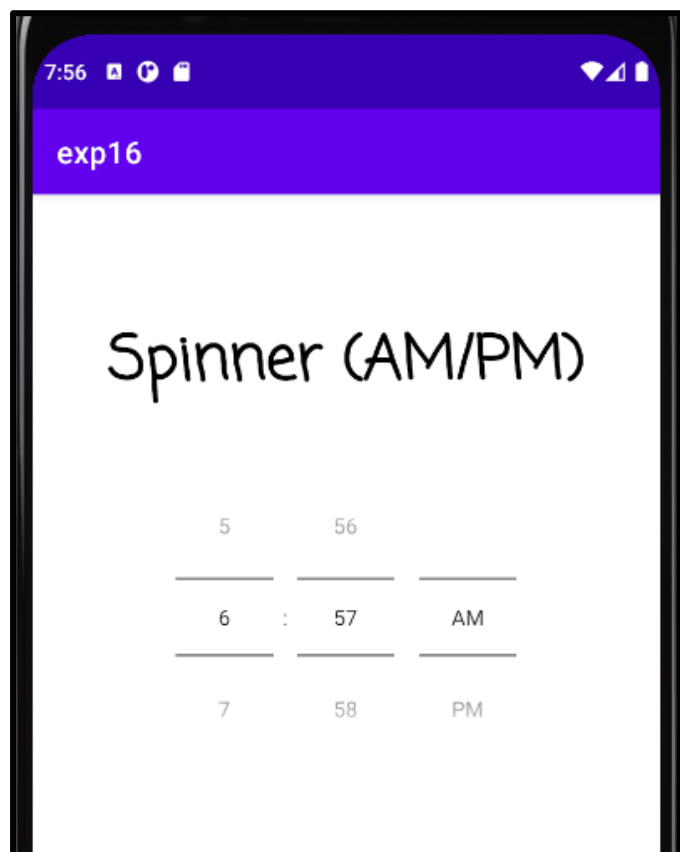# Practical 16

1. Write a Program to display the time picker in spinner mode.

Xml Code :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical">
    <TextView
        android:layout_marginTop="80dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Spinner (AM/PM)"
        android:textSize="40dp"
        android:textStyle="bold"
        android:fontFamily="casual"
        android:textColor="@color/black"
        android:layout_gravity="center"
        />
    <TimePicker
        android:id="@+id/timepik1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:timePickerMode="spinner"
        android:layout_gravity="center"
        android:layout_marginTop="30dp" />
</LinearLayout>
```

Output:

2. Write a Program to display time picker in 24 hours format in spinner mode.

Xml Code :

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical">
    <TextView
        android:layout_marginTop="80dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Spinner 24 Hours Format"
        android:textSize="30dp"
        android:textStyle="bold"
        android:fontFamily="casual"
        android:textColor="@color/black"
        android:layout_gravity="center"
        />
    <TimePicker
        android:id="@+id/t2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:timePickerMode="spinner"
        android:layout_gravity="center"
        android:layout_marginTop="30dp"/>
</LinearLayout>
```

Java Code :

```java
package com.example.exp16;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TimePicker;
public class MainActivity2 extends AppCompatActivity
{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        TimePicker tp1 = (TimePicker)
findViewById(R.id.t2);
        tp1.setIs24HourView(true);
    }
}
```

Output:

3. Write a Program to display the following output.Select and display date and time on click of "select date" and "select time" buttons respectively.

Xml Code :

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity3"
    >
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Date and Time picker"
        android:textSize="40dp"
        android:gravity="center"
        android:layout_marginTop="30dp"
        android:textColor="@color/black"
        android:id="@+id/txt"
        />
    <EditText
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:layout_below="@id/txt"
        android:layout_marginTop="50dp"
        android:layout_marginLeft="15dp"
        android:id="@+id/datetextview"
        />
    <EditText
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:layout_below="@id/datetextview"
        android:layout_marginTop="20dp"
        android:layout_marginLeft="15dp"
        android:id="@+id/timetextview"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Date"
        android:textSize="20dp"
        android:layout_toRightOf="@id/datetextview"
        android:layout_marginTop="135dp"
        android:layout_marginLeft="10dp"
        android:backgroundTint="@color/black"
```

```xml
        android:id="@+id/datebutton"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Time"
        android:textSize="20dp"
        android:layout_toRightOf="@id/timetextview"
        android:layout_marginTop="205dp"
        android:layout_marginLeft="10dp"
        android:backgroundTint="@color/black"
        android:layout_alignBottom="@id/timetextview"
        android:id="@+id/timebutton"
        />
</RelativeLayout>
```

Java Code :

```java
package com.example.exp16;
import androidx.appcompat.app.AppCompatActivity;
import android.app.DatePickerDialog;
import android.app.TimePickerDialog;
import android.os.Bundle;
import android.text.format.DateFormat;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.TimePicker;
import java.util.Calendar;
public class MainActivity3 extends AppCompatActivity {
    Button date,time;
    EditText dtext,ttext;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main3);
        date = (Button) findViewById(R.id.datebutton);
        time = (Button) findViewById(R.id.timebutton);
        dtext = (EditText) findViewById(R.id.datetextview);
        ttext = (EditText) findViewById(R.id.timetextview);
        date.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                dispdate();
            }
        });
        time.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                disptime();
            }
        });
    }
```

```
   public void dispdate(){
      Calendar calendar = Calendar.getInstance();
      int year = calendar.get(Calendar.YEAR);
      int month = calendar.get(Calendar.MONTH);
      int date = calendar.get(Calendar.DATE);

      DatePickerDialog datePickerDialog = new DatePickerDialog(this, new
DatePickerDialog.OnDateSetListener() {
          @Override
         public void onDateSet(DatePicker datePicker, int YEAR, int MONTH, int DATE) {
            Calendar calendar1 = Calendar.getInstance();
            calendar1.set(Calendar.YEAR,YEAR);
            calendar1.set(Calendar.MONTH,MONTH);
            calendar1.set(Calendar.DATE,DATE);
            CharSequence charSequence = DateFormat.format("dd/MM/yyyy",calendar1);
            dtext.setText(charSequence);
          }
      },year,month,date);
      datePickerDialog.show();
   }
   public void disptime(){
      Calendar calendar = Calendar.getInstance();
      int hour = calendar.get(Calendar.HOUR);
      int minute = calendar.get(Calendar.MINUTE);
      boolean is24hour = DateFormat.is24HourFormat(this);
      TimePickerDialog timePickerDialog = new TimePickerDialog(this, new
TimePickerDialog.OnTimeSetListener() {
          @Override
         public void onTimeSet(TimePicker timePicker, int H, int M) {
            Calendar calendar1 = Calendar.getInstance();
            calendar1.set(Calendar.HOUR,H);
            calendar1.set(Calendar.MINUTE,M);
            CharSequence charSequence = DateFormat.format("h : mm a",calendar1);
            ttext.setText(charSequence);
          }
      }, hour, minute, is24hour);
      timePickerDialog.show();
   }
}
```

Output: