Practical No. 4

X. Exercise

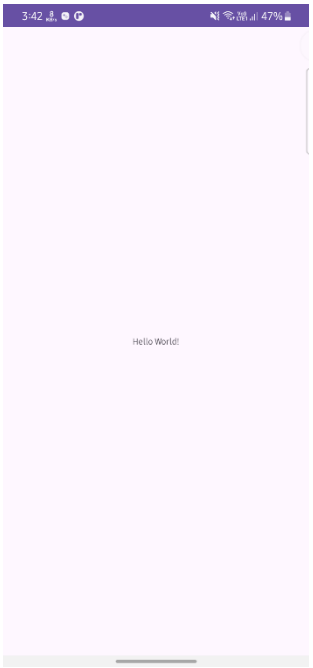
1. Write a program to display HelloWorld.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:gravity="center"
    >

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

</LinearLayout>
```

MainActivity.java

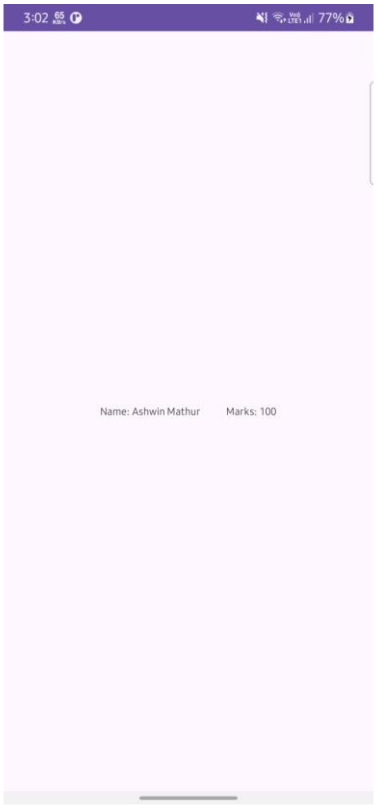
```
package com.example.helloworld;

import android.app.Activity;
import android.os.Bundle;

public class MainActivity extends Activity {
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
    }
}
```



```
Hello World!
```

2. Write a program to display student name and marks.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="horizontal">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name: Ashwin Mathur" />

    <TextView
        android:layout_marginLeft="50px"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Marks: 100" />

</LinearLayout>
```



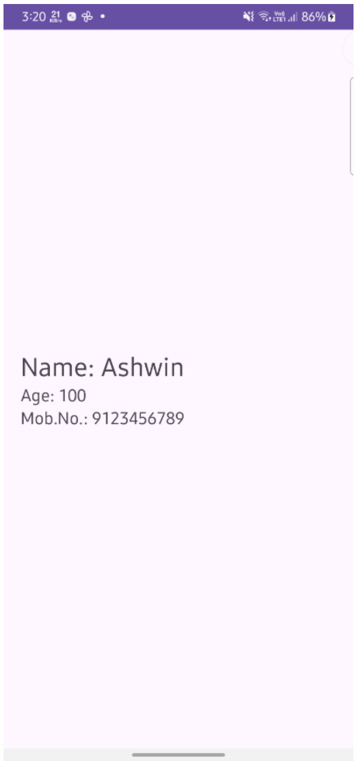
Name: Ashwin Mathur   Marks: 100

Practical No. 5

1. Write a program to place Name, Age and mobile number linearly (Vertical) on the display screen using Linear layout.

XML Code:

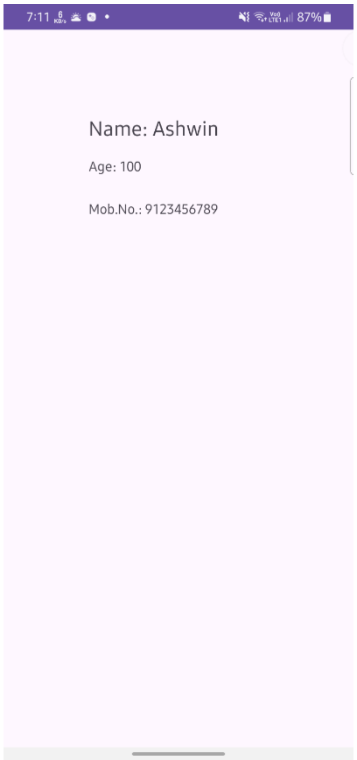
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="20dp"
    android:gravity="center"
    android:orientation="vertical">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Name: Ashwin"
android:textSize="30dp"/>
    <TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Age: 100"
android:textSize="20dp"/>/>
    <TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Mob.No.: 9123456789"
android:textSize="20dp"/>
</LinearLayout>
```



Name: Ashwin
Age: 100
Mob.No.: 9123456789

2. Write a program to place Name, Age and mobile number centrally on the display screen using Absolute layout.

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name: Ashwin"
        android:textSize="30sp"
        android:layout_x="100dp"
        android:layout_y="100dp"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Age: 100"
        android:textSize="20sp"
        android:layout_x="100dp"
        android:layout_y="150dp"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mob.No.: 9123456789"
        android:textSize="20sp"
        android:layout_x="100dp"
        android:layout_y="200dp"/>

</AbsoluteLayout>
```



Name: Ashwin

Age: 100

Mob.No.: 9123456789

Practical No. 6

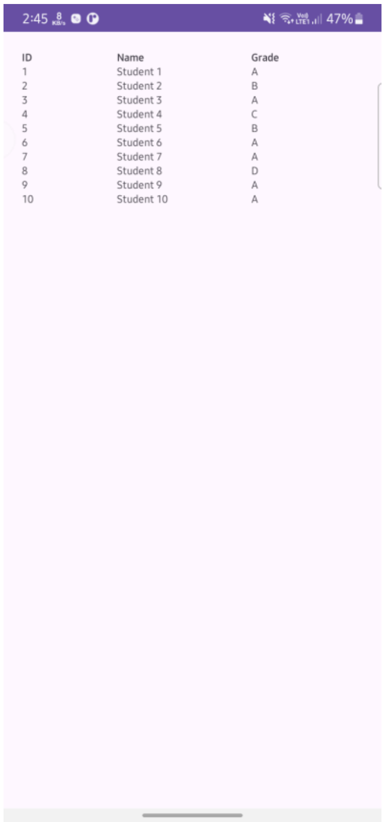
X. Exercise

1. Write a program to display 10 students' basic information in a table form using Table layout.

activity_main.xml

```
<TableLayout
xmlns:android="http://schemas.android.com
/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:stretchColumns="*">

    <TableRow>
        <TextView
            android:text="ID"
            android:textStyle="bold"/>
        <TextView
            android:text="Name"
            android:textStyle="bold"/>
        <TextView
            android:text="Grade"
            android:textStyle="bold"/>
    </TableRow>

    <TableRow>
        <TextView
            android:text="1"/>
        <TextView
            android:text="Student 1"/>
        <TextView
            android:text="A"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="2"/>
        <TextView
            android:text="Student 2"/>
        <TextView
            android:text="B"/>
    </TableRow>
   <TableRow>
        <TextView
            android:text="3"/>
        <TextView
            android:text="Student 3"/>
        <TextView
            android:text="A"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="4"/>
        <TextView
            android:text="Student 4"/>
        <TextView
            android:text="C"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="5"/>
        <TextView
            android:text="Student 5"/>
        <TextView
            android:text="B"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="6"/>
        <TextView
            android:text="Student 6"/>
        <TextView
            android:text="A"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="7"/>
        <TextView
            android:text="Student 7"/>
        <TextView
            android:text="A"/>
    </TableRow>
```

Output:



```
ID          Name          Grade
1           Student 1     A
2           Student 2     B
3           Student 3     A
4           Student 4     C
5           Student 5     B
6           Student 6     A
7           Student 7     A
8           Student 8     D
9           Student 9     A
10          Student 10    A
```

```
    <TableRow>
        <TextView
            android:text="8"/>
        <TextView
            android:text="Student 8"/>
        <TextView
            android:text="D"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="9"/>
        <TextView
            android:text="Student 9"/>
        <TextView
            android:text="A"/>
    </TableRow>
    <TableRow>
        <TextView
            android:text="10"/>
        <TextView
            android:text="Student 10"/>
        <TextView
            android:text="A"/>
    </TableRow>
</TableLayout>
```

2. Write a program to display all the data types in object-oriented programming using Frame layout.

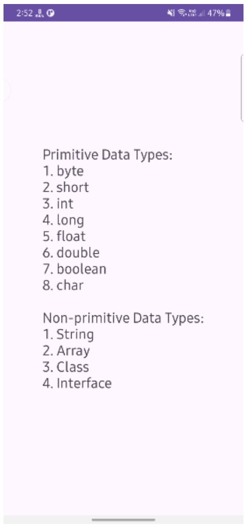
activity_main.xml

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:textSize="30sp"
        android:layout_gravity="center"/>

</FrameLayout>
```

MainActivity.java

```java
package com.example.framelayout;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class MainActivity extends Activity {

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

        TextView textView = findViewById(R.id.textView);
        String dataTypes = "Primitive Data Types:\n" +
                "1. byte\n" +
                "2. short\n" +
                "3. int\n" +
                "4. long\n" +
                "5. float\n" +
                "6. double\n" +
                "7. boolean\n" +
                "8. char\n\n" +
                "Non-primitive Data Types:\n" +
                "1. String\n" +
                "2. Array\n" +
                "3. Class\n" +
                "4. Interface";
        textView.setText(dataTypes);
    }
}
```

Output:



```
Primitive Data Types:
1. byte
2. short
3. int
4. long
5. float
6. double
7. boolean
8. char

Non-primitive Data Types:
1. String
2. Array
3. Class
4. Interface
```

Practical No. 7

X. Exercise

1. Write a program to accept username and password from the end user using Text View and Edit Text

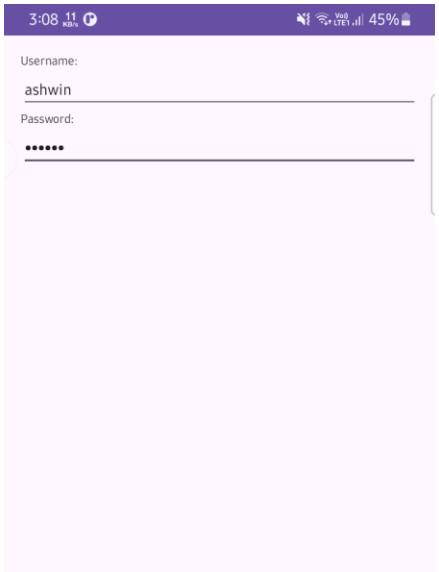
activity_main.xml

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

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

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text" />

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

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword" />

</LinearLayout>
```

MainActivity.java

```
package com.example.edittextdemo;
import android.app.Activity;
import android.os.Bundle;
import android.widget.EditText;

public class MainActivity extends Activity {
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);

        EditText username = findViewById(R.id.username);
        EditText password = findViewById(R.id.password);

        String enteredUsername = username.getText().toString();
        String enteredPassword = password.getText().toString();
    }
}
```

output:



Username:

ashwin

Password:

••••••

2. Write a program to accept and display personal information of the student.

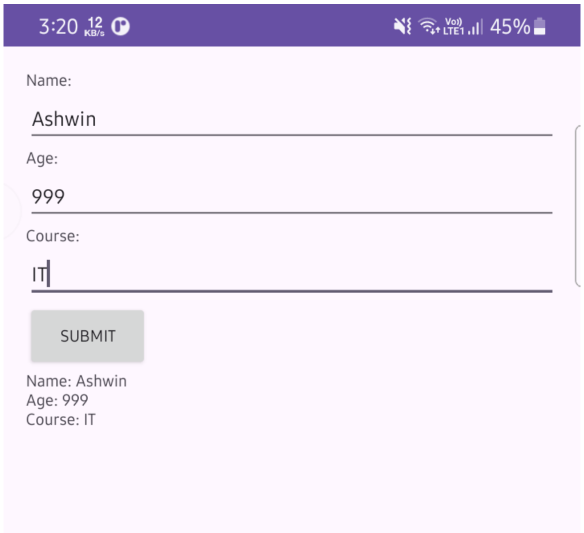
android_activity.xml

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

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

    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text" />

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

    <EditText
        android:id="@+id/age"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="number" />

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

    <EditText
        android:id="@+id/course"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="text" />

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

    <TextView
        android:id="@+id/display"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</LinearLayout>
```

MainActivity.java

```
package com.example.edittextdemo;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {

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

        final EditText name = findViewById(R.id.name);
        final EditText age = findViewById(R.id.age);
        final EditText course = findViewById(R.id.course);
        Button button = findViewById(R.id.button);
        final TextView display = findViewById(R.id.display);

        button.setOnClickListener(new View.OnClickListener() {

            public void onClick(View v) {
                String enteredName = name.getText().toString();
                String enteredAge = age.getText().toString();
                String enteredCourse = course.getText().toString();

                String displayText = "Name: " + enteredName + "\n" +
                        "Age: " + enteredAge + "\n" +
                        "Course: " + enteredCourse;
                display.setText(displayText);
            }
        });
    }
}
```

Output:



Name:

Ashwin

Age:

999

Course:

IT

Name: Ashwin
Age: 999
Course: IT

Practical No. 8

X. Exercise

1. Write a program to create a first display screen of any search engine using Auto Complete Text View.

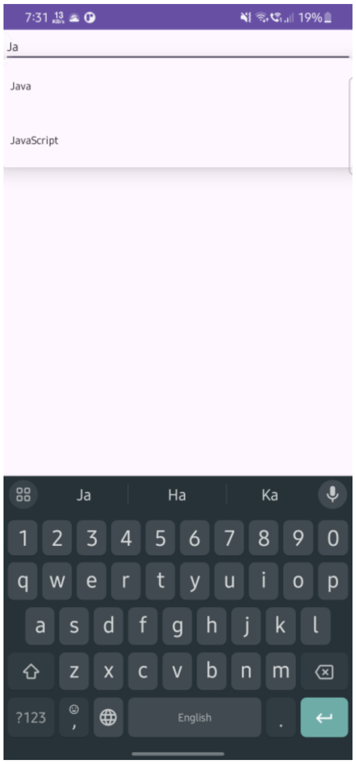
activity_main.xml

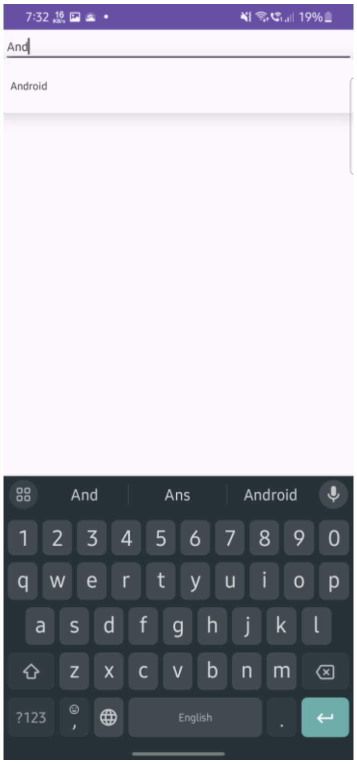
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <AutoCompleteTextView
        android:id="@+id/actv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Search here..." />

</LinearLayout>
```

MainActivity.java

```
package com.example.autocomplete;
import android.os.Bundle;
import android.app.*;
import android.widget.*;

public class MainActivity extends Activity {

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

        AutoCompleteTextView actv = (AutoCompleteTextView) findViewById(R.id.actv);

        String[] suggestions = new String[] {
                "Android", "Java", "Python", "JavaScript", "C++", "C#", "Kotlin",
"Swift", "Ruby", "Go"
        };
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(
                this,
                android.R.layout.simple_dropdown_item_1line,
                suggestions
        );
        actv.setAdapter(adapter);
    }
}
```

Output:





2. Write a program to display all the subjects of sixth semester using Auto Complete Text View

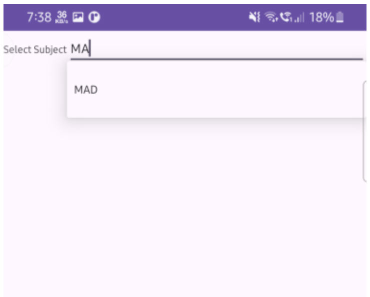
activity_main.xml

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

    <TextView
        android:id="@+id/tv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Subject" />

    <AutoCompleteTextView
        android:id="@+id/actv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Subject" />

</LinearLayout>
```

MainActivity.java

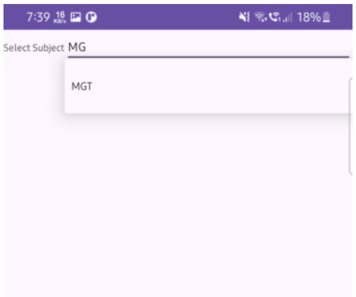
```
package com.example.autocomplete;
import android.os.Bundle;
import android.app.*;
import android.widget.*;

public class MainActivity extends Activity {

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

        AutoCompleteTextView actv = (AutoCompleteTextView) findViewById(R.id.actv);

        String[] suggestions = new String[] {
                "MAD", "NIS", "WMN", "ETI", "MGT"
        };
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(
                this,
                android.R.layout.simple_dropdown_item_1line,
                suggestions
        );
        actv.setAdapter(adapter);
    }
}
```

Output:



Select Subject MA

MAD



Select Subject MG

MGT

Practical No. 9

X. Exercise

1. Write a program to create a toggle button to display ON / OFF Bluetooth on the display screen.

activity_main.xml

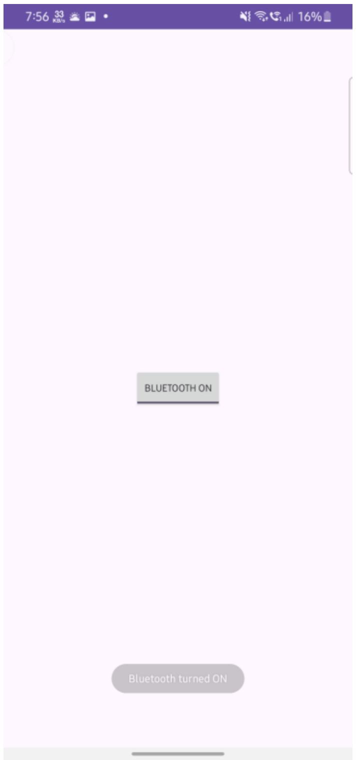
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <ToggleButton
        android:id="@+id/tB"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:textOff="Bluetooth OFF"
        android:textOn="Bluetooth ON" />

</RelativeLayout>
```

MainActivity.java

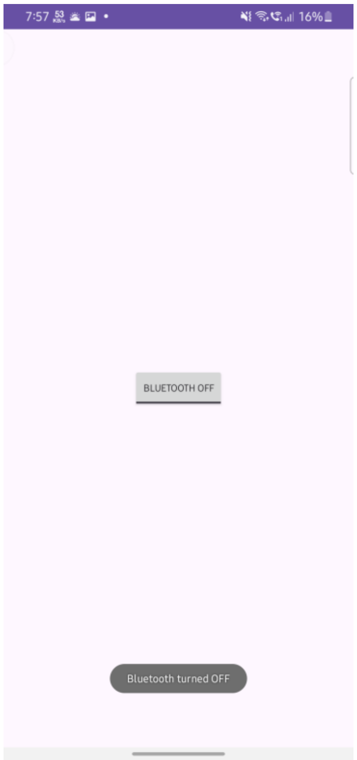
```
package com.example.togglebutton;
import android.os.Bundle;
import android.app.Activity;
import android.bluetooth.*;
import android.widget.*;
import android.view.View;

public class MainActivity extends Activity {

    private BluetoothAdapter btA;

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

        btA = BluetoothAdapter.getDefaultAdapter();
        ToggleButton tB = (ToggleButton) findViewById(R.id.tB);

        tB.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (btA == null) {
                    Toast.makeText(getApplicationContext(), "Device doesn't support
Bluetooth", Toast.LENGTH_SHORT).show();
                } else {
                    if (tB.isChecked()) {
                        btA.enable();
                        Toast.makeText(getApplicationContext(), "Bluetooth turned
ON", Toast.LENGTH_SHORT).show();
                    } else {
                        btA.disable();
                        Toast.makeText(getApplicationContext(), "Bluetooth turned
OFF", Toast.LENGTH_SHORT).show();
                    }
                }
            }
        });
    }
}
```

AndroidManifest.xml (BT and location permissions)

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
</manifest>
```



Bluetooth turned ON



Bluetooth turned OFF

2. Write a program to create a simple calculator.

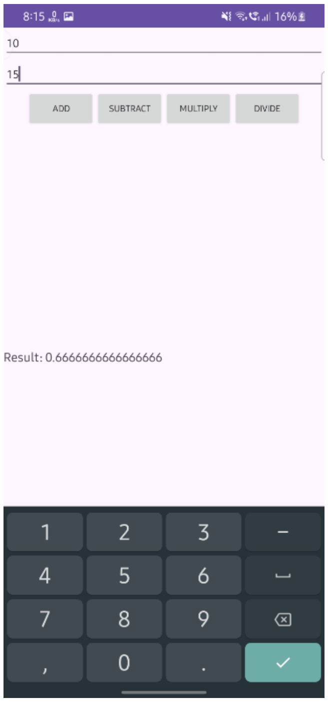
activity_main.xml

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

    <EditText
        android:id="@+id/number1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="numberDecimal"
        android:hint="Enter first number" />

    <EditText
        android:id="@+id/number2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="numberDecimal"
        android:layout_below="@id/number1"
        android:hint="Enter second number" />

    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/number2"
        android:layout_centerInParent="true">
        <Button
            android:id="@+id/addButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"

            android:text="Add" />

        <Button
            android:id="@+id/subtractButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/addButton"
            android:text="Subtract" />

        <Button
            android:id="@+id/multiplyButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/subtractButton"
            android:text="Multiply" />
        <Button
            android:id="@+id/divideButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_toRightOf="@id/multiplyButton"
            android:text="Divide" />
    </RelativeLayout>

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:textSize="20sp"/>
</RelativeLayout>
```

MainActivity.java

```
package com.example.calculator;

import android.app.*;
import android.os.Bundle;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {
    EditText number1, number2;
    Button addButton, subtractButton, multiplyButton, divideButton;
    TextView result;
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);

        number1 = findViewById(R.id.number1);
        number2 = findViewById(R.id.number2);
        addButton = findViewById(R.id.addButton);
        subtractButton = findViewById(R.id.subtractButton);
        multiplyButton = findViewById(R.id.multiplyButton);
        divideButton = findViewById(R.id.divideButton);
        result = findViewById(R.id.result);

        addButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double num1 = Double.parseDouble(number1.getText().toString());
                double num2 = Double.parseDouble(number2.getText().toString());
                double res = num1 + num2;
                result.setText("Result: " + res);
            }
        });
        subtractButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double num1 = Double.parseDouble(number1.getText().toString());
                double num2 = Double.parseDouble(number2.getText().toString());
                double res = num1 - num2;
                result.setText("Result: " + res);
            }
        });

        multiplyButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double num1 = Double.parseDouble(number1.getText().toString());
                double num2 = Double.parseDouble(number2.getText().toString());
                double res = num1 * num2;
                result.setText("Result: " + res);
            }
        });

        divideButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double num1 = Double.parseDouble(number1.getText().toString());
                double num2 = Double.parseDouble(number2.getText().toString());
                double res = num1 / num2;
                result.setText("Result: " + res);
            }
        });
    }
}
```

Output:



10

15

Result: 0.6666666666666666

Practical No. 10

X. Exercise

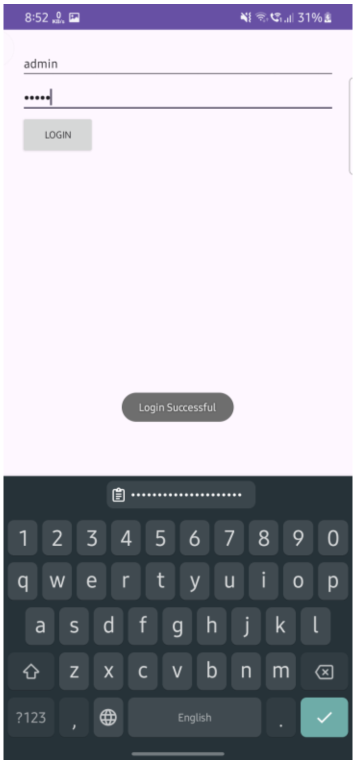
1. Write a program to create a login form for a social networking site.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding = "20dp">

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username" />

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword" />

    <Button
        android:id="@+id/login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick = "login"
        android:text="Login" />

</LinearLayout>
```

MainActivity.java

```
package com.example.forms;

import android.app.*;
import android.os.Bundle;
import android.widget.*;
import android.view.View;

public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        Button btn = findViewById(R.id.login);
        btn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                EditText username = findViewById(R.id.username);
                EditText password = findViewById(R.id.password);
                String user = username.getText().toString();
                String pass = password.getText().toString();
                if (user.equals("admin") && pass.equals("admin")) {
                    Toast.makeText(getApplicationContext(), "Login Successful",
Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(getApplicationContext(), "Login Failed",
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}
```



admin

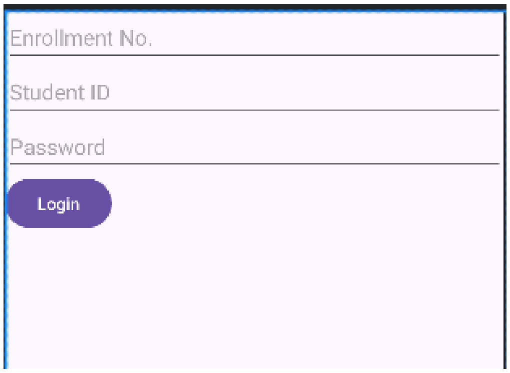
•••••

Login Successful

2. Write a program to create a login form for student registration system

activity_main.xml

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

    <EditText
        android:id="@+id/studentId"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enrollment No." />

    <EditText
        android:id="@+id/studentId"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Student ID" />

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword" />

    <Button
        android:id="@+id/login"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login" />

</LinearLayout>
```



Output:

Practical No. 11

X. Exercise

1. Write a program to show five checkboxes and toast selected checkboxes.

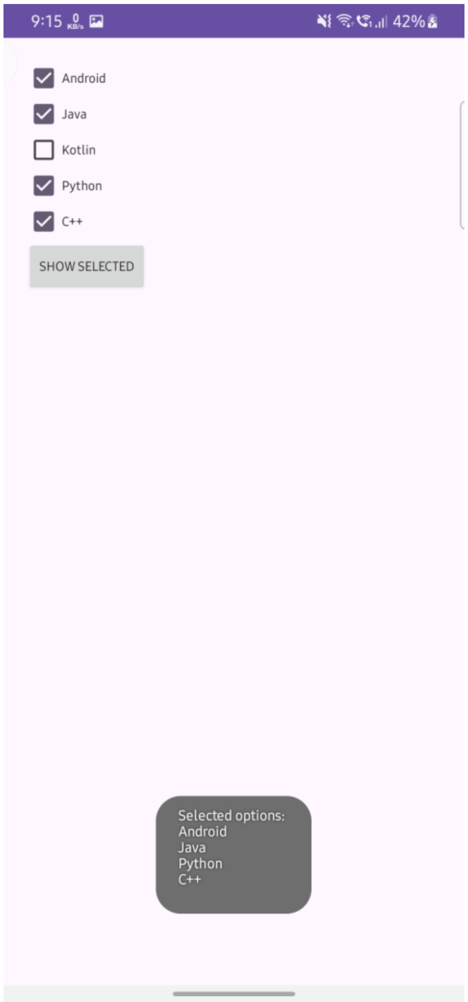
activity_main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding = "20dp">

    <CheckBox
        android:id="@+id/cb1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Android" />

    <CheckBox
        android:id="@+id/cb2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Java" />
    <CheckBox
        android:id="@+id/cb3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Kotlin" />
    <CheckBox
        android:id="@+id/cb4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Python" />
    <CheckBox
        android:id="@+id/cb5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="C++" />

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

</LinearLayout>
```

MainActivity.java

```java
package com.example.checkbox;
import android.os.Bundle;
import android.app.Activity;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {

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

        CheckBox cb1 = (CheckBox) findViewById(R.id.cb1);
        CheckBox cb2 = (CheckBox) findViewById(R.id.cb2);
        CheckBox cb3 = (CheckBox) findViewById(R.id.cb3);
        CheckBox cb4 = (CheckBox) findViewById(R.id.cb4);
        CheckBox cb5 = (CheckBox) findViewById(R.id.cb5);


        Button btnShowSelected = (Button) findViewById(R.id.btnShowSelected);
        btnShowSelected.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                StringBuilder selected = new StringBuilder();
                if (cb1.isChecked()) {
                    selected.append("\n" + cb1.getText().toString() + "\n");
                }
                if (cb2.isChecked()) {
                    selected.append(cb2.getText().toString() + "\n");
                }
                if (cb3.isChecked()) {
                    selected.append(cb3.getText().toString() + "\n");
                }
                if (cb4.isChecked()) {
                    selected.append(cb4.getText().toString() + "\n");
                }
                if (cb5.isChecked()) {
                    selected.append(cb5.getText().toString() + "\n");
                }

                Toast.makeText(getApplicationContext(), "Selected options: " +
selected, Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Output:



☑ Android

☑ Java

☐ Kotlin

☑ Python

☑ C++

```
Selected options:
Android
Java
Python
C++
```

Practical No. 12

X. Exercise

1. Write a program to show the following output. First two radio buttons are without using radio group and next two radio buttons are using radio group. Note the changes between these two. Also toast which radio button has been selected.

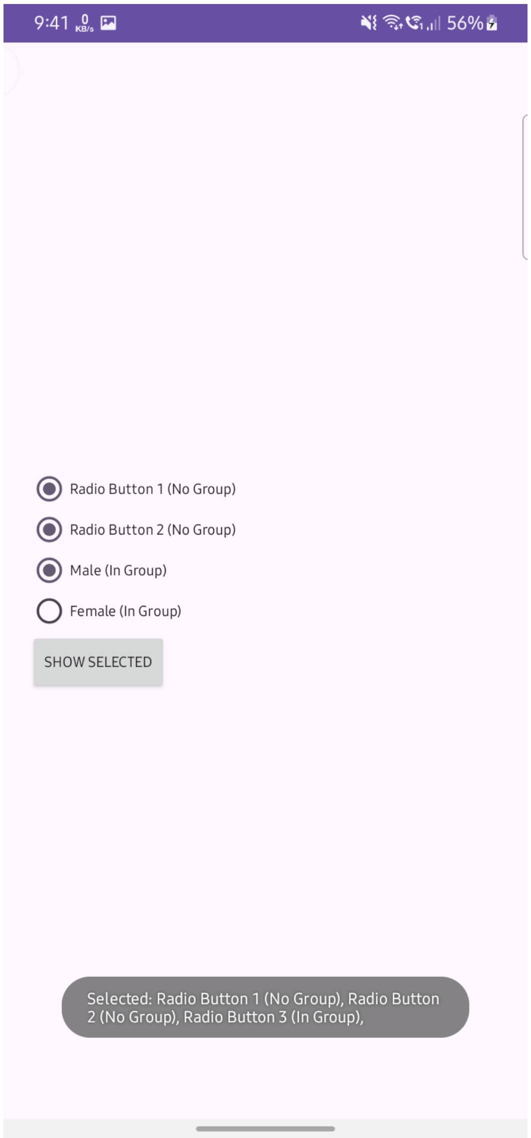
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="start|center"
    android:padding="20dp">

    <RadioButton
        android:id="@+id/rb1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Radio Button 1 (No Group)" />

    <RadioButton
        android:id="@+id/rb2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Radio Button 2 (No Group)" />

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

        <RadioButton
            android:id="@+id/rb3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Male (In Group)" />

        <RadioButton
            android:id="@+id/rb4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Female (In Group)" />

    </RadioGroup>

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

</LinearLayout>
```

MainActivity.java

```
package com.example.checkbox;
import android.os.Bundle;
import android.app.Activity;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {

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

        RadioButton rb1 = (RadioButton) findViewById(R.id.rb1);
        RadioButton rb2 = (RadioButton) findViewById(R.id.rb2);
        RadioButton rb3 = (RadioButton) findViewById(R.id.rb3);
        RadioButton rb4 = (RadioButton) findViewById(R.id.rb4);
        Button buttonShowSelected = (Button) findViewById(R.id.buttonShowSelected);

        buttonShowSelected.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String selected = "";
                if (rb1.isChecked()) {
                    selected += "Radio Button 1 (No Group), ";
                }
                if (rb2.isChecked()) {
                    selected += "Radio Button 2 (No Group), ";
                }
                if (rb3.isChecked()) {
                    selected += "Radio Button 3 (In Group), ";
                }
                if (rb4.isChecked()) {
                    selected += "Radio Button 4 (In Group), ";
                }
                Toast.makeText(getApplicationContext(), "Selected: " + selected,
Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Output:



● Radio Button 1 (No Group)

● Radio Button 2 (No Group)

● Male (In Group)

○ Female (In Group)

Selected: Radio Button 1 (No Group), Radio Button 2 (No Group), Radio Button 3 (In Group),

Practical No. 13

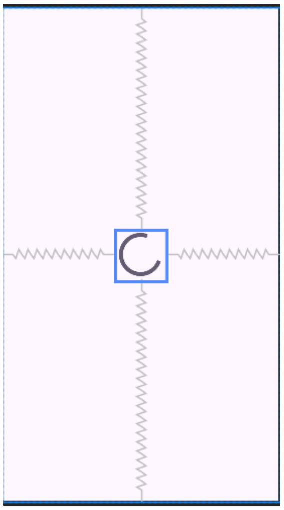
X. Exercise

1. Write a program to display circular progress bar

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" >

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:indeterminate="true" />

</RelativeLayout>
```

Output:



2. Program to show the given output

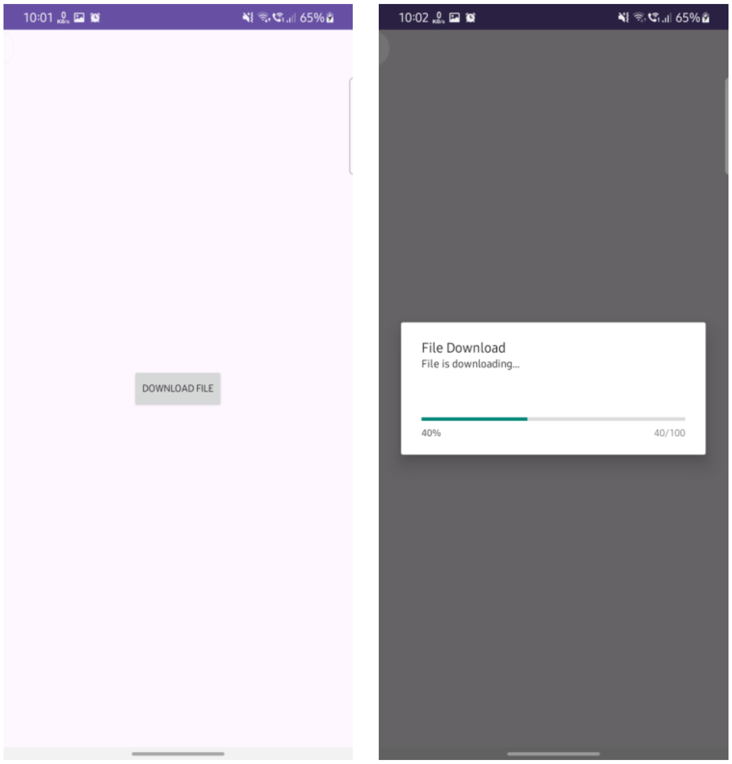
activity_main.xml

```
<?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:gravity = "center">

    <Button
        android:id="@+id/downloadButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Download File" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.progressbar;
import android.os.Bundle;
import android.app.Activity;
import android.app.ProgressDialog;
import android.view.View;
import android.widget.Button;

public class MainActivity extends Activity {

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

        Button downloadButton = (Button) findViewById(R.id.downloadButton);
        downloadButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final ProgressDialog pb = new ProgressDialog(MainActivity.this);
                pb.setTitle("File Download");
                pb.setMessage("File is downloading...");
                pb.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);
                pb.setProgress(0);
                pb.setMax(100);
                pb.show();

                new Thread(new Runnable() {
                    public void run() {
                        for (int progress = 0; progress <= 100; progress += 10) {
                            final int finalProgress = progress;
                            pb.setProgress(finalProgress);
                            try {
                                Thread.sleep(1000);
                            } catch (InterruptedException e) {
                                e.printStackTrace();
                            }
                        }
                        pb.dismiss();
                    }
                }).start();
            }
        });
    }
}
```

Output:



File Download

File is downloading...

40%

40/100

Practical No. 14

X. Exercise

1. Program to show given output

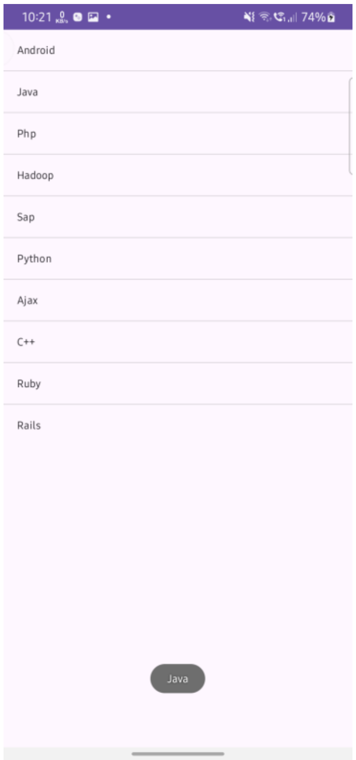
activity_main.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <ListView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/listView"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.helloworld;

import android.app.Activity;
import android.os.Bundle;
import android.widget.*;
import android.view.*;

public class MainActivity extends Activity {

    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        ListView lv = (ListView) findViewById(R.id.listView);
        String[] names = {"Android", "Java", "Php", "Hadoop",
"Sap","Python","Ajax","C++","Ruby","Rails"};
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, names);

        lv.setAdapter(adapter);

        lv.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View view, int position,
long id) {
                Toast.makeText(getApplicationContext(), ((TextView) view).getText(),
Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Output:



```
Android
Java
Php
Hadoop
Sap
Python
Ajax
C++
Ruby
Rails
```

```
Java
```

2. Write a program to display an image using Image View and a button named as "Change Image". Once you click on button another image should get displayed.

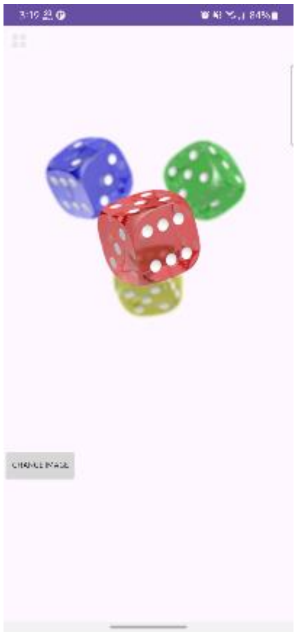
activity_main.xml

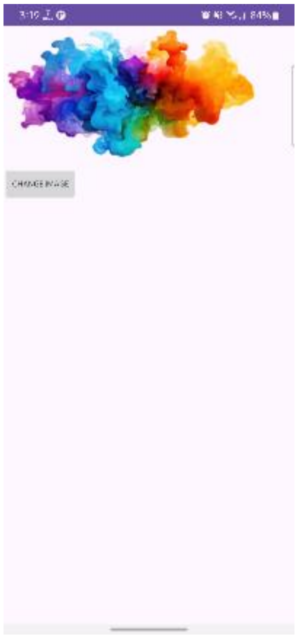
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <ImageView
        android:id="@+id/iv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/img1" />

    <Button
        android:id="@+id/btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Change Image" />
```

```
</LinearLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.app.Activity;
import android.view.*;
import android.widget.*;

public class MainActivity extends Activity {
    boolean isImg1 = true;

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

        ImageView iv = (ImageView) findViewById(R.id.iv);
        Button btn = (Button) findViewById(R.id.btn);

        btn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                if (isImg1) {
                    iv.setImageResource(R.drawable.img2);
                    isImg1 = false;
                } else {
                    iv.setImageResource(R.drawable.img1);
                    isImg1 = true;
                }
            }
        });
    }
}
```





3. Write a program to display 15 buttons using grid view.

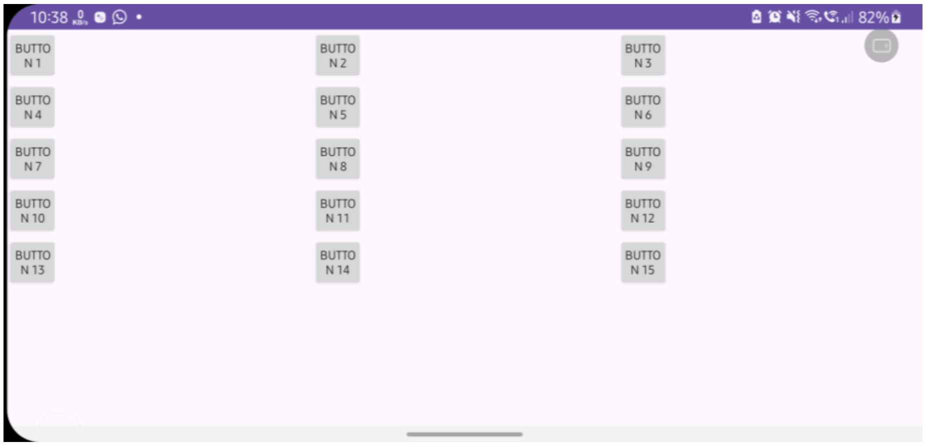
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<GridView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/gridView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:numColumns="3" />
```

MainActivity.java

```
import android.os.Bundle;
import android.app.Activity;
import android.view.*;
import android.widget.*;

public class MainActivity extends Activity {

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

        GridView gridView = (GridView) findViewById(R.id.gridView);
        gridView.setAdapter(new ButtonAdapter());
    }

    public class ButtonAdapter extends BaseAdapter {
        @Override
        public int getCount() {
            return 15;
        }

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

        @Override
        public long getItemId(int position) {
            return 0;
        }

        @Override
        public View getView(int position, View convertView, ViewGroup parent) {
            Button button;
            if (convertView == null) {
                button = new Button(MainActivity.this);
                button.setLayoutParams(new GridView.LayoutParams(85, 85));
                button.setPadding(8, 8, 8, 8);
            } else {
                button = (Button) convertView;
            }
            button.setText("Button " + (position + 1));
            return button;
        }
    }
}
```

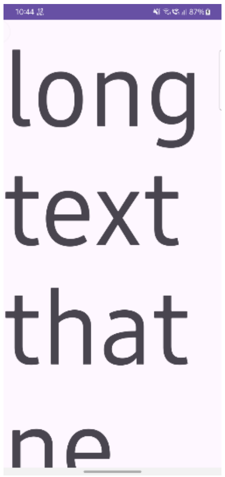
Output:



4. Write a program to display a text view using vertical scroll view.

activity_main.xml

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text = "long text that needs scrolling"
        android:textSize="240sp"/>

</ScrollView>
```



long text that ne

Practical No. 15

X. Exercise

1. Display the given custom toast message

custom_toast.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/custom_toast_layout"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#000000"
    android:orientation="horizontal"
    android:padding="8dp" >

    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="#FFFFFF"
        android:textSize="18sp"
        android:text = "Message for you: \n You've got mail."/>

</LinearLayout>
```

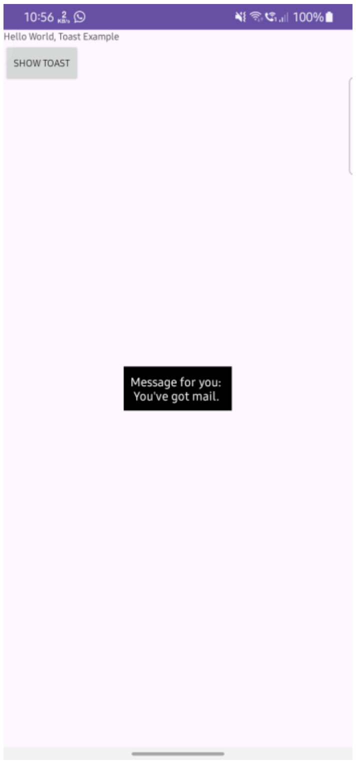
activity_main.xml

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

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

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

</LinearLayout>
```

MainActivity.java

```
package com.example.customtoast;

import android.os.Bundle;
import android.app.Activity;
import android.view.*;
import android.widget.Button;
import android.widget.Toast;

public class MainActivity extends Activity {

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

        Button btn = (Button) findViewById(R.id.btn);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                LayoutInflater inflater = getLayoutInflater();
                View customToastroot = inflater.inflate(R.layout.custom_toast, null);

                Toast customtoast = new Toast(getApplicationContext());

                customtoast.setView(customToastroot);
                customtoast.setGravity(Gravity.CENTER_HORIZONTAL |
Gravity.CENTER_VERTICAL, 0, 0);
                customtoast.setDuration(Toast.LENGTH_LONG);
                customtoast.show();
            }
        });


    }
}
```

```
Output:
```



Hello World, Toast Example

Message for you:
You've got mail.

2. Write a program to display three checkboxes and one button named "Order "as shown below. Once you click on button it should toast different selected checkboxes along with items individual and total price.

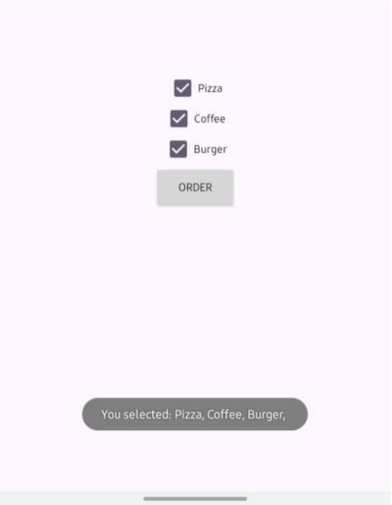
activity_main.xml

```
<?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" >

    <CheckBox
        android:id="@+id/checkBoxPizza"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Pizza"
        android:layout_centerInParent="true" />

    <CheckBox
        android:id="@+id/checkBoxCoffee"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Coffee"
        android:layout_below="@id/checkBoxPizza"
        android:layout_centerHorizontal="true" />
    <CheckBox
        android:id="@+id/checkBoxBurger"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Burger"
        android:layout_below="@id/checkBoxCoffee"
        android:layout_centerHorizontal="true" />

    <Button
        android:id="@+id/buttonOrder"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Order"
        android:layout_below="@id/checkBoxBurger"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.checkbox;
import android.os.Bundle;
import android.app.Activity;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {

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

        CheckBox checkBoxPizza = (CheckBox) findViewById(R.id.checkBoxPizza);
        CheckBox checkBoxCoffee = (CheckBox) findViewById(R.id.checkBoxCoffee);
        CheckBox checkBoxBurger = (CheckBox) findViewById(R.id.checkBoxBurger);
        Button buttonOrder = (Button) findViewById(R.id.buttonOrder);

        buttonOrder.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String selected = "You selected: ";
                if (checkBoxPizza.isChecked()) {
                    selected += "Pizza, ";
                }
                if (checkBoxCoffee.isChecked()) {
                    selected += "Coffee, ";
                }
                if (checkBoxBurger.isChecked()) {
                    selected += "Burger, ";
                }
                Toast.makeText(getApplicationContext(), selected, 
Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Output:



☑ Pizza

☑ Coffee

☑ Burger

You selected: Pizza, Coffee, Burger,

Practical No. 16

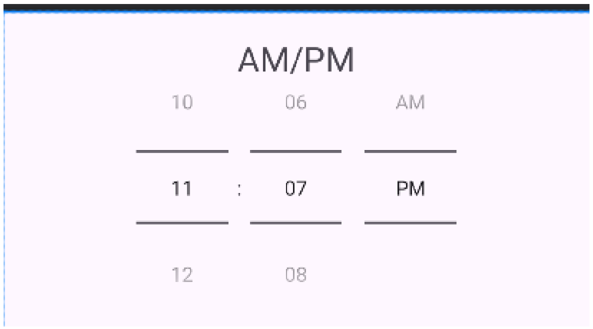
X. Exercise

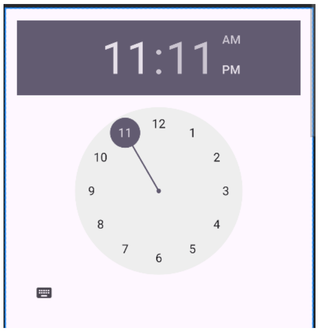
1. Write a program to display the given output

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:padding="16dp">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="AM/PM"
        android:textSize="24sp"
        android:layout_centerHorizontal="true"/>
    <TimePicker
        android:id="@+id/timePicker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:timePickerMode="spinner"
        android:layout_centerHorizontal="true"
        />
</RelativeLayout>
```

Output:





2. Write a program to display following output. Select and display date and time on click of "select date", "select time" buttons respectively.

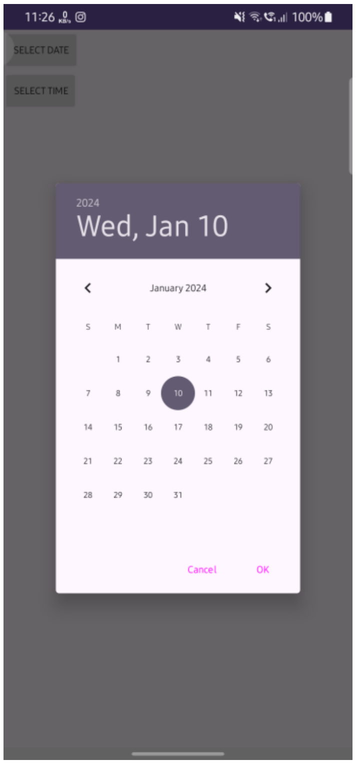
activity_main.xml

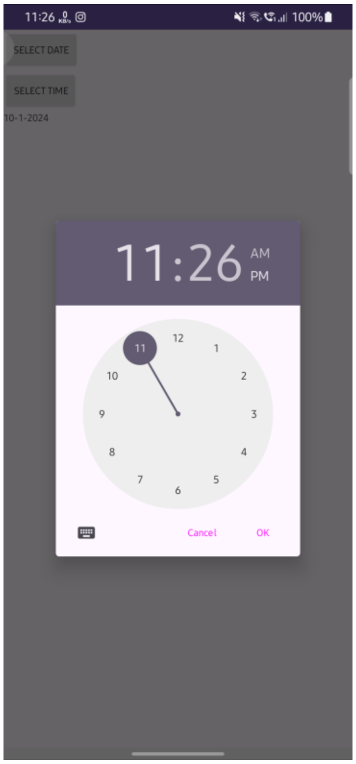
```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Date" />

    <Button
        android:id="@+id/b2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Time"
        android:layout_below="@id/b1" />

    <TextView
        android:id="@+id/tv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/b2" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.datepicker;
import android.app.*;
import android.os.*;
import android.widget.*;
import android.view.*;

import java.sql.Time;

import java.util.Calendar;

public class MainActivity extends Activity {

    private TextView tv;

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

        tv = (TextView) findViewById(R.id.tv);
        Button b1 = (Button) findViewById(R.id.b1);
        Button b2 = (Button) findViewById(R.id.b2);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final Calendar c = Calendar.getInstance();
                int y = c.get(Calendar.YEAR);
                int m = c.get(Calendar.MONTH);
                int d = c.get(Calendar.DAY_OF_MONTH);

                DatePickerDialog dpd = new DatePickerDialog(MainActivity.this,
                        new DatePickerDialog.OnDateSetListener() {
                            @Override
                            public void onDateSet(DatePicker view, int y, int m, int
d) {
                                tv.setText(d + "-" + (m + 1) + "-" + y);
                            }
                        }, y, m, d);
                dpd.show();
            }
        });

        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final Calendar c = Calendar.getInstance();
                int h = c.get(Calendar.HOUR_OF_DAY);
                int m = c.get(Calendar.MINUTE);

                TimePickerDialog tpd = new TimePickerDialog(MainActivity.this,
                        new TimePickerDialog.OnTimeSetListener() {
                            @Override
                            public void onTimeSet(TimePicker view, int h, int m) {
                                tv.setText(h + ":" + m);
                            }
                        }, h, m, false);
                tpd.show();
            }
        });
    }
}
```

Output:


11:26
100%
SELECT DATE
SELECT TIME
2024
Wed, Jan 10
January 2024
S M T W T F S
1 2 3 4 5 6
7 8 9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30 31
Cancel
OK


11:26
100%
SELECT DATE
SELECT TIME
10-1-2024
11:26
AM
PM
12
1
2
3
4
5
6
7
8
9
10
11
Cancel
OK

Practical No. 17

X. Exercise

1. Program to create a HelloWorld Activity using all lifecycle methods to display messages using Log.d

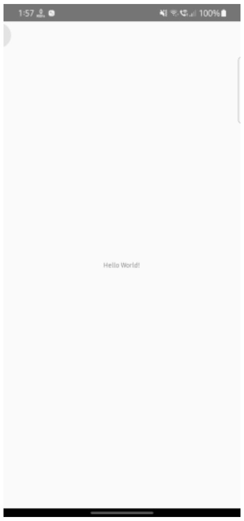
activity_main.xml

```
<?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">

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

</RelativeLayout>
```

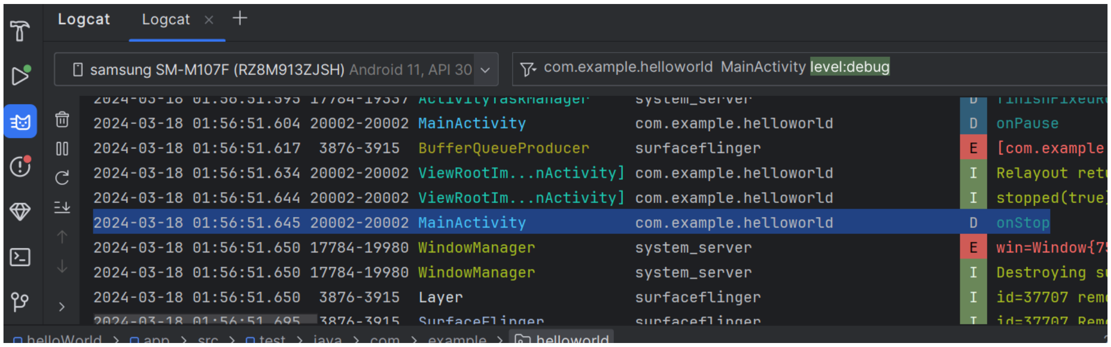
MainActivity.java

```
package com.example.helloworld;
import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
public class MainActivity extends Activity {
    private static final String TAG = "MainActivity";

    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        Log.d(TAG, "onCreate");
    }

    @Override
    protected void onStart() {
        super.onStart();
        Log.d(TAG, "onStart");
    }

    @Override
    protected void onResume() {
        super.onResume();
        Log.d(TAG, "onResume");
    }

    @Override
    protected void onPause() {
        super.onPause();
        Log.d(TAG, "onPause");
    }

    @Override
    protected void onStop() {
        super.onStop();
        Log.d(TAG, "onStop");
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        Log.d(TAG, "onDestroy");
    }

    @Override
    protected void onRestart() {
        super.onRestart();
        Log.d(TAG, "onRestart");
    }
}
```

Output:



Logcat output:


Logcat
Logcat
samsung SM-M107F (RZ8M913ZJSH) Android 11, API 30
com.example.helloworld MainActivity level:debug
helloWorld > app > src > test > java > com > example > helloworld

```
2024-03-18 01:56:51.595 17784-19337 ActivityTaskManager      system_server          D finishFixedR
2024-03-18 01:56:51.604 20002-20002 MainActivity             com.example.helloworld D onPause
2024-03-18 01:56:51.617  3876-3915  BufferQueueProducer      surfaceflinger         E [com.example
2024-03-18 01:56:51.634 20002-20002 ViewRootIm...nActivity] com.example.helloworld I Relayout ret
2024-03-18 01:56:51.644 20002-20002 ViewRootIm...nActivity] com.example.helloworld I stopped(true
2024-03-18 01:56:51.645 20002-20002 MainActivity             com.example.helloworld D onStop
2024-03-18 01:56:51.650 17784-19980 WindowManager            system_server          E win=Window{7
2024-03-18 01:56:51.650 17784-19980 WindowManager            system_server          I Destroying s
2024-03-18 01:56:51.650  3876-3915  Layer                    surfaceflinger         I id=37707 rem
2024-03-18 01:56:51.695  3876-3915  SurfaceFlinger           surfaceflinger         I id=37707 Rem
```

Practical No. 18

1. Write a program to create a text field and a button “Navigate”. When you enter “www.google.com” and press navigate button it should open google page

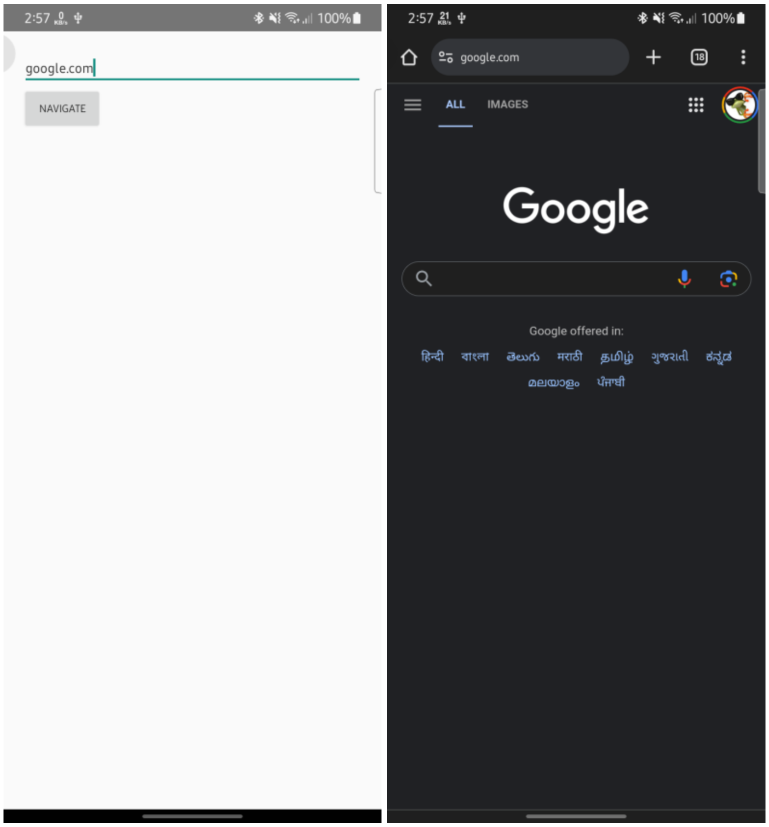
activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding = "20dp">

    <EditText
        android:id="@+id/urlField"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter URL"/>

    <Button
        android:id="@+id/navigateButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Navigate"
        android:layout_below="@+id/urlField"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.intent;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.*;
import android.app.*;

public class MainActivity extends Activity {
    EditText urlField;
    Button navigateButton;

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

        urlField = findViewById(R.id.urlField);
        navigateButton = findViewById(R.id.navigateButton);

        navigateButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String url = urlField.getText().toString();
                Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://" +
url));

                startActivity(intent);
            }
        });
    }
}
```

Output:



google.com

2. Write a program to create button “Start Dialer”. When you click on this button it should open the phone dialer.

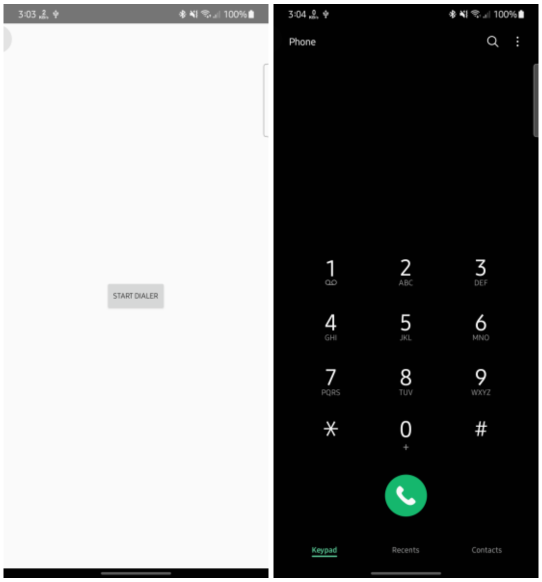
activity_main.xml

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

    <Button
        android:id="@+id/dialerBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start Dialer"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.intent;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.app.Activity;
public class MainActivity extends Activity {
    Button startDialerButton;

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

        startDialerButton = findViewById(R.id.dialerBtn);

        startDialerButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Intent.ACTION_DIAL);
                startActivity(intent);
            }
        });
    }
}
```

Output:



3. Write a program to create two screens. First screen will take one number input from user. After click on Factorial button, second screen will open and it should display factorial of the same number. Also specify which type of intent you will use in this case.

Here, we will use explicit intents because we need to start an activity from a user defined class.

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding = "20dp"
    android:gravity="center">

    <EditText
        android:id="@+id/numberField"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="number"
        android:hint="Enter a number"/>

    <Button
        android:id="@+id/factorialButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Factorial"
        android:layout_below="@+id/numberField"/>

</RelativeLayout>
```

activity_second.xml

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

    <TextView
        android:id="@+id/factRes"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.intent;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.*;
import android.app.Activity;
public class MainActivity extends Activity {
    EditText numberField;
    Button factorialButton;

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

        numberField = findViewById(R.id.numberField);
        factorialButton = findViewById(R.id.factorialButton);

        factorialButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int number = Integer.parseInt(numberField.getText().toString());
                Intent intent = new Intent(MainActivity.this, SecondActivity.class);
                intent.putExtra("number", number);
                startActivity(intent);
            }
        });
    }
}
```

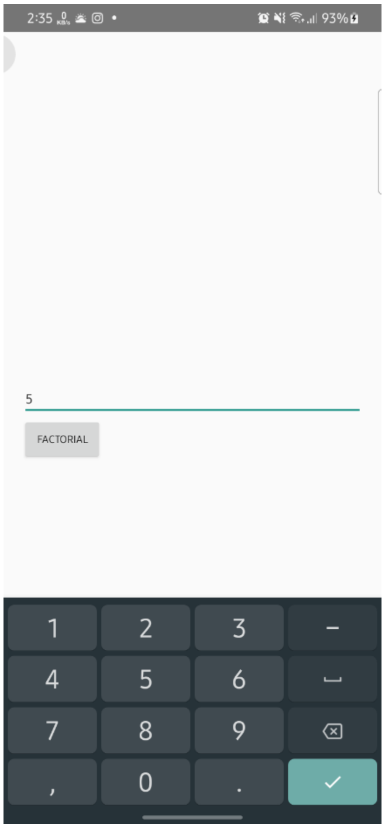
SecondActivity.java

```
package com.example.intent;
import android.os.Bundle;
import android.widget.TextView;
import android.app.Activity;
public class SecondActivity extends Activity {
    TextView factorialResult;

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

        factorialResult = findViewById(R.id.factRes);

        int number = getIntent().getIntExtra("number", 1);
        int factorial = factorial(number);

        factorialResult.setText("Factorial: " + factorial);
    }

    private int factorial(int n) {
        if (n == 0) {
            return 1;
        } else {
            return n * factorial(n - 1);
        }
    }
}
```

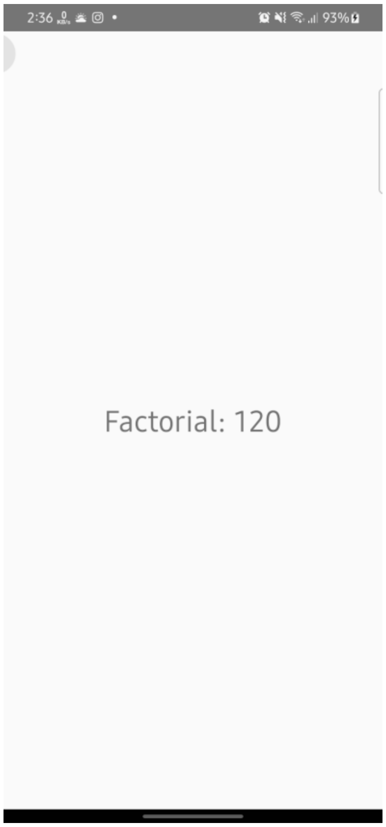
Defining the second activity in AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    <application
        <activity android:name=".SecondActivity">

        </activity>
    </application>
</manifest>
```

Output:



5



Factorial: 120

Practical No. 19

X. Exercise

1. Write a program to create your own content provider to insert and access data in android application

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="20dp">

    <EditText
        android:id="@+id/dataInput"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Data to Store" />

    <Button
        android:id="@+id/insertButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Insert" />

    <Button
        android:id="@+id/queryButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Query" />

    <TextView
        android:id="@+id/queryResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"/>

</LinearLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.content.ContentValues;
import android.database.Cursor;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    EditText dataInput;
    Button insertButton, queryButton;
    TextView queryResult;

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

        dataInput = findViewById(R.id.dataInput);
        insertButton = findViewById(R.id.insertButton);
        queryButton = findViewById(R.id.queryButton);
        queryResult = findViewById(R.id.queryResult);

        insertButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String data = dataInput.getText().toString();
                ContentValues values = new ContentValues();
                values.put("column1", data); // Replace 'column1' with your actual
column name

                getContentResolver().insert(MyContentProvider.CONTENT_URI, values);
                dataInput.setText("");
            }
        });

        queryButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Cursor cursor =
getContentResolver().query(MyContentProvider.CONTENT_URI, null, null, null, null);

                if (cursor.moveToFirst()) {
                    StringBuilder builder = new StringBuilder();
                    do {
                        String data =
cursor.getString(cursor.getColumnIndex("column1"));
                        builder.append(data).append("\n");
                    } while (cursor.moveToNext());
                    queryResult.setText(builder.toString());
                } else {
                    queryResult.setText("No data found");
                }
                cursor.close();
            }
        });
    }
}
```

MyContentProvider.java

```
import android.content.ContentProvider;
import android.content.ContentUris;
import android.content.ContentValues;
import android.content.UriMatcher;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.net.Uri;

public class MyContentProvider extends ContentProvider {

    // Define constants
    private static final String AUTHORITY = "com.example.mycontentauthority";
    private static final String TABLE_NAME = "my_data";
    public static final Uri CONTENT_URI = Uri.parse("content://" + AUTHORITY + "/" +
TABLE_NAME);

    private static final int DATA = 1;
    private static final int DATA_ID = 2;
    private static final UriMatcher uriMatcher = buildUriMatcher();

    private SQLiteOpenHelper dbHelper;

    private static UriMatcher buildUriMatcher() {
        UriMatcher matcher = new UriMatcher(UriMatcher.NO_MATCH);
        matcher.addURI(AUTHORITY, TABLE_NAME, DATA);
        matcher.addURI(AUTHORITY, TABLE_NAME + "/#", DATA_ID);
        return matcher;
    }

    @Override
    public boolean onCreate() {
        dbHelper = new MyDatabaseHelper(getContext());
        return true;
    }

    @Override
    public Cursor query(Uri uri, String[] projection, String selection, String[]
selectionArgs, String sortOrder) {
        SQLiteDatabase db = dbHelper.getReadableDatabase();
        Cursor cursor;
        switch (uriMatcher.match(uri)) {
            case DATA:
                cursor = db.query(TABLE_NAME, projection, selection, selectionArgs,
null, null, sortOrder);
                break;
            case DATA_ID:
                long id = ContentUris.parseId(uri);
                cursor = db.query(TABLE_NAME, projection, "_id = ?", new
String[]{String.valueOf(id)}, null, null, sortOrder);
                break;
            default:
                throw new IllegalArgumentException("Unknown URI: " + uri);
        }
        cursor.setNotificationUri(getContext().getContentResolver(), uri);
        return cursor;
    }

    @Override
    public Uri insert(Uri uri, ContentValues values) {
        SQLiteDatabase db = dbHelper.getWritableDatabase();
        if (uriMatcher.match(uri) != DATA) {
            throw new IllegalArgumentException("Unknown URI: " + uri);
        }
        long rowId = db.insert(TABLE_NAME, null, values);
        if (rowId > 0) {
            Uri itemUri = ContentUris.withAppendedId(CONTENT_URI, rowId);
            getContext().getContentResolver().notifyChange(itemUri, null);
            return itemUri;
        }
        return null;
    }

    // ... (Implement update and delete if needed)

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

    @Override
    public int update(Uri uri, ContentValues values, String selection, String[]
selectionArgs) {
        throw new UnsupportedOperationException("Not yet implemented");
    }

    @Override
    public int delete(Uri uri, String selection, String[] selectionArgs) {
        throw new UnsupportedOperationException("Not yet implemented");
    }

    // ========= SQLiteOpenHelper =========
    private static class MyDatabaseHelper extends SQLiteOpenHelper {
        private static final String DATABASE_NAME = "my_data.db";
        private static final int DATABASE_VERSION = 1;

        public MyDatabaseHelper(Context context) {
            super(context, DATABASE_NAME, null, DATABASE_VERSION);
        }
        @Override
        public void onCreate(SQLiteDatabase db) {
            String CREATE_TABLE_SQL = "CREATE TABLE " + TABLE_NAME +
                    " (_id INTEGER PRIMARY KEY, data TEXT)"; // Data is your column
name
            db.execSQL(CREATE_TABLE_SQL);
        }

        @Override
        public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
            // Handle database upgrades (if needed)
        }
    }
}
```

MyDatabaseHelper.java

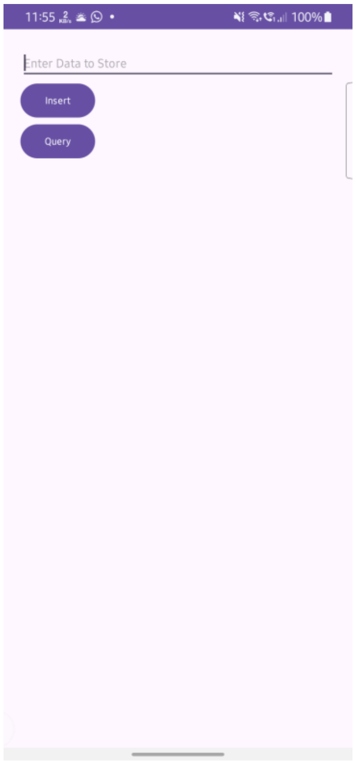
```
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.content.Context;

public class MyDatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "my_data.db";
    private static final int DATABASE_VERSION = 1;

    public MyDatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_TABLE_SQL = "CREATE TABLE " + TABLE_NAME +
                                  "(_id INTEGER PRIMARY KEY, column1 TEXT)";
        db.execSQL(CREATE_TABLE_SQL);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        // Handle schema changes if needed
    }
}
```

AndroidManifest.xml

```
<manifest>
<application>
<provider
    android:name=".MyContentProvider"
    android:authorities="com.example.mycontentprovider"
    android:exported="true" />
</application>
```

```
</manifest>
```

Output:



Practical No. 20

X. Exercise

1. Write a program to start Wi-Fi using service.

Starting Android 10 (API Level 29), apps are not allowed to enable or disable Wi-Fi. The WifiManager.setWifiEnabled() method was deprecated in API level 29.

Following is an approach to guide users to the Wi-Fi settings page on their device using intent.

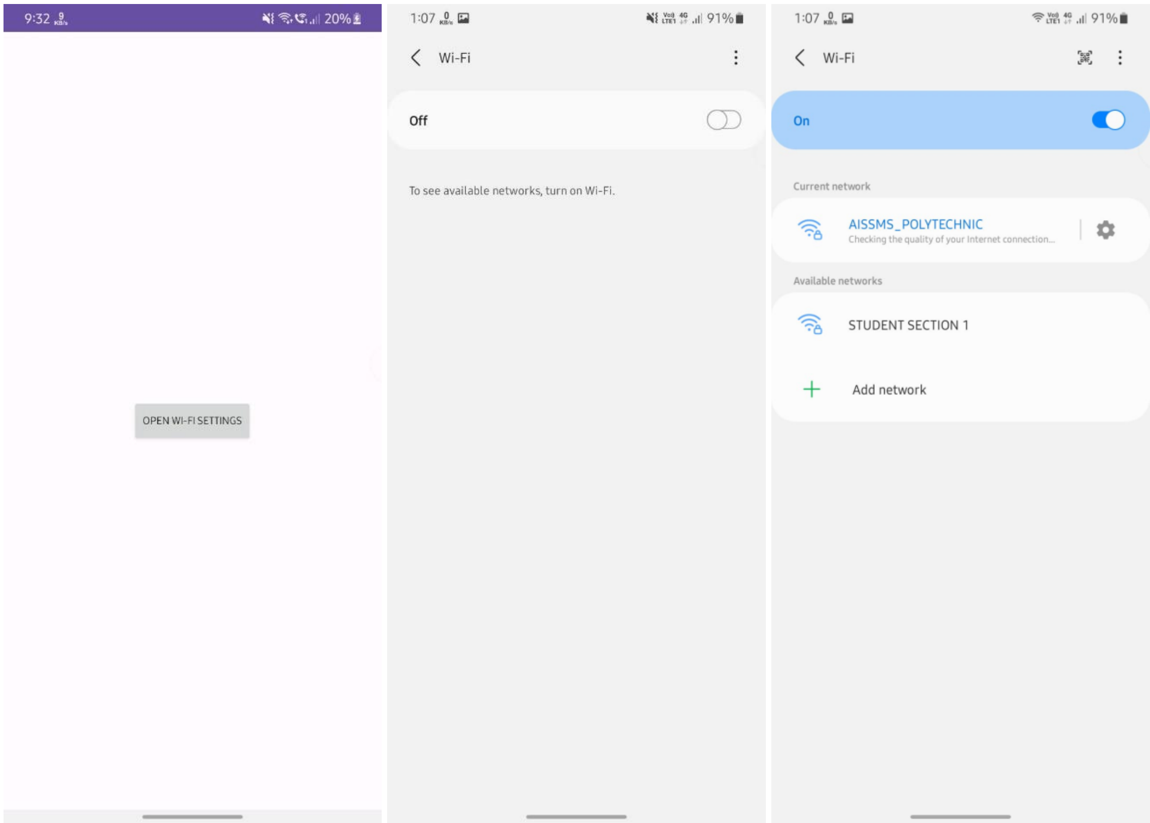
activity_main.xml

```
<?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">

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Open Wi-Fi Settings"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

ActivityMain.java

```
package com.example.wifiservice;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.app.Activity;

public class MainActivity extends Activity {

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

        Button button = findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new
Intent(android.provider.Settings.ACTION_WIFI_SETTINGS));
            }
        });
    }
}
```

Output:



To see available networks, turn on Wi-Fi.

Checking the quality of your Internet connection...

2. Write a program to display the given output

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">

    <Button
        android:id="@+id/startServiceButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start Service"
        android:layout_centerInParent="true"/>

    <Button
        android:id="@+id/stopServiceButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Stop Service"
        android:layout_below="@+id/startServiceButton"
        android:layout_centerHorizontal="true"/>
</RelativeLayout>
```

MainActivity.java

```java
package com.example.demo;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.app.Activity;

public class MainActivity extends Activity {
    Button startServiceButton, stopServiceButton;

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

        startServiceButton = findViewById(R.id.startServiceButton);
        stopServiceButton = findViewById(R.id.stopServiceButton);

        startServiceButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startService(new Intent(MainActivity.this, MyService.class));
            }
        });

        stopServiceButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                stopService(new Intent(MainActivity.this, MyService.class));
            }
        });
    }
}
```

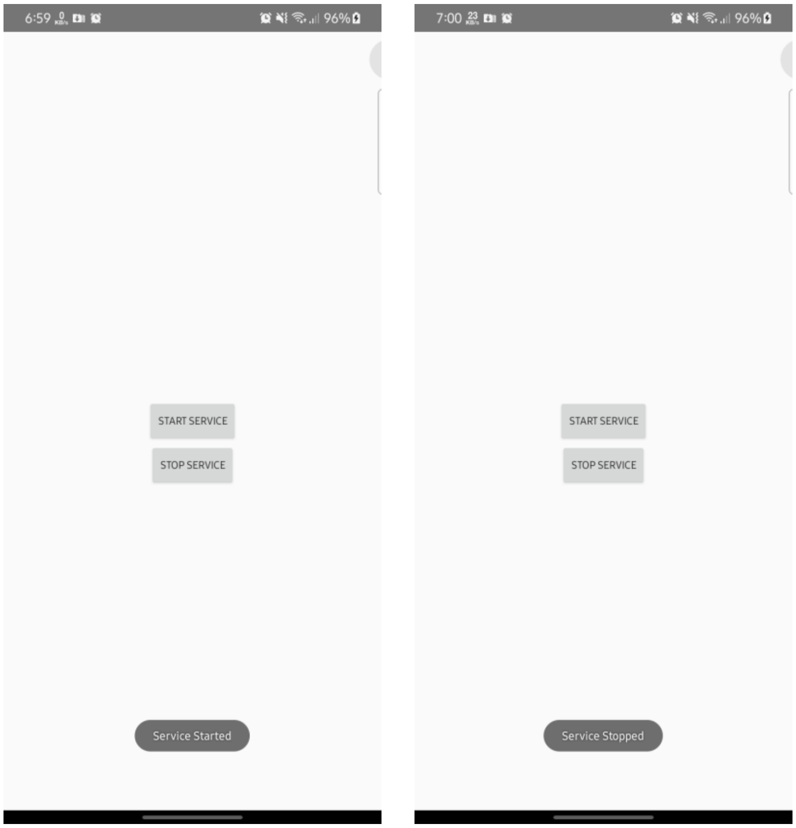
MyService.java (service class)

```java
package com.example.demo;
import android.app.Service;
import android.content.Intent;
import android.os.IBinder;
import android.widget.Toast;

public class MyService extends Service {
    @Override
    public IBinder onBind(Intent intent) {
        return null;
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        Toast.makeText(this, "Service Started", Toast.LENGTH_LONG).show();
        return START_STICKY;
    }
    @Override
    public void onDestroy() {
        super.onDestroy();
        Toast.makeText(this, "Service Stopped", Toast.LENGTH_LONG).show();
    }
}
```

AndroidManifest.xml (add inside <application> tag)

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.demo">
    <application
        ...
        <service android:name=".MyService"/>
         ...
    </application>
</manifest>
```

Output:



Service Started

Service Stopped

Practical No. 21

X. Exercise

1. Write a program to demonstrate system broadcast messages

Broadcast receiver that listens for CONECTIVITY_ACTION broadcast, , which is sent whenever the device's network connectivity changes

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">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Network Status"
        android:textSize="20sp"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

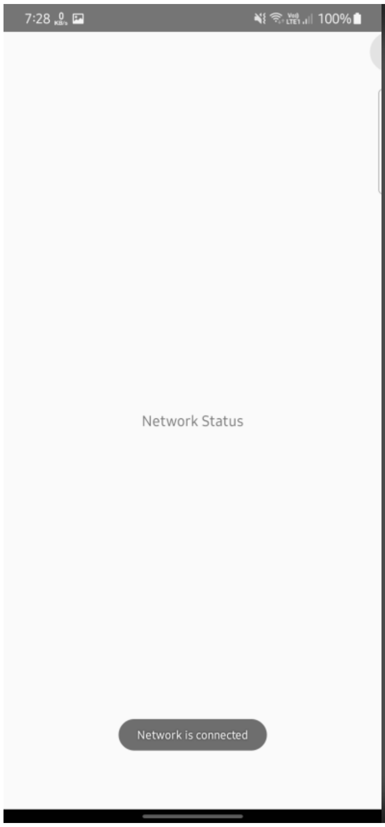
MainActivity.java

```
package com.example.broadcast;

import android.content.*;
import android.net.*;
import android.widget.Toast;
import android.app.Activity;
import android.os.*;

public class MainActivity extends Activity {
    public void onCreate(Bundle b)
    {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
    }
    final BroadcastReceiver networkChangeReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            ConnectivityManager cmgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo activeNetwork = cmgr.getActiveNetworkInfo();
            boolean isConnected = activeNetwork != null &&
activeNetwork.isConnectedOrConnecting();

            if (isConnected) {
                Toast.makeText(context, "Network is connected",
Toast.LENGTH_LONG).show();
            } else {
                Toast.makeText(context, "Network is disconnected",
Toast.LENGTH_LONG).show();
            }
        }
    };

    @Override
    protected void onResume() {
        super.onResume();
        registerReceiver(networkChangeReceiver, new
IntentFilter(ConnectivityManager.CONNECTIVITY_ACTION));
    }

    @Override
    protected void onPause() {
        super.onPause();
        unregisterReceiver(networkChangeReceiver);
    }
}
```

AndroidManifest.xml (permission)

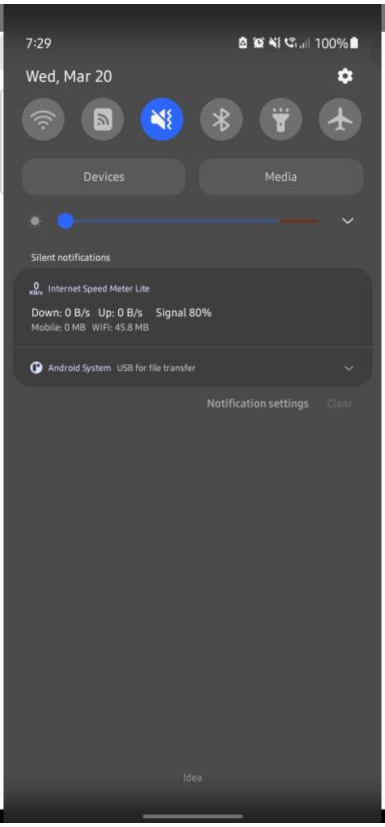
```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Output:



Network Status

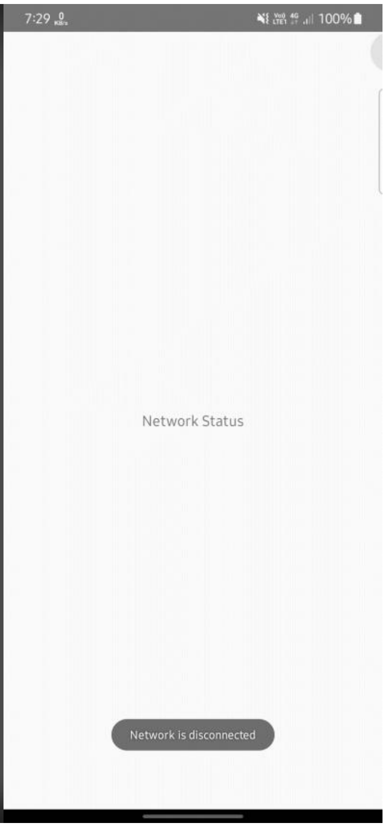
Network is connected



Silent notifications

Internet Speed Meter Lite
Down: 0 B/s Up: 0 B/s Signal 80%
Mobile: 0 MB WiFi: 45.8 MB

Android System USB for file transfer



Network Status

Network is disconnected

Practical No. 22

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/rl"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
</RelativeLayout>
```

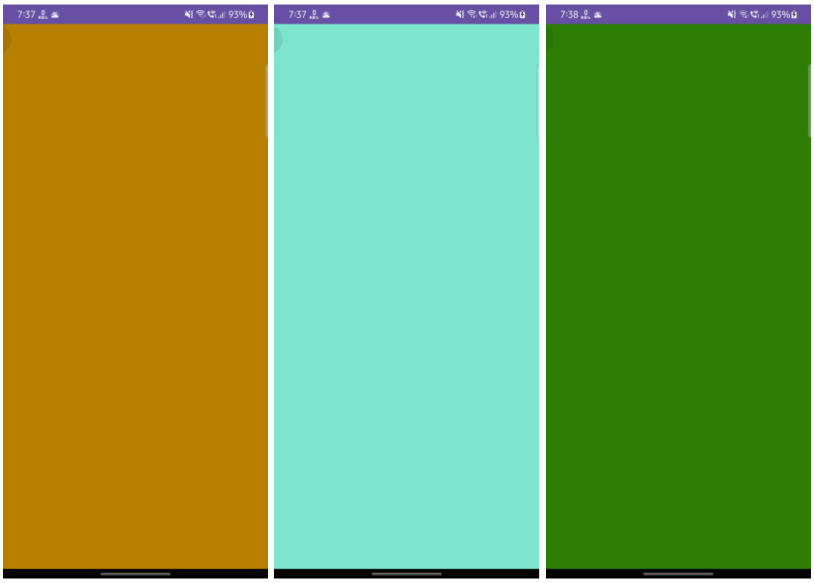
MainActivity.java

```
package com.example.sensors;
import android.app.Activity;
import android.content.Context;
import android.hardware.*;
import android.os.Bundle;
import android.view.*;
import android.widget.RelativeLayout;
import java.util.Random;

public class MainActivity extends Activity implements SensorEventListener {
    private SensorManager sm;
    private Sensor acc;
    private RelativeLayout rl;
    private float lx, ly, lz;
    private long lu;
    private static final int ST = 600;

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

        rl = findViewById(R.id.rl);
        sm = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
        acc = sm.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
        sm.registerListener(this, acc, SensorManager.SENSOR_DELAY_NORMAL);
        lu = System.currentTimeMillis();
    }

    @Override
    public void onSensorChanged(SensorEvent event) {
        Sensor s = event.sensor;

        if (s.getType() == Sensor.TYPE_ACCELEROMETER) {
            float x = event.values[0];
            float y = event.values[1];
            float z = event.values[2];

            long ct = System.currentTimeMillis();

            if ((ct - lu) > 100) {
                long dt = (ct - lu);
                lu = ct;

                float sp = Math.abs(x + y + z - lx - ly - lz) / dt * 10000;
                if (sp > ST) {
                    rl.setBackgroundColor(getRandomColor());
                }

                lx = x;
                ly = y;
                lz = z;
            }
        }
    }

    @Override
    public void onAccuracyChanged(Sensor sensor, int accuracy) {
    }
    private int getRandomColor() {
        Random rnd = new Random();
        return android.graphics.Color.argb(255, rnd.nextInt(256),
rnd.nextInt(256), rnd.nextInt(256));
    }
}
```

AndroidManifest.xml (add line inside <manifest> tag)

```
<uses-feature android:name="android.hardware.sensor.accelerometer"
android:required="true" />
```

Output:



2. Write a program to display the list of sensors supported by the mobile device.

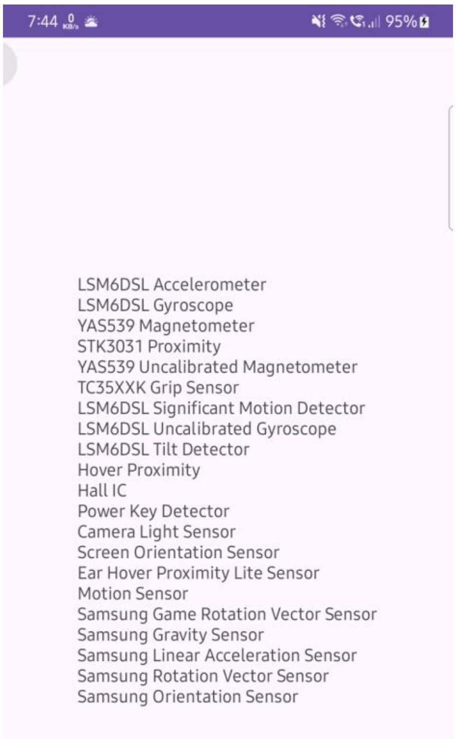
activity_main.xml

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:textSize = "20sp"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.sensors;
import android.app.Activity;
import android.hardware.Sensor;
import android.hardware.SensorManager;
import android.os.Bundle;
import android.widget.TextView;
import java.util.List;

public class MainActivity extends Activity {
    private SensorManager sm;
    private TextView tv;

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

        tv = findViewById(R.id.textView);
        sm = (SensorManager) getSystemService(SENSOR_SERVICE);

        List<Sensor> sensorList = sm.getSensorList(Sensor.TYPE_ALL);

        StringBuilder sensorText = new StringBuilder();
        for (Sensor currentSensor : sensorList ) {

sensorText.append(currentSensor.getName()).append(System.getProperty("line.separat
or"));
        }

        tv.setText(sensorText);
    }
}
```

Output:



```
LSM6DSL Accelerometer
LSM6DSL Gyroscope
YAS539 Magnetometer
STK3031 Proximity
YAS539 Uncalibrated Magnetometer
TC35XXK Grip Sensor
LSM6DSL Significant Motion Detector
LSM6DSL Uncalibrated Gyroscope
LSM6DSL Tilt Detector
Hover Proximity
Hall IC
Power Key Detector
Camera Light Sensor
Screen Orientation Sensor
Ear Hover Proximity Lite Sensor
Motion Sensor
Samsung Game Rotation Vector Sensor
Samsung Gravity Sensor
Samsung Linear Acceleration Sensor
Samsung Rotation Vector Sensor
Samsung Orientation Sensor
```

Practical No. 23

X. Exercise

1. Write a program to capture an image and display it using image view.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp">

    <!-- Button to open the camera -->
    <Button
        android:id="@+id/camera_button"
        android:layout_width="100dp"
        android:layout_height="50dp"
        android:layout_marginStart="150dp"
        android:text="Capture"
        android:layout_alignParentBottom="true"/>

    <!-- ImageView to display the captured image -->
    <ImageView
        android:id="@+id/click_image"
        android:layout_width="match_parent"
        android:layout_height="600dp"
        android:layout_marginBottom="10dp" />
</RelativeLayout>
```

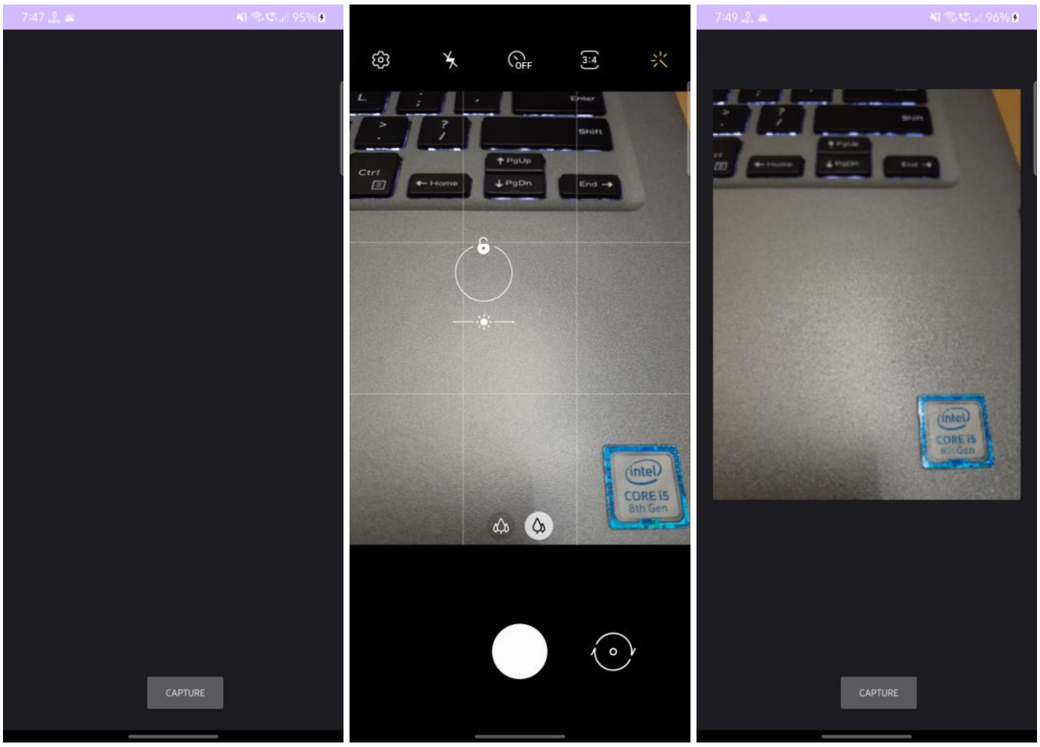
MainActivity.java

```
package com.example.camerabuilder;
import android.app.Activity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {

    private static final int CAMERA_REQUEST = 1;
    private ImageView iv;

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

        // Initialize views
        Button cameraButton = findViewById(R.id.camera_button);
        iv = findViewById(R.id.click_image);
        // Set click listener for the camera button
        cameraButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent takePictureIntent = new
Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                startActivityForResult(takePictureIntent,CAMERA_REQUEST);
            }
        });
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode, Intent data)
{
        if (requestCode == CAMERA_REQUEST) {
            Bitmap imageBitmap = (Bitmap) data.getExtras().get("data");
            iv.setImageBitmap(imageBitmap);
        }
    }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-feature android:name = "android.hardware.camera"/>
    <uses-permission android:name="android.permission.CAMERA" />
</manifest>
```

Output:



2. Write a program to record a video using various camera methods

activity_main.xml

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

    <Button
        android:id="@+id/recordButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Record Video"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

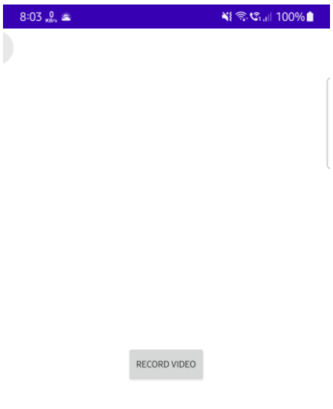
MainActivity.java

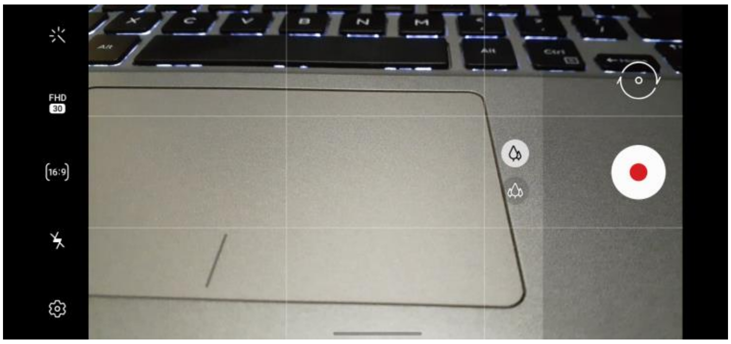
```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <Button
        android:id="@+id/recordButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Record Video"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

AndroidManifest.xml (add lines inside <manifest> tag)

```
<uses-feature android:name="android.hardware.camera" android:required="true" />
<uses-feature android:name="android.hardware.microphone" android:required="true"
/>
```

Output:





Practical No. 24

X. Exercise

1. Write a program to turn on, get visible, list devices, and turnoff Bluetooth

activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp">

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Turn On Bluetooth"/>

    <Button
        android:id="@+id/b2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Make Bluetooth Visible"
        android:layout_below="@+id/b1"/>

    <Button
        android:id="@+id/b3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="List Bluetooth Devices"
        android:layout_below="@+id/b2"/>

    <Button
        android:id="@+id/b4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Turn Off Bluetooth"
        android:layout_below="@+id/b3"/>
    <TextView
        android:id="@+id/tv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Paired Devices: "
        android:layout_below="@+id/b4"
        android:textSize = "15sp"/>

</RelativeLayout>
```

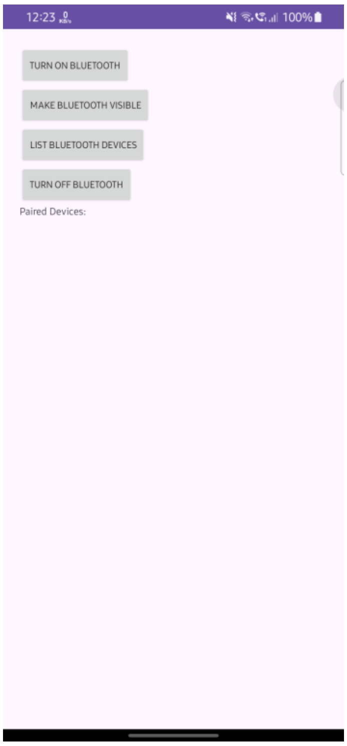
MainActivity.java

```
package com.example.bluetooth;
import android.annotation.SuppressLint;
import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.Intent;
import android.os.Bundle;
import android.view.*;
import android.widget.*;


import java.util.Set;

public class MainActivity extends Activity {
    private BluetoothAdapter ba;
    private Button b1, b2, b3, b4;
    private TextView tv;

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

        ba = BluetoothAdapter.getDefaultAdapter();

        b1 = findViewById(R.id.b1);
        b2 = findViewById(R.id.b2);
        b3 = findViewById(R.id.b3);
        b4 = findViewById(R.id.b4);
        tv = findViewById(R.id.tv);

        b1.setOnClickListener(new View.OnClickListener() {
            @SuppressLint("MissingPermission")
            @Override
            public void onClick(View v) {
                if (ba != null && !ba.isEnabled()) {
                    Intent i = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                    startActivityForResult(i, 1);
                }
            }
        });

        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            @SuppressLint("MissingPermission")
            public void onClick(View v) {
                if (ba != null && ba.isEnabled()) {
                    Intent i = new
Intent(BluetoothAdapter.ACTION_REQUEST_DISCOVERABLE);
                    i.putExtra(BluetoothAdapter.EXTRA_DISCOVERABLE_DURATION, 300);
                    startActivity(i);
                }
            }
        });

        b3.setOnClickListener(new View.OnClickListener() {
            @Override
            @SuppressLint("MissingPermission")
            public void onClick(View v) {
                if (ba != null && ba.isEnabled()) {
                    Set<BluetoothDevice> devices = ba.getBondedDevices();
                    for (BluetoothDevice device : devices) {
                        tv.append("\n"+device.getName() + " " +
device.getAddress() + "\n");
                    }
                }
            }
        });

        b4.setOnClickListener(new View.OnClickListener() {
            @Override
            @SuppressLint("MissingPermission")
            public void onClick(View v) {
                if (ba != null && ba.isEnabled()) {
                    ba.disable();
                }
            }
        });
    }
}
```

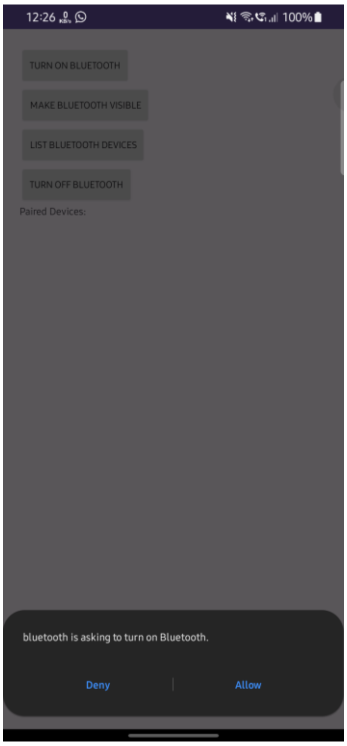
AndroidManifest.xml (add lines inside <manifest> tag)

```
<uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE" />
<uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
<uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
```

Output:

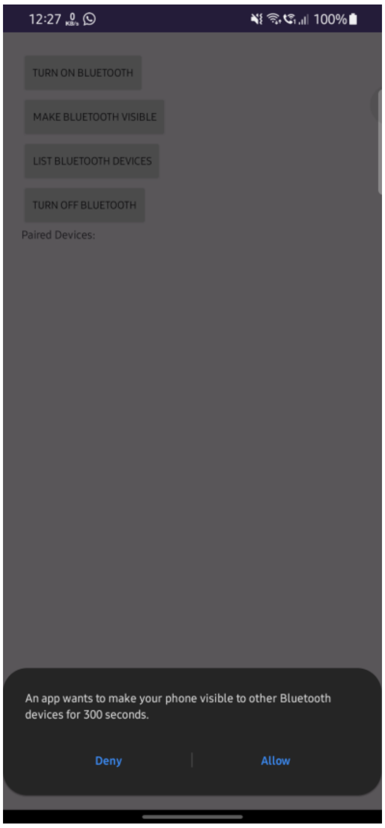


Paired Devices:



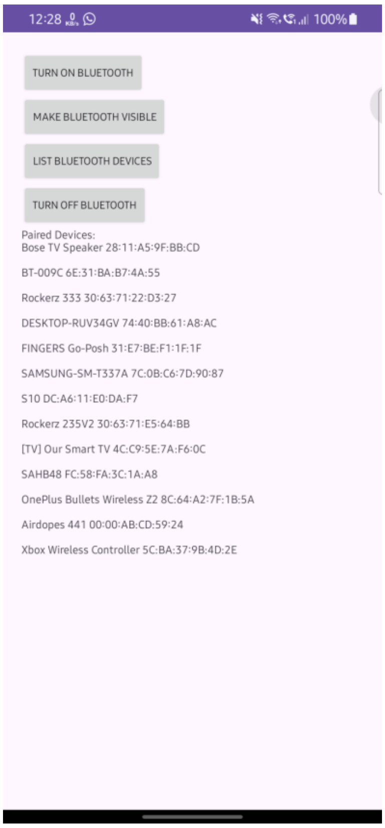
Paired Devices:

bluetooth is asking to turn on Bluetooth.



Paired Devices:

An app wants to make your phone visible to other Bluetooth devices for 300 seconds.



```
Paired Devices:
Bose TV Speaker 28:11:A5:9F:BB:CD

BT-009C 6E:31:BA:B7:4A:55

Rockerz 333 30:63:71:22:D3:27

DESKTOP-RUV34GV 74:40:BB:61:A8:AC

FINGERS Go-Posh 31:E7:BE:F1:1F:1F

SAMSUNG-SM-T337A 7C:0B:C6:7D:90:87

S10 DC:A6:11:E0:DA:F7

Rockerz 235V2 30:63:71:E5:64:BB

[TV] Our Smart TV 4C:C9:5E:7A:F6:0C

SAHB48 FC:58:FA:3C:1A:A8

OnePlus Bullets Wireless Z2 8C:64:A2:7F:1B:5A

Airdopes 441 00:00:AB:CD:59:24

Xbox Wireless Controller 5C:BA:37:9B:4D:2E
```

Practical No. 25

X. Exercise

1. Write a program to Rotate the image, Zoom in/out, Fade in/out using given GUI

activity_main.xml

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

    <ImageView
        android:id="@+id/img"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:src="@drawable/android"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.animation;

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

public class MainActivity extends AppCompatActivity  {
    ImageView img;

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

        img = findViewById(R.id.img);
    }

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

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();

        if (id == R.id.rotate) {
            animate(R.anim.rotate);
        } else if (id == R.id.zoom_in) {
            animate(R.anim.zoom_in);
        } else if (id == R.id.zoom_out) {
            animate(R.anim.zoom_out);
        } else if (id == R.id.fade_in) {
            animate(R.anim.fade_in);
        } else if (id == R.id.fade_out) {
            animate(R.anim.fade_out);
        } else {
            return super.onOptionsItemSelected(item);
        }

        return true;
    }

    private void animate(int animation) {
        Animation anim = AnimationUtils.loadAnimation(this, animation);
        img.startAnimation(anim);
    }
}
```

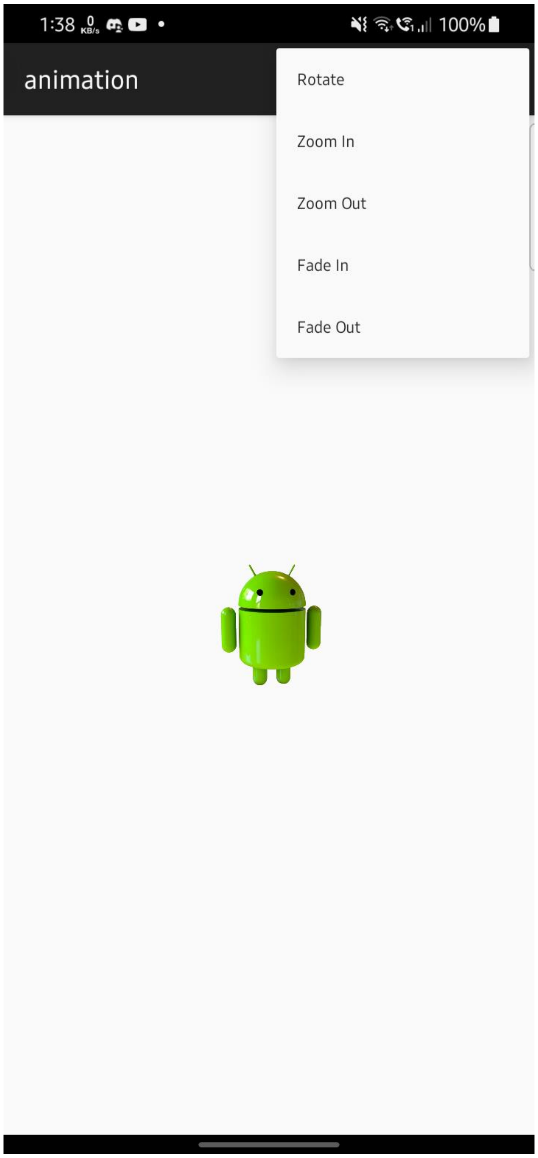
Resource Files

res/menu/menu_main.xml

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/rotate"
        android:title="Rotate"/>
    <item
        android:id="@+id/zoom_in"
        android:title="Zoom In"/>
    <item
        android:id="@+id/zoom_out"
        android:title="Zoom Out"/>
    <item
        android:id="@+id/fade_in"
        android:title="Fade In"/>
    <item
        android:id="@+id/fade_out"
        android:title="Fade Out"/>
</menu>
```

```
<!-- rotate.xml -->

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

    android:duration="2000"

    android:fromDegrees="0"

    android:pivotX="50%"

    android:pivotY="50%"
    android:toDegrees="360" />

<!-- zoom_in.xml -->
<scale xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="2000"
    android:fromXScale="1.0"
    android:fromYScale="1.0"
    android:pivotX="50%"
    android:pivotY="50%"
    android:toXScale="2.0"
    android:toYScale="2.0" />

<!-- zoom_out.xml -->
<scale xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="2000"
    android:fromXScale="2.0"
    android:fromYScale="2.0"
    android:pivotX="50%"
    android:pivotY="50%"
    android:toXScale="1.0"
    android:toYScale="1.0" />

<!-- fade_in.xml -->
<alpha xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="2000"
    android:fromAlpha="0.0"
    android:toAlpha="1.0" />

<!-- fade_out.xml -->
<alpha xmlns:android="http://schemas.android.com/apk/res/android"
    android:duration="2000"
    android:fromAlpha="1.0"
    android:toAlpha="0.0" />
```

Output:



Practical No. 26

1. Write a program to insert data in SQLite database using AsyncTask

activity_main.xml

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

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Data" />

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

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

</LinearLayout>
```

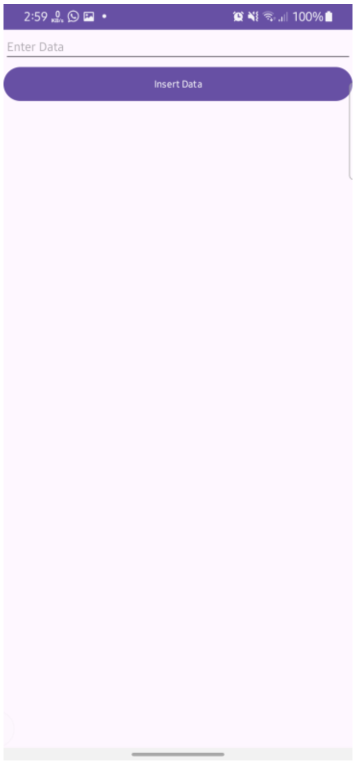
MainActivity.java

```
package com.example.myapplication;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText editText;
    Button button;
    TextView textView;
    DatabaseHelper databaseHelper;

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

        editText = findViewById(R.id.editText);
        button = findViewById(R.id.button);
        textView = findViewById(R.id.textView);

        databaseHelper = new DatabaseHelper(this);

        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                new InsertDataTask().execute(editText.getText().toString());
            }
        });
    }

    class InsertDataTask extends AsyncTask<String, Void, Boolean> {

        @Override
        protected Boolean doInBackground(String... params) {
            return databaseHelper.insertData(params[0]);
        }

        @Override
        protected void onPostExecute(Boolean result) {
            textView.setText(result ? "Data inserted successfully!" : "Data insertion
failed.");
        }
    }
}
```

DatabaseHelper.java

```
package com.example.myapplication;

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

public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "database.db";
    private static final String TABLE_NAME = "table";
    private static final String COL_1 = "DATA";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME + " (ID INTEGER PRIMARY KEY
AUTOINCREMENT, DATA TEXT)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    public boolean insertData(String data) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL_1, data);
        long result = db.insert(TABLE_NAME, null, contentValues);
        return result != -1;
    }
}
```

Output:



Practical No. 27

X. Exercise

1. Write a program to create the login form and display login successful/ Unsuccessful toast message.

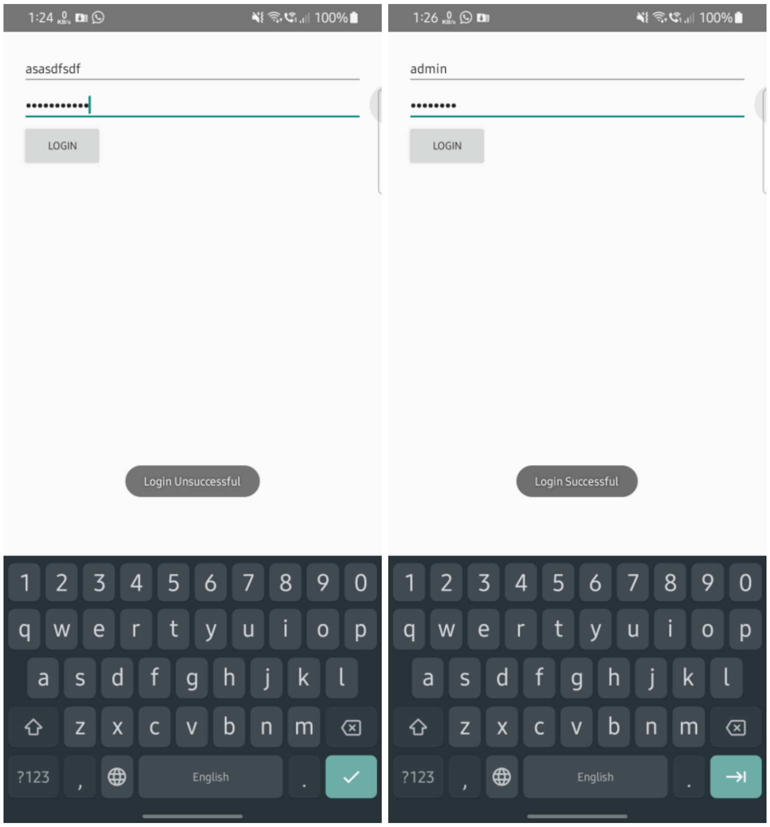
activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding = "20dp">

    <EditText
        android:id="@+id/et1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"/>

    <EditText
        android:id="@+id/et2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword"
        android:layout_below="@+id/et1"/>

    <Button
        android:id="@+id/b1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_below="@+id/et2"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.demo;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.*;

public class MainActivity extends Activity {
    private EditText et1, et2;
    private Button b1;

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

        et1 = findViewById(R.id.et1);
        et2 = findViewById(R.id.et2);
        b1 = findViewById(R.id.b1);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = et1.getText().toString();
                String password = et2.getText().toString();

                if (username.equals("admin") && password.equals("password")) {
                    Toast.makeText(getApplicationContext(), "Login Successful",
Toast.LENGTH_LONG).show();
                } else {
                    Toast.makeText(getApplicationContext(), "Login Unsuccessful",
Toast.LENGTH_LONG).show();
                }
            }
        });
    }
}
```

Output:



asasdfsdf

•••••••••••

Login Unsuccessful

admin

••••••••

Login Successful

Practical No. 28

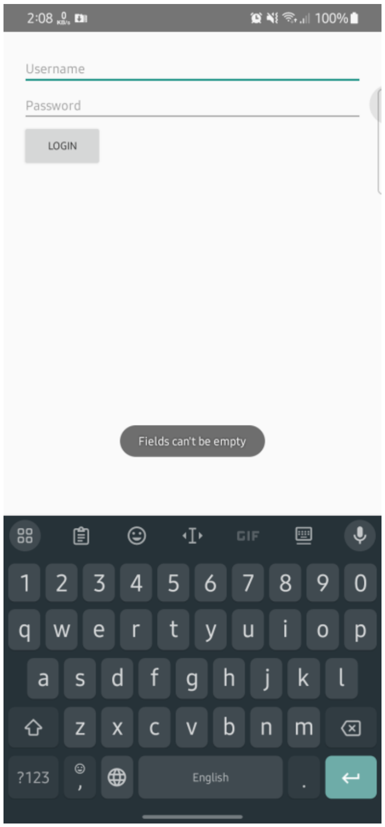
activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding = "20dp">

    <EditText
        android:id="@+id/u"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"/>

    <EditText
        android:id="@+id/p"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword"
        android:layout_below="@+id/u"/>

    <Button
        android:id="@+id/b"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_below="@+id/p"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.demo;
import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends Activity {
    private EditText u, p;
    private Button b;
    private int c = 0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        u = findViewById(R.id.u);
        p = findViewById(R.id.p);
        b = findViewById(R.id.b);

        b.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String un = u.getText().toString();
                String pw = p.getText().toString();

                if (un.isEmpty() || pw.isEmpty()) {
                    Toast.makeText(getApplicationContext(), "Fields can't be empty",
Toast.LENGTH_LONG).show();
                } else if (un.length() < 5 || pw.length() < 5) {
                    Toast.makeText(getApplicationContext(), "Username and password
must be at least 5 characters long", Toast.LENGTH_LONG).show();
                } else if (un.equals("admin") && pw.equals("password")) {
                    Toast.makeText(getApplicationContext(), "Login Successful",
Toast.LENGTH_LONG).show();
                    c = 0;
                } else {
                    c++;
                    if (c >= 3) {
                        Toast.makeText(getApplicationContext(), "Too many
unsuccessful attempts", Toast.LENGTH_LONG).show();
                    } else {
                        Toast.makeText(getApplicationContext(), "Login Unsuccessful",
Toast.LENGTH_LONG).show();
                    }
                }
            }
        });
    }
}
```

Output:



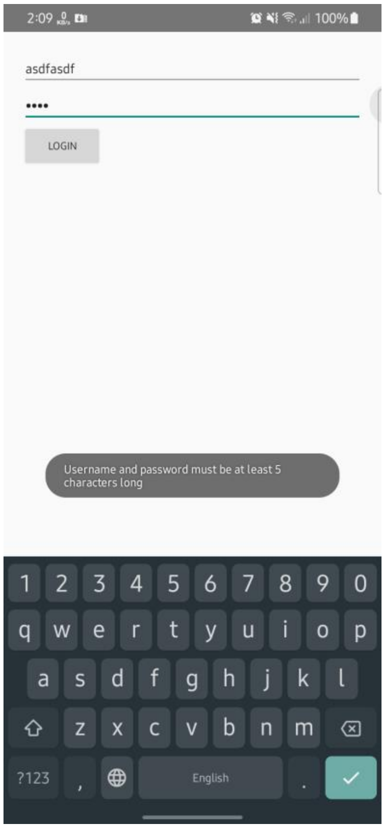
2:08 100%

Username

Password

LOGIN

Fields can't be empty



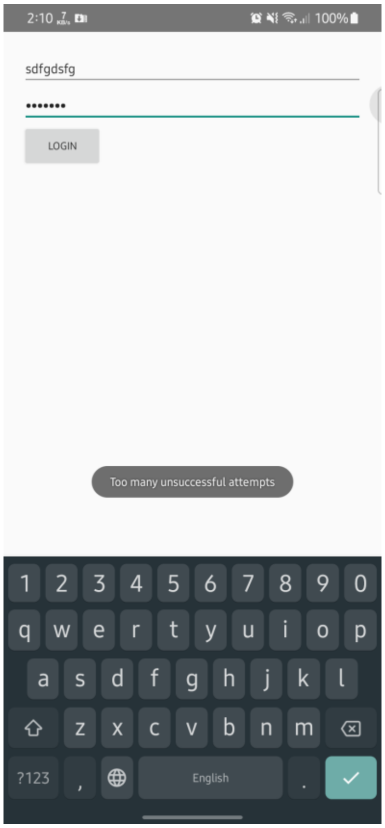
2:09 100%

asdfasdf

••••

LOGIN

Username and password must be at least 5 characters long



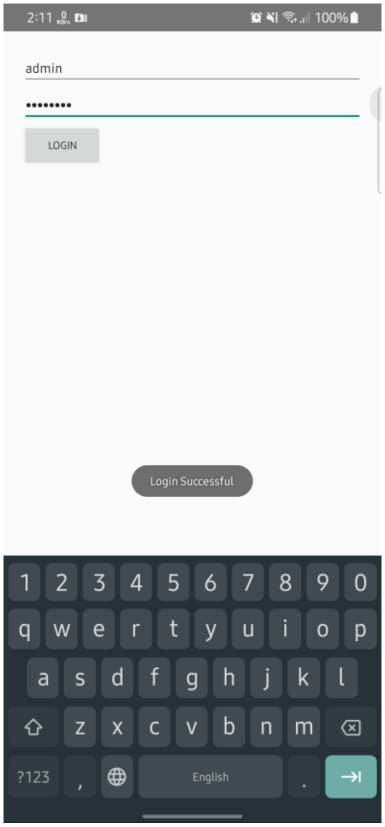
2:10 100%

sdfgdsfg

•••••••

LOGIN

Too many unsuccessful attempts



2:11 100%

admin

••••••••

LOGIN

Login Successful

Practical No. 29

X. Exercise

1. Write a program to send and receive SMS

AndroidManifest.xml

```
<uses-feature
    android:name="android.hardware.telephony"
    android:required="false" />

<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.READ_SMS"/>
```

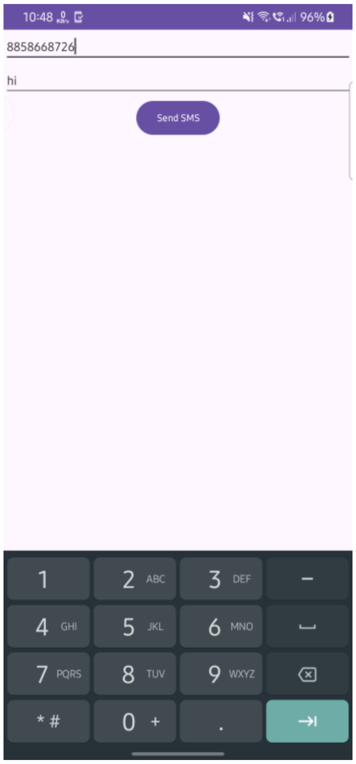
Then inside <application> tag

```
<activity android:name=".MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>

<receiver android:name=".SmsReceiver" android:enabled="true" android:exported="true"
    android:permission="android.permission.BROADCAST_SMS">
    <intent-filter>
        <action android:name="android.provider.Telephony.SMS_RECEIVED"/>
    </intent-filter>
</receiver>
```

activity_main.xml

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

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="phone"
        android:hint="Phone Number"/>

    <EditText
```

```
        android:id="@+id/editText2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textMultiLine"
        android:layout_below="@id/editText"
        android:hint="Message"/>

    <Button
        android:id="@+id/btnSendSMS"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send SMS"
        android:layout_below="@id/editText2"
        android:layout_centerHorizontal="true"/>

</RelativeLayout>
```



10:48 96%

8858668726

hi

MainActivity.java

```
package com.example.sms;



import android.app.PendingIntent;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    Button sendBtn;
    EditText txtphoneNo;
    EditText txtMessage;

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

        sendBtn = (Button) findViewById(R.id.btnSendSMS);
        txtphoneNo = (EditText) findViewById(R.id.editText);
        txtMessage = (EditText) findViewById(R.id.editText2);

        sendBtn.setOnClickListener(new View.OnClickListener() {
            public void onClick(View view) {
                sendSMSMessage();
            }
        });
    }

    protected void sendSMSMessage() {
        String phoneNo = txtphoneNo.getText().toString();
        String message = txtMessage.getText().toString();

        try {
            SmsManager smsManager = SmsManager.getDefault();
            smsManager.sendTextMessage(phoneNo, null, message, null, null);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

SmsReceiver.java

```
package com.example.sms;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.telephony.SmsMessage;
import android.widget.Toast;

public class SmsReceiver extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {
        Bundle bundle = intent.getExtras();
        SmsMessage[] msgs;
        String str = "";
        if (bundle != null) {
            Object[] pdus = (Object[]) bundle.get("pdus");
            msgs = new SmsMessage[pdus.length];
            for (int i=0; i<msgs.length; i++){
                msgs[i] = SmsMessage.createFromPdu((byte[])pdus[i]);
                str += "SMS from " + msgs[i].getOriginatingAddress();
                str += " :";
                str += msgs[i].getMessageBody().toString();
                str += "\n";
            }
            Toast.makeText(context, str, Toast.LENGTH_SHORT).show();
        }
    }
}
```

Practical No. 30

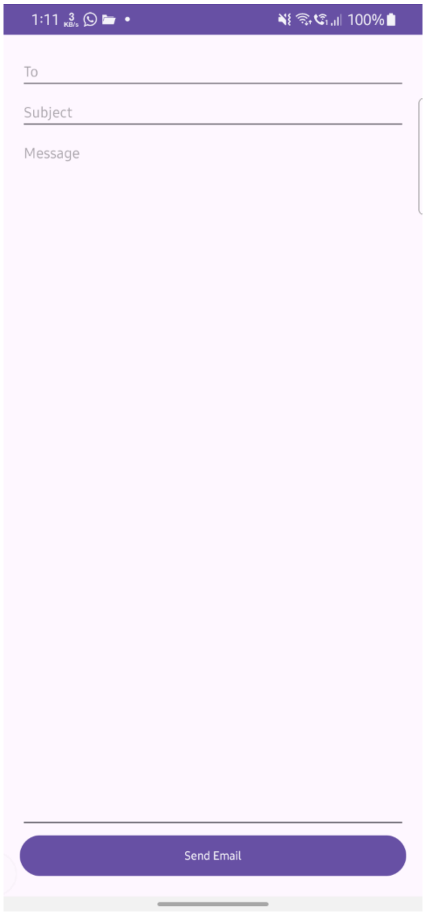
X. Exercise

1. Write a program to send email.

activity_main.xml

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

    <EditText
        android:id="@+id/editTextTo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="To" />

    <EditText
        android:id="@+id/editTextSubject"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Subject" />

    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:gravity="top"
        android:hint="Message" />

    <Button
        android:id="@+id/buttonSend"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Send Email" />

</LinearLayout>
```

Output:



MainActivity.java

```
package com.example.email;

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

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private EditText editTextTo;
    private EditText editTextSubject;
    private EditText editTextMessage;

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

        editTextTo = findViewById(R.id.editTextTo);
        editTextSubject = findViewById(R.id.editTextSubject);
        editTextMessage = findViewById(R.id.editTextMessage);
        Button buttonSend = findViewById(R.id.buttonSend);

        buttonSend.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String to = editTextTo.getText().toString();
                String subject = editTextSubject.getText().toString();
                String message = editTextMessage.getText().toString();

                Intent intent = new Intent(Intent.ACTION_SEND);
                intent.setData(Uri.parse("mailto:")); // only email apps should
handle this
                intent.putExtra(Intent.EXTRA_EMAIL, new String[]{to});
                intent.putExtra(Intent.EXTRA_SUBJECT, subject);
                intent.putExtra(Intent.EXTRA_TEXT, message);
                if (intent.resolveActivity(getPackageManager()) != null) {
                    startActivity(intent);
                }
            }
        });
    }
}
```

Practical No. 31

X. Exercise

1. Write a program to locate the user’s current location

```
package com.example.maps;


import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
public
class MapsActivity extends FragmentActivity implements OnMapReadyCallback,
    Locati onListener, GoogleApiClient.ConnectionCallbacks,
    GoogleApiClient.OnConnectionFailedListener {
 private
  GoogleMap mMap;
  Location mLastLocation;
  Marker mCurrLocationMarker;
  GoogleApiClient mGoogleApiClient;
  LocationRequest mLocationRequest;
  @Override protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);
    SupportMapFragment mapFragment =
        (SupportMapFragment)getSupportFragmentManager().findFragmentById(
            R.id.map);
    mapFragment.getMapAsync(this);
  }
  @Override public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    if (android.os.Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
      if (ContextCompat.checkSelfPermission(
              this, Manifest.permission.ACCESS_FINE_LOCATION) ==
          PackageManager.PERMISSION_GRANTED) {
        buildGoogleApiClient();
        mMap.setMyLocationEnabled(true);
      }
    } else {
      buildGoogleApiClient();
      mMap.setMyLocationEnabled(true);
    }
  }
 protected
  synchronized void buildGoogleApiClient() {
    mGoogleApiClient = new GoogleApiClient.Builder(this)
                           .addConnectionCallbacks(this)
                           .addOnConnectionFailedListener(this)
                           .addApi(LocationServices.API)
                           .build();
    mGoogleApiClient.connect();
  }
  @Override public void onConnected(Bundle bundle) {
    mLocationRequest = new LocationRequest();
    mLocationRequest.setInterval(1000);
    mLocationRequest.setFastestInterval(1000);
    mLocationRequest.setPriority(
        LocationRequest.PRIORITY_BALANCED_POWER_ACCUR ACY);
    if (ContextCompat.checkSelfPermission(
            this, Manifest.permission.ACCESS_FINE_LOCATION) ==
        PackageManager.PERMISSION_GRANTED) {
      LocationServices.FusedLocationApi.requestLocationUpdates(
          mGoogleApiClient, mLocati onRequest, this);
    }
  }
  @Override public void onConnectionSuspended(int i) {
  }
  @Override public void onLocationChanged(Location location) {
    mLastLocation = location;
    if (mCurrLocationMarker != null) {
      mCurrLocationMarker.remove();
    }
    // Place current location marker
    LatLng latLng = new LatLng(location.getLatitude(), location.getLongitude());
    MarkerOptions markerOptions = new MarkerOptions();
    markerOptions.position(latLng);
```

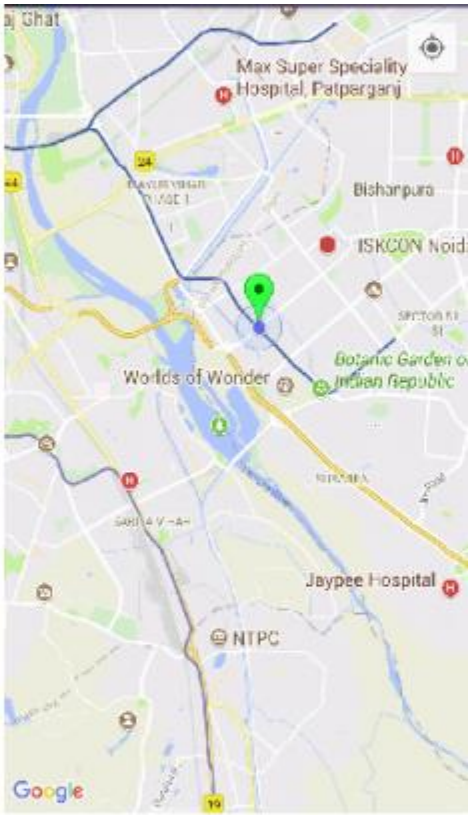
```
    markerOptions.title("Current Position");
    markerOptions.icon(BitmapDescriptorFactory.defaultMarker(
        BitmapDescriptorFactory.HUE _GREEN));
    mCurrLocationMarker = mMap.addMarker(markerOptions);
    // move map camera
    mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
    mMap.animateCamera(CameraUpdateFactory.zoomTo(11));
    // stop location updates
    if (mGoogleApiClient != null) {
      LocationServices.FusedLocationApi.removeLocationUpdates(mGoogleApiClient,
                                                              this);
    }
  }
  @Override public void onConnectionFailed(ConnectionResult connectionResult) {
  }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="example.com.mapexample">
    <!--
    The ACCESS_COARSE/FINE_LOCATION permissions are not required to use
    Google Maps Android API v2, but you must specify either coarse or fine
    location permissions for the 'MyLocation' functionality.
    -->
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-p
        ermission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <!--
        The API key for Google Maps-based APIs is defined as a string resource.
        (See the file "res/values/google_maps_api.xml").
        Note that the API key is linked to the encryption key used to sign the APK.
        You need a different API key for each encryption key, including the release
key that is use
       d to
```

```
        sign the APK for publishing.
        You can define the keys for the debug and release targets in src/debug/ and
src/release/.
        -->
        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />
        <activity
            android:name=".MapsActivity"
            android:label="@string/title_activity_maps">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Output:



Practical No. 32

X. Exercise

1. Write a program to draw a route between two locations

MainActivity.java

```
package com.example.mapdirectiondemo;
import android.graphics.Color;
import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.maps.model.PolylineOptions;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.IOException;
public
class MapsActivity extends FragmentActivity implements OnMapReadyCallback {
 private
  GoogleMap mMap;
  MarkerOptions origin, destination;
  @Override protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_maps);
    SupportMapFragment mapFragment =
        (SupportMapFragment)getSupportFragmentManager().findFragmentById(
            R.id.map);
    destination = new MarkerOptions()
                      .position(new LatLng(12.9304, 77.6784))
                      .title("Bellandur")
                      .snippet("destination");
    String url =
        getDirectionsUrl(origin.getPosition(), destination.getPosition());
    DownloadTask downloadTask = new DownloadTask();
    downloadTask.execute(url);
  }
  @Override public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    mMap.addMarker(origin);
    mMap.addMarker(destination);
    mMap.animateCamera(
        CameraUpdateFactory.newLatLngZoom(origin.getPosition(), 10));
  }
 private
  class DownloadTask extends AsyncTask<String, Void, String> {
    @Override protected String doInBackground(String... url) {
      String data = "";
      try {
        data = downloadUrl(url[0]);
      } catch (Exception e) {
        Log.d("Background Task", e.toString());
      }
      return data;
    }
```

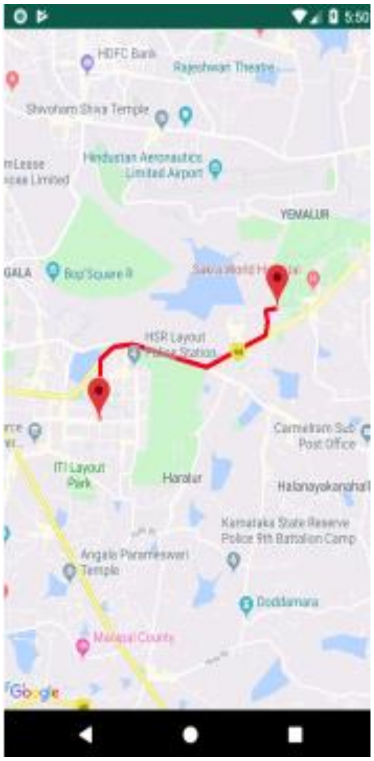
```
   @Override protected void onPostExecute(String result) {
     super.onPostExecute(result);
     ParserTask parserTask = new ParserTask();
     parserTask.execute(result);
   }
 } private class ParserTask extends
     AsyncTask<String, Integer, List<List<HashMap<String, String>>>> {
   @Override protected List<List<HashMap<String, String>>> doInBackground(
       String... jsonData) {
     JSONObject jObject;
     List<List<HashMap<String, String>>> routes = null;
     try {
       jObject = new JSONObject(jsonData[0]);
       DirectionsJSONParser parser = new DirectionsJSONParser();
       routes = parser.parse(jObject);
     } catch (Exception e) {
       e.printStackTrace();
     }
     return routes;
   }
   @Override protected void onPostExecute(
       List<List<HashMap<String, String>>> result) {
     ArrayList points = new ArrayList();
     PolylineOptions lineOptions = new PolylineOptions();
     for (int i = 0; i < result.size(); i++) {
       List<HashMap<String, String>> path = result.get(i);
       for (int j = 0; j < path.size(); j++) {
         HashMap<String, String> point = path.get(j);
         double lat = Double.parseDouble(point.get("lat"));
         double lng = Double.parseDouble(point.get("lng"));
         LatLng position = new LatLng(lat, lng);
         points.add(position);
       }
       lineOptions.addAll(points);
       lineOptions.width(12);
       lineOptions.color(Color.RED);
       lineOptions.geodesic(true);
     }
     if (points.size() != 0) mMap.addPolyline(lineOptions);
   }
 } private String getDirectionsUrl(LatLng origin, LatLng dest) {
   String str_origin = "origin=" + origin.latitude + "," + origin.longitude;
   String str_dest = "destination=" + dest.latitude + "," + dest.longitude;
   String mode = "mode=driving";
   String parameters = str_origin + "&" + str_dest + "&" + sensor + "&" + mode;
   String output = "json";
   String url = "https://maps.googleapis.com/maps/api/directions/" + output +
                "?" + parameters +
                "&key=" + "AIzaSyD_L8g3AcwXBKnEjhvLJwBXwI3L51LjQUU";
   return url;
 }
private
 String downloadUrl(String strUrl) throws IOException {
   String data = "";
   InputStream iStream = null;
   HttpURLConnection urlConnection = null;
```

```
    try {
      URL url = new URL(strUrl);
      urlConnection = (HttpURLConnection)url.openConnection();
      urlConnection.connect();
      iStream = urlConnection.getInputStream();
      BufferedReader br = new BufferedReader(new InputStreamReader(iStream));
      StringBuffer sb = new StringBuffer();
      String line = "";
      while ((line = br.readLine()) != null) {
        sb.append(line);
      }
      data = sb.toString();
      br.close();
    } catch (Exception e) {
      Log.d("Exception", e.toString());
    } finally {
      iStream.close();
      urlConnection.disconnect();
    }
    return data;
  }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.mapdirectiondemo">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".MapsActivity"
            android:label="@string/title_activity_maps">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <meta-data
            android:name="com.google.android.geo.API_KEY"
            android:value="@string/google_maps_key" />
    </application>
</manifest>
```

Output:


5:50
HDFC Bank
Rajeshwari Theatre
Shivoham Shiva Temple
Hindustan Aeronautics Limited Airport
YEMALUR
Bop'Square
HSR Layout Police Station
Carmelram Sub Post Office
ITI Layout Park
Haralur
Halanayakanahalli
Karnataka State Reserve Police 9th Battalion Camp
Angala Parameswari Temple
Doddamara
Google