and Button	04
Practical-3: Multiple Activities and Intent	09
Practical-4: Checkbox and Radio Button	16
Practical-5: Date Time Dialog	22
Practical-6: Spinner-1	27
Practical-7: Spinner-2	31
Practical-8: List Activity	35
Practical-9: Intent	38
Practical-10: Draw Shape Programmatically	41
Practical-11: Draw Shape Defined as an XML Resource	46
Practical-12: Tween Animation	49
Practical-13: Frame by Frame Animation	58
Practical-14: Auto Complete Text View	61
Practical-15: Currency Converter	64
Practical-16: Multiple Choice List Activity	69
Practical-17: Gallery View	72
Practical-18: Service	76
Practical-19: Alert Dialogbox	81
Practical-20: Multiple Activities	88
21: Self Assessment Exercises and Additional Resources	94

Practical-1: Activity Life Cycle

1

Create an application with default layout as shown below to demonstrate Android Activity Life Cycle.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - a) **Name:** "Activity Life Cycle"
 - b) **Package name:** "in.edu.baou.activitylifecycle"
 - c) Select language as Java
 - d) Change the project location as per your requirement.
5. Click **Finish**.
6. Write following XML code to design required activity_main.xml layout file

```

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

<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" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

7. Replace MainActivity.java file with following code

```
package in.edu.baou.activitylifecycle;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
import android.util.Log;
```

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

```
    @Override
```

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

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

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

```
        Log.d("Event", "In the onCreate() event");
```

```
    }
```

```
    @Override
```

```
    protected void onStart() {
```

```
        super.onStart();
```

```
        Log.d("Event", "In the onStart() event");
```

```
    }
```

```
    @Override
```

```
    protected void onResume() {
```

```
        super.onResume();
```

```
        Log.d("Event", "In the onResume() event");
```

```
    }
```

```
    @Override
```

```
protected void onPause() {
    super.onPause();
    Log.d("Event", "In the onPause() event");
}

@Override
protected void onStop() {
    super.onStop();
    Log.d("Event", "In the onStop() event");
}

@Override
protected void onRestart() {
    super.onRestart();
    Log.d("Event", "In the onRestart() event");
}

@Override
protected void onDestroy() {
    super.onDestroy();
    Log.d("Event", "In the onDestroy() event");
}
}
```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you
9. When the activity is first loaded, you should see the message depending on the event occur in the LogCat window.

Practical-2: TextView, EditText and Button

2

Write a mobile application to create simple calculator with interface as shown below and implement following functionalities:

- When user press ADDITION Button add two numbers and show result
- When user press SUBTRACT Button subtract two numbers and show result
- When user press MULTIPLY Button multiply two numbers and show result
- When user press DIVIDE Button divide number 1 by numbers 2 and show result



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - a) **Name:** "Simple Calc"
 - b) **Package name:** "in.edu.baou.simplecalc"
 - c) Select language as Java
 - d) Change the project location as per your requirement.

5. Click **Finish**.
6. Write following XML code to design required activity_main.xml layout file

```
<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">

    <TextView android:text="@string/app_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="30dp"
        android:id="@+id/textView1"
        android:textColor="#FF0000" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Number 1"
        android:id="@+id/textView"
        android:layout_below="@+id/textView1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="69dp"
        android:textSize="25dp" />
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="numberDecimal"
        android:ems="10"
        android:id="@+id/editText"
        android:layout_alignTop="@+id/textView"
        android:layout_toRightOf="@+id/textView"
        android:layout_toEndOf="@+id/textView"
        android:textSize="25dp" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Number 2"
        android:id="@+id/textView2"
        android:textSize="25dp"
        android:layout_below="@+id/editText"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true" />
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:inputType="numberDecimal"
        android:ems="10"
        android:id="@+id/editText2"
```

```

        android:layout_alignTop="@+id/textView2"
        android:layout_toRightOf="@+id/textView2"
        android:layout_toEndOf="@+id/textView2"
        android:textSize="25dp" />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Result"
    android:id="@+id/textView3"
    android:textSize="25dp"
    android:textColor="#00FF00"
    android:layout_below="@+id/editText2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="numberDecimal"
    android:ems="10"
    android:id="@+id/editText3"
    android:layout_alignBottom="@+id/textView3"
    android:layout_alignLeft="@+id/editText2"
    android:layout_alignStart="@+id/editText2"
    android:textSize="25dp"
    android:editable="false"
    android:enabled="false"
    android:clickable="false" />
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Addition"
    android:id="@+id/button"
    android:layout_below="@+id/editText3"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Subtract"
    android:id="@+id/button2"
    android:layout_above="@+id/button3"
    android:layout_centerHorizontal="true" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Multiply"
    android:id="@+id/button3"
    android:layout_below="@+id/button"
    android:layout_alignParentLeft="true"

```

```

        android:layout_alignParentStart="true" />
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Divide"
    android:id="@+id/button4"
    android:layout_alignTop="@+id/button3"
    android:layout_alignLeft="@+id/button2"
    android:layout_alignStart="@+id/button2" />

</RelativeLayout>

```

7. Replace MainActivity.java file with following code

```

package in.edu.baou.simplecalc;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    Button add,sub,mul,div;
    EditText no1,no2,no3;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        add = (Button) findViewById(R.id.button);
        sub = (Button) findViewById(R.id.button2);
        mul = (Button) findViewById(R.id.button3);
        div = (Button) findViewById(R.id.button4);

        no1 = (EditText)findViewById(R.id.editText);
        no2 = (EditText)findViewById(R.id.editText2);
        no3 = (EditText)findViewById(R.id.editText3);

        add.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double n1,n2,res;
                n1 = Double.parseDouble(no1.getText().toString());
                n2 = Double.parseDouble(no2.getText().toString());
                res=n1+n2;
                no3.setText(String.valueOf(res));
            }
        }
    }
}

```

```

});

sub.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        double n1,n2,res;
        n1 = Double.parseDouble(no1.getText().toString());
        n2 = Double.parseDouble(no2.getText().toString());
        res=n1-n2;
        no3.setText(String.valueOf(res));
    }
});

mul.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        double n1, n2, res;
        n1 = Double.parseDouble(no1.getText().toString());
        n2 = Double.parseDouble(no2.getText().toString());
        res = n1 * n2;
        no3.setText(String.valueOf(res));
    }
});

div.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        double n1, n2, res;
        n1 = Double.parseDouble(no1.getText().toString());
        n2 = Double.parseDouble(no2.getText().toString());
        res = n1 / n2;
        no3.setText(String.valueOf(res));
    }
});
}
}
}

```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-3: Multiple Activities and Intent

3

Write a mobile application to create login interface as shown below and implement following functionalities:

- If username is "abc@baou.edu.in" and password is "abc@123" then enable login button and when login button is pressed open another activity and display welcome <user-name> message
- If username or password is not correct then login button should remain disabled



Enter User Name (Email)

Enter Password

LOGIN



Welcome

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - a) **Name:** "Login Demo"
 - b) **Package name:** "in.edu.baou.logindemo"

- c) Select language as Java
 - d) Change the project location as per your requirement.
5. Click **Finish**.
 6. Write following XML code to design required activity_main.xml layout file

```
<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: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=".LoginActivity"
    android:orientation="vertical"
    android:id="@+id/linearlayout1"
    android:background="#fffc3fc"
    android:baselineAligned="false">
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Enter User Name (Email) "
    android:id="@+id/textView"
    android:textColor="#fff1e17" />
```

```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textEmailAddress"
    android:ems="10"
    android:id="@+id/txtUserID" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Enter Password"
    android:id="@+id/textView2"
    android:textColor="#fff1e17" />
```

```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/txtPassword" />
```

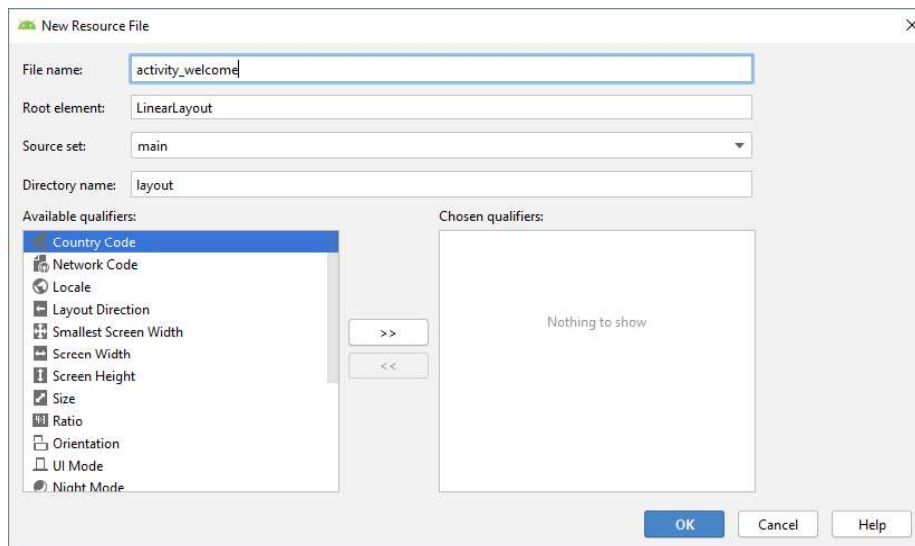
```
<Button
    android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:text="Login"
        android:id="@+id/btnLogin"
        android:enabled="false"
        android:background="#ffffffc3"
        android:textColor="#ffff1e17" />
</LinearLayout>

```

7. Create new layout for welcome activity by right clicking on app→res→layout and select New→ Layout Resource File



8. Write following XML code to design required activity_welcome.xml layout file.

```

<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="in.edu.baou.logindemo.WelcomeActivity"
    android:background="#ffffffc3">

```

```

    <TextView android:text="Welcome" android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/txtWelcome"
        android:textColor="#ffff1e17" />
</RelativeLayout>

```

9. Replace MainActivity.java file with following code

```
package in.edu.baou.logindemo;

import android.content.Intent;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText uname,pwd;
    Button login;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        uname = (EditText)findViewById(R.id.txtUserID);
        pwd = (EditText) findViewById(R.id.txtPassword);
        login = (Button) findViewById(R.id.btnLogin);

        uname.setOnKeyListener(new View.OnKeyListener() {
            @Override
            public boolean onKey(View v, int keyCode, KeyEvent event) {
                String u="abc@baou.edu.in";
                String p="abc@123";
                if(keyCode== KeyEvent.KEYCODE_DEL || keyCode==
                KeyEvent.KEYCODE_BACK) {
                    login.setEnabled(false);
                    return false;
                }
                else if(uname.getText().toString().equals(u) &&
                pwd.getText().toString().equals(p))
                {
                    login.setEnabled(true);
                    return true;
                }
                else
                {
                    return false;
                }
            }
        });
    }
}
```



```

pwd.setOnKeyListener(new View.OnKeyListener() {
    @Override
    public boolean onKey(View v, int keyCode, KeyEvent event) {
        String u="abc@baou.edu.in";
        String p="abc@123";
        if(keyCode== KeyEvent.KEYCODE_DEL || keyCode==
KeyEvent.KEYCODE_BACK) {
            login.setEnabled(false);
            return false;
        }
        else if(uname.getText().toString().equals(u) &&
pwd.getText().toString().equals(p))
        {
            login.setEnabled(true);
            return true;
        }
        else
        {
            return false;
        }
    }
});

```

```

login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent i;
        i = new Intent(getApplicationContext(),WelcomeActivity.class);
        i.putExtra("username",uname.getText().toString());
        startActivity(i);
    }
});
}

```

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {

    getMenuInflater().inflate(R.menu.menu_login, menu);
    return true;
}

```

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {

    int id = item.getItemId();

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

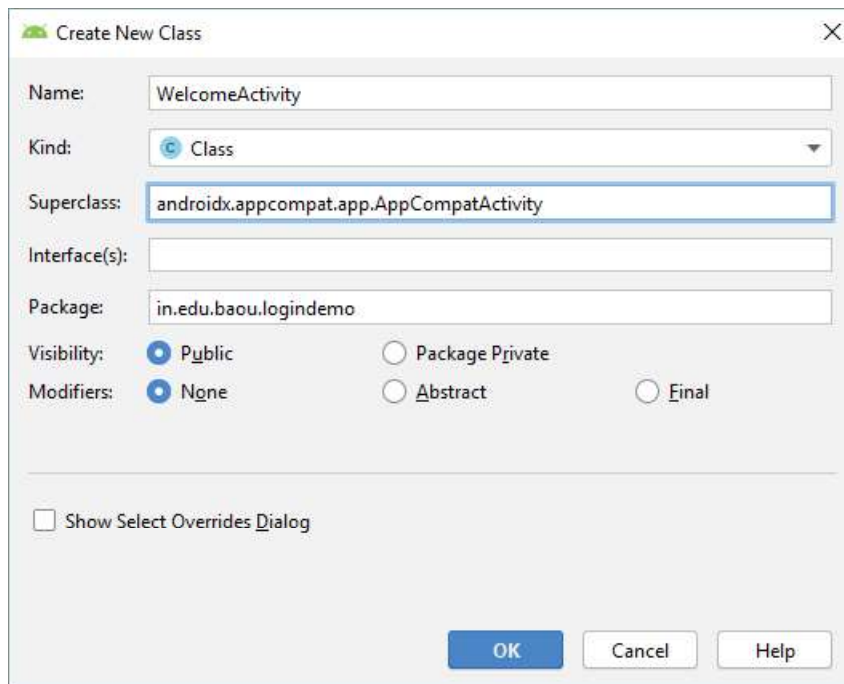
```

```

    }
    return super.onOptionsItemSelected(item);
}
}

```

10. Right clicking on app→java→ in.edu.baou.logindemo and enter following details to create WelcomeActivity class



11. Enter following code in WelcomeActivity.java file

```
package in.edu.baou.logindemo;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.TextView;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
public class WelcomeActivity extends AppCompatActivity {
    TextView uname;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_welcome);
        uname=(TextView)findViewById(R.id.txtWelcome);
        Intent i = getIntent();
    }
}

```

```

        uname.setText("Welcome "+i.getStringExtra("username"));
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_welcome, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

        int id = item.getItemId();

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

        return super.onOptionsItemSelected(item);
    }
}

```

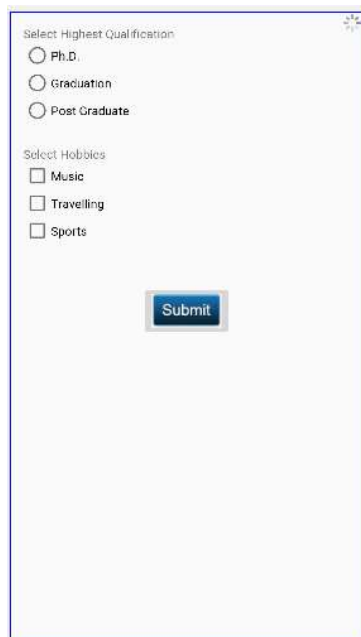
- Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-4: Checkbox and Radio Button

4

Create mobile interface similar to shown in figure given below with following functionalities.

- 1) When user press the submit button app should display highest qualification and hobbies selected by the user in Toast.



Select Highest Qualification

Ph.D.

Graduation

Post Graduate

Select Hobbies

Music

Travelling

Sports

Submit

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Personnel Information"
 - 2) **Package name:** "in.edu.baou. personalinformation"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.

5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```
<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">
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Highest Qualification"
    android:id="@+id/textView"
    android:layout_alignParentTop="true" />
```

```
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:id="@+id/radioGroup">
```

```
<RadioButton
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Ph.D."
    android:id="@+id/radioButton3"
    android:checked="false"
    android:layout_below="@+id/radioGroup"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" />
```

```
<RadioButton
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Graduation"
    android:id="@+id/radioButton1"
    android:layout_below="@+id/textView"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:checked="false" />
```

```
<RadioButton
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
    android:text="Post Graduate"
    android:id="@+id/radioButton2"
    android:layout_below="@+id/radioButton1"
    android:checked="false" />
```

```
</RadioGroup>
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Hobbies"
    android:id="@+id/textView2"
    android:layout_below="@+id/radioGroup"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_marginTop="25dp" />
```

```
<CheckBox
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Music"
    android:id="@+id/checkbox1"
    android:checked="false"
    android:layout_below="@+id/textView2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<CheckBox
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Sports"
    android:id="@+id/checkbox2"
    android:checked="false"
    android:layout_below="@+id/checkbox3"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<CheckBox
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Travelling"
    android:id="@+id/checkbox3"
    android:checked="false"
    android:layout_below="@+id/checkbox1"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<ImageButton
    android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
android:id="@+id/imageButton1"
android:src="@drawable/submit"
android:layout_below="@+id/checkbox2"
android:layout_centerHorizontal="true"
android:layout_marginTop="48dp" />
```

</RelativeLayout>

7. Replace MainActivity.java file with following code

```
package in.edu.baou.personalinformation;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.CheckBox;
import android.widget.ImageButton;
import android.widget.RadioButton;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    RadioButton r1,r2,r3;
    CheckBox c1,c2,c3;
    ImageButton b1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        r1 = (RadioButton) findViewById(R.id.radioButton1);
        r2 = (RadioButton) findViewById(R.id.radioButton2);
        r3 = (RadioButton) findViewById(R.id.radioButton3);

        c1 = (CheckBox) findViewById(R.id.checkBox1);
        c2 = (CheckBox) findViewById(R.id.checkBox2);
        c3 = (CheckBox) findViewById(R.id.checkBox3);

        b1 = (ImageButton)findViewById(R.id.imageButton1);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String msg="";

                if(r1.isChecked())
```

```

    {
        msg="Qualification : "+r1.getText().toString();
    }
    else if(r2.isChecked())
    {
        msg="Qualification : "+r2.getText().toString();
    }
    else if(r3.isChecked())
    {
        msg="Qualification : "+r3.getText().toString();
    }
}

msg=msg+"\nHobby : ";
if(c1.isChecked())
{
    msg=msg+c1.getText().toString()+" ";
}
if(c2.isChecked())
{
    msg=msg+c2.getText().toString()+" ";
}
if(c3.isChecked())
{
    msg=msg+c3.getText().toString()+" ";
}
}

```

Toast.makeText(getApplicationContext(),msg,Toast.LENGTH_LONG).show();

```

    }
});
}

```

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_display, menu);
    return true;
}

```

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {

    int id = item.getItemId();

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

    return super.onOptionsItemSelected(item);
}

```



```
}  
}
```

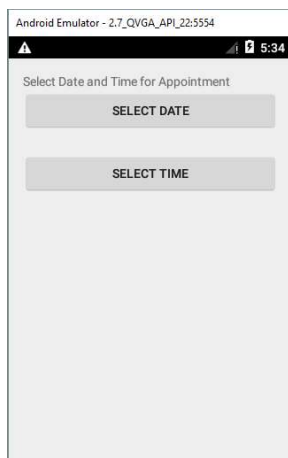
8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-5: Date and Time Dialog

5

Create mobile interface similar to shown in figure given below with following functionalities.

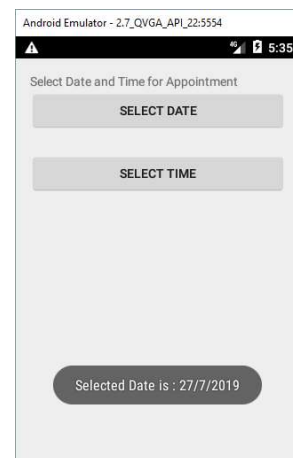
- 1) When user press the **SELECT DATE** button (figure a) app should display date dialog (figure b) and when user select date and press OK button it should display selected date in Toast (figure c).
- 2) When user press the **SELECT TIME** button (figure a) app should display time dialog (figure d) and when user select time and press OK button it should display selected time in Toast (figure e).
- 3) If user CANCEL button the dialog should be closed and return to home screen



(a)



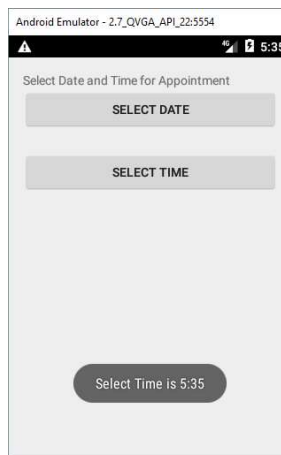
(b)



(c)



(d)



(e)

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 5) **Name:** "Personnel Information"
 - 6) **Package name:** "in.edu.baou.appointmentdialog"
 - 7) Select language as Java
 - 8) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```
<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=".ActivityMain">
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Date and Time for Appointment"
    android:id="@+id/textView"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Date"
    android:id="@+id/btnDate"
    android:layout_below="@+id/textView"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<Button
    android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:text="Select Time"
        android:id="@+id/btnTime"
        android:layout_below="@+id/btnDate"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginTop="23dp" />
</RelativeLayout>

```

7. Replace MainActivity.java file with following code

```
package in.edu.baou.appointmentdialog;
```

```

import android.app.Activity;
import android.app.DatePickerDialog;
import android.app.Dialog;
import android.app.TimePickerDialog;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.TimePicker;
import android.widget.Toast;

```

```
import java.util.Calendar;
```

```
public class ActivityMain extends Activity {
```

```

    Button date;
    Button time;
    int dd,mm,yy,hr,mi;

```

```
@Override
```

```

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

```

```

    date = (Button)findViewById(R.id.btnDate);
    time = (Button)findViewById(R.id.btnTime);

```

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

```
@Override
```

```

public void onClick(View v) {
    Calendar today= Calendar.getInstance();
    dd = today.get(Calendar.DAY_OF_MONTH);
    mm = today.get(Calendar.MONTH);

```

```

        yy = today.get(Calendar.YEAR);
        showDialog(0);
    }
});

time.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Calendar today= Calendar.getInstance();
        hr = today.get(Calendar.HOUR);
        mi = today.get(Calendar.MINUTE);
        showDialog(1);
    }
});
}

DatePickerDialog.OnDateSetListener dateSetListener = new
DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker view, int year, int monthOfYear, int
dayOfMonth) {
        String msg = "Selected Date is :
"+dayOfMonth+"/"+monthOfYear+"/"+year;

Toast.makeText(getApplicationContext(),msg,Toast.LENGTH_LONG).show();
    }
};

TimePickerDialog.OnTimeSetListener timeSetListener = new
TimePickerDialog.OnTimeSetListener() {
    @Override
    public void onTimeSet(TimePicker view, int hourOfDay, int minute) {
        String msg = "Select Time is "+hourOfDay+": "+minute;

Toast.makeText(getApplicationContext(),msg,Toast.LENGTH_LONG).show();
    }
};

@Override
protected Dialog onCreateDialog(int id) {
    switch (id)
    {
        case 0 : return new DatePickerDialog(this,dateSetListener,yy,mm,dd);
        case 1 : return new TimePickerDialog(this,timeSetListener,hr,mi,true);
    }
    return null;
}

@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_appointment_dialog, 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);
    }
}

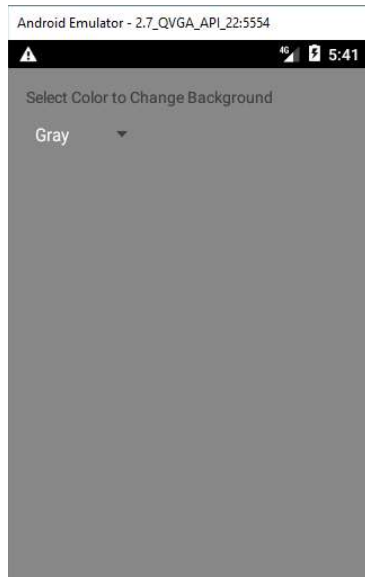
```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-6: Spinner-1

6

Create mobile interface similar to shown in figure given below with following functionalities



- 1) The spinner should display list of standard colors defined as string array in string.xml file
- 2) Whenever user selects color from spinner, it should change the background color as per user choice.

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 9) **Name:** "Spinner Demo"
 - 10) **Package name:** "in.edu.baou.spinnerdemo"
 - 11) Select language as Java
 - 12) Change the project location as per your requirement.
5. Click finish

6. Write following XML code to design required activity_main.xml layout file

```
<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:id="@+id/RelativeLayout1"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">

    <TextView android:text="@string/title"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/textview1" />

    <Spinner
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/spinner1"
        android:layout_below="@+id/textview1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:prompt="@string/title" />

</RelativeLayout>
```

7. Write following code to string.xml file in values folder

```
<resources>
    <string name="app_name">SpinnerDemo</string>
    <string name="hello_world">Hello world!</string>
    <string name="action_settings">Settings</string>
    <string name="title">Select Color to Change Background</string>
    <string-array name="clr">
        <item>Red</item>
        <item>Green</item>
        <item>Blue</item>
        <item>Cyan</item>
        <item>Magenta</item>
        <item>Yellow</item>
        <item>Black</item>
        <item>White</item>
        <item>Gray</item>
    </string-array>
</resources>
```


8. Replace MainActivity.java file with following code

```
package in.edu.baou.spinnerdemo;

import android.app.Activity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.RelativeLayout;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends Activity {

    Spinner bkgrnd;
    String colornames[];
    int
    colors[]={Color.RED,Color.GREEN,Color.BLUE,Color.CYAN,Color.MAGENTA
,Color.YELLOW,Color.BLACK,Color.WHITE,Color.GRAY};
    RelativeLayout relativeLayout;

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

        colornames =getResources().getStringArray(R.array.clr);
        relativeLayout = (RelativeLayout)findViewById(R.id.relativelayout1);

        bkgrnd = (Spinner)findViewById(R.id.spinner1);
        bkgrnd.setAdapter(new ArrayAdapter<String>(getApplicationContext(),
android.R.layout.simple_spinner_dropdown_item, colornames));

        bkgrnd.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
                relativeLayout.setBackgroundColor(colors[position]);
            }

            @Override
            public void onNothingSelected(AdapterView<?> parent) {

            }
        }
    }
}
```

```

    });
}

@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);
}
}
}

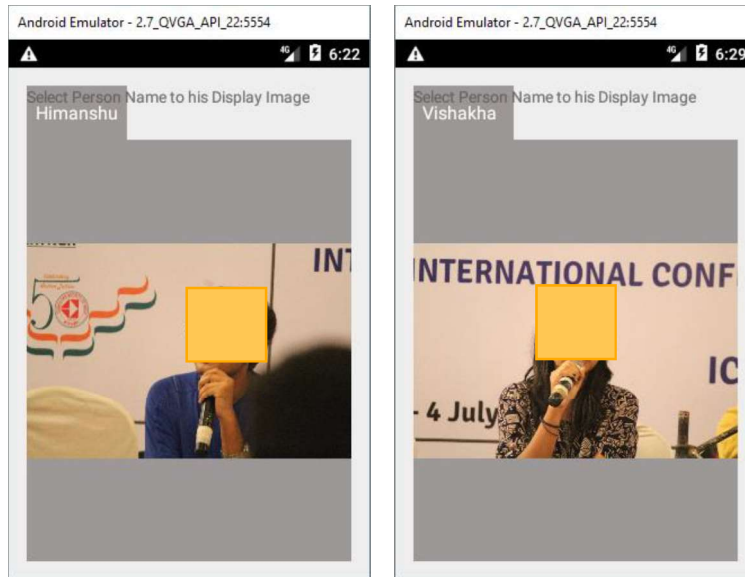
```

9. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-7: Spinner-2

7

Create mobile interface similar to shown in figure given below with following functionalities



- 1) The spinner should display list of person names defined as string array in string.xml file
- 2) Whenever user selects person name from spinner, it should display his/her image available in drawable image resource

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 13) **Name:** "Spinner Demo"
 - 14) **Package name:** "in.edu.baou.spinnerdemo"
 - 15) Select language as Java
 - 16) Change the project location as per your requirement.
5. Click finish

6. Write following XML code to design required activity_main.xml layout file

```
<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="in.edu.baou.spinnerdemo.MainActivity">

    <TextView android:text="Select Person Name to his Display Image"
    android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/texview1" />

    <Spinner
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/spinner1"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:background="#95615A5A"/>

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/spinner1"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:background="#95615A5A" />

</RelativeLayout>
```

7. Write following code to string.xml file in values folder

```
<resources>
    <string name="app_name">SpinnerDemo</string>
    <string name="hello_world">Hello world!</string>
    <string name="action_settings">Settings</string>
    <string name="title">Select Color to Change Background</string>
    <string-array name="persons">
        <item>Vishakha</item>
        <item>Himanshu</item>
        <item>Nikhil</item>
        <item>Rutvij</item>
        <item>Akshar</item>
    </string-array>
</resources>
```

```

    </string-array>
    <string name="title_activity_main_activity2">MainActivity2</string>
</resources>

```

8. Copy persons image in res → drawable folder

9. Replace MainActivity.java file with following code

```
package in.edu.baou.spinnerdemo;
```

```

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ImageView;
import android.widget.Spinner;

```

```

public class MainActivity extends Activity {
    Spinner s;
    ImageView iv;

    int
    images[]={R.drawable.b,R.drawable.a,R.drawable.c,R.drawable.d,R.drawable
.e};
    String persons[];
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        persons = getResources().getStringArray(R.array.persons);

        iv = (ImageView)findViewById(R.id.imageView1);
        s = (Spinner)findViewById(R.id.spinner1);
        s.setAdapter(new
        ArrayAdapter<String>(getApplicationContext(),android.R.layout.simple_spinn
er_dropdown_item,persons));

        s.setOnItemSelectedListener(new
        AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
                iv.setImageResource(images[position]);
            }
        }
    }
}

```

```

        @Override
        public void onNothingSelected(AdapterView<?> parent) {

        }
    });
}

@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_activity2, 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);
}
}

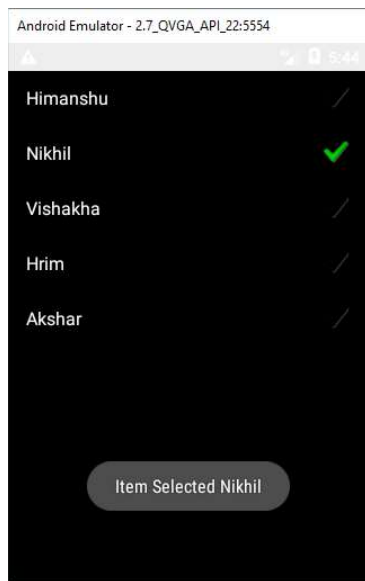
```

10. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-8: List Activity

8

Create mobile interface similar to shown in figure given below with following functionalities



- 1) The app should display list of person names defined as string array
- 2) Whenever user selects person name from list, it should display his/her name in Toast.

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "List Activity"
 - 2) **Package name:** "in.edu.baou.listactivity"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.
5. Click finish

6. Write following XML code to design required activity_main.xml layout file

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

    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/list"
        android:layout_alignParentTop="true" />
</RelativeLayout>
```

7. Replace MainActivity.java file with following code

```
package in.edu.baou.listactivity;

import android.app.ListActivity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.Toast;

public class MainActivity extends ListActivity {

    String person[]={ "Himanshu", "Nikhil", "Vishakha", "Hrim", "Akshar"};
    ListView lv;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        //setContentView(R.layout.activity_main);
        setListAdapter(new ArrayAdapter<String>(getApplicationContext(),
        android.R.layout.simple_list_item_checked, person));

        lv=getListView();
        lv.setBackgroundColor(Color.BLACK);
        lv.setTextFilterEnabled(true);
        lv.setChoiceMode(ListView.CHOICE_MODE_SINGLE);

        lv.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int
```



```

position, long id) {
    Toast.makeText(getApplicationContext(), "Item Selected " +
person[position], Toast.LENGTH_LONG).show();
    }
});

}

@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);
}
}
}

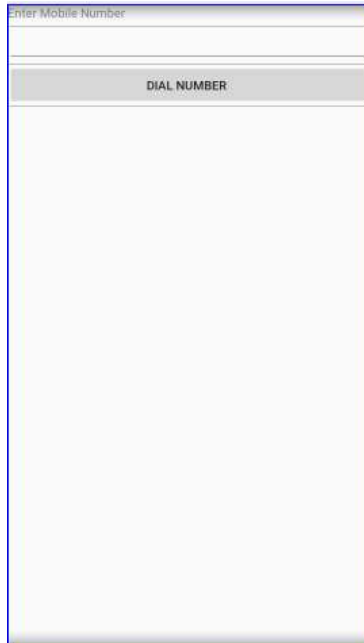
```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-9: Intent

9

Create an application to call specific entered number by user in the Edit Text



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 5) **Name:** "Call Activity"
 - 6) **Package name:** "in.edu.baou.callactivity"
 - 7) Select language as Java
 - 8) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file

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

```

xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".MainActivity">

<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="25dp"
    android:text="Enter Mobile Number" />

<EditText
    android:id="@+id/txtNo"
    android:layout_width="match_parent"
    android:layout_height="43dp"
    android:ems="10"
    android:inputType="phone" />

<Button
    android:id="@+id/btnDial"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Dial Number " />

</LinearLayout>

```

7. Replace MainActivity.java file with following code

```

package in.edu.baou.callactivity;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

    EditText no;
    Button dial;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

```

```

setContentView(R.layout.activity_main);

dial = (Button)findViewById(R.id.btnDial);
no = (EditText)findViewById(R.id.txtNo);

no.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent i = new Intent(Intent.ACTION_CALL);
        i.setData(Uri.parse("tel:"+no.getText()));
        startActivity(i);
    }
});
}
}

```

8. Add following permission to AndroidManifest.xml file

```

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

```

9. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

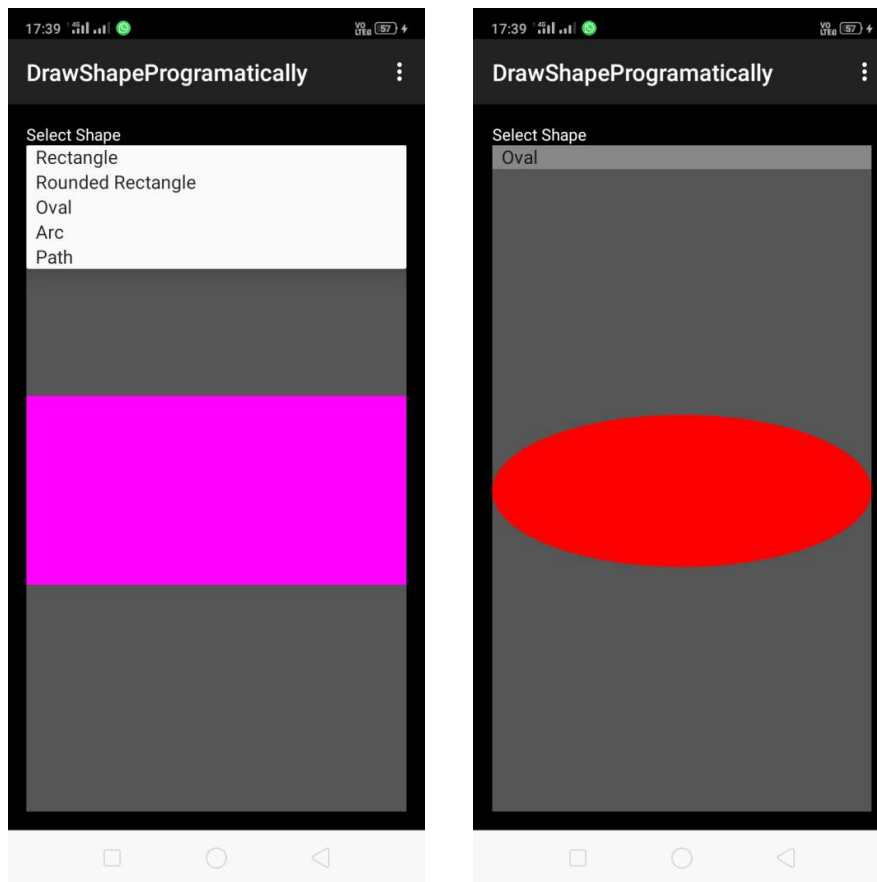
Note: You must explicitly grant permission for Phone Call where this app is installed.

Call requires permission which may be rejected by user: code should explicitly check to see if permission is available (with checkPermission) or explicitly handle a potential SecurityException

Practical-10: Draw Shape Programmatically

10

Create an application that designs a layout with spinner and an image view as shown below. The spinner should contain various shapes like circle, rectangle and rounded rectangle. When a shape is selected from the spinner that shape should be drawn in the image view. (Hint: Use shape Drawable or xml files for shapes on image view)



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "DrawShapeProgrammatically"

- 2) **Package name: "in.edu.baou.graphicsdemo"**
- 3) Select language as Java
- 4) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```

<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="in.edu.baou.graphicsdemo.DrawShapeProgramatically"
    android:background="#000">

    <TextView android:text="Select Shape"
    android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/textView2"
        android:textColor="#FFF" />

    <Spinner
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/spinner"
        android:layout_below="@+id/textView2"
        android:background="#AAA" />

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/imageView2"
        android:layout_below="@+id/spinner"
        android:background="#555" />

</RelativeLayout>

```

7. Replace MainActivity.java file with following code

```

package in.edu.baou.graphicsdemo;

import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Path;
import android.graphics.drawable.ShapeDrawable;
import android.graphics.drawable.shapes.ArcShape;

```

```

import android.graphics.drawable.shapes.OvalShape;
import android.graphics.drawable.shapes.PathShape;
import android.graphics.drawable.shapes.RectShape;
import android.graphics.drawable.shapes.RoundRectShape;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.Adapter;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.AdapterView.OnItemSelectedListener;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    Spinner spinner;
    ImageView imageView;
    String shapes[]={"Rectangle","Rounded Rectangle","Oval","Arc","Path"};

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

        spinner = (Spinner)findViewById(R.id.spinner);
        imageView = (ImageView) findViewById(R.id.imageView2);

        spinner.setBackgroundColor(Color.GRAY);
        spinner.setAdapter(new
        ArrayAdapter<String>(getApplicationContext(),android.R.layout.simple_spinner_item,shapes));

        spinner.setOnItemSelectedListener(new
        AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int
            position, long id) {
                switch (position) {
                    case 0:
                        ShapeDrawable rect =
                        new ShapeDrawable(new RectShape());
                        rect.setIntrinsicHeight(50);
                        rect.setIntrinsicWidth(100);
                        rect.getPaint().setColor(Color.MAGENTA);
                        imageView.setImageDrawable(rect);
                        break;
                }
            }
        });
    }
}

```

```

case 1:
    ShapeDrawable rndrect =
        new ShapeDrawable(new RoundRectShape( new float[] { 5, 5,
            5, 5, 5, 5, 5, 5 },null, null));
    rndrect.setIntrinsicHeight(50);
    rndrect.setIntrinsicWidth(100);
    rndrect.getPaint().setColor(Color.CYAN);
    imageView.setImageDrawable(rndrect);
    break;
case 2:
    ShapeDrawable oval =
        new ShapeDrawable(new OvalShape());
    oval.setIntrinsicHeight(40);
    oval.setIntrinsicWidth(100);
    oval.getPaint().setColor(Color.RED);
    imageView.setImageDrawable(oval);
    break;
case 3:
    ShapeDrawable arc =
        new ShapeDrawable(new ArcShape(0, 345));
    arc.setIntrinsicHeight(100);
    arc.setIntrinsicWidth(100);
    arc.getPaint().setColor(Color.CYAN);
    imageView.setImageDrawable(arc);
    break;
case 4:
    Path p = new Path();
    p.moveTo(50, 0);
    p.lineTo(25, 100);
    p.lineTo(100, 50);
    p.lineTo(0, 50);
    p.lineTo(75, 100);
    p.lineTo(50, 0);
    ShapeDrawable star =
        new ShapeDrawable(new PathShape(p, 100, 100));
    star.setIntrinsicHeight(100);
    star.setIntrinsicWidth(100);
    star.getPaint().setColor(Color.YELLOW);
    star.getPaint().setStyle(Paint.Style.STROKE);
    imageView.setImageDrawable(star);
    break;
    }
}
@Override
public void onNothingSelected(AdapterView<?> parent) {
}
});

```

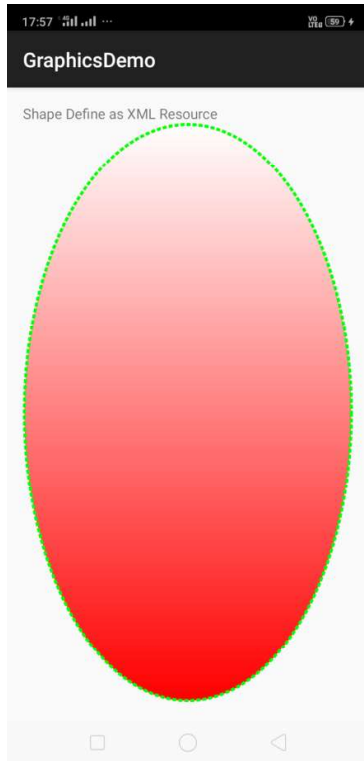

}

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-11: Draw Shape Defined as an XML Resource

11

Create an application that load shape defined as an XML resource and display it in an image view as shown below.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Graphics Demo"
 - 2) **Package name:** "in.edu.baou.graphicsdemo"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.

5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```

<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=".DrawingXMLResource">

    <TextView android:text="Shape Define as XML Resource"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/textView" />

    <ImageView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/imageView"
    android:layout_below="@+id/textView" />

</RelativeLayout>

```

7. Create following drawabpe resource file oval.xml by right click on app > resource> drawable and select New>Drawable resource file option

```

<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="oval">
    <solid android:color="#F00"></solid>
    <stroke android:color="#0F0" android:width="3dp" android:dashGap="2dp"
    android:dashWidth="4dp"></stroke>
    <gradient android:angle="90" android:startColor="#F00"
    android:endColor="#FFF"></gradient>
</shape>

```

8. Replace MainActivity.java file with following code

```

package in.edu.baou.graphicsdemo;

import android.os.Bundle;
import android.widget.ImageView;
import androidx.appcompat.app.AppCompatActivity;

```

```
public class MainActivity extends AppCompatActivity {
    ImageView iv;

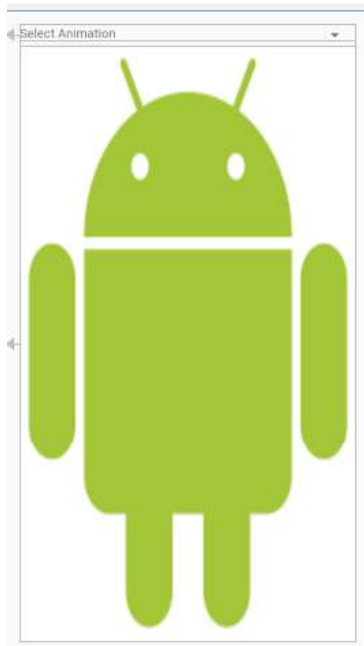
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        iv = (ImageView)findViewById(R.id.imageView);
        iv.setImageResource(R.drawable.oval);
    }
}
```

9. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-12: Tween Animation

12

Create an application that will have spinner with list of animation names as shown below. On selecting animation name, that animation should affect on the images displayed below.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "AnimationDemo"
 - 2) **Package name:** "in.edu.baou.animationdemo"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.
5. Click finish

6. Write following XML code to design required activity_main.xml layout file

```
<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=".Animation">

    <TextView android:text="Select Animation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/textview" />

    <Spinner
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/spinner"
    android:spinnerMode="dialog" />

    <ImageView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/imageView"
    android:layout_below="@+id/spinner"
    android:background="@drawable/android_logo" />

</RelativeLayout>
```

7. Create folder with “anim” name inside app>res folder.

8. Add following animation resource files by right click on app > res > anim folder and select New>Animation resource file option with file name given below one by one.

blink.xml

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

bouncing.xml

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

composite.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true"
    android:interpolator="@android:anim/linear_interpolator" >

    <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
        android:duration="500"
        android:fromDegrees="0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:repeatCount="infinite"
        android:repeatMode="restart"
        android:toDegrees="360" />

</set>
```

rotation.xml

```

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

```

fadein.xml

```

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

```

fadout.xml

```

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

```

onebyone.xml

```

<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
  android:fillAfter="true"
  android:interpolator="@android:anim/linear_interpolator" >
  <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
    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>

```

rotation.xml

```

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

```

translate.xml

```

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

zoomin.xml

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

zoomout.xml

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



```

        case 1:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.bouncing);
            iv.startAnimation(animation);
            break;
        case 2:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.fadein);
            iv.startAnimation(animation);
            break;
        case 3:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.fadeout);
            iv.startAnimation(animation);
            break;
        case 4:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.rotation);
            iv.startAnimation(animation);
            break;
        case 5:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.translate);
            iv.startAnimation(animation);
            break;
        case 6:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.zoomin);
            iv.startAnimation(animation);
            break;
        case 7:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.zoomout);
            iv.startAnimation(animation);
            break;
        case 8:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.onebyone);
            iv.startAnimation(animation);
            break;
        case 9:
            animation =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.composite);
            iv.startAnimation(animation);
            break;
    }
}

```

```

@Override
public void onNothingSelected(AdapterView<?> parent) {

```

```

    }
  });
}

@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_animation, 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);
}
}

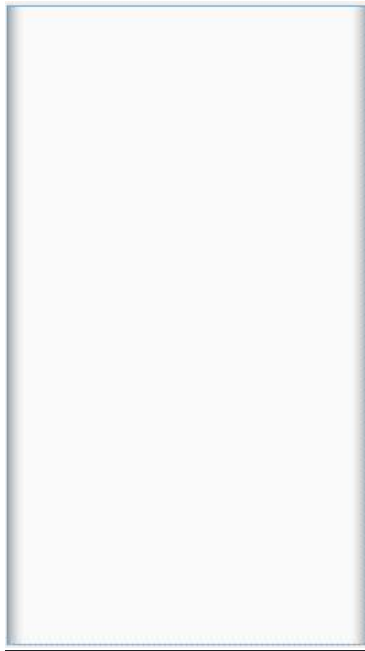
```

10. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-13: Frame by Frame Animation

13

Create an application that will start frame by frame animation defined in an XML file. The application has one ImageView that will host the animation and set its background to AnimationDrawable XML resource. The background should be changed at every one second.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Frame by Frame Animation"
 - 2) **Package name:** "in.edu.baou.framebyframeanimation"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.
5. Click finish

6. Write following XML code to design required activity_main.xml layout file

```
<?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" >

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_weight="1"
        app:srcCompat="@drawable/i2" />
</LinearLayout>
```

7. Add seven background images in app > res > drawable folder required for frame by frame animation with name i1.jpeg, i2.jpeg...i7.jpeg

8. Create animation file "framebyfram.xml" in app > res > drawable folder to display every image at one second one by one using frame by frame animation technique as shown below

```
<?xml version="1.0" encoding="utf-8"?>
<animation-list xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/selected" android:oneshot="false">
    <item android:drawable="@drawable/i1" android:duration="1000" />
    <item android:drawable="@drawable/i2" android:duration="1000" />
    <item android:drawable="@drawable/i3" android:duration="1000" />
    <item android:drawable="@drawable/i4" android:duration="1000" />
    <item android:drawable="@drawable/i5" android:duration="1000" />
    <item android:drawable="@drawable/i6" android:duration="1000" />
    <item android:drawable="@drawable/i7" android:duration="1000" />
</animation-list>
```

9. Replace MainActivity.java file with following code

```
package in.edu.baou.framebyframeanimation;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.drawable.AnimationDrawable;
import android.os.Bundle;
import android.widget.ImageView;
```

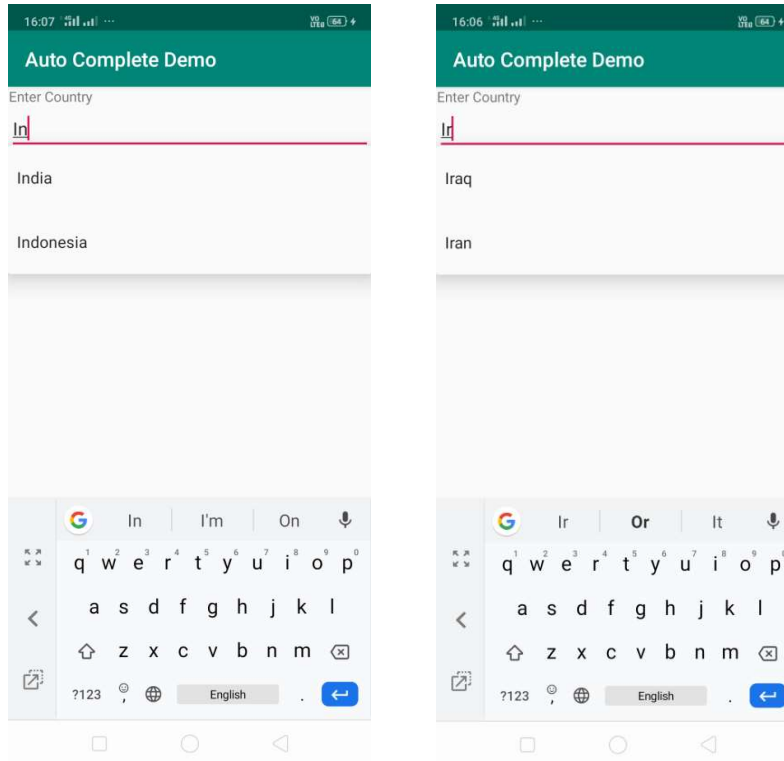
```
public class MainActivity extends AppCompatActivity {  
  
    ImageView img;  
    AnimationDrawable fbyf;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        img = (ImageView)findViewById(R.id.imageView2);  
        img.setBackgroundResource(R.drawable.framebyframe);  
  
        fbyf = (AnimationDrawable) img.getBackground();  
        fbyf.start();  
    }  
}
```

10. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-14: Auto Complete Text View

14

Create an auto complete text view which suggests various countries names while the user is typing as shown below.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Auto Complete Demo"
 - 2) **Package name:** "in.edu.baou.autocompletedemo"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.

5. Click finish

6. Write following XML code to design required activity_main.xml layout file

```
<?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="wrap_content"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Enter Country" />

    <AutoCompleteTextView
        android:id="@+id/autoCompleteTextView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="" />
</LinearLayout>
```

7. Replace MainActivity.java file with following code

```
package in.edu.baou.autocompletedemo;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.AutoCompleteTextView;

public class MainActivity extends AppCompatActivity {

    private static final String[] COUNTRY = new String[] {
        "India", "America", "Shri lanka", "Japan", "Russia", "Nepal",
        "Swizerland", "Belgium", "France", "Italy", "Germany", "Spain",
        "Bhutan", "Iraq", "Iran", "Saudi Arabia", "Poland", "Indonesia"
    };

    AutoCompleteTextView tv;
    ArrayAdapter<String> adapter;
    @Override
```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    adapter = new ArrayAdapter<String>  
        (this,android.R.layout.simple_dropdown_item_1line, COUNTRY);  
    tv = findViewById(R.id.autoCompleteTextView);  
    tv.setAdapter(adapter);  
  
    }  
}
```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-15: Currency Converter

15

Create an application that designs a layout having radio buttons and 2 text boxes for currency converter which allows user to select a particular conversion from following options.

- a) Rupees to dollars
- b) Dollars to Rupees
- c) Rupees to pound
- d) Pound to Rupees

The data for conversion should be entered by the user in textbox. When user clicking on the option, appropriate operation should be performed and correct result should be displayed. You can assume suitable rates for conversion between rupees and dollars and rupees and pounds and vice versa.



The screenshot shows a mobile application interface for a currency converter. At the top, there is a text input field labeled "Enter Amount". Below the input field, there are four radio button options: "Rupees to dollars", "Dollars to Rupees", "Rupees to pound", and "Pound to Rupees". A "CONVERT" button is positioned below the radio buttons. At the bottom of the interface, the conversion rates are displayed: "1 Dollar = 71.20 Indian Rupees" and "1 Pound = 88.75 Indian Rupee".

Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:

- 5) **Name:** "Currency Conversion"
 - 6) **Package name:** " in.edu.baou.currencyconversion"
 - 7) Select language as Java
 - 8) Change the project location as per your requirement.
5. Click finish
 6. Write following XML code to design required activity_main.xml layout file

```

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        tools:text="Enter Amoun" />

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Amount"
        android:inputType="number" />

    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <RadioButton
            android:id="@+id/rtod"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Rupees to Dollars" />

        <RadioButton
            android:id="@+id/dtor"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Dollars to Rupees" />
    </RadioGroup>

```

```

<RadioButton
    android:id="@+id/rtop"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Rupees to Pounds" />

<RadioButton
    android:id="@+id/ptor"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Pounds to Rupees" />
</RadioGroup>

<Button
    android:id="@+id/button"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Convert" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="1 Dollar = 71.20 Indian Rupees" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="1 Pound = 88.75 Indian Rupee" />

</LinearLayout>

```

7. Replace MainActivity.java file with following code

```

package in.edu.baou.currencyconversion;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

```

```

EditText amt;
Button con;
RadioButton rtod,dtor,rtop,ptor;
private static final double DOLLAR = 71.20;
private static final double POUND = 88.75;
double input,output;
String message;

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

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

    rtod = (RadioButton)findViewById(R.id.rtod);
    dtor = (RadioButton)findViewById(R.id.dtor);
    rtop = (RadioButton)findViewById(R.id.rtop);
    ptor = (RadioButton)findViewById(R.id.ptor);

    con.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if (rtod.isChecked()) {
                input= Double.parseDouble(amt.getText().toString());
                output=input/DOLLAR;
                message=input+" rupees is equals "+output+" dollars";

                Toast.makeText(getApplicationContext(),message,Toast.LENGTH_LONG).show();
            }
            else if (dtor.isChecked()){
                input= Double.parseDouble(amt.getText().toString());
                output=input*DOLLAR;
                message=input+" dollars is equals "+output+" rupees";

                Toast.makeText(getApplicationContext(),message,Toast.LENGTH_LONG).show();
            }
            else if (rtop.isChecked()){
                input= Double.parseDouble(amt.getText().toString());
                output=input/POUND;
                message=input+" rupees is equals "+output+" pounds";

                Toast.makeText(getApplicationContext(),message,Toast.LENGTH_LONG).show();
            }
            else if (ptor.isChecked()){

```

```
input= Double.parseDouble(amt.getText().toString());
output=input*POUND;
message=input+" pounds is equals "+output+" rupees";
```

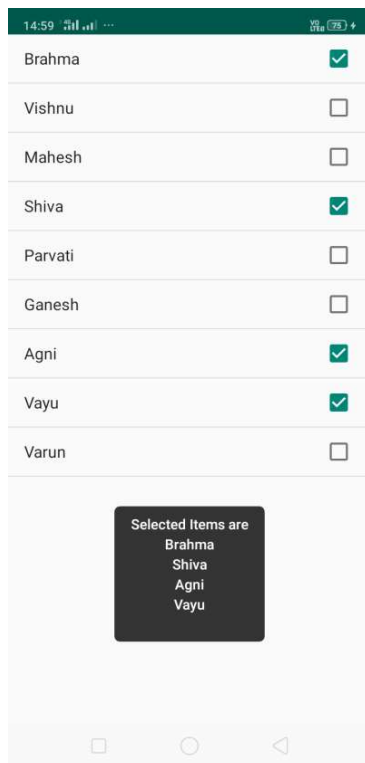
```
Toast.makeText(getApplicationContext(),message,Toast.LENGTH_LONG).show();
    }
    });
}
```

8. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-16: Multiple Choice List

16

Create List with strings taken from resource folder (res >> value folder) as shown below. On selecting item from list display in Toast, also provide facility for checking and un-checking of items.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Multiple Choice List"
 - 2) **Package name:** "in.edu.baou.multiplechoicelist"
 - 3) Select language as Java

- 4) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file
7. Replace MainActivity.java file with following code

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="in.edu.baou.multiplechoicelist.MultipleChoiceListActivity">

    <TextView android:text="Hello World" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

8. Create following entry in app > res > string.xml file

```
<resources>
    <string name="app_name">Multiple Choice List Demo</string>
    <string-array name="names">
        <item>Brahma</item>
        <item>Vishnu</item>
        <item>Mahesh</item>
        <item>Shiva</item>
        <item>Parvati</item>
        <item>Ganesh</item>
        <item>Agni</item>
        <item>Vayu</item>
        <item>Varun</item>
    </string-array>
</resources>
```

9. Replace MainActivity.java file with following code

```
package in.edu.baou.multiplechoicelist;

import android.app.ListActivity;
import android.graphics.Color;
import android.os.Bundle;
import android.util.SparseBooleanArray;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
```

```

import android.widget.Toast;

public class MultipleChoiceListActivity extends ListActivity {

    String names[];
    ListView lv;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        names = getResources().getStringArray(R.array.names);

        setListAdapter(new ArrayAdapter<String>(getApplicationContext(),
android.R.layout.simple_list_item_multiple_choice, names));

        lv = getListView();
        lv.setTextFilterEnabled(true);
        lv.setChoiceMode(ListView.CHOICE_MODE_MULTIPLE);

        lv.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
                String msg = "Selected Items are\n";
                SparseBooleanArray ids = lv.getCheckedItemPositions();
                for (int i = 0; i < ids.size(); i++) {
                    msg=msg+lv.getItemAtPosition(ids.keyAt(i))+"\n";
                }
                Toast.makeText(getApplicationContext(), msg,
Toast.LENGTH_SHORT).show();
            }
        });
    }
}

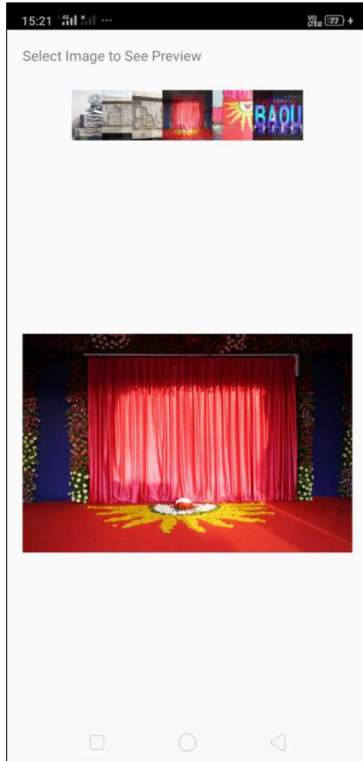
```

10. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-17: Gallery View

17

Create an application in which display image from app > res > drawable folder using Gallery View as shown below



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Image Gallery"
 - 2) **Package name:** " in.edu.baou.imagegallery"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.

5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```
<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">
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Image to See Preview"
    android:id="@+id/textView"
    android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<Gallery
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/gallery"
    android:layout_below="@+id/textView"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:minHeight="100dp" />
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/textView2"
    android:layout_below="@+id/gallery"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
<ImageView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/imageView"
    android:layout_below="@+id/textView2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true" />
```

```
</RelativeLayout>
```

7. Replace MainActivity.java file with following code

```
package in.edu.baou.imagegallery;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.BaseAdapter;
import android.widget.Gallery;
import android.widget.ImageView;

public class MainActivity extends Activity {

    Gallery gallery;
    ImageView iv;
    int
    ids[]={R.drawable.img1,R.drawable.img2,R.drawable.img3,R.drawable.img4,R
    .drawable.img5,R.drawable.img6,R.drawable.img7};

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

        gallery = (Gallery)findViewById(R.id.gallery);
        iv = (ImageView) findViewById(R.id.imageView);

        gallery.setAdapter(new BaseAdapter() {
            @Override
            public int getCount() {
                return ids.length;
            }

            @Override
            public Object getItem(int position) {
                return position;
            }

            @Override
            public long getItemId(int position) {
                return position;
            }

            @Override
            public View getView(int position, View convertView, ViewGroup
parent) {
```

```

        ImageView imageView = new ImageView(getApplicationContext());
        imageView.setImageResource(ids[position]);
        imageView.setScaleType(ImageView.ScaleType.FIT_XY);
        imageView.setLayoutParams(new Gallery.LayoutParams(100,
100));
        return imageView;
    }
});

    gallery.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
        @Override
        public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
            iv.setImageResource(ids[position]);
        }

        @Override
        public void onNothingSelected(AdapterView<?> parent) {

        }
    });
}
}
}

```

8. Copy seven Images of your choice into app > res > drawable folder with file name img1.jpeg, img2.jpeg, img3.jpeg, img4.jpeg, img5.jpeg, img6.jpeg and img7.jpeg
9. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-18: Service

18

Create an application that will display toast (Message) at every 3 seconds using service at shown below. The service should start when start timer service button is pressed and start displaying number from 1, 2, 3 ... and when stop timer service button is pressed the service should stop.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Service Demo"
 - 2) **Package name:** " in.edu.baou.servicedemo"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file

```
<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"
    tools:context=".ActivityMain">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Service Demonstration" />

    <Button
        android:id="@+id/btnStartTimer"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Start Timer Service" />

    <Button
        android:id="@+id/btnStopTimer"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Stop Timer Service" />

</LinearLayout>
```

7. Create class TimerService.java inside the main package with following code

```
package in.edu.baou.servicesdemo;

import android.app.Service;
import android.content.Intent;
import android.os.Handler;
import android.os.IBinder;
import android.util.Log;
import android.widget.Toast;

import java.util.Timer;
import java.util.TimerTask;

public class TimerService extends Service {

    int counter = 0;
    final Handler handler=new Handler();
    Runnable myrunnable;
    public TimerService() {
    }

    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        Toast.makeText(this, "Service Started!", Toast.LENGTH_LONG).show();
        myrunnable = new Runnable() {
            @Override
            public void run() {
                Toast.makeText(getApplicationContext(), String.valueOf(++counter),
Toast.LENGTH_LONG).show();
                handler.postDelayed(this,3000);
            }
        };
        handler.postDelayed(myrunnable,3000);
        return START_STICKY;
    }

    @Override
    public void onDestroy() {
        super.onDestroy();
        handler.removeCallbacks(myrunnable);
        Toast.makeText(this, "Service Destroyed!",
Toast.LENGTH_LONG).show();
    }
}
```

```
}  
}
```

8. Write following code inside MainActivity.java file

```
package in.edu.baou.servicesdemo;
```

```
import android.app.Activity;  
import android.content.BroadcastReceiver;  
import android.content.Context;  
import android.content.Intent;  
import android.content.IntentFilter;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;
```

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

```
    Button startTimer,stopTimer;
```

```
    @Override
```

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

```
        startTimer = (Button)findViewById(R.id.btnStartTimer);  
        stopTimer = (Button)findViewById(R.id.btnStopTimer);
```

```
        startTimer.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                startService(new Intent(getApplicationContext(), TimerService.class));  
            }  
        });
```

```
        stopTimer.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View v) {  
                stopService(new Intent(getApplicationContext(), TimerService.class));  
            }  
        });
```

```
    }  
}
```

9. Register TimerService in AndroidManifest.xml file using code given below

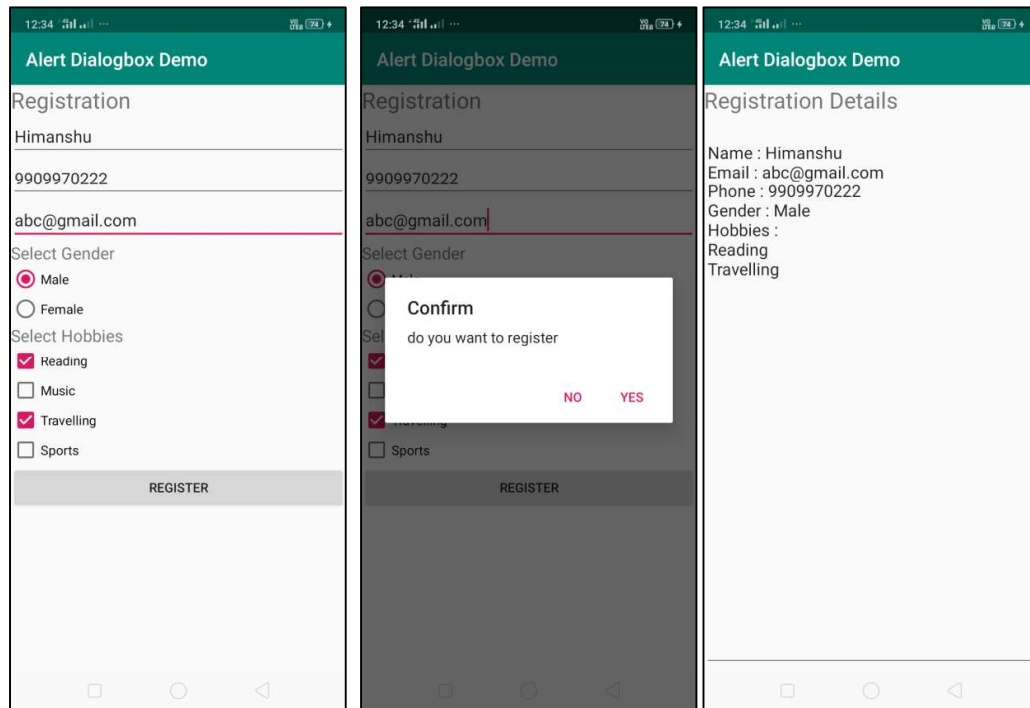
```
<service  
    android:name=".TimerService"  
    android:enabled="true"  
    android:exported="true" >  
</service>
```

10. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-19: Alert Dialog Box

19

Create an application that designs a layout of a student registration form. The layout should contain fields like full name, phone no, email, gender hobbies and a button register as shown in left side figure. On clicking register an alert dialog box should appear with a message “do you want to register” and with two buttons yes or no as shown in middle image. If yes is clicked then a new layout should appear where all the entered details should be displayed as shown in right figure. If no is clicked than the alert dialog box should disappear.



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**
3. Click **Next**.
4. In the **Configure your project** window, enter the following values:

- 5) **Name:** "Alert Dialog Demo"
 - 6) **Package name:** "in.edu.baou.alertdialogdemo"
 - 7) Select language as Java
 - 8) Change the project location as per your requirement.
5. Click finish
 6. Write following XML code to design required activity_main.xml layout file

```

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

    <TextView
        android:id="@+id/textView3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Registration"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/txtName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Full Name"
        android:inputType="textPersonName" />

    <EditText
        android:id="@+id/txtPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Phone"
        android:inputType="textPersonName" />

    <EditText
        android:id="@+id/txtEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Email"
        android:inputType="textPersonName" />

```

```

<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <TextView
        android:id="@+id/textView4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Select Gender"
        android:textSize="18sp" />

    <RadioButton
        android:id="@+id/optMale"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Male" />

    <RadioButton
        android:id="@+id/optFemale"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Female" />
</RadioGroup>

<TextView
    android:id="@+id/textView5"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Hobbies"
    android:textSize="18sp" />

<CheckBox
    android:id="@+id/chkReading"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Reading" />

<CheckBox
    android:id="@+id/chkMusic"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Music" />

<CheckBox
    android:id="@+id/chkTravelling"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Travelling" />

<CheckBox

```

```

        android:id="@+id/chkSports"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Sports" />

<Button
    android:id="@+id/btnRegistration"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Register" />
</LinearLayout>

```

7. Write following code in MainActivity.java file

```

package in.edu.baou.alertdialogboxdemo;

import androidx.appcompat.app.AppCompatActivity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.Toast;

import static android.provider.AlarmClock.EXTRA_MESSAGE;

public class MainActivity extends AppCompatActivity {

    Button register;
    RadioButton male,female;
    CheckBox reading,travelling,music,sports;
    EditText name,email,phone;
    String details;

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

        try {

            name = (EditText) findViewById(R.id.txtName);
            email = (EditText) findViewById(R.id.txtEmail);
            phone = (EditText) findViewById(R.id.txtPhone);

```



```

male = (RadioButton) findViewById(R.id.optMale);
female = (RadioButton) findViewById(R.id.optFemale);

reading = (CheckBox) findViewById(R.id.chkReading);
travelling = (CheckBox) findViewById(R.id.chkTravelling);
music = (CheckBox) findViewById(R.id.chkMusic);
sports = (CheckBox) findViewById(R.id.chkSports);

register = (Button) findViewById(R.id.btnRegistration);

register.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        AlertDialog.Builder builder = new
AlertDialog.Builder(MainActivity.this);
        AlertDialog alert;
        builder.setTitle("Confirm")
            .setMessage("do you want to register")
            .setCancelable(false)
            .setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialogInterface, int i) {
                    details="";
                    details=details+"Name :
"+name.getText().toString()+"\n";
                    details=details+"Email :
"+email.getText().toString()+"\n";
                    details=details+"Phone :
"+phone.getText().toString()+"\n";

                    if (male.isChecked())
                        details=details+"Gender :
"+male.getText().toString()+"\n";
                    else
                        details=details+"Gender :
"+female.getText().toString()+"\n";

                    details=details+"Hobbies : \n";
                    if(reading.isChecked())
                        details=details+reading.getText().toString()+"\n";
                    if(travelling.isChecked())
                        details=details+travelling.getText().toString()+"\n";
                    if(music.isChecked())
                        details=details+music.getText().toString()+"\n";
                    if(sports.isChecked())
                        details=details+sports.getText().toString()+"\n";

                    Intent intent = new Intent(getApplicationContext(),

```

```

ShowRegistration.class);
        intent.putExtra("DETAILS", details);
        startActivity(intent);
    }
}
    .setNegativeButton("No", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
        dialogInterface.cancel();
    }
});
    alert = builder.create();
    alert.show();
}
});
}
catch(Exception ex)
{
    Toast.makeText(getApplicationContext(), ex.getMessage(),
Toast.LENGTH_SHORT).show();
}
}
}
}

```

8. Add activity by right clicking on package name and select New > Activity > Empty Activity and give name of activity as ShowRegistration and press OK.
9. Write following code to design required activity_show_registration.xml layout file

```

<?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=".ShowRegistration">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:layout_weight="1"
        android:text="Registration Details"
        android:textSize="24sp" />

```

```
<EditText
    android:id="@+id/txtRegistration"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:ems="10"
    android:gravity="start|top"
    android:inputType="textMultiLine" />
</LinearLayout>
```

10. Write following code in ShowRegistration.java file

```
package in.edu.baou.alertdialogboxdemo;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.EditText;

public class ShowRegistration extends AppCompatActivity {

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

        details = (EditText)findViewById(R.id.txtRegistration);

        Intent i = getIntent();
        String str = i.getStringExtra("DETAILS");
        details.setText(str);
    }
}
```

11. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

Practical-20: Multiple Activities

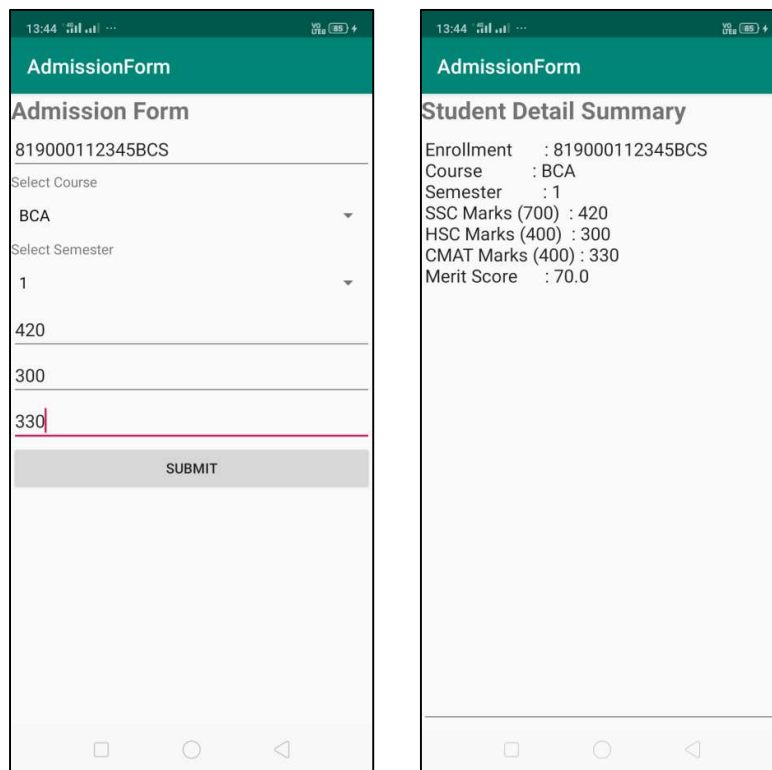
20

Create an application that designs a layout for making an Admission Form.

The layout should contain details like student enrollment no, Course, Semester, S.S.C marks, H.S.C Marks, C.M.A.T Score and a button named Submit as shown in left image.

When the submit button is clicked than a new layout should be opened that displays the student details as a summary with Merit Score as shown in right image. Merit Score should be calculated when the submit button is clicked.

Merit score = Average of (SSC marks, HSC Marks, CMAT score)



Solution

1. In the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
2. In the **Choose your project** window, select **Empty Activity**

3. Click **Next**.
4. In the **Configure your project** window, enter the following values:
 - 1) **Name:** "Admission Form"
 - 2) **Package name:** "in.edu.baou.admissionform"
 - 3) Select language as Java
 - 4) Change the project location as per your requirement.
5. Click finish
6. Write following XML code to design required activity_main.xml layout file

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

    <TextView
        android:id="@+id/textView2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Admission Form"
        android:textSize="24sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/txtEnroll"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Enrollment Number"
        android:inputType="textPersonName" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Select Course" />

    <Spinner
        android:id="@+id/cmbCourse"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
```

```

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Select Semester" />

<Spinner
    android:id="@+id/cmbSem"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<EditText
    android:id="@+id/txtSSC"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="S.S.C Marks"
    android:inputType="number" />

<EditText
    android:id="@+id/txtHSC"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="H.S.C Marks"
    android:inputType="number" />

<EditText
    android:id="@+id/txtCMAT"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="C.M.A.T Score"
    android:inputType="number" />

<Button
    android:id="@+id/btnSubmit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Submit" />
</LinearLayout>

```

7. Write following code in MainActivity.java file

```
package in.edu.baou.admissionform;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.content.Intent;
```

```

import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    EditText enroll,ssc,hsc,cmat;
    Spinner course,semester;
    Button submit;
    String summary;
    double average;
    String [] courselist = {"BCA", "BBA","PGDCA", "PGDMAD", "CCC-BAOU"};
    String [] semlist = {"1","2","3","4","5","6"};

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

        enroll = (EditText)findViewById(R.id.txtEnroll);
        ssc = (EditText) findViewById(R.id.txtSSC);
        hsc = (EditText) findViewById(R.id.txtHSC);
        cmat = (EditText) findViewById(R.id.txtCMAT);

        course = (Spinner) findViewById(R.id.cmbCourse);
        course.setAdapter(new
        ArrayAdapter<String>(getApplicationContext(),android.R.layout.simple_spinner_dropdown_item,courselist));

        semester = (Spinner)findViewById(R.id.cmbSem);
        semester.setAdapter(new
        ArrayAdapter<String>(getApplicationContext(),android.R.layout.simple_spinner_dropdown_item,semlist));

        submit = (Button)findViewById(R.id.btnSubmit);

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

                try {
                    average =
                    (Double.parseDouble(ssc.getText().toString())+Double.parseDouble(hsc.getText().toString())+Double.parseDouble(cmat.getText().toString()))/15;

                    summary = new String();

```

```

        summary = summary + "Enrollment    :
"+enroll.getText().toString()+"\n";
        summary = summary + "Course      :
"+course.getSelectedItem().toString()+"\n";
        summary = summary + "Semester    :
"+semester.getSelectedItem().toString()+"\n";
        summary = summary + "SSC Marks (700) :
"+ssc.getText().toString()+"\n";
        summary = summary + "HSC Marks (400) :
"+hsc.getText().toString()+"\n";
        summary = summary + "CMAT Marks (400) :
"+cmat.getText().toString()+"\n";
        summary = summary + "Merit Score   :
"+String.valueOf(average)+"\n";

        Intent i = new
Intent(getApplicationContext(),ShowStudentMerit.class);
        i.putExtra("summary",summary);
        startActivity(i);
    }
    catch (Exception ex){

Toast.makeText(getApplicationContext(),ex.getMessage(),Toast.LENGTH_L
ONG).show();
    }
}
});
}
}

```

8. Add activity by right clicking on package name and select New > Activity > Empty Activity and give name of activity as ShowStudentMerit and press OK.

9. Write following code to design required activity_show_student_merit.xml layout file

```

<?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="match_parent"
    android:orientation="vertical">

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

```



```

        android:text="Student Detail Summary"
        android:textSize="24sp"
        android:textStyle="bold" />

<EditText
    android:id="@+id/txtSummary"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:ems="10"
    android:gravity="start|top"
    android:inputType="textMultiLine" />

</LinearLayout>

```

10. Write following code in ShowStudentMerit.java file

```

package in.edu.baou.admissionform;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.widget.EditText;

public class ShowStudentMerit extends AppCompatActivity {

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

        summary = (EditText)findViewById(R.id.txtSummary);

        Intent i = getIntent();
        summary.setText(i.getStringExtra("summary"));

    }
}

```

11. Click "Run App" from Run menu or press Shift+F10 to run App in virtual device (AVD) created by you

21: Self Assessment Exercises and Additional Resources

20

No.	Practical
1	Installing "Android Studio IDE" and "Android SDK" Step 0: Pre-Installation Check List Step 1: Install "Android Studio IDE" Step 2: Installing Android SDK Step-3: Create AVD
2	Create "Hello World" application. That will display "Hello World" in the middle of the screen in the red color with white background.
3	Create an application that designs a layout with a text box and button named Submit. The user should enter the text in the text box. When the submit button is clicked then the text in the text box should be displayed in the toast.
4	Create an application to demonstrate Android Activity Life Cycle.
5	To understand Activity, Intent Create sample application with login module. (Check username and password) On successful login, go to next screen. And on failing login, alert user using Toast. Also pass username to next screen.
6	Create an application that designs a layout having two text boxes user name and password. The user and password will be taken from the user. The username and password will be verified from the username and password stored in the resources folder. If the verification is successful then a new layout will appear which will open any URL in browser
7	Create an application to call specific entered number by user in the Edit Text
8	Write Android App to Perform all Operations using Calculators
9	Understand resource folder & Adapter : a. Create spinner with strings taken from resource folder (res >>

No. Practical	
	value folder). b. On changing spinner value, change image.
10	Create an application that designs a layout having two text boxes user name and password. The user and password will be taken from the user. The username and password will be verified from database. If the verification is successful then a new layout will appear which will contain text "Welcome ,<UserName>" otherwise displays error message.
11	Android Program to Demonstrate Layouts in an Activity and Nesting of Layouts and Demonstrate List View Activity
12	Create application that works like an alarm.
13	Create an application that will have spinner with list of animation names. On selecting animation name, that animation should affect on the images displayed below.
14	Create an application to take picture using native application.
15	Create an application to pick up any image from the native application gallery and display it on the screen.
16	Write Android Program to Demonstrate Shape Drawables, Gradient Drawables, Radial Gradient, Sweep Gradient in Android
17	Create an application that designs a layout with spinner and an image view. The spinner should contain various shapes like circle, rectangle and rounded rectangle. When a shape is selected from the spinner that shape should be drawn in the image view. (Hint: Use shape Drawable or xml files for shapes on image view)
18	Create an application that designs a layout with 3 text boxes and an options menu. The options menu should contain options like Simple Interest and compound Interest. The text boxes should be used for the input of information like the principle amount, rate of interest and number of years. Then clicking on the option from menu appropriate operation should be performed and correct result should be displayed.
19	Create an application that designs a layout having gallery of images and an image view. The gallery contains number of images. Whenever

No. Practical

an image is selected from the gallery it should be displayed in the image view. The images should be fetched from the resources folder. (You can take any readymade image available or you can create your own)

20 Create an application that designs a layout having options menu and 2 text boxes for distance converter which allows user to select a particular conversion from following options.

- a. Feet to Meter
- b. Meter to Feet
- c. Feet to Yard
- d. Yard to Feet

The data for conversion should be entered by the user in textbox. Then clicking on the option from menu appropriate operation should be performed and correct result should be displayed.

21 Create an application that designs a layout of a student registration form. The layout should contain fields like first name, last name, phone no, date of birth (use date control) gender (use radio buttons), hobbies (use check boxes) and a button register. On clicking register an alert dialog box should appear with a message “do you want to register” and with two buttons yes or no. If yes is clicked then a new layout should appear where all the entered details should be displayed and also calculate the age (in years only) of the student and display it. If no is clicked than the alert dialog box should disappear.

22 Create an application that designs a layout for making an Admission Form. The layout should contain details like student enrollment no (textbox), Sem (spinner), Course (spinner), Date (date control), DOB (Date picker), SSC marks (textbox), HSC Marks (spinner), CMAT Score (Textbox) and a button named Submit. When the button submit is clicked than a new layout should be opened that displays all the student details as a summary and also display Merit Score. Merit Score should be calculated when the submit button is clicked.

Merit score = Average of (SSC marks, HSC Marks, CMAT score)

No. Practical	
23	<p>Create an application that designs a layout to store Student Details. The layout should contain information like: Roll number (textbox), Name (textbox), Birth date (Date Picker), Marks1 (textbox), Marks2 (textbox), Marks3 (textbox) and button submit. When button submit is clicked than all the details, percentage and total should be calculated automatically and displayed on another screen.</p>
24	<p>Create an application that designs a layout to store Result Details for BCA. The layout must contain details like name (textbox), Sem 1 Marks(spinner), Sem 2 Marks(spinner), Sem 3 Marks(spinner) , Sem 4 Marks(spinner), Sem 5 Marks(spinner) , Sem 6 Marks(spinner) and a Button named Submit.</p> <p>Note: Marks must be from (AA, AB, BB, BC, CC and FF)</p> <p>When the submit button is selected all the details should be displayed in next screen.</p>
24	<p>Create an application that will calculate percentage and class of any Student using following guidelines.</p> <p>Percentage= (Sub1+Sub2+Sub3)/3</p> <p>If percentage < 50 then fail</p> <p>If percentage > 50 and < 60 then pass class</p> <p>If percentage > 60 and < 70 then first class</p> <p>If percentage > 70 then distinction class</p>
25	<p>Create login application where you will have to validate Email ID (User Name). Till the username and password is not validated, login button should remain disabled. The user name and password should be stored in database. On successful login, open welcome screen.</p>
26	<p>Create an application that will change color of the screen, based on selected options from the spinner control.</p>
27	<p>Display List of following color in spinner control: Red, Green, Blue, Yellow, Orange, Pink, Black, White, Light Gray, Gray, Dark Gray, Magenta and Cyan. When user selects color it should change color of Image View displayed below spinner and display name of color in toast.</p>

No. Practical	
28	Create List with strings taken from resource folder (res >> value folder). On selecting item from list display in Toast, also provide facility for checking and un-checking of items.
29	Write an application to display five images in image view at interval of 5 seconds one by one in circular fashion.
30	Create an application in which display image from asset folder using Gallery View
31	Create an application that will remind anniversary of his/her friend.

* * * * *

References

- 1) https://www.gtu.ac.in/syllabus/NEW_Diploma/Sem6/3360704.pdf
- 2) https://www.gtu.ac.in/syllabus/NEW_Diploma/Sem6/3361602.pdf
- 3) <https://www.gtu.ac.in/syllabus/MCA/Effective%20from%202018/4649303.pdf>

Additional Resources

- 1) <https://www.udacity.com/course/new-android-fundamentals--ud851>
- 2) <https://www.udacity.com/course/ud853>
- 3) <http://developer.android.com/training/basics/firstapp/index.html>
- 4) <http://developer.android.com/tools/sdk/eclipse-adt.html>
- 5) <http://www.tutorialspoint.com/android/>
- 6) http://www.tutorialspoint.com/android/android_overview.htm
- 7) <http://www.codelearn.org/android-tutorial>
- 8) <http://pl.cs.jhu.edu/oose/resources/android/Android-Tutorial.pdf>
- 9) <http://mobisys.in/blog/2012/01/introduction-to-android-sqlite-database/>
- 10) www.appmakr.com/Android
- 11) www.telerik.com/android-development
- 12) www.developer.android.com/training/basics/firstapp

યુનિવર્સિટી ગીત

સ્વાધ્યાય: પરમં તપ:

સ્વાધ્યાય: પરમં તપ:

સ્વાધ્યાય: પરમં તપ:

શિક્ષણ, સંસ્કૃતિ, સદ્ભાવ, દિવ્યબોધનું ધામ
ડૉ. બાબાસાહેબ આંબેડકર ઓપન યુનિવર્સિટી નામ;
સૌને સૌની પાંખ મળે, ને સૌને સૌનું આભ,
દશે દિશામાં સ્મિત વહે હો દશે દિશે શુભ-લાભ.

અભણ રહી અજ્ઞાનના શાને, અંધકારને પીવો ?
કહે બુદ્ધ આંબેડકર કહે, તું થા તારો દીવો;
શારદીય અજવાળા પહોંચ્યાં ગુર્જર ગામે ગામ
ધ્રુવ તારકની જેમ ઝળહળે એકલવ્યની શાન.

સરસ્વતીના મયૂર તમારે ફળિયે આવી ગહેકે
અંધકારને હડસેલીને ઉજાસના ફૂલ મહેંકે;
બંધન નહીં કો સ્થાન સમયના જવું ન ઘરથી દૂર
ઘર આવી મા હરે શારદા દૈન્ય તિમિરના પૂર.

સંસ્કારોની સુગંધ મહેંકે, મન મંદિરને ધામે
સુખની ટપાલ પહોંચે સૌને પોતાને સરનામે;
સમાજ કેરે દરિયે હાંકી શિક્ષણ કેરું વહાણ,
આવો કરીયે આપણ સૌ
ભવ્ય રાષ્ટ્ર નિર્માણ...
દિવ્ય રાષ્ટ્ર નિર્માણ...
ભવ્ય રાષ્ટ્ર નિર્માણ



DR. BABASAHEB AMBEDKAR OPEN UNIVERSITY

(Established by Government of Gujarat)

'Jyotirmay' Parisar,

Sarkhej-Gandhinagar Highway, Chharodi, Ahmedabad-382 481

Website : www.baou.edu.in