

Android application development

Practical No:2

Date:

Aim: Develop an android app which displays “Hello, welcome to Android Lab” message.

Exercises:

1. Develop an android app which displays “Hello, Welcome to Android Lab” message.

MainActivity.java

```
package com.example.admin.helloworld;

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

public class MainActivity extends AppCompatActivity {

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

activity_main.xml

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

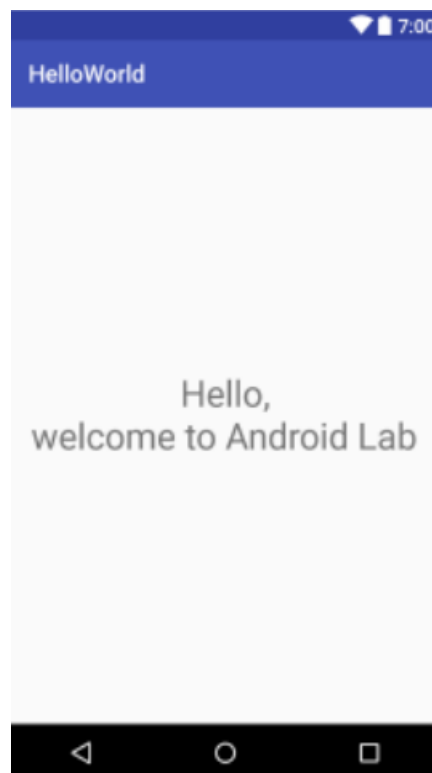
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello,"
        android:textSize="30sp"
        android:layout_gravity="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="welcome to Android Lab"
```

```
android:textSize="30sp"  
android:layout_gravity="center"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintLeft_toLeftOf="parent"  
app:layout_constraintRight_toRightOf="parent"  
app:layout_constraintTop_toTopOf="parent" />
```

</LinearLayout>

Output :-



Develop form in android application

Practical No:3

Date:

Aim: Develop an android app which displays a form to get following information from user. Username, Password, Email Address, Phone Number, Country, State, Gender, Interests, Birthdate, Birth time.

Exercises:

1. Develop an android app which displays a form to get Username, Password, Email Address, Phone Number, Country, State, Gender, Interests, Birthdate, Birth time from user.

Activity_prac3.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView android:layout_height="match_parent"
android:layout_width="match_parent"
xmlns:android="http://schemas.android.com/apk/res/android" >
<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:layout_marginTop="16dp"
android:layout_marginBottom="16dp"
```

```
android:layout_marginLeft="16dp"
android:layout_marginRight="16dp"
tools:context="iwt.waytowebsite.practicals.Prac6"
android:orientation="vertical" >
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Registration"
    android:textColor="@android:color/holo_blue_dark"
    android:textSize="30dp"
    android:layout_gravity="center"
    android:layout_marginTop="20dp"
    android:textStyle="bold|italic"
    android:id="@+id/title"/>
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="User Name"
    android:inputType="text"
    android:layout_marginTop="10dp"
    android:textColorHint="@android:color/black"
    android:textColor="@android:color/holo_red_light"
    android:id="@+id/unm" />
<EditText
    android:textColorHint="@android:color/black"
    android:textColor="@android:color/holo_red_light"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
```

```
android:hint="Password"
android:inputType="textPassword"
android:id="@+id/pwd" />
<EditText
android:textColorHint="@android:color/black"
android:textColor="@android:color/holo_red_light"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Email ID"
android:inputType="textEmailAddress"
android:id="@+id/eid" />
<EditText
android:textColorHint="@android:color/black"
android:textColor="@android:color/holo_red_light"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Phone Number"
android:inputType="phone"
android:maxLength="10"
android:id="@+id/pno" />
<AutoCompleteTextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:textColorHint="@android:color/black"
android:textColor="@android:color/holo_red_light"
android:hint="Country"
android:inputType="phone"
android:maxLength="10"
```

```

android:id="@+id/country" />
<Spinner
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:id="@+id/state">
</Spinner>
<RadioGroup
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <RadioButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Male"
        android:textColor="@android:color/black" />
    <RadioButton
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Female"
        android:textColor="@android:color/black" />
    </RadioGroup>
    <EditText
        android:textColorHint="@android:color/black"
        android:textColor="@android:color/holo_red_light"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Interests"
        android:maxLength="10"

```

```

android:id="@+id/interaset"/>
<EditText
android:textColorHint="@android:color/black"
android:textColor="@android:color/holo_red_light"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Birthdate"
android:maxLength="10"
android:id="@+id/birthdate" />
<EditText
android:textColorHint="@android:color/black"
android:textColor="@android:color/holo_red_light"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Birth Time"
android:maxLength="10"
android:id="@+id/birthtime" />
<Button
android:layout_width="250dp"
android:layout_height="wrap_content"
android:layout_marginTop="40dp"
android:layout_marginLeft="60dp"
android:text="Register"
android:id="@+id/regi" />
</LinearLayout>
</ScrollView>

```

Prac 3.java:


```
package iwt.waytowebsite.practicals;

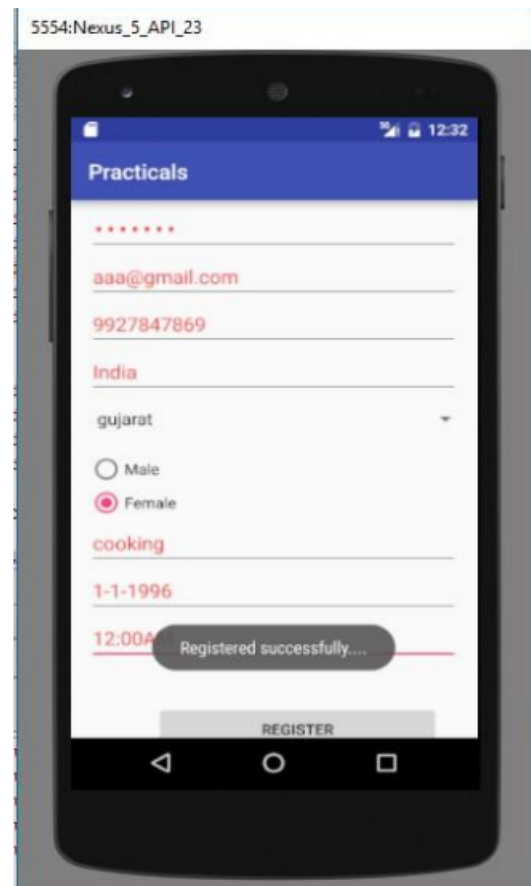
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AutoCompleteTextView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

public class Prac2 extends AppCompatActivity {
    EditText username,password,email,phone,interest,birthdate,birthtime;
    AutoCompleteTextView country;
    Spinner states;
    Button submit;
    ArrayAdapter arrayAdapter,arrayAdapter1;
    String[] Country={"India","Indonesia","Africa","Afghanistan"};
    String[] state={"gujarat","goa","maharashtra","rajasthan","aasam","bihar","west bangol"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_prac6);
        submit= (Button) findViewById(R.id.regi);
        states= (Spinner) findViewById(R.id.state);
        country= (AutoCompleteTextView) findViewById(R.id.country);
    }
}
```

```
arrayAdapter=new  
ArrayAdapter(Prac6.this,android.R.layout.simple_spinner_item,state);  
states.setAdapter(arrayAdapter);  
country.setThreshold(1);  
country.setAdapter(arrayAdapter1);  
submit.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Toast.makeText(Prac2.this, "Registered successfully... ",  
        Toast.LENGTH_SHORT).show();  
    }  
});  
}  
}
```

Output :-



Create login activity using android application

Practical No:4

Date:

Aim: Using Android, Create a login Activity. It asks “username” and “password” from user. If username and password are valid, it displays Welcome message using new activity.

Exercises:

Activity_prac4.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:layout_centerHorizontal="true"
    tools:context="iwt.waytowe.practicals.Prac7">
    <TextView
        android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:text="Login"
android:textColor="#FF212355"
android:textStyle="italic"
android:textSize="30dp"
android:gravity="center"
android:layout_marginTop="20dp" />
<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="enter email:"
android:id="@+id/email"
android:layout_marginTop="60dp" />
<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="enter password:"
android:id="@+id/pwd"
android:layout_marginTop="120dp"
android:inputType="textPassword" />
<Button
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="login"
android:id="@+id/login_btn"
android:layout_marginTop="180dp" />
</RelativeLayout>
Activity_WelcomePage.xml
```

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="iwt.waytowebsite.practicals.WelcomePage">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to the new page!"
        android:textSize="25dp"
        android:textStyle="bold"
        android:textColor="@color/colorAccent"
        android:layout_marginTop="20dp"/>
    </RelativeLayout>

```

Prac4.java

```

package iwt.waytowebsite.practicals;

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

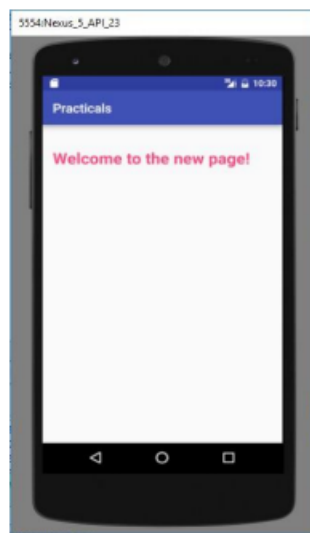
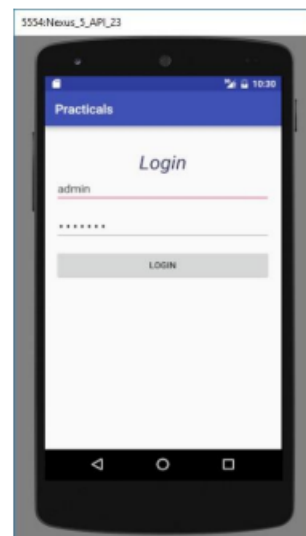
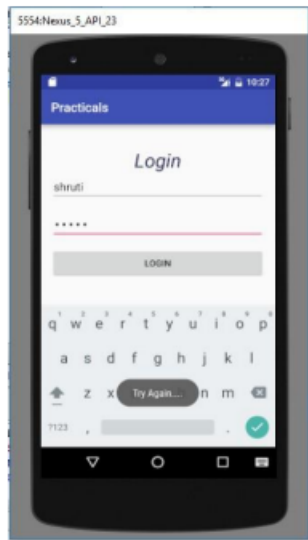
```

```

import android.widget.EditText;
import android.widget.Toast;
public class Prac3 extends AppCompatActivity {
    EditText email,password;
    Button login;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_prac7);
        email= (EditText) findViewById(R.id.email);
        password= (EditText) findViewById(R.id.pwd);
        login= (Button) findViewById(R.id.login_btn);
        login.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (email.getText().toString().equals("admin")) {
                    if( password.getText().toString().equals("1234567"))
                    {
                        Intent intent=new Intent(getApplicationContext(),WelcomePage.class);
                        startActivity(intent);
                    }
                    else
                    {
                        Toast.makeText(Prac3.this, "Try Again. .. ",
                        Toast.LENGTH_SHORT).show(); } } }); } }

```

Output :-



Android application development

Practical No:5

Date:

Aim: Develop calculator Android Application.

Exercises:

1. Develop calculator Android Application.

Activity_prac5.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity"
    android:id="@+id/relative1">
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/edt1"/>
    <Button
```

```

style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="1"
android:id="@+id/button1"
android:layout_marginTop="94dp"
android:layout_below="@+id/edt1"
android:layout_toStartOf="@+id/button4"
android:layout_alignRight="@+id/button4"
android:layout_alignEnd="@+id/button4" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="2"
android:id="@+id/button2"
android:layout_alignTop="@+id/button1"
android:layout_toLeftOf="@+id/button3"
android:layout_toStartOf="@+id/button3" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="3"
android:id="@+id/button3"
android:layout_alignTop="@+id/button2"
android:layout_centerHorizontal="true" />
<Button

```

```

style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="4"
android:id="@+id/button4"
android:layout_below="@+id/button1"
android:layout_toLeftOf="@+id/button2" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="5"
android:id="@+id/button5"
android:layout_alignBottom="@+id/button4"
android:layout_alignLeft="@+id/button2"
android:layout_alignStart="@+id/button2" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="6"
android:id="@+id/button6"
android:layout_below="@+id/button3"
android:layout_alignLeft="@+id/button3"
android:layout_alignStart="@+id/button3" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"

```

```

android:layout_height="wrap_content"
android:text="7"
android:id="@+id/button7"
android:layout_below="@+id/button4"
android:layout_toLeftOf="@+id/button2" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="8"
android:id="@+id/button8"
android:layout_below="@+id/button5"
android:layout_alignLeft="@+id/button5"
android:layout_alignStart="@+id/button5" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="9"
android:id="@+id/button9"
android:layout_below="@+id/button6"
android:layout_alignLeft="@+id/button6"
android:layout_alignStart="@+id/button6" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="+"

```

```

android:id="@+id/buttonadd"
android:layout_alignTop="@+id/button3"
android:layout_toRightOf="@+id/button3"
android:layout_marginLeft="46dp"
android:layout_marginStart="46dp"
android:layout_alignRight="@+id/edt1"
android:layout_alignEnd="@+id/edt1" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="-"
android:id="@+id/buttonsub"
android:layout_below="@+id/buttonadd"
android:layout_alignLeft="@+id/buttonadd"
android:layout_alignStart="@+id/buttonadd"
android:layout_alignRight="@+id/buttonadd"
android:layout_alignEnd="@+id/buttonadd" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="*"
android:id="@+id/buttonmul"
android:layout_below="@+id/buttonsub"
android:layout_alignLeft="@+id/buttonsub"
android:layout_alignStart="@+id/buttonsub"
android:layout_alignParentRight="true"

```

```

android:layout_alignParentEnd="true" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="."
android:id="@+id/button10"
android:layout_below="@+id/button7"
android:layout_toLeftOf="@+id/button2" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="0"
android:id="@+id/button0"
android:layout_below="@+id/button8"
android:layout_alignLeft="@+id/button8"
android:layout_alignStart="@+id/button8" />
<Button
style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="C"
android:id="@+id/buttonC"
android:layout_below="@+id/button9"
android:layout_alignLeft="@+id/button9"
android:layout_alignStart="@+id/button9" />
<Button

```

```

style="?android:attr/buttonStyleSmall"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="/"
android:id="@+id/buttondiv"
android:layout_below="@+id/buttonmul"
android:layout_alignLeft="@+id/buttonmul"
android:layout_alignStart="@+id/buttonmul"
android:layout_alignRight="@+id/buttonmul"
android:layout_alignEnd="@+id/buttonmul" />
<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text=""
android:id="@+id/buttoneql"
android:layout_below="@+id/button0"
android:layout_marginTop="37dp"
android:layout_alignRight="@+id/buttondiv"
android:layout_alignEnd="@+id/buttondiv"
android:layout_alignLeft="@+id/button10"
android:layout_alignStart="@+id/button10" />
</RelativeLayout>

```

Prac4.java

```

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;

```

```
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    Button button0 , button1 , button2 , button3 , button4 , button5 , button6 ,
    button7 , button8 , button9 , buttonAdd , buttonSub , buttonDivision ,
    buttonMul , button10 , buttonC , buttonEqual ;
    EditText edt1 ;
    float mValueOne , mValueTwo ;
    boolean mAddition , mSubtract ,mMultiplication ,mDivision ;

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

        button0 = (Button) findViewById(R.id.button0);
        button1 = (Button) findViewById(R.id.button1);
        button2 = (Button) findViewById(R.id.button2);
        button3 = (Button) findViewById(R.id.button3);
        button4 = (Button) findViewById(R.id.button4);
        button5 = (Button) findViewById(R.id.button5);
        button6 = (Button) findViewById(R.id.button6);
        button7 = (Button) findViewById(R.id.button7);
        button8 = (Button) findViewById(R.id.button8);
        button9 = (Button) findViewById(R.id.button9);
        button10 = (Button) findViewById(R.id.button10);
        buttonAdd = (Button) findViewById(R.id.buttonadd);
        buttonSub = (Button) findViewById(R.id.buttonsub);
        buttonMul = (Button) findViewById(R.id.buttonmul);
        buttonDivision = (Button) findViewById(R.id.buttondiv);
    }
}
```



```
buttonC = (Button) findViewById(R.id.buttonC);
buttonEqual = (Button) findViewById(R.id.buttoneq1);
edt1 = (EditText) findViewById(R.id.edt1);
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"1");
    }
});
button2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"2");
    }
});
button3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"3");
    }
});
button4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"4");
    }
});
button5.setOnClickListener(new View.OnClickListener() {
```

```
@Override
public void onClick(View v) {
    edt1.setText(edt1.getText()+"5");
}

});
button6.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"6");
    }

});
button7.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"7");
    }

});
button8.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"8");
    }

});
button9.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        edt1.setText(edt1.getText()+"9");
    }

});
```

```

});
button0.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
edt1.setText(edt1.getText()+"0");
}
});
buttonAdd.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
if (edt1 == null){
edt1.setText("");
}else {
mValueOne = Float.parseFloat(edt1.getText() + "");
mAddition = true;
edt1.setText(null);
}
}
});
buttonSub.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
mValueOne = Float.parseFloat(edt1.getText() + "");
mSubtract = true ;
edt1.setText(null);
}
});
buttonMul.setOnClickListener(new View.OnClickListener() {

```

```

@Override
public void onClick(View v) {
    mValueOne = Float.parseFloat(edt1.getText() + "");
    mMultiplication = true ;
    edt1.setText(null);
}
});
buttonDivision.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    mValueOne = Float.parseFloat(edt1.getText()+"");
    mDivision = true ;
    edt1.setText(null);
}
});
buttonEqual.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    mValueTwo = Float.parseFloat(edt1.getText() + "");
    if (mAddition == true){
        edt1.setText(mValueOne + mValueTwo + "");
        mAddition=false;
    }
    if (mSubtract == true){
        edt1.setText(mValueOne - mValueTwo+"");
        mSubtract=false;
    }
    if (mMultiplication == true){

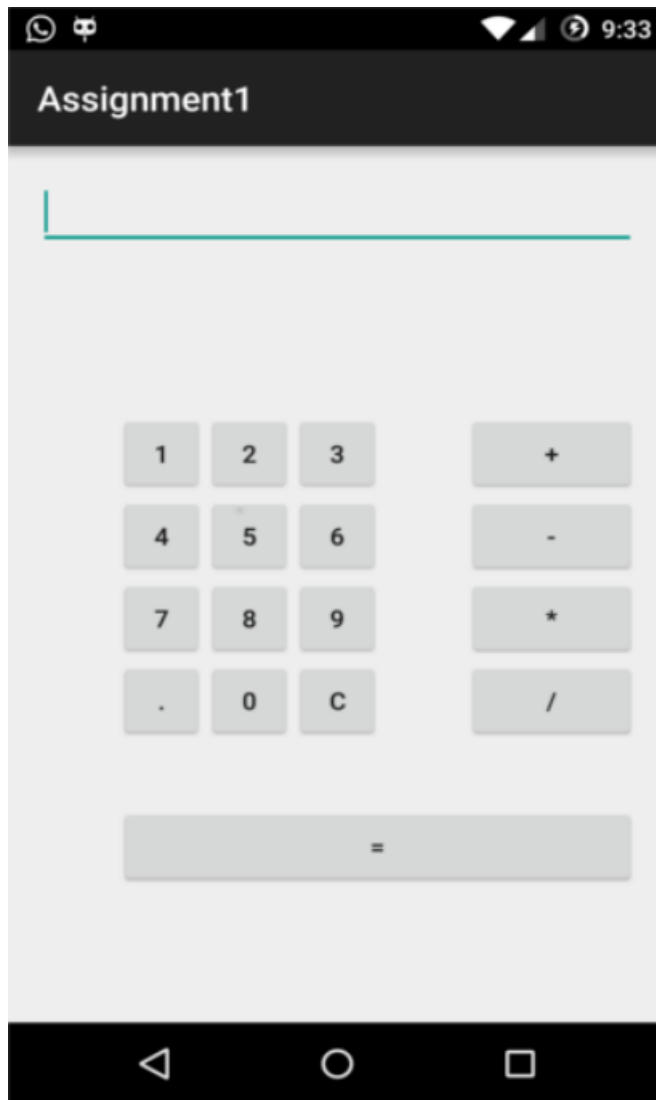
```

```

edt1.setText(mValueOne * mValueTwo+"");
mMultiplication=false;
}
if (mDivision == true){
edt1.setText(mValueOne / mValueTwo+"");
mDivision=false;
}
}
});
buttonC.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
edt1.setText("");
}
});
button10.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
edt1.setText(edt1.getText()+".");
}
});
}}

```

Output :-



Android application to convert into different Currencies

Practical No:6

Date:

Aim: Write an Android application to convert into different currencies for example, Rupees to dollar.

Exercises:

1. Write an Android application to convert into different currencies for example, Rupees to dollar.

- **Activity_main.xml Source Code**

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

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

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

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

android:layout_width="match_parent"

android:layout_height="match_parent"

tools:context=".MainActivity">
```

<ImageView

```
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:layout_alignEnd="@+id/button"
    android:layout_alignParentTop="true"
    android:layout_marginTop="30dp"
    android:layout_marginEnd="-14dp"
    android:src="@drawable/icon"
    android:layout_alignRight="@+id/button"
    android:layout_marginRight="-14dp" />
```

<TextView

```
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="41dp"
    android:layout_marginLeft="41dp"
    android:layout_marginTop="164dp"
    android:text="Euro" />
```


<EditText

```
    android:id="@+id/editText4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/textView4"
    android:layout_centerHorizontal="true"

    android:layout_marginBottom="-16dp"
    android:ems="10"
    android:inputType="number" />
```

<TextView

```
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/textView4"
    android:layout_alignParentTop="true"
    android:layout_marginStart="0dp"
    android:layout_marginTop="231dp"
    android:text="Curency" />
```

<Spinner

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

android:layout_width="220dp"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/textView5"
android:layout_marginLeft="70dp"
android:layout_marginTop="1dp" />
```

```
<Button
```

```
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="125dp"
    android:layout_marginLeft="125dp"
    android:layout_marginTop="297dp"
    android:text="Converte" />
```

```
<TextView
```

```
android:id="@+id/textView6"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentStart="true"
android:layout_alignParentLeft="true"
android:layout_alignParentBottom="true"
android:layout_marginStart="51dp"
android:layout_marginLeft="51dp"
android:layout_marginBottom="113dp"
android:text="Value" />
```

```
<TextView
```

```
android:id="@+id/textView7"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignTop="@+id/textView6"
android:layout_marginStart="44dp"
android:layout_marginTop="0dp"
android:layout_toEndOf="@+id/textView5"
android:text="TextView" />
```

```
</RelativeLayout>
```

- **Code implementation in MainActivity.java**

```
package com.example.currency_converter;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.AdapterView;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Spinner;

import android.widget.TextView;

import android.widget.Toast;


import com.example.currency_converter.R;
```

```
import com.squareup.okhttp.Callback;

import com.squareup.okhttp.OkHttpClient;

import com.squareup.okhttp.Request;

import com.squareup.okhttp.Response;


import org.json.JSONException;

import org.json.JSONObject;

import java.io.IOException;

import java.text.BreakIterator;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;


    public class MainActivity extends AppCompatActivity {

        public static BreakIterator data;

        List<String> keysList;
```

Spinner toCurrency;

TextView textView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_main);

toCurrency = (Spinner)findViewById(R.id.planets_spinner);

final EditText edtEuroValue =
(EditText)findViewById(R.id.editText4);

final Button btnConvert =
(Button)findViewById(R.id.button);

textView =(TextView) findViewById(R.id.textView7);

try {

loadConvTypes();

```
    } catch (IOException e) {  
  
        e.printStackTrace();  
  
    }  
  
    btnConvert.setOnClickListener(new  
View.OnClickListener() {  
  
        @Override  
  
        public void onClick(View v) {  
  
            if(!edtEuroValue.getText().toString().isEmpty())  
  
            {  
  
                String toCurr =  
toCurrency.getSelectedItemAt().toString();  
  
                double euroVlaue =  
Double.valueOf(edtEuroValue.getText().toString());  
  
                Toast.makeText(MainActivity.this, "Please Wait..",  
Toast.LENGTH_SHORT).show();  
  
                try {  
  
                    convertCurrency(toCurr, euroVlaue);  
  
                }  
            }  
        }  
    }  
}
```

```

        } catch (IOException e) {

            e.printStackTrace();

            Toast.makeText(MainActivity.this,
e.getMessage(), Toast.LENGTH_SHORT).show();

        }

    }

    else

    {

        Toast.makeText(MainActivity.this,

            "Please Enter a Value to Convert..",
            Toast.LENGTH_SHORT).show();

        } }));

    } public void loadConvTypes() throws IOException {

        String url = "https://api.exchangeratesapi.io/latest";
        OkHttpClient client = new OkHttpClient();

        Request request = new Request.Builder()

            .url(url)

```



```
.header("Content-Type", "application/json")

.build();

client.newCall(request).enqueue(new Callback() {

@Override

public void onFailure(Request request, IOException e) {

    String mMessage = e.getMessage().toString();

    Log.w("failure Response", mMessage);

    Toast.makeText(MainActivity.this, mMessage,
Toast.LENGTH_SHORT).show();

}

@Override

public void onResponse(Response response) throws
IOException {

    final String mMessage = response.body().string();

    MainActivity.this.runOnUiThread(new Runnable() {

@Override
```

```
public void run() {  
  
    //Toast.makeText(MainActivity.this, mMessage,  
    Toast.LENGTH_SHORT).show();  
  
    try {  
  
        JSONObject obj = new  
        JSONObject(mMessage);  
  
        JSONObject b = obj.getJSONObject("rates");  
  
        Iterator keysToCopyIterator = b.keys();  
  
        keysList = new ArrayList<String>();  
  
        while(keysToCopyIterator.hasNext()) {  
  
            String key = (String)  
            keysToCopyIterator.next();  
            keysList.add(key) }  
  
            ArrayAdapter<String> spinnerArrayAdapter  
            = new  
            ArrayAdapter<String>(getApplicationConte  
            xt(), android.R.layout.simple_spinner_item,  
            keysList );  
  
            toCurrency.setAdapter(spinnerArrayAdapter)
```

```

        } catch (JSONException e) {

            e.printStackTrace();

        }

    });

} });

}

public void convertCurrency(final String toCurr, final double
euroVlaue) throws IOException {

    String url = "https://api.exchangeratesapi.io/latest";

    OkHttpClient client = new OkHttpClient();

    Request request = new Request.Builder()

        .url(url)

        .header("Content-Type", "application/json")

        .build();

    client.newCall(request).enqueue(new Callback() {

        @Override

```

```
public void onFailure(Request request, IOException e) {  
  
    String mMessage = e.getMessage().toString();  
  
    Log.w("failure Response", mMessage);  
  
    Toast.makeText(MainActivity.this, mMessage,  
        Toast.LENGTH_SHORT).show();  
  
}
```

@Override

```
public void onResponse(Response response) throws  
IOException {  
  
    final String mMessage = response.body().string();  
  
    MainActivity.this.runOnUiThread(new Runnable() {  
  
        @Override  
  
        public void run() {  
  
            //Toast.makeText(MainActivity.this, mMessage,  
                Toast.LENGTH_SHORT).show();  
  
            try {
```

```
        JSONObject obj = new
JSONObject(mMessage);

        JSONObject b = obj.getJSONObject("rates");

        String val = b.getString(toCurr);

        double output =
euroVlaue*Double.valueOf(val);
textView.setText(String.valueOf(output));

    } catch (JSONException e) {

        e.printStackTrace();

    } }

});

}






});

}


}
```

Output : -




6:19     

Currency Converter



Euro

Currency BGN 

CONVERTE

Value 19.558

Android application to count Library Overdue

Practical No:7

Date:

Aim: Write an Android application to count library overdue.

Exercises:

1. Write an Android application to count library overdue.

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

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

activity_main.xml file

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

```
<RelativeLayout
```

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

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

```
android:layout_width="match_parent"
```

```
android:layout_height="match_parent"
```

```
tools:context=".MainActivity">
```

```
<LinearLayout
```

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

```
android:layout_width="match_parent"
```



```

        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:weightSum="5">

```

```

<!--edit text for getting the search
      query for book from user-->
<EditText
    android:id="@+id/idEdtSearchBooks"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="4" />

```

```

<!--image button for our search button -->
<ImageButton
    android:id="@+id/idBtnSearch"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:src="@drawable/ic_search" />

```

```

</LinearLayout>

```

```

<!--recycler view for displaying our list of books-->
<androidx.recyclerview.widget.RecyclerView
    android:id="@+id/idRVBooks"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_below="@id/idLLsearch" />

```

```
<!--progressbar for displaying our loading indicator-->
```

```
<ProgressBar
```

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

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_centerInParent="true"
```

```
    android:visibility="gone" />
```

```
</RelativeLayout>
```

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

```
public class BookInfo {
```

```
    // creating string, int and array list
```

```
    // variables for our book details
```

```
    private String title;
```

```
    private String subtitle;
```

```
    private ArrayList<String> authors;
```

```
    private String publisher;
```

```
    private String publishedDate;
```

```
    private String description;
```

```
    private int pageCount;
```

```
    private String thumbnail;
```

```
    private String previewLink;
```

```
    private String infoLink;
```

```
private String buyLink;

// creating getter and setter methods
public String getTitle() {
    return title;
}

public void setTitle(String title) {
    this.title = title;
}

public String getSubtitle() {
    return subtitle;
}

public void setSubtitle(String subtitle) {
    this.subtitle = subtitle;
}

public ArrayList<String> getAuthors() {
    return authors;
}

public void setAuthors(ArrayList<String> authors) {
    this.authors = authors;
}

public String getPublisher() {
```

```
        return publisher;
    }

    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }

    public String getPublishedDate() {
        return publishedDate;
    }

    public void setPublishedDate(String publishedDate) {
        this.publishedDate = publishedDate;
    }

    public String getDescription() {
        return description;
    }

    public void setDescription(String description) {
        this.description = description;
    }

    public int getPageCount() {
        return pageCount;
    }

    public void setPageCount(int pageCount) {
```

```
        this.pageCount = pageCount;
    }

    public String getThumbnail() {
        return thumbnail;
    }

    public void setThumbnail(String thumbnail) {
        this.thumbnail = thumbnail;
    }

    public String getPreviewLink() {
        return previewLink;
    }

    public void setPreviewLink(String previewLink) {
        this.previewLink = previewLink;
    }

    public String getInfoLink() {
        return infoLink;
    }

    public void setInfoLink(String infoLink) {
        this.infoLink = infoLink;
    }

    public String getBuyLink() {
```

```

        return buyLink;
    }

    public void setBuyLink(String buyLink) {
        this.buyLink = buyLink;
    }

    // creating a constructor class for our BookInfo
    public BookInfo(String title, String subtitle, ArrayList<String>
authors, String publisher,
                                String publishedDate, String description,
int pageCount, String thumbnail,
                                String previewLink, String infoLink,
String buyLink) {
        this.title = title;
        this.subtitle = subtitle;
        this.authors = authors;
        this.publisher = publisher;
        this.publishedDate = publishedDate;
        this.description = description;
        this.pageCount = pageCount;
        this.thumbnail = thumbnail;
        this.previewLink = previewLink;
        this.infoLink = infoLink;
        this.buyLink = buyLink;
    }
}

```

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

```
<androidx.cardview.widget.CardView
```

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

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

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_margin="4dp"
```

```
    app:cardCornerRadius="8dp"
```

```
    app:cardElevation="8dp">
```

```
    <RelativeLayout
```

```
        android:layout_width="match_parent"
```

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

```
        <ImageView
```

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

```
            android:layout_width="130dp"
```

```
            android:layout_height="160dp"
```

```
            android:layout_margin="10dp" />
```

```
        <TextView
```

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

```
            android:layout_width="match_parent"
```

```
            android:layout_height="wrap_content"
```

```
            android:layout_marginTop="10dp"
```

```
            android:layout_toEndOf="@id/idIVbook"
```

```
            android:padding="3dp"
```

```
            android:text="Book Title"
```

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

```
android:textSize="11sp" />
```

```
<TextView
```

```
    android:id="@+id/idTVpublisher"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/idTVBookTitle"
    android:layout_marginTop="3dp"
    android:layout_toEndOf="@id/idIVbook"
    android:padding="3dp"
    android:text="Publisher"
    android:textColor="@color/black"
    android:textSize="11sp" />
```

```
<TextView
```

```
    android:id="@+id/idTVPageCount"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/idTVpublisher"
    android:layout_marginTop="3dp"
    android:layout_toEndOf="@id/idIVbook"
    android:padding="3dp"
    android:text="Page count"
    android:textColor="@color/black"
    android:textSize="11sp" />
```

```
<TextView
```

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



```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/idTVPageCount"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="5dp"
        android:padding="3dp"
        android:text="date"
        android:textColor="@color/black"
        android:textSize="11sp" />
```

```
</RelativeLayout>
```

```
</androidx.cardview.widget.CardView>
```

```
import android.content.Context;
import android.content.Intent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import com.squareup.picasso.Picasso;

import java.util.ArrayList;
```

```
public class BookAdapter extends
RecyclerView.Adapter<BookAdapter.BookViewHolder> {

    // creating variables for arraylist and context.
    private ArrayList<BookInfo> bookInfoArrayList;
    private Context mContext;

    // creating constructor for array list and context.
    public BookAdapter(ArrayList<BookInfo> bookInfoArrayList,
Context mContext) {
        this.bookInfoArrayList = bookInfoArrayList;
        this.mContext = mContext;
    }

    @NonNull
    @Override
    public BookViewHolder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType) {
        // inflating our layout for item of recycler view item.
        View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.book_rv_item,
parent, false);
        return new BookViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull BookViewHolder
holder, int position) {
```

```

        // inside on bind view holder method we are
        // setting ou data to each UI component.
        BookInfo bookInfo = bookInfoArrayList.get(position);
        holder.nameTV.setText(bookInfo.getTitle());
        holder.publisherTV.setText(bookInfo.getPublisher());
        holder.pageCountTV.setText("No of Pages : " +
bookInfo.getPageCount());
        holder.dateTV.setText(bookInfo.getPublishedDate());

        // below line is use to set image from URL in our image
view.

        Picasso.get().load(bookInfo.getThumbnail()).into(holder.bookIV);

        // below line is use to add on click listener for our item of
recycler view.

        holder.itemView.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // inside on click listener method we are calling
a new activity
                // and passing all the data of that item in next
intent.

                Intent i = new Intent(mcontext,
BookDetails.class);

                i.putExtra("title", bookInfo.getTitle());
                i.putExtra("subtitle", bookInfo.getSubtitle());
                i.putExtra("authors", bookInfo.getAuthors());

```

```

        i.putExtra("publisher",
bookInfo.getPublisher());

        i.putExtra("publishedDate",
bookInfo.getPublishedDate());

        i.putExtra("description",
bookInfo.getDescription());

        i.putExtra("pageCount",
bookInfo.getPageCount());

        i.putExtra("thumbnail",
bookInfo.getThumbnail());

        i.putExtra("previewLink",
bookInfo.getPreviewLink());

        i.putExtra("infoLink", bookInfo.getInfoLink());
        i.putExtra("buyLink", bookInfo.getBuyLink());

        // after passing that data we are
        // starting our new intent.
        mcontext.startActivity(i);
    }

});
}

@Override
public int getItemCount() {
    // inside get item count method we
    // are returning the size of our array list.
    return bookInfoArrayList.size();
}

```

```

public class BookViewHolder extends RecyclerView.ViewHolder {
    // below line is use to initialize
    // our text view and image views.
    TextView nameTV, publisherTV, pageCountTV, dateTV;
    ImageView bookIV;

    public BookViewHolder(View itemView) {
        super(itemView);
        nameTV =
itemView.findViewById(R.id.idTVBookTitle);
        publisherTV =
itemView.findViewById(R.id.idTVpublisher);
        pageCountTV =
itemView.findViewById(R.id.idTVPageCount);
        dateTV = itemView.findViewById(R.id.idTVDate);
        bookIV = itemView.findViewById(R.id.idIVbook);
    }
}
}

```

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

```
<ScrollView
```

```
    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=".BookDetails">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical">
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:orientation="horizontal">
```

```
<!--Image view for displaying our book image-->
```

```
<ImageView
```

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

```
    android:layout_width="130dp"
```

```
    android:layout_height="160dp"
```

```
    android:layout_margin="18dp" />
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_marginTop="20dp"
```

```
    android:orientation="vertical">
```

```
<!--Text view for displaying book publisher-->
```

```
<TextView
```

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

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```

        android:padding="4dp"
        android:text="Publisher"
        android:textColor="@color/black"
        android:textSize="15sp" />

```

book-->

```

<TextView
    android:id="@+id/idTVNoOfPages"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    android:padding="4dp"
    android:text="Number of Pages"
    android:textColor="@color/black"
    android:textSize="15sp" />

```

>

```

<TextView
    android:id="@+id/idTVPublishDate"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="4dp"
    android:padding="4dp"
    android:text="Publish Date"
    android:textColor="@color/black"
    android:textSize="15sp" />

```

```
</LinearLayout>
```

```
</LinearLayout>
```

```
<!--text view for displaying book title-->
```

```
<TextView
```

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

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_margin="8dp"
```

```
    android:padding="4dp"
```

```
    android:text="title"
```

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

```
    android:textSize="15sp" />
```

```
<!--text view for displaying book subtitle-->
```

```
<TextView
```

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

```
    android:layout_width="match_parent"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_margin="8dp"
```

```
    android:padding="4dp"
```

```
    android:text="subtitle"
```

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

```
    android:textSize="12sp" />
```

```
<!--text view for displaying book description-->
```



```
<TextView
```

```
    android:id="@+id/idTVDescription"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="8dp"  
    android:padding="4dp"  
    android:text="description"  
    android:textColor="@color/black"  
    android:textSize="12sp" />
```

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="8dp"  
    android:orientation="horizontal"  
    android:weightSum="2">
```

```
<!--button for displaying book preview-->
```

```
<Button
```

```
    android:id="@+id/idBtnPreview"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_margin="4dp"  
    android:layout_weight="1"  
    android:text="Preview"  
    android:textAllCaps="false" />
```

```
<!--button for opening buying page of the book-->
```

```
<Button
    android:id="@+id/idBtnBuy"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_margin="4dp"
    android:layout_weight="1"
    android:text="Buy"
    android:textAllCaps="false" />

</LinearLayout>
</LinearLayout>
</ScrollView>
```

```
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.squareup.picasso.Picasso;

import java.util.ArrayList;
```

```
public class BookDetails extends AppCompatActivity {

    // creating variables for strings,text view, image views and button.
    String title, subtitle, publisher, publishedDate, description,
    thumbnail, previewLink, infoLink, buyLink;

    int pageCount;

    private ArrayList<String> authors;

    TextView titleTV, subtitleTV, publisherTV, descTV, pageTV,
    publishDateTV;

    Button previewBtn, buyBtn;

    private ImageView bookIV;

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

        // initializing our views..
        titleTV = findViewById(R.id.idTVTitle);
        subtitleTV = findViewById(R.id.idTVSubTitle);
        publisherTV = findViewById(R.id.idTVpublisher);
        descTV = findViewById(R.id.idTVDescription);
        pageTV = findViewById(R.id.idTVNoOfPages);
        publishDateTV = findViewById(R.id.idTVPublishDate);
        previewBtn = findViewById(R.id.idBtnPreview);
        buyBtn = findViewById(R.id.idBtnBuy);
        bookIV = findViewById(R.id.idIVbook);
    }
}
```

```

class.
    // getting the data which we have passed from our adapter
    title = getIntent().getStringExtra("title");
    subtitle = getIntent().getStringExtra("subtitle");
    publisher = getIntent().getStringExtra("publisher");
    publishedDate =
getIntent().getStringExtra("publishedDate");
    description = getIntent().getStringExtra("description");
    pageCount = getIntent().getIntExtra("pageCount", 0);
    thumbnail = getIntent().getStringExtra("thumbnail");
    previewLink = getIntent().getStringExtra("previewLink");
    infoLink = getIntent().getStringExtra("infoLink");
    buyLink = getIntent().getStringExtra("buyLink");

    // after getting the data we are setting
    // that data to our text views and image view.
    titleTV.setText(title);
    subtitleTV.setText(subtitle);
    publisherTV.setText(publisher);
    publishDateTV.setText("Published On : " + publishedDate);
    descTV.setText(description);
    pageTV.setText("No Of Pages : " + pageCount);
    Picasso.get().load(thumbnail).into(bookIV);

    // adding on click listener for our preview button.
    previewBtn.setOnClickListener(new
View.OnClickListener() {
        @Override

```

```

        public void onClick(View v) {
            if (previewLink.isEmpty()) {
                // below toast message is displayed when
                preview link is not present.
                Toast.makeText(BookDetails.this, "No
                preview Link present", Toast.LENGTH_SHORT).show();
                return;
            }
            // if the link is present we are opening
            // that link via an intent.
            Uri uri = Uri.parse(previewLink);
            Intent i = new Intent(Intent.ACTION_VIEW,
            uri);
            startActivity(i);
        }
    });

    // initializing on click listener for buy button.
    buyBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if (buyLink.isEmpty()) {
                // below toast message is displaying when
                buy link is empty.
                Toast.makeText(BookDetails.this, "No
                buy page present for this book", Toast.LENGTH_SHORT).show();
                return;
            }
            // if the link is present we are opening

```

```
        // the link via an intent.  
        Uri uri = Uri.parse(buyLink);  
        Intent i = new Intent(Intent.ACTION_VIEW,  
uri);  
  
        startActivity(i);  
    }  
});  
}  
}
```

```
import android.os.Bundle;  
import android.view.View;  
import android.widget.EditText;  
import android.widget.ImageButton;  
import android.widget.ProgressBar;  
import android.widget.Toast;
```

```
import androidx.appcompat.app.AppCompatActivity;  
import androidx.recyclerview.widget.LinearLayoutManager;  
import androidx.recyclerview.widget.RecyclerView;
```

```
import com.android.volley.Request;  
import com.android.volley.RequestQueue;  
import com.android.volley.Response;  
import com.android.volley.VolleyError;  
import com.android.volley.toolbox.JsonObjectRequest;  
import com.android.volley.toolbox.Volley;
```

```
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    // creating variables for our request queue,
    // array list, progressbar, edittext,
    // image button and our recycler view.
    private RequestQueue mRequestQueue;
    private ArrayList<BookInfo> bookInfoArrayList;
    private ProgressBar progressBar;
    private EditText searchEdt;
    private ImageButton searchBtn;

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

        // initializing our views.
        progressBar = findViewById(R.id.idLoadingPB);
        searchEdt = findViewById(R.id.idEdtSearchBooks);
        searchBtn = findViewById(R.id.idBtnSearch);

        // initializing on click listener for our button.
```

```

searchBtn.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View v) {
        progressBar.setVisibility(View.VISIBLE);

        // checking if our edittext field is empty or not.
        if (searchEdt.getText().toString().isEmpty()) {
            searchEdt.setError("Please enter search
query");

            return;
        }
        // if the search query is not empty then we are
        // calling get book info method to load all
        // the books from the API.
        getBooksInfo(searchEdt.getText().toString());
    }
});
}

```

```

private void getBooksInfo(String query) {

    // creating a new array list.
    bookInfoArrayList = new ArrayList<>();

    // below line is use to initialize
    // the variable for our request queue.
    mRequestQueue =
Volley.newRequestQueue(MainActivity.this);

```



```

// below line is use to clear cache this
// will be use when our data is being updated.
mRequestQueue.getCache().clear();

// below is the url for getting data from API in json format.
String url = "https://www.googleapis.com/books/v1/
volumes?q=" + query;

// below line we are creating a new request queue.
RequestQueue queue =
Volley.newRequestQueue(MainActivity.this);

// below line is use to make json object request inside that we
// are passing url, get method and getting json object. .
JsonObjectRequest booksObjrequest = new
JsonObjectRequest(Request.Method.GET, url, null, new
Response.Listener<JSONObject>() {
    @Override
    public void onResponse(JSONObject response) {
        progressBar.setVisibility(View.GONE);
        // inside on response method we are extracting
all our json data.
        try {
            JSONArray itemsArray =
response.getJSONArray("items");
            for (int i = 0; i < itemsArray.length(); i+
+) {

```

```

                                JSONObject itemsObj =
itemsArray.getJSONObject(i);
                                JSONObject volumeObj =
itemsObj.getJSONObject("volumeInfo");
                                String title =
volumeObj.optString("title");
                                String subtitle =
volumeObj.optString("subtitle");
                                JSONArray authorsArray =
volumeObj.getJSONArray("authors");
                                String publisher =
volumeObj.optString("publisher");
                                String publishedDate =
volumeObj.optString("publishedDate");
                                String description =
volumeObj.optString("description");
                                int pageCount =
volumeObj.optInt("pageCount");
                                JSONObject imageLinks =
volumeObj.optJSONObject("imageLinks");
                                String thumbnail =
imageLinks.optString("thumbnail");
                                String previewLink =
volumeObj.optString("previewLink");
                                String infoLink =
volumeObj.optString("infoLink");
                                JSONObject saleInfoObj =
itemsObj.optJSONObject("saleInfo");
                                String buyLink =
saleInfoObj.optString("buyLink");
                                ArrayList<String>
authorsArrayList = new ArrayList<>();
                                if (authorsArray.length() != 0) {

```

```

                                for (int j = 0; j <
authorsArray.length(); j++) {

authorsArrayList.add(authorsArray.optString(i));

                                }
                                }
                                // after extracting all the data we
are
                                // saving this data in our modal
class.

                                BookInfo bookInfo = new
BookInfo(title, subtitle, authorsArrayList, publisher, publishedDate,
description, pageCount, thumbnail, previewLink, infoLink, buyLink);

                                // below line is use to pass our
modal
                                // class in our array list.
                                bookInfoArrayList.add(bookInfo);

                                // below line is use to pass our
                                // array list in adapter class.
                                BookAdapter adapter = new
BookAdapter(bookInfoArrayList, MainActivity.this);

                                // below line is use to add linear
layout
                                // manager for our recycler view.
                                LinearLayoutManager
linearLayoutManager = new LinearLayoutManager(MainActivity.this,
RecyclerView.VERTICAL, false);

```

```

        RecyclerView mRecyclerView =
(RecyclerView) findViewById(R.id.idRVBooks);

        // in below line we are setting
layout manager and

        // adapter to our recycler view.

mRecyclerView.setLayoutManager(linearLayoutManager);

mRecyclerView.setAdapter(adapter);
    }
    } catch (JSONException e) {
        e.printStackTrace();
        // displaying a toast message when we get
any error from API

        Toast.makeText(MainActivity.this, "No
Data Found" + e, Toast.LENGTH_SHORT).show();
    }
}

}, new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        // also displaying error message in toast.

        Toast.makeText(MainActivity.this, "Error found
is " + error, Toast.LENGTH_SHORT).show();
    }
});

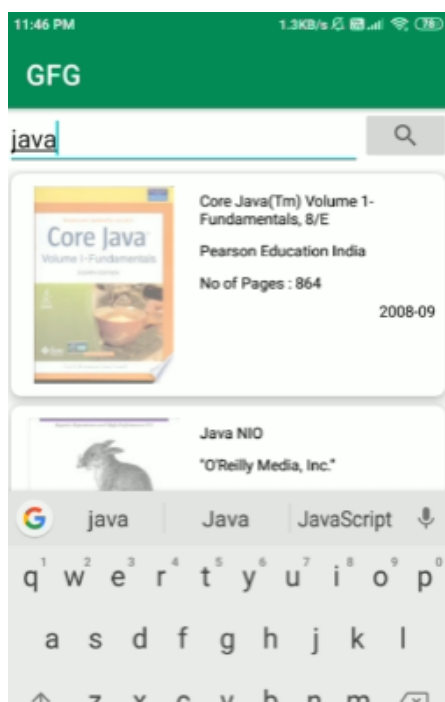
// at last we are adding our json object
// request in our request queue.
queue.add(booksObjrequest);

```

}

}

Output: -



2. Write code to create context menu in android.

- **Activity_main.xml**

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

<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="example.javatpoint.com.contextmenu.MainActivity"
">

    <ListView
        android:layout_width="368dp"
        android:layout_height="495dp"
        android:id="@+id/listView"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</android.support.constraint.ConstraintLayout>
```

- **Main_menu.xml**

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

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

    <item android:id="@+id/call"
        android:title="Call" />

    <item android:id="@+id/sms"
        android:title="SMS" />

</menu>
```

- **Activity class**

```
package example.javatpoint.com.contextmenu;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ListView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
```

```

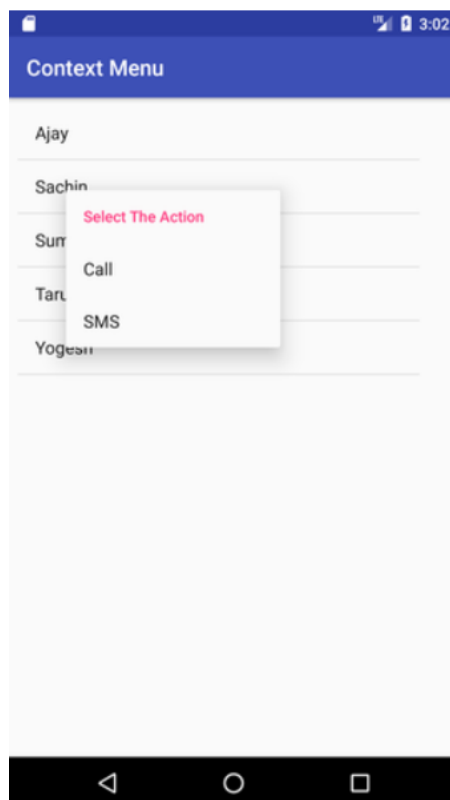
ListView listView;
String contacts[]={"Ajay","Sachin","Sumit","Tarun","Yogesh"};
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    listView=(ListView)findViewById(R.id.listView);
    ArrayAdapter<String> adapter=new
ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,co
ntacts);
    listView.setAdapter(adapter);
    // Register the ListView for Context menu
    registerForContextMenu(listView);
}
@Override
public void onCreateContextMenu(ContextMenu menu, View v,
ContextMenu.ContextMenuInfo menuInfo)
{
    super.onCreateContextMenu(menu, v, menuInfo);
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.menu_main, menu);
    menu.setHeaderTitle("Select The Action");
}
@Override
public boolean onContextItemSelected(MenuItem item){
    if(item.getItemId()==R.id.call){
        Toast.makeText(getApplicationContext(),"calling
code",Toast.LENGTH_LONG).show();
    }
}

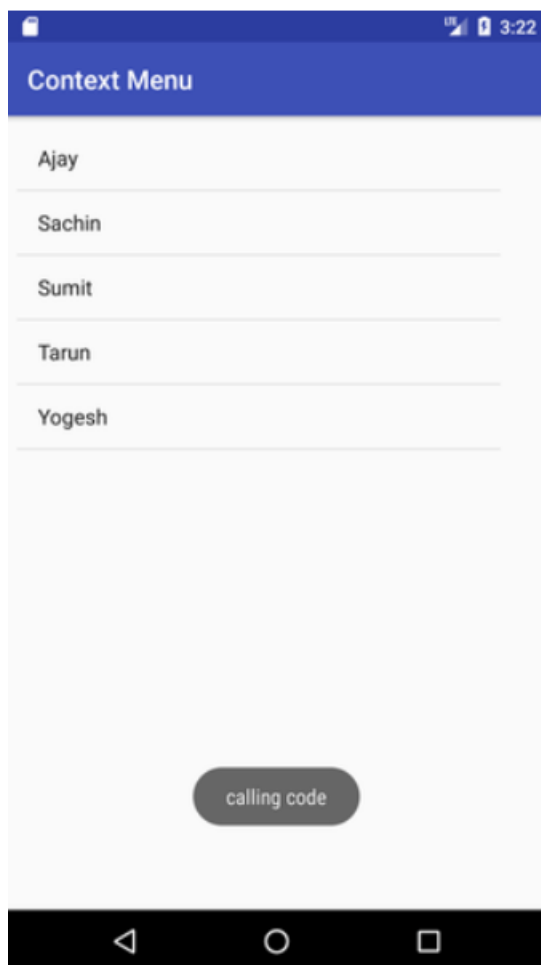
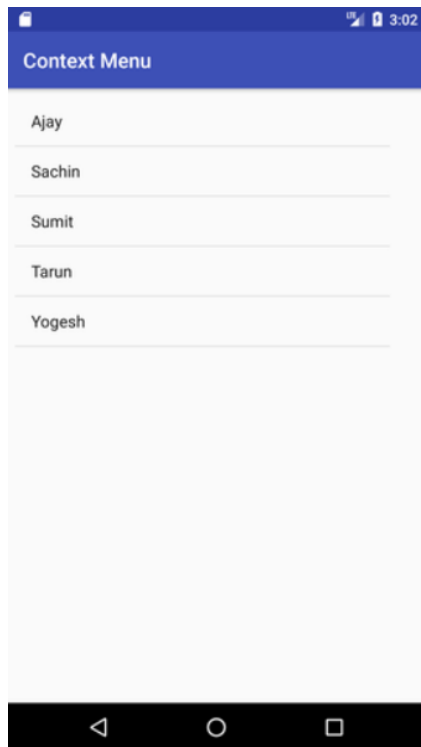
```



```
else if(item.getItemId()==R.id.sms){  
    Toast.makeText(getApplicationContext(),"sending sms  
code",Toast.LENGTH_LONG).show();  
    }else{  
        return false;  
    }  
    return true;  
}  
}
```

Output : -





Android application to create and insert data into SQLite Database

Practical No:8

Date:

Aim: Write an Android application to create and insert data into SQLite Database.

Exercises:

1. Write an Android application to create and insert data into SQLite Database.

- **Adding permissions to access the storage in the AndroidManifest.xml file**

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

- **Working with the activity_main.xml file**

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

```
<LinearLayout
```

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

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

```
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".MainActivity">

<!--Edit text to enter course name-->
<EditText
    android:id="@+id/idEdtCourseName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Enter course Name" />

<!--edit text to enter course duration-->
<EditText
    android:id="@+id/idEdtCourseDuration"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Enter Course Duration" />

<!--edit text to display course tracks-->
<EditText
    android:id="@+id/idEdtCourseTracks"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Enter Course Tracks" />
```

```

<!--edit text for course description-->
<EditText
    android:id="@+id/idEdtCourseDescription"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:hint="Enter Course Description" />

<!--button for adding new course-->
<Button
    android:id="@+id/idBtnAddCourse"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="10dp"
    android:text="Add Course"
    android:textAllCaps="false" />

</LinearLayout>

```

- **Creating a new Java class for performing SQLite operations**

```

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

public class DBHandler extends SQLiteOpenHelper {

```

```
// creating a constant variables for our database.  
// below variable is for our database name.  
private static final String DB_NAME = "coursedb";  
  
// below int is our database version  
private static final int DB_VERSION = 1;  
  
// below variable is for our table name.  
private static final String TABLE_NAME = "mycourses";  
  
// below variable is for our id column.  
private static final String ID_COL = "id";  
  
// below variable is for our course name column  
private static final String NAME_COL = "name";  
  
// below variable id for our course duration column.  
private static final String DURATION_COL = "duration";  
  
// below variable for our course description column.  
private static final String DESCRIPTION_COL =  
"description";  
  
// below variable is for our course tracks column.  
private static final String TRACKS_COL = "tracks";  
  
// creating a constructor for our database handler.
```

```

public DBHelper(Context context) {
    super(context, DB_NAME, null, DB_VERSION);
}

// below method is for creating a database by running a sqlite
query
@Override
public void onCreate(SQLiteDatabase db) {
    // on below line we are creating
    // an sqlite query and we are
    // setting our column names
    // along with their data types.
    String query = "CREATE TABLE " + TABLE_NAME
+ "("
+ ID_COL + " INTEGER PRIMARY
KEY AUTOINCREMENT, "
+ NAME_COL + " TEXT,"
+ DURATION_COL + " TEXT,"
+ DESCRIPTION_COL + " TEXT,"
+ TRACKS_COL + " TEXT)";

    // at last we are calling a exec sql
    // method to execute above sql query
    db.execSQL(query);
}

// this method is use to add new course to our sqlite database.
public void addNewCourse(String courseName, String
courseDuration, String courseDescription, String courseTracks) {

```

```
// on below line we are creating a variable for
// our sqlite database and calling writable method
// as we are writing data in our database.
SQLiteDatabase db = this.getWritableDatabase();

// on below line we are creating a
// variable for content values.
ContentValues values = new ContentValues();

// on below line we are passing all values
// along with its key and value pair.
values.put(NAME_COL, courseName);
values.put(DURATION_COL, courseDuration);
values.put(DESCRIPTION_COL, courseDescription);
values.put(TRACKS_COL, courseTracks);

// after adding all values we are passing
// content values to our table.
db.insert(TABLE_NAME, null, values);

// at last we are closing our
// database after adding database.
db.close();
}
```

@Override


```

        public void onUpgrade(SQLiteDatabase db, int oldVersion,
int newVersion) {
            // this method is called to check if the table exists
already.
            db.execSQL("DROP TABLE IF EXISTS " +
TABLE_NAME);
            onCreate(db);
        }
    }
}

```

- **Working with the MainActivity.java file**

```

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

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    // creating variables for our edittext, button and dbhandler
    private EditText courseNameEdt, courseTracksEdt,
courseDurationEdt, courseDescriptionEdt;

    private Button addCourseBtn;
    private DBHelper dbHelper;

    @Override

```

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    // initializing all our variables.  
    courseNameEdt =  
    findViewById(R.id.idEdtCourseName);  
    courseTracksEdt =  
    findViewById(R.id.idEdtCourseTracks);  
    courseDurationEdt =  
    findViewById(R.id.idEdtCourseDuration);  
    courseDescriptionEdt =  
    findViewById(R.id.idEdtCourseDescription);  
    addCourseBtn =  
    findViewById(R.id.idBtnAddCourse);  
  
    // creating a new dbhandler class  
    // and passing our context to it.  
    dbHandler = new DBHandler(MainActivity.this);  
  
    // below line is to add on click listener for our add  
    course button.  
    addCourseBtn.setOnClickListener(new  
    View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
  
            // below line is to get data from all edit  
            text fields.
```

```

        String courseName =
courseNameEdt.getText().toString();

        String courseTracks =
courseTracksEdt.getText().toString();

        String courseDuration =
courseDurationEdt.getText().toString();

        String courseDescription =
courseDescriptionEdt.getText().toString();

        // validating if the text fields are empty or
not.

        if (courseName.isEmpty() &&
courseTracks.isEmpty() && courseDuration.isEmpty() &&
courseDescription.isEmpty()) {

            Toast.makeText(MainActivity.this,
"Please enter all the data..", Toast.LENGTH_SHORT).show();

            return;

        }

        // on below line we are calling a method
to add new

        // course to sqlite data and pass all our
values to it.

        dbHandler.addNewCourse(courseName,
courseDuration, courseDescription, courseTracks);

        // after adding the data we are displaying
a toast message.

        Toast.makeText(MainActivity.this,
"Course has been added.", Toast.LENGTH_SHORT).show();

        courseNameEdt.setText("");

        courseDurationEdt.setText("");

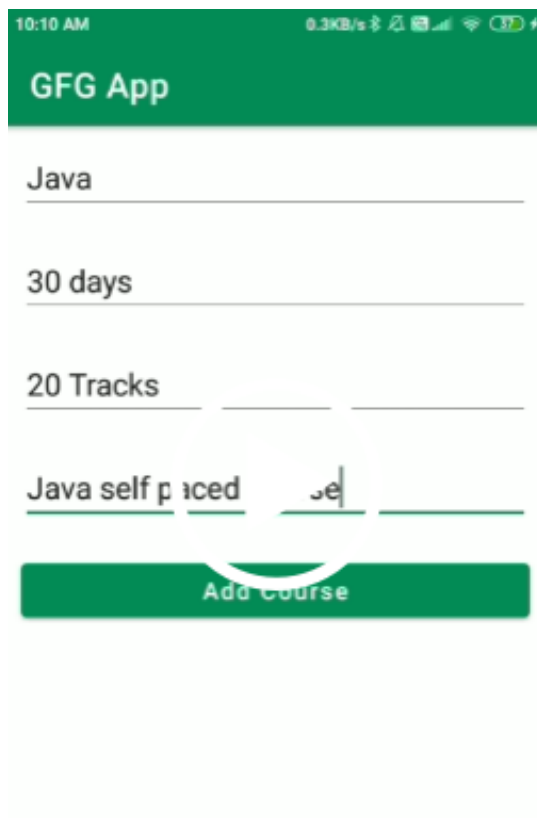
```

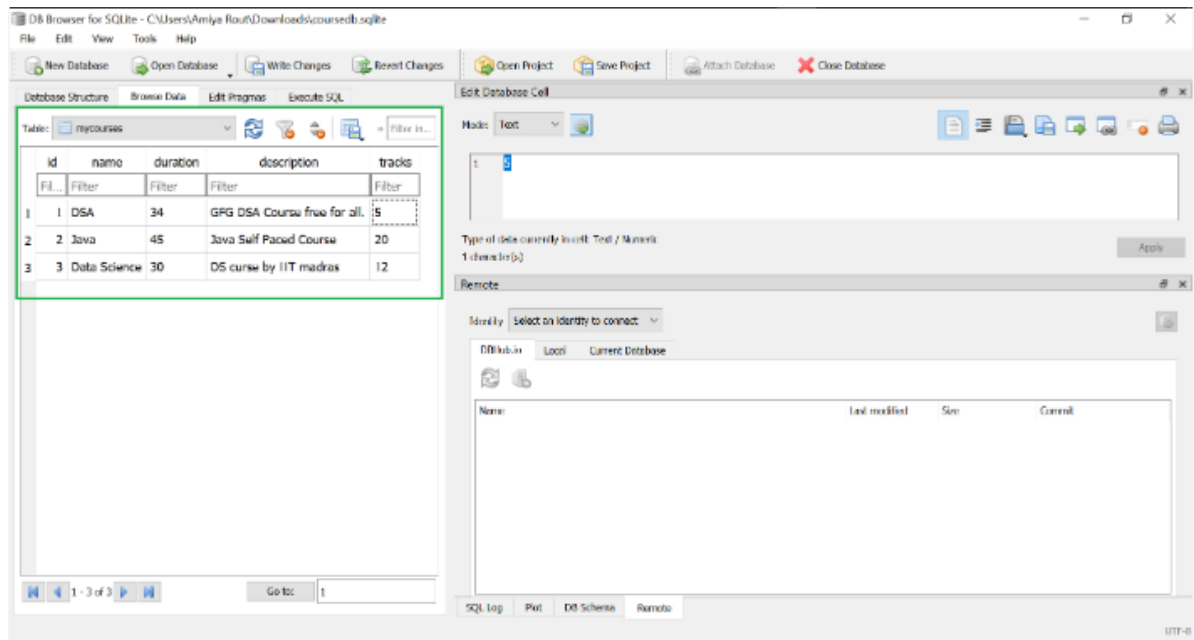
```

courseTracksEdt.setText("");
courseDescriptionEdt.setText("");
    }
});
}
}

```

Output: -





Android application to Animate Ball.

Practical No:9

Date:

Aim: Write an android application to convert a ball from size of radius 2(colour red) to radius 4(colour blue) to radius 6 (colour green). The ball must rotate in circle for 1 minute before changing size and colour.

Exercises:

1. Write an Android application to convert a ball from size of radius 2(colour red) to radius 4(colour blue) to radius 6 (colour green). The ball must rotate in circle for 1 minute before changing size and colour.

```
import java.awt.*;

/**
 * A rectangular container box, containing the bouncing ball.
 */

public class ContainerBox {
    int minX, maxX, minY, maxY; // Box's bounds (package access)
    private Color colorFilled; // Box's filled color (background)
    private Color colorBorder; // Box's border color
    private static final Color DEFAULT_COLOR_FILLED =
        Color.BLACK;
```

```
private static final Color DEFAULT_COLOR_BORDER =  
Color.YELLOW;
```

```
/** Constructors */
```

```
public ContainerBox(int x, int y, int width, int height, Color  
colorFilled, Color colorBorder) {  
    minX = x;  
    minY = y;  
    maxX = x + width - 1;  
    maxY = y + height - 1;  
    this.colorFilled = colorFilled;  
    this.colorBorder = colorBorder;  
}
```

```
/** Constructor with the default color */
```

```
public ContainerBox(int x, int y, int width, int height) {  
    this(x, y, width, height, DEFAULT_COLOR_FILLED,  
DEFAULT_COLOR_BORDER);  
}
```

```
/** Set or reset the boundaries of the box. */
```

```
public void set(int x, int y, int width, int height) {  
    minX = x;  
    minY = y;  
    maxX = x + width - 1;  
    maxY = y + height - 1;  
}
```

```
/** Draw itself using the given graphic context. */
```

```

public void draw(Graphics g) {
    g.setColor(colorFilled);
    g.fillRect(minX, minY, maxX - minX - 1, maxY - minY - 1);
    g.setColor(colorBorder);
    g.drawRect(minX, minY, maxX - minX - 1, maxY - minY - 1);
}

```

The Ball Class

```

import java.awt.*;
import java.util.Formatter;

/**
 * The bouncing ball.
 */
public class Ball {
    float x, y;          // Ball's center x and y (package access)
    float speedX, speedY; // Ball's speed per step in x and y (package
access)
    float radius;        // Ball's radius (package access)
    private Color color; // Ball's color
    private static final Color DEFAULT_COLOR = Color.BLUE;

    /**
     * Constructor: For user friendliness, user specifies velocity in speed
and
     * moveAngle in usual Cartesian coordinates. Need to convert to
speedX and
     * speedY in Java graphics coordinates for ease of operation.
     */
}

```



```

    public Ball(float x, float y, float radius, float speed, float
angleInDegree,
        Color color) {
        this.x = x;
        this.y = y;
        // Convert (speed, angle) to (x, y), with y-axis inverted
        this.speedX = (float)(speed *
Math.cos(Math.toRadians(angleInDegree)));
        this.speedY = (float)(-speed *
(float)Math.sin(Math.toRadians(angleInDegree)));
        this.radius = radius;
        this.color = color;
    }

    /** Constructor with the default color */
    public Ball(float x, float y, float radius, float speed, float
angleInDegree) {
        this(x, y, radius, speed, angleInDegree, DEFAULT_COLOR);
    }

    /** Draw itself using the given graphics context. */
    public void draw(Graphics g) {
        g.setColor(color);
        g.fillOval((int)(x - radius), (int)(y - radius), (int)(2 * radius), (int)
(2 * radius));
    }

    /**
     * Make one move, check for collision and react accordingly if
collision occurs.

```

```

*
* @param box: the container (obstacle) for this ball.
*/

public void moveOneStepWithCollisionDetection(ContainerBox
box) {
    // Get the ball's bounds, offset by the radius of the ball
    float ballMinX = box.minX + radius;
    float ballMinY = box.minY + radius;
    float ballMaxX = box.maxX - radius;
    float ballMaxY = box.maxY - radius;

    // Calculate the ball's new position
    x += speedX;
    y += speedY;

    // Check if the ball moves over the bounds. If so, adjust the
    position and speed.
    if (x < ballMinX) {
        speedX = -speedX; // Reflect along normal
        x = ballMinX;    // Re-position the ball at the edge
    } else if (x > ballMaxX) {
        speedX = -speedX;
        x = ballMaxX;
    }

    // May cross both x and y bounds
    if (y < ballMinY) {
        speedY = -speedY;
        y = ballMinY;
    } else if (y > ballMaxY) {

```

```

        speedY = -speedY;
        y = ballMaxY;
    }
}

```

```

/** Return the magnitude of speed. */
public float getSpeed() {
    return (float)Math.sqrt(speedX * speedX + speedY * speedY);
}

```

```

/** Return the direction of movement in degrees (counter-
clockwise). */
public float getMoveAngle() {
    return (float)Math.toDegrees(Math.atan2(-speedY, speedX));
}

```

```

/** Return mass */
public float getMass() {
    return radius * radius * radius / 1000f; // Normalize by a factor
}

```

```

/** Return the kinetic energy (0.5mv^2) */
public float getKineticEnergy() {
    return 0.5f * getMass() * (speedX * speedX + speedY * speedY);
}

```

```

/** Describe itself. */
public String toString() {

```

```

sb.delete(0, sb.length());
formatter.format("@(%3.0f,%3.0f) r=%3.0f V=(%2.0f,%2.0f) " +
    "S=%4.1f \u0398=%4.0f KE=%3.0f",
    x, y, radius, speedX, speedY, getSpeed(), getMoveAngle(),
    getKineticEnergy()); // \u0398 is theta
return sb.toString();
}
// Re-use to build the formatted string for toString()
private StringBuilder sb = new StringBuilder();
private Formatter formatter = new Formatter(sb);
}

```

The Main class

```

import javax.swing.JFrame;

/**
 * Main Program for running the bouncing ball as a standalone
 * application.
 */
public class Main {
    // Entry main program
    public static void main(String[] args) {
        // Run UI in the Event Dispatcher Thread (EDT), instead of Main
        thread
        javax.swing.SwingUtilities.invokeLater(new Runnable() {
            public void run() {
                JFrame frame = new JFrame("A World of Balls");
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            }
        });
    }
}

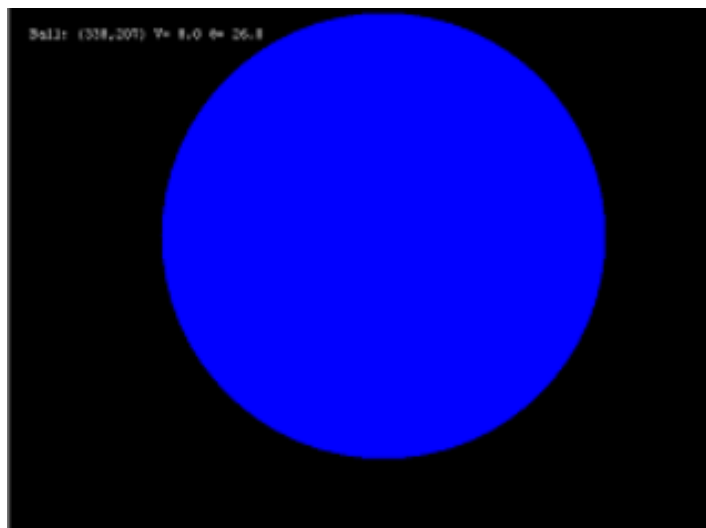
```

```

        frame.setContentPane(new BallWorld(640, 480)); // BallWorld
is a JPanel
        frame.pack();          // Preferred size of BallWorld
        frame.setVisible(true); // Show it
    }
});
}
}

```

Output: -



2. Write an Android application using property animation to rotate an Image for 10 seconds.

- **Create XML File to Define Animation**

```

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

```

```

android" android:interpolator="@android:anim/cycle_interpolator">
  <rotate android:fromDegrees="0"
    android:toDegrees="360"
    android:pivotX="50%"
    android:pivotY="50%"
    android:duration="5000" />
</set>

```

```

<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/
android" android:interpolator="@android:anim/cycle_interpolator">
  <rotate android:fromDegrees="360"
    android:toDegrees="0"
    android:pivotX="50%"
    android:pivotY="50%"
    android:duration="5000" />
</set>

```

- **activity_main.xml**

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/
android"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:paddingLeft="10dp"
  android:paddingRight="10dp">
  <ImageView android:id="@+id/imgvw"
    android:layout_width="wrap_content"
    android:layout_height="250dp"
    android:src="@drawable/bangkok"/>
  <Button
    android:id="@+id/btnRCIk"
    android:layout_below="@+id/imgvw"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

```

```

        android:text="Clockwise" android:layout_marginLeft="100dp" /
    >
    <Button
        android:id="@+id/btnRAClk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@+id/btnRCIk"
        android:layout_toRightOf="@+id/btnRCIk"
        android:text="Anti Clockwise" />
</RelativeLayout>

```

- **rotate_clockwise.xml**

```

<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/cycle_interpolator">
    <rotate android:fromDegrees="0"
        android:toDegrees="360"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="5000" />
</set>

```

- **rotate_anticlockwise.xml**

```

<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/cycle_interpolator">
    <rotate android:fromDegrees="360"
        android:toDegrees="0"
        android:pivotX="50%"

```

```

        android:pivotY="50%"
        android:duration="5000" />
    </set>

```

- **MainActivity.java**

```

package com.tutlane.rotateexample;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private Button btnrclock;
    private Button btnrantick;
    private ImageView img;

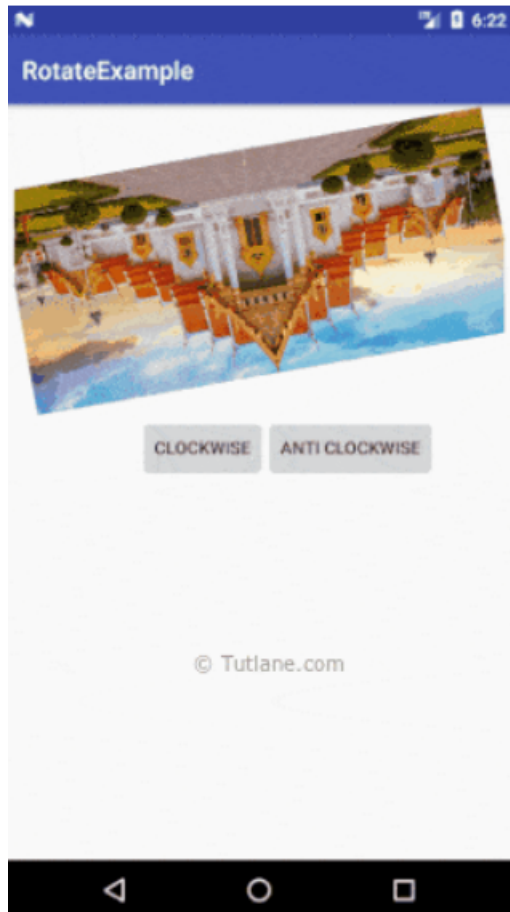
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnrclock = (Button)findViewById(R.id.btnRClk);
        btnrantick = (Button)findViewById(R.id.btnRAClk);
        img = (ImageView)findViewById(R.id.imgvw);
        btnrclock.setOnClickListener(new View.OnClickListener() {
            @Override

```



```
public void onClick(View v) {  
    Animation aniRotateClk =  
        AnimationUtils.loadAnimation(getApplicationContext(),R.anim.ro  
tate_clockwise);  
    img.startAnimation(aniRotateClk);  
}  
});  
  
btnrantick.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Animation animRotateAclk =  
            AnimationUtils.loadAnimation(getApplicationContext(),R.anim.ro  
tate_anticlockwise);  
        img.startAnimation(animRotateAclk);  
    }  
}); }}
```

Output: -



Android application to record audio and Video

Practical No:10

Date:

Aim: Write an application to record video and audio on topic “Intent” and play the audio and video.

Exercises:

1. Write an application to record video and audio on topic “Intent” and play the audio and video.

File: MainActivity.java

```
package com.example.audiomediaplayer1;

import android.media.MediaPlayer;
import android.net.Uri;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.widget.MediaController;
import android.widget.VideoView;

public class MainActivity extends Activity {
```

```

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

    MediaPlayer mp=new MediaPlayer();
    try{
        mp.setDataSource("/sdcard/Music/main.mp3");//Write
your location here
        mp.prepare();
        mp.start();

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

}

```

```

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

}

```

activity_main.xml

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

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

```
<TextView
```

```
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_marginTop="30dp"
    android:text="Audio Controller" />
```

```
<Button
```

```
    android:id="@+id/button1"
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView1"
    android:layout_below="@+id/textView1"
    android:layout_marginTop="48dp"
```

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

```
<Button
```

```
    android:id="@+id/button2"
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/button1"
    android:layout_toRightOf="@+id/button1"
    android:text="pause" />
```

```
<Button
```

```
    android:id="@+id/button3"
    style="?android:attr/buttonStyleSmall"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/button2"
    android:layout_toRightOf="@+id/button2"
    android:text="stop" />
```

```
</RelativeLayout>
```

Activity class

```
package com.example.audioplay;

import android.media.MediaPlayer;
import android.os.Bundle;
```

```
import android.os.Environment;
import android.app.Activity;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;

public class MainActivity extends Activity {
    Button start,pause,stop;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

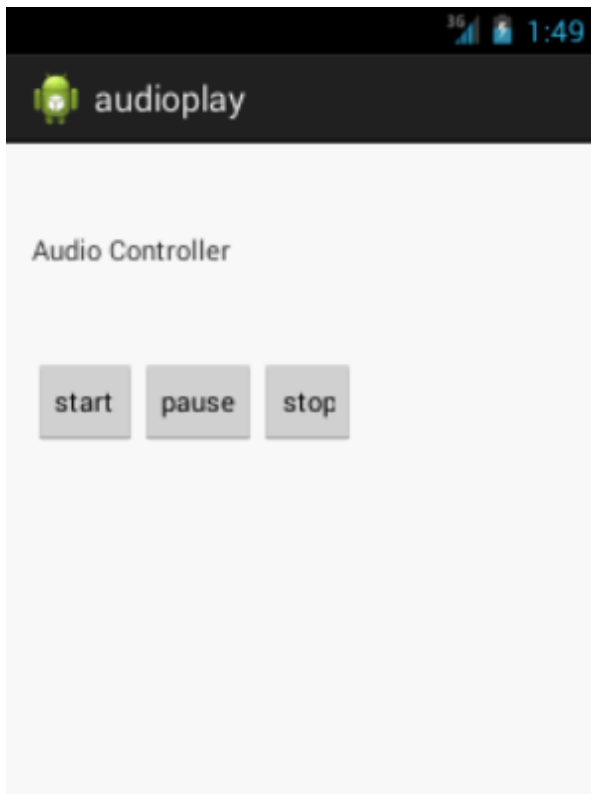
        start=(Button)findViewById(R.id.button1);
        pause=(Button)findViewById(R.id.button2);
        stop=(Button)findViewById(R.id.button3);
        //creating media player
        final MediaPlayer mp=new MediaPlayer();
        try{
            //you can change the path, here path is external directory(e.g.
sdcard) /Music/main.mp3

mp.setDataSource(Environment.getExternalStorageDirectory().getPath()+"/
Music/main.mp3");

        mp.prepare();
        }catch(Exception e){e.printStackTrace();}
```

```
start.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        mp.start();  
    }  
});  
pause.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        mp.pause();  
    }  
});  
stop.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        mp.stop();  
    }  
});  
}  
}
```

Output: -



2. Create a soundpool and play the different sounds on clicking different buttons.

activity_main.xml

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

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

```
<Button  
    android:id="@+id/button_sound1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="playSound"  
    android:text="Game Over"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent" />
```

```
<Button  
    android:id="@+id/button_sound2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="playSound"  
    android:text="Player Died"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.472" />
```

```
<Button  
    android:id="@+id/button_sound3"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"
```

```
        android:onClick="playSound"
        android:text="Level Complete"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.472" />

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

MainActivity.java

```
package org.geeksforgeeks.gfgsoundpool;

import androidx.appcompat.app.AppCompatActivity;
import android.media.AudioAttributes;
import android.media.AudioManager;
import android.media.SoundPool;
import android.os.Build;
import android.os.Bundle;
import android.view.View;

public class MainActivity
```

```
extends AppCompatActivity {  
    SoundPool soundPool;  
    int game_over,  
        level_complete,  
        player_died;  
  
    @Override  
    protected void onCreate(  
        Bundle savedInstanceState)  
    {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        if (Build.VERSION.SDK_INT  
            >= Build.VERSION_CODES.LOLLIPOP) {  
            AudioAttributes  
                audioAttributes  
            = new AudioAttributes  
                .Builder()  
                .setUsage(  
                    AudioAttributes  
                        .USAGE_ASSISTANCE_SONIFICATION)  
                .setContentType(  
                    AudioAttributes  
                        .CONTENT_TYPE_SONIFICATION)  
                .build();  
            soundPool  
                = new SoundPool
```

```
        .Builder()
        .setMaxStreams(3)
        .setAudioAttributes(
            audioAttributes)
        .build();
    }
    else {
        soundPool
            = new SoundPool(
                3,
                AudioManager.STREAM_MUSIC,
                0);
    }
```

```
// This load function takes
// three parameter context,
// file_name and priority.
```

```
game_over
    = soundPool
        .load(
            this,
            R.raw.game_over,
            1);

level_complete
    = soundPool.load(
        this,
        R.raw.level_complete,
        1);
```

```
player_died
    = soundPool.load(
        this,
        R.raw.player_died,
        1);
}

public void playSound(View v)
{
    switch (v.getId()) {

        case R.id.button_sound1:

            // This play function
            // takes five parameter
            // leftVolume, rightVolume,
            // priority, loop and rate.
            soundPool.play(
                game_over, 1, 1, 0, 0, 1);
            soundPool.autoPause();
            break;

        case R.id.button_sound2:
            soundPool.play(
                player_died, 1, 1, 0, 0, 1);
            break;

        case R.id.button_sound3:
```

```
        soundPool.play(  
            level_complete, 1, 1, 0, 0, 1);  
        break;  
    }  
}  
}
```

Output: -

