

**Name :- Rohan Kamble**

**Class :- S. Y. B. Sc. CS**

**PRN NO :- 2020420004**

**Subject Code :- USCS407**

## **Practical:- 1**

**Aim :- Install Android Studio and Run Hello World Program.**

**XML Code :-**

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

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

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

**Source Code :-**

```
package com.example.college_journal;

import androidx.appcompat.app.AppCompatActivity;

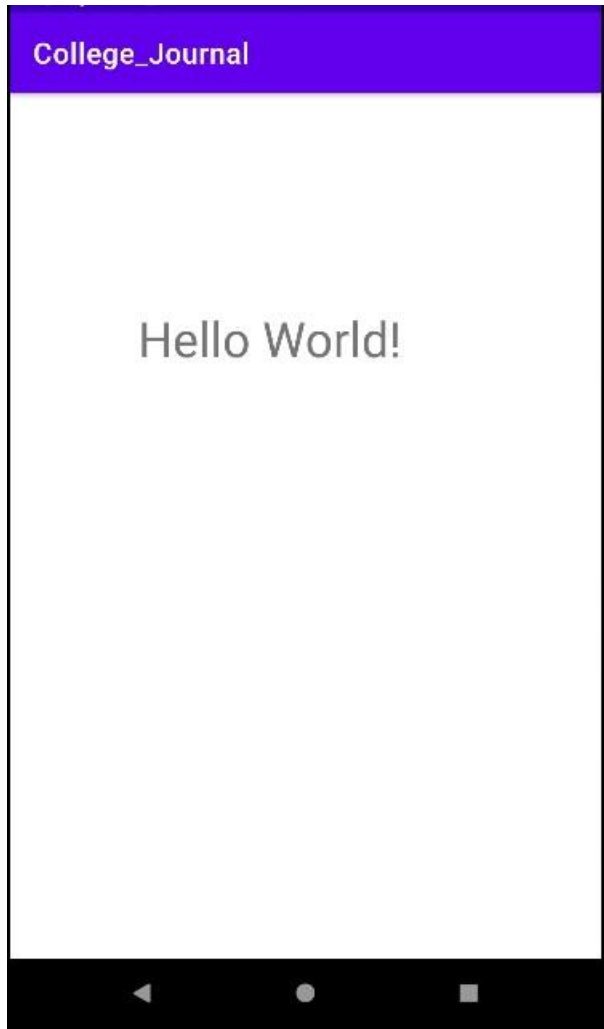
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

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

**Output :-**



**Practical:- 2**

**Aim :- Create an android app with Interactive User Interface using Layouts.**

**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:orientation="vertical" >      <TextView
android:id="@+id/text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Hello, I am a TextView" />
    <Button android:id="@+id/button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Hello, I am a Button" />
</LinearLayout>
```

**Source Code :-**

```
package com.example.college_journal;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

**Output :-**



## Practical:- 3

**Aim :- Create an android app that demonstrates working with TextView Elements.**

**XML Code :-**

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

    <EditText
        android:id="@+id/editTextTextPersonName2"
        android:layout_width="250dp"
        android:layout_height="48dp"
        android:layout_marginTop="28dp"        android:ems="10"
        android:inputType="textPersonName"
        android:text="Enter The First Name"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.173"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextTextPersonName3"
        android:layout_width="246dp"
        android:layout_height="53dp"
        android:layout_marginTop="56dp"        android:ems="10"
        android:inputType="textPersonName"
        android:text="Enter The Middle Name"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.193"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName2" />

    <EditText
        android:id="@+id/editTextTextPersonName4"
        android:layout_width="249dp"
        android:layout_height="48dp"
        android:layout_marginTop="20dp"        android:ems="10"
        android:inputType="textPersonName"
        android:text="Enter The Last Name"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.197"
        app:layout_constraintStart_toStartOf="parent">
```

```
app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName3"
app:layout_constraintVertical_bias="0.068" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

### Source Code :-

```
package com.example.college_journal;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

### Output :-



## College\_Journal

Enter The First Name

Enter The Middle Name

Enter The Last Name



# Practical:- 4

**Aim :- Create an android app that demonstrates Activity Lifecycle and Instance State.**

**XML Code :-**

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

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

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

**Source Code :-**

```
package com.example.college_journal;

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

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Toast toast = Toast.makeText(getApplicationContext(), "onCreate
        Called", Toast.LENGTH_LONG).show();
    }
}
```



```
    protected void onStart() {
super.onStart();
    Toast toast = Toast.makeText(getApplicationContext(), "onStart Called",
Toast.LENGTH_LONG).show();
    }

    @Override protected
    void onRestart() {
super.onRestart();
    Toast toast = Toast.makeText(getApplicationContext(), "onRestart
Called", Toast.LENGTH_LONG).show();
    }

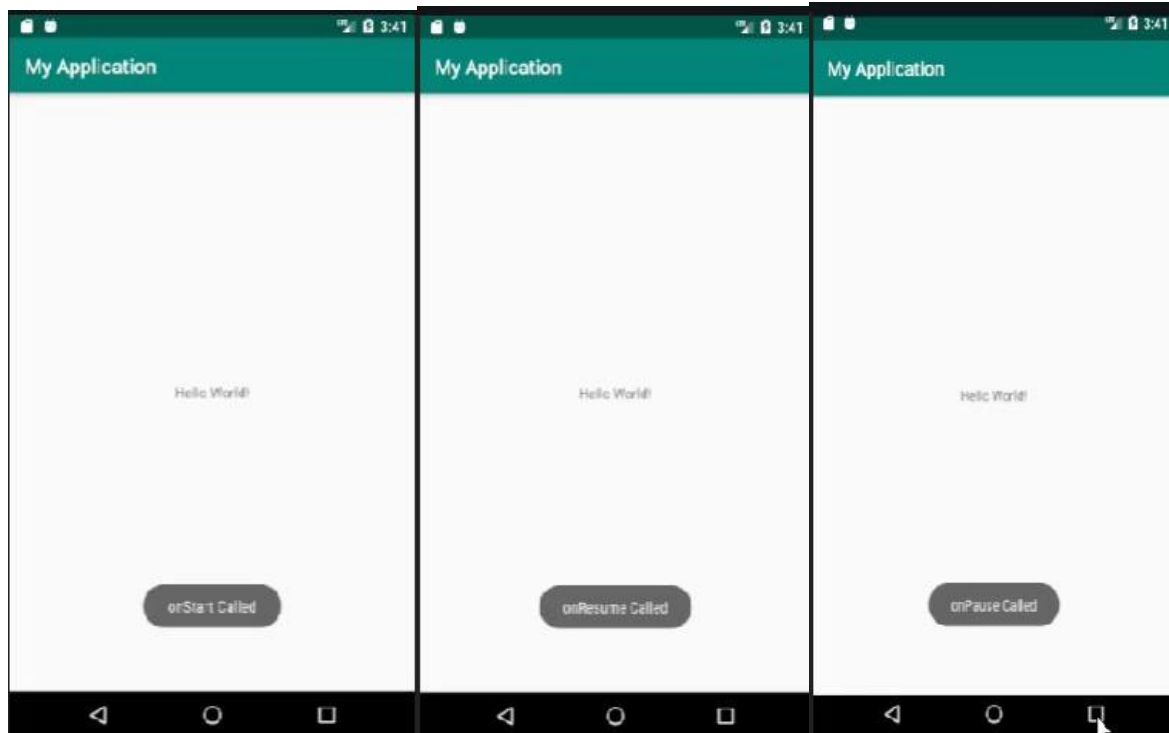
    protected void onPause() {
super.onPause();
    Toast toast = Toast.makeText(getApplicationContext(), "onPause
Called", Toast.LENGTH_LONG).show();
    }

    protected void onResume() {
super.onResume();
    Toast toast = Toast.makeText(getApplicationContext(), "onResume
Called", Toast.LENGTH_LONG).show();
    }

    protected void onStop() {
super.onStop();
    Toast toast = Toast.makeText(getApplicationContext(), "onStop Called",
Toast.LENGTH_LONG).show();
    }

    protected void onDestroy() {
super.onDestroy();
    Toast toast = Toast.makeText(getApplicationContext(), "onDestroy
Called", Toast.LENGTH_LONG).show();
    }
}
```

## Output :-



## **Practical:- 5**

**Aim :- Create an android app that demonstrates the use of Keyboards, Input Controls, Alerts and Pickers.**

### **1) Keyboard :- XML Code :-**

```
<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="com.example.android.keyboardsamples.MainActivity">

<Button android:id="@+id/button_main"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentBottom="true"
" android:layout_alignParentRight="true"
android:onClick="showText"
android:text="@string/show" />

<EditText
android:id="@+id/editText_main" android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_alignParentBottom="true"
android:inputType="textPassword"
android:layout_toLeftOf="@id/button_main"
android:hint="@string/enter" />

</RelativeLayout>
```

**Source Code :-**

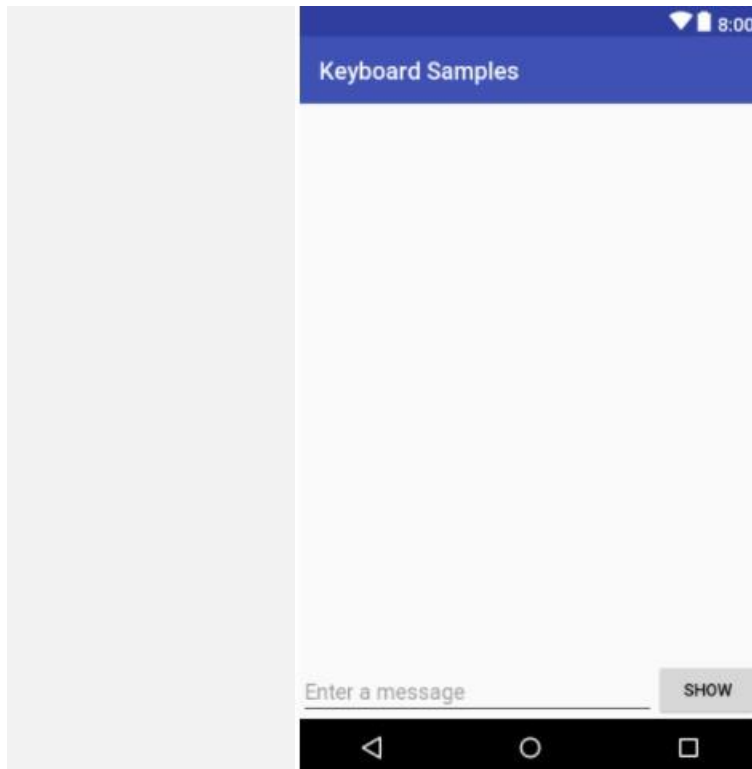
```
package com.example.android.keyboardsamples;
import
android.support.v7.app.AppCompatActivity;
import android.os.Bundle; import
android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    public void showText(View view) {
        EditText editText = (EditText) findViewById(R.id.editText_main); if
        (editText != null) {
            String showString = editText.getText().toString();
            Toast.makeText(this, showString, Toast.LENGTH_SHORT).show();
        }
    }
}
```

**Output :-**



## 2) Input Controls :- XML Code :-

```
<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="com.example.android.phonenumberspinner.MainActivity">

<LinearLayout android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="horizontal">

<EditText
android:id="@+id/editText_main"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:inputType="phone"
android:hint="@string/hint_phonenumber" />
```

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

```
<Button android:id="@+id/button_main"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:onClick="showText"  
    android:text="@string/show_button" />
```

```
</LinearLayout>  
<LinearLayout  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal"  
    android:layout_alignParentBottom="true">
```

```
<TextView  
    android:id="@+id/title_phonelabel"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/phonenummer_label"/>
```

```
<TextView  
    android:id="@+id/text_phonelabel"  
    android:layout_width="wrap_content" android:layout_height="wrap_content"  
    android:text="@string/nothing_entered"/>  
</LinearLayout>
```

```
</RelativeLayout>
```

```
<resources>  
    <string name="app_name">Phone Number Spinner</string>  
    <string name="show">Show</string>  
    <string name="enter">Enter a message</string>  
    <string name="hint_phonenumber">Enter phone number</string>  
    <string name="show_button">Show</string>  
    <string name="phonenummer_label">Phone Number:</string>  
    <string name="nothing_entered">Nothing entered.</string>  
    <string name="nothing_selected">onNothingSelected:</string>  
    <string-array name="labels_array">  
        <item>Home</item>  
        <item>Work</item>  
        <item>Mobile</item>
```

```
<item>Other</item>
</string-array>
</resources>
```

## Source Code :-

```
package com.example.android.phonenumberspinner;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle; import android.util.Log;
import android.view.View; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
android.widget.AdapterView; import
import android.widget.AdapterView;

public class MainActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener {

    private static final String TAG = MainActivity.class.getSimpleName();
    private String mSpinnerLabel = "";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Spinner spinner = (Spinner) findViewById(R.id.label_spinner); if
        (spinner != null) {
            spinner.setOnItemSelectedListener(this);
        }
        // Create ArrayAdapter using the string array and default spinner layout.
        ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
            R.array.labels_array, android.R.layout.simple_spinner_item); // Specify the
            layout to use when the list of choices appears.
            adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
            item);
            // Apply the adapter to the spinner.
            if (spinner != null) { spinner.setAdapter(adapter);
            }

        }

        public void showText(View view) {
```

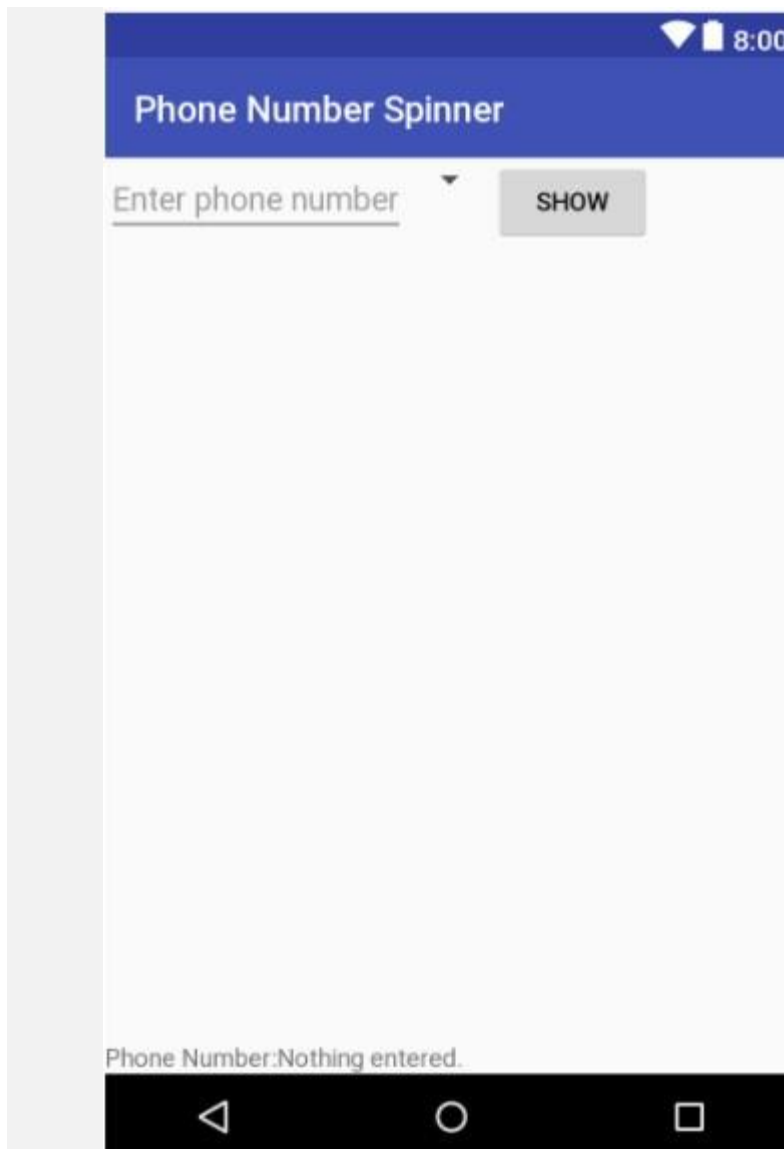
```
EditText editText = (EditText) findViewById(R.id.editText_main); if  
(editText != null) {  
String showString = (editText.getText().toString() + " - " + mSpinnerLabel);  
Toast.makeText(this, showString, Toast.LENGTH_SHORT).show(); TextView  
textView = (TextView)findViewById(R.id.text_phonelabel);  
textView.setText(showString);  
}  
}
```

```
@Override  
public void onItemSelected(AdapterView<?> adapterView, View view, int i, long l)  
{ mSpinnerLabel = adapterView.getItemAtPosition(i).toString(); showText(view);  
}
```

```
@Override  
public void onNothingSelected(AdapterView<?> adapterView) {  
Log.d(TAG, getString(R.string.nothing_selected));  
}  
}
```

**Output :-**





### 3) Alert :- XML Code :-

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

    <TextView
        android:id="@+id/top_message"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/tap_test" />
    <Button
```

```

        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/top_message"
        android:layout_marginTop="@dimen/button_top_margin"
        android:text="@string/alert_button"
        android:onClick="onClickShowAlert"/>

</RelativeLayout>

```

## Source Code :-

```

package com.example.android.alertsample;

import android.content.DialogInterface; import
android.support.v7.app.AlertDialog; import
android.support.v7.app.AppCompatActivity;
import android.os.Bundle; import
android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    public void onClickShowAlert(View view) {
        AlertDialog.Builder myAlertBuilder = new AlertDialog.Builder(MainActivity.this); //
        Set the dialog title.
        myAlertBuilder.setTitle(R.string.alert_title); //
        Set the dialog message.
        myAlertBuilder.setMessage(R.string.alert_message); //
        Add the buttons.
        myAlertBuilder.setPositiveButton(R.string.ok, new DialogInterface.OnClickListener()
        {
            public void onClick(DialogInterface dialog, int which) { //
                User clicked OK button.
                Toast.makeText(getApplicationContext(), R.string.pressed_ok,
                Toast.LENGTH_SHORT).show();
            }
        });
        myAlertBuilder.setNegativeButton(R.string.cancel, new
        DialogInterface.OnClickListener() { public void

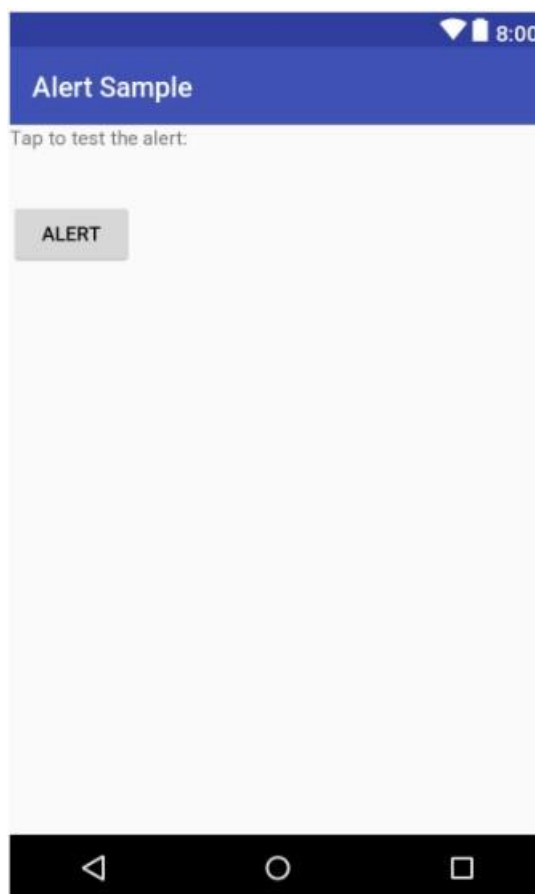
```

```

onClick(DialogInterface dialog, int which) { // User
cancelled the dialog.
Toast.makeText(getApplicationContext(), R.string.pressed_cancel,
Toast.LENGTH_SHORT).show();
}
});
// Create and show the AlertDialog.
myAlertBuilder.show();
}
}

```

### Output :-



### 4) Pickers :- XML Code :-

```

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

```

```

<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textSize="@dimen/text_size"
android:text="@string/choose_datetime"/>

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

<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:id="@+id/button_date"
android:text="@string/date_button"
android:onClick="showDatePickerDialog"/>

<Button android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:id="@+id/button_time"
android:layout_alignBottom="@id/button_date"
" android:layout_toRightOf="@id/button_date"
android:text="@string/time_button"
android:onClick="showTimePickerDialog"/>

</RelativeLayout>
</LinearLayout>

```

## Source Code:-

```

package com.example.android.datetimepickers;

import android.support.v4.app.DialogFragment;
import
android.support.v7.app.AppCompatActivity;
import android.os.Bundle; import
android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    public void showDatePickerDialog(View view) {

```

```

DialogFragment newFragment = new DatePickerFragment();
newFragment.show(getSupportFragmentManager(),
getString(R.string.date_picker));
}

public void showTimePickerDialog(View view) {
DialogFragment newFragment = new TimePickerFragment();
newFragment.show(getSupportFragmentManager(),
getString(R.string.time_picker));
}

public void processDatePickerResult(int year, int month, int day) {
String month_string = Integer.toString(month+1);
String day_string = Integer.toString(day);
String year_string = Integer.toString(year); String
dateMessage = (month_string + "/" +
day_string + "/" + year_string);
Toast.makeText(this, getString(R.string.date) + dateMessage,
Toast.LENGTH_SHORT).show();
}

public void processTimePickerResult(int hourOfDay, int minute) {
String hour_string = Integer.toString(hourOfDay);
String minute_string = Integer.toString(minute);
String timeMessage = (hour_string + ":" + minute_string);
Toast.makeText(this, getString(R.string.time) + timeMessage,
Toast.LENGTH_SHORT).show();
}
}

```

DatePickerFragment.java

```

package com.example.android.datetimepickers;

import
android.app.DatePickerDialog;
import android.app.Dialog; import
android.os.Bundle;
import android.support.annotation.NonNull;
import
android.support.v4.app.DialogFragment;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater; import
android.view.View; import
android.view.ViewGroup; import
android.widget.DatePicker;
import android.widget.TextView;

```

```

import java.util.Calendar;

/**
 * A simple {@link Fragment} subclass.
 */
public class DatePickerFragment extends DialogFragment implements
    DatePickerDialog.OnDateSetListener {

    @NonNull @Override public Dialog
    onCreateDialog(Bundle savedInstanceState) { // Use the
        current date as the default date in the picker. final
        Calendar c = Calendar.getInstance(); int year =
        c.get(Calendar.YEAR); int month =
        c.get(Calendar.MONTH);
        int day = c.get(Calendar.DAY_OF_MONTH);

        // Create a new instance of DatePickerDialog and return it.
        return new DatePickerDialog(getActivity(), this, year, month, day);
    }

    @Override public void onDateSet(DatePicker view, int year, int
        month, int day) { // Do something with the date chosen by the user.
        // Set the activity to the Main Activity.
        MainActivity activity = (MainActivity) getActivity(); // Invoke
        Main Activity's processDatePickerResult() method.
        activity.processDatePickerResult(year, month, day);
    }
}

```

TimePickerFragment.java

```

package com.example.android.datetimepickers;

import android.app.Dialog; import
    android.os.Bundle; import
    android.support.annotation.NonNull; import
    android.support.v4.app.Fragment; import
    android.text.format.DateFormat; import
    android.support.v4.app.DialogFragment;
    import android.app.TimePickerDialog; import
    android.widget.TimePicker; import
    java.util.Calendar;

/**
 * A simple {@link Fragment} subclass.
 */
public class TimePickerFragment extends DialogFragment
    implements TimePickerDialog.OnTimeSetListener {

```

```

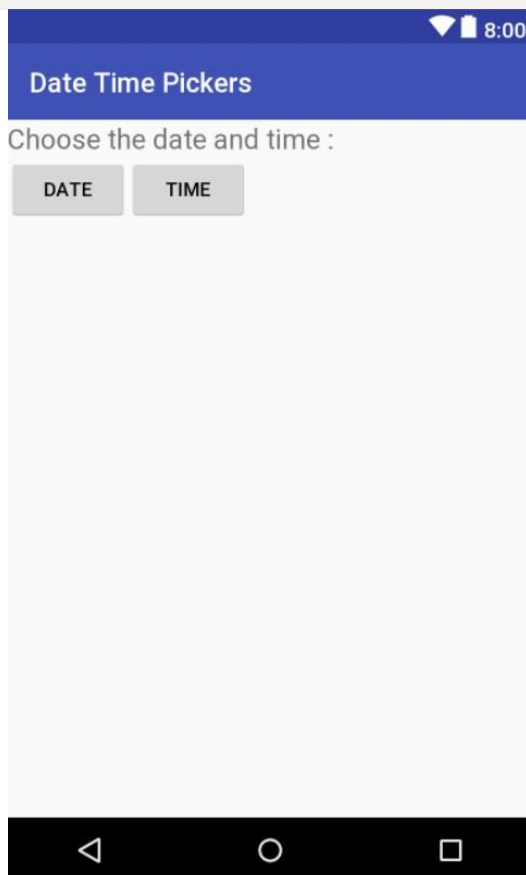
@NonNull @Override public Dialog
onCreateDialog(Bundle savedInstanceState) { // Use the
current time as the default values for the picker.
final Calendar c = Calendar.getInstance(); int
hour = c.get(Calendar.HOUR_OF_DAY); int
minute = c.get(Calendar.MINUTE);

// Create a new instance of TimePickerDialog and return it. return
new TimePickerDialog(getActivity(), this, hour, minute,
DateFormat.is24HourFormat(getActivity()));
}

public void onTimeSet(TimePicker view, int hourOfDay, int minute) {
// Do something with the time chosen by the user.
// Set the activity to the Main Activity.
MainActivity activity = (MainActivity) getActivity();
// Invoke Main Activity's processTimePickerResult() method.
activity.processTimePickerResult(hourOfDay, minute);
}
}

```

## Output :-



# Practical - 6

**Aim :- Create an android app that demonstrates the use of an Options Menu.**

**XML Code :-**

**Main\_activity.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="example.javatpoint.com.optionmenu.MainActivity">

    <android.support.design.widget.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/AppTheme.AppBarOverlay">

        <android.support.v7.widget.Toolbar
            android:id="@+id/toolbar"            android:layout_width="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:popupTheme="@style/AppTheme.PopupOverlay" />

    </android.support.design.widget.AppBarLayout>

    <include layout="@layout/content_main" />

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



### context\_main.xml:

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

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.a
ndroid.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"

    tools:context="example.javatpoint.com.optionmenu.MainActivity"

    tools:showIn="@layout/activity_main">
```

### <TextView

```
    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:text="Hello World!"

    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

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

### Menu\_main.xml:

```
    <menu 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"
        tools:context="example.javatpoint.com.optionmenu.MainActivity">

        <item    android:id="@+id/item1"
        android:title="Item 1"/>        <item
        android:id="@+id/item2"
        android:title="Item 2"/>        <item
        android:id="@+id/item3"
```

```

        android:title="Item 3"
        app:showAsAction="withText"/>
    </menu>

```

## Source Code :-

```

package example.javatpoint.com.optionmenu;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.Menu; import
android.view.MenuItem;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

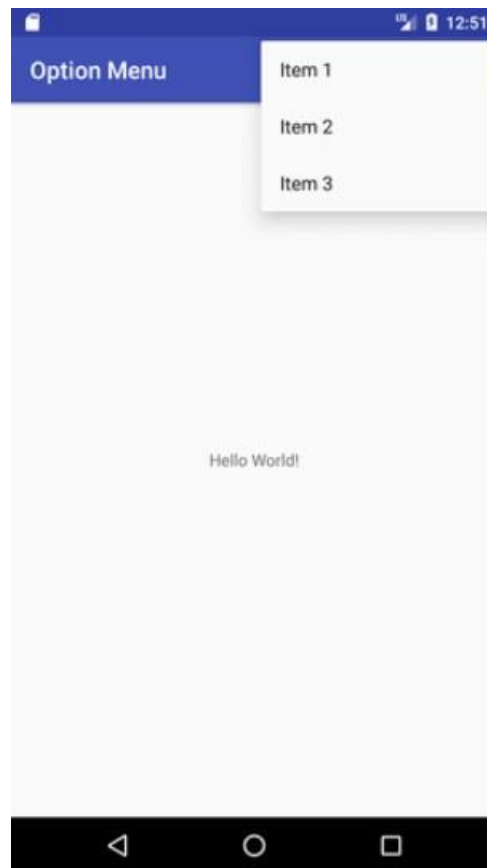
    @Override    public boolean
    onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        switch (id){
        case R.id.item1:
            Toast.makeText(getApplicationContext(),"Item 1 Selected",Toast.LENGTH_LONG).s
            how();
            return true;
        case R.id.item2:
            Toast.makeText(getApplicationContext(),"Item 2 Selected",Toast.LENGTH_LONG).s
            how();
            return true;
        case R.id.item3:
            Toast.makeText(getApplicationContext(),"Item 3 Selected",Toast.LENGTH_LONG).s
            how();

```

```
        return true;
    default:
        return super.onOptionsItemSelected(item);
    }
}
```

**Output :-**



## Practical:- 7

**Aim :- Create an android a-pp that demonstrate Screen Navigation Using the App Bar and Tabs. XML Code :-**

**Activity\_main.xml:**

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

    <com.google.android.material.tabs.TabLayout
        android:layout_width="match_parent"
        android:layout_height="?actionBarSize"
        android:id="@+id/tab_layout"          android:background="@color/white"
        app:tabInlineLabel="true"
        app:tabTextColor="@color/teal_700"
        app:tabIndicatorColor="@color/teal_700"
        app:tabSelectedTextColor="@color/teal_700"
        app:tabTextAppearance="@style/TextAppearance.AppCompat.Small"/>

    <androidx.viewpager.widget.ViewPager
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/view_pager"/>

</LinearLayout>
```

**Fragment\_main.xml:**

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

<FrameLayout
```

```

        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">

<TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/text_view"
        android:textSize="32sp"
        android:textStyle="bold"
        android:gravity="center"/>

</FrameLayout>

```

## Source Code: -

```

package com.example.tablayoutwithicon;

import androidx.annotation.ContentView; import
androidx.annotation.NonNull; import
androidx.annotation.Nullable; import
androidx.appcompat.app.AppCompatActivity; import
androidx.core.content.ContextCompat; import
androidx.fragment.app.Fragment; import
androidx.fragment.app.FragmentManager; import
androidx.fragment.app.FragmentManager; import
import androidx.viewpager.widget.ViewPager;

import
android.graphics.drawable.Drawable;
import android.os.Bundle; import

```

```

        android.text.Spannable; import
        android.text.SpannableString; import
        android.text.Spanned; import
        android.text.style.ImageSpan;

import com.google.android.material.tabs.TabLayout;

import java.lang.reflect.Array;
import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    // Initialize variables
    TabLayout tabLayout;
    ViewPager viewPager;

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

        // assign variable
        tabLayout=findViewById(R.id.tab_layout);
        viewPager=findViewById(R.id.view_pager);

        // Initialize array list
        ArrayList<String> arrayList=new ArrayList<>(0);

        // Add title in array list
        arrayList.add("Basic");        arrayList.add("Advance");
        arrayList.add("Pro");

        // Setup tab layout
        tabLayout.setupWithViewPager(viewPager);

```

```

        // Prepare view pager
prepareViewPager(viewPager,arrayList);

    }

    private void prepareViewPager(ViewPager viewPager, ArrayList<String> arrayList) {

        // Initialize main adapter

        MainAdapter adapter=new MainAdapter(getSupportFragmentManager());
// Initialize main fragment

        MainFragment mainFragment=new MainFragment();

        // Use for loop
        for(int i=0;i<arrayList.size();i++)
        {

            // Initialize bundle

            Bundle bundle=new Bundle();

            // Put title

            bundle.putString("title",arrayList.get(i));

            // set argument

            mainFragment.setArguments(bundle);

            // Add fragment

            adapter.addFragment(mainFragment,arrayList.get(i));
mainFragment=new MainFragment();

        }

        // set adapter

        viewPager.setAdapter(adapter);

    }

    private class MainAdapter extends FragmentPagerAdapter {

```

```

// Initialize arrayList
ArrayList<Fragment> fragmentArrayList= new ArrayList<>();
ArrayList<String> stringArrayList=new ArrayList<>();

int[] imageList={R.drawable.basic,R.drawable.advance,R.drawable.pro};

// Create constructor
public void addFragment(Fragment fragment,String s)
{
    // Add fragment
    fragmentArrayList.add(fragment);
    // Add title
    stringArrayList.add(s);
}

public MainAdapter(FragmentManager supportFragmentManager) {
    super(supportFragmentManager);
}

@Override
public Fragment getItem(int position) {
    // return fragment position
    return fragmentArrayList.get(position);
}

@Override
public int getCount() {
    // Return fragment array list size
    return fragmentArrayList.size();
}

```



```

        @Nullable

@Override

        public CharSequence getPageTitle(int position) {

            // Initialize drawable   Drawable drawable=
ContextCompat.getDrawable(getApplicationContext()

            ,imageList[position]);

            // set bound

            drawable.setBounds(0,0,drawable.getIntrinsicWidth(),

            drawable.getIntrinsicHeight());

            // Initialize spannable image

            SpannableString spannableString=new

SpannableString(""+stringArrayList.get(position));

            // Initialize image span

            ImageSpan imageSpan=new

ImageSpan(drawable,ImageSpan.ALIGN_BOTTOM);

            // Set span

            spannableString.setSpan(imageSpan,0,1,

Spanned.SPAN_EXCLUSIVE_EXCLUSIVE);

            // return spannable string

            return spannableString;

        }

    }

}

```

Fragment\_main.java:

```

package com.example.tablayoutwithicon;

import android.os.Bundle;

```

```

import androidx.fragment.app.Fragment;

import android.view.LayoutInflater;
import android.view.View; import
android.view.ViewGroup; import
android.widget.TextView;

public class MainFragment extends Fragment {

    // Initialize variable
    TextView textView;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {

        // Initialize view
        View view =inflater.inflate(R.layout.fragment_main, container, false);

        // Assign variable
        textView=view.findViewById(R.id.text_view);

        // Get Title
        String sTitle=getArguments().getString("title");

        // Set title on text view
        textView.setText(sTitle);

        // return view    return view;
    }
}

```

**Output:-**



## Practical:- 8

**Aim :- Create an android app to Connect to the Internet and use BroadcastReceiver.**

**XML Code :-**

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

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

```
</androidx.constraintlayout.widget.ConstraintLayout> Main_activity.java:
```

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

import android.content.Intent; import
android.content.IntentFilter; import
android.os.Bundle;

public class MainActivity extends AppCompatActivity {
```

```

        AirplaneModeChangeReceiver airplaneModeChangeReceiver = new
        AirplaneModeChangeReceiver();

        @Override

        protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity_main);

        }

        @Override

        protected void onStart() {

super.onStart();

        IntentFilter filter = new IntentFilter(Intent.ACTION_AIRPLANE_MODE_CHANGED);
        registerReceiver(airplaneModeChangeReceiver, filter);

        }

        @Override

        protected void onStop() {

            super.onStop();

            unregisterReceiver(airplaneModeChangeReceiver);

        }

    }

```

#### **AiropplanModeReciver.java:**

```

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.provider.Settings;

import android.widget.Toast;

public class AirplaneModeChangeReceiver extends BroadcastReceiver {

        @Override

```

```

public void onReceive(Context context, Intent intent) {

    if (isAirplaneModeOn(context.getApplicationContext())) {

        Toast.makeText(context, "AirPlane mode is on",
Toast.LENGTH_SHORT).show();

    } else {

        Toast.makeText(context, "AirPlane mode is off",
Toast.LENGTH_SHORT).show();

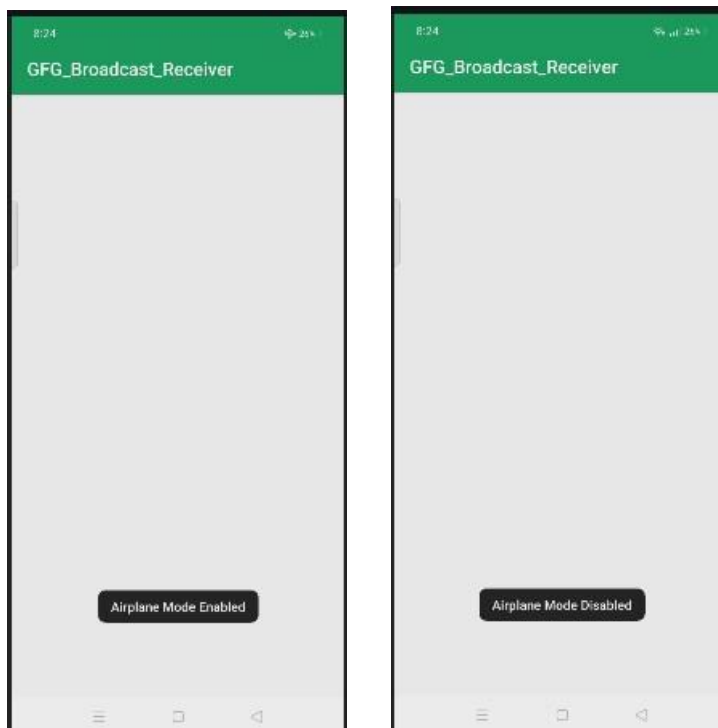
    }

}

private static boolean isAirplaneModeOn(Context context) {
return Settings.System.getInt(context.getContentResolver(),
Settings.Global.AIRPLANE_MODE_ON, 0) != 0;
}
}

```

## Output:-



## **Practical:- 9**

**Aim :- Create an android app to show Notifications and Alarm manager.**

**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:orientation="vertical">

    <TimePicker

        android:id="@+id/timePicker"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_gravity="center" />

    <ToggleButton
```

```

        android:id="@+id/toggleButton"

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_margin="20dp"
        android:checked="false"
        android:onClick="OnToggleClicked" />

        <!--"OnToggleClicked" method will be implemented in MainActivity.java -->

    </LinearLayout>

<resources>

    <color name="colorPrimary">#0F9D58</color>
    <color name="colorPrimaryDark">#0F4C2E</color>

    <color name="colorAccent">#9D0F9B</color>

</resources>

<resources>

    <!-- Base application theme. -->

    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">

        <!-- Customize your theme here. -->

        <item name="colorPrimary">@color/colorPrimary</item>

        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>

        <item name="colorAccent">@color/colorAccent</item>

    </style>

</resources>

```

## Source Code :-

```

import android.app.AlarmManager;

import android.app.PendingIntent;

```

```

import android.content.Intent;

import android.os.Bundle; import

android.view.View; import

android.widget.TimePicker; import

android.widget.Toast; import

android.widget.ToggleButton;

import androidx.appcompat.app.AppCompatActivity;

import java.util.Calendar;


    public class MainActivity extends AppCompatActivity {

        TimePicker alarmTimePicker;

        PendingIntent pendingIntent;

        AlarmManager alarmManager;


        @Override

        protected void onCreate(Bundle savedInstanceState) {

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


            alarmTimePicker = (TimePicker) findViewById(R.id.timePicker);

alarmManager = (AlarmManager) getSystemService(ALARM_SERVICE);


        }


        // OnToggleClicked() method is implemented the time functionality

public void OnToggleClicked(View view) {

            long time;

            if (((ToggleButton) view).isChecked()) {

                Toast.makeText(MainActivity.this, "ALARM ON",

Toast.LENGTH_SHORT).show();

                Calendar calendar = Calendar.getInstance();


                // calendar is called to get current time in hour and minute

```



```

        calendar.set(Calendar.HOUR_OF_DAY,
alarmTimePicker.getCurrentHour());

        calendar.set(Calendar.MINUTE,
alarmTimePicker.getCurrentMinute());

        // using intent i have class AlarmReceiver class which inherits
        // BroadcastReceiver
        Intent intent = new Intent(this, AlarmReceiver.class);

        // we call broadcast using pendingIntent
        pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0);

        time = (calendar.getTimeInMillis() - (calendar.getTimeInMillis() %
60000));

        if (System.currentTimeMillis() > time) {
            // setting time as AM and PM
            if (calendar.AM_PM == 0)
                time = time + (1000 * 60 * 60 * 12);
            else
                time = time + (1000 * 60 * 60 * 24);
        }

        // Alarm rings continuously until toggle button is turned off
        alarmManager.setRepeating(AlarmManager.RTC_WAKEUP, time,
10000, pendingIntent);

        // alarmManager.set(AlarmManager.RTC_WAKEUP,
System.currentTimeMillis() + (time * 1000), pendingIntent);
    } else {
        alarmManager.cancel(pendingIntent);

        Toast.makeText(MainActivity.this, "ALARM OFF",
Toast.LENGTH_SHORT).show();
    }
}
}
}

```

### AlarmReceiver.java:

```
import
    android.content.BroadcastReceiver;

import android.content.Context; import
    android.content.Intent; import
    android.media.Ringtone; import
    android.media.RingtoneManager; import
    android.net.Uri; import android.os.Build;
import android.os.Vibrator; import
    android.widget.Toast;

import androidx.annotation.RequiresApi;

public class AlarmReceiver extends BroadcastReceiver {
    @RequiresApi(api = Build.VERSION_CODES.Q)
    @Override
    // implement onReceive() method
    public void onReceive(Context context, Intent intent) {

        // we will use vibrator first
        Vibrator vibrator = (Vibrator)
            context.getSystemService(context.VIBRATOR_SERVICE);
        vibrator.vibrate(4000);

        Toast.makeText(context, "Alarm! Wake up! Wake up!",
            Toast.LENGTH_LONG).show();

        Uri alarmUri =
            RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);
        if (alarmUri == null) {
            alarmUri =
                RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
        }

        // setting default ringtone
```

```
Ringtone ringtone = RingtoneManager.getRingtone(context, alarmUri);
```

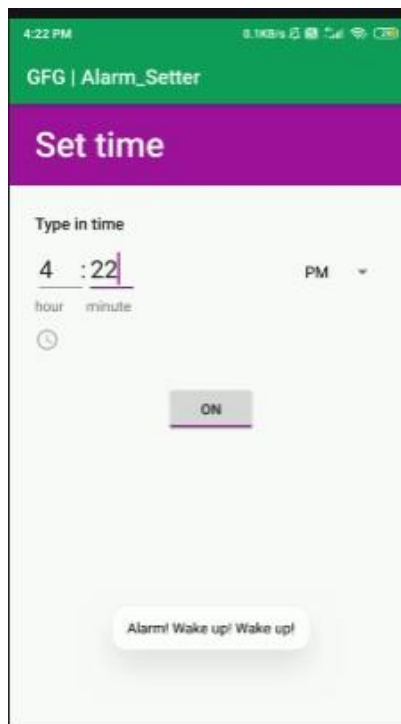
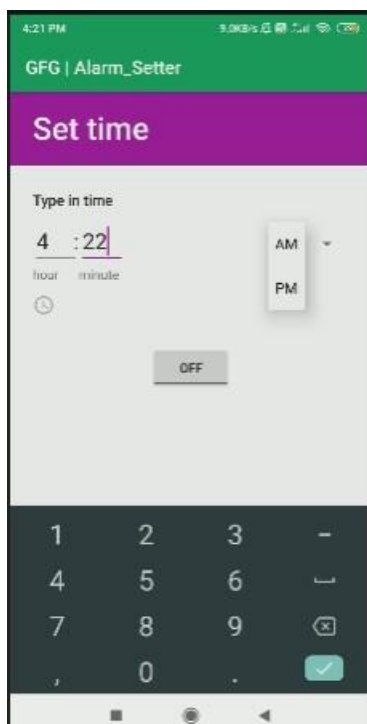
```
// play ringtone
```

```
ringtone.play();
```

```
}
```

```
}
```

## Output:-



## **Practical:- 10**

**Aim : Create an android app to save user data in a database and use of different queries.**

**XML Code :**

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

<LinearLayout

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

    <!--Edit text to enter course name-->

    <EditText

        android:id="@+id/idEdtCourseName"

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
android:layout_margin="10dp"
android:hint="Enter course Name" />
```

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

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

```
        <EditText
            android:id="@+id/idEdtCourseDescription"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="10dp"
            android:hint="Enter Course Description" />
```

```
        <Button
            android:id="@+id/idBtnAddCourse"
            android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:text="Add Course"
        android:textAllCaps="false" />

</LinearLayout>
```

#### **DataBaseHandler:**

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

public class DBHandler extends SQLiteOpenHelper {

    // creating a constant variables for our database.

    // below variable is for our database name.
    private static final String DB_NAME = "coursedb";

    // below int is our database version private
    static final int DB_VERSION = 1;

    // below variable is for our table name.
    private static final String TABLE_NAME = "mycourses";

    // below variable is for our id column.
    private static final String ID_COL = "id";

    // below variable is for our course name column
    private static final String NAME_COL = "name";
```

```

        // below variable id for our course duration column.
private static final String DURATION_COL = "duration";

        // below variable for our course description column.
private static final String DESCRIPTION_COL = "description";

        // below variable is for our course tracks column.
private static final String TRACKS_COL = "tracks";

        // creating a constructor for our database handler.
public DBHandler(Context context) {
super(context, DB_NAME, null, DB_VERSION);
    }

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

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

```

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

```
        // on below line we are creating a variable for  
// our sqlite database and calling writable method  
// as we are writing data in our database.
```

```
        SQLiteDatabase db = this.getWritableDatabase();
```

```
        // on below line we are creating a  
// variable for content values.
```

```
        ContentValues values = new ContentValues();
```

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

```
        // after adding all values we are passing  
// content values to our table.
```

```
        db.insert(TABLE_NAME, null, values);
```

```
        // at last we are closing our  
// database after adding database.
```

```
        db.close();
```

```
    }
```

```
@Override
```



```

        public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

            // this method is called to check if the table exists already.

            db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);

            onCreate(db);

        }

    }

```

#### **Mian\_activity.java:**

```

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

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

    @Override

    protected void onCreate(Bundle savedInstanceState) {

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

        // initializing all our variables.

        courseNameEdt = findViewById(R.id.idEdtCourseName);

        courseTracksEdt = findViewById(R.id.idEdtCourseTracks);

        courseDurationEdt = findViewById(R.id.idEdtCourseDuration);

```

```

courseDescriptionEdt = findViewById(R.id.idEdtCourseDescription);
addCourseBtn = findViewById(R.id.idBtnAddCourse);

        // creating a new dbhandler class
        // and passing our context to it.

        dbHandler = new DBHandler(MainActivity.this);

        // below line is to add on click listener for our add course button.
addCourseBtn.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

                // below line is to get data from all edit text fields.

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

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

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

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

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

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

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

                                return;

                        }

                // on below line we are calling a method to add new // course to sqlite data and pass
all our values to it. dbHandler.addNewCourse(courseName, courseDuration,
courseDescription, courseTracks);

```

```
        // after adding the data we are displaying a toast message.
        Toast.makeText(MainActivity.this, "Course has been added.",
        Toast.LENGTH_SHORT).show();

                courseNameEdt.setText("");

        courseDurationEdt.setText("");                courseTracksEdt.setText("");

                courseDescriptionEdt.setText("");

                }

        });

    }

}
```

**Output:-**

10:10 AM 4.7KB/s

### GFG App

Enter course Name

Enter Course Duration

Enter Course Tracks

Enter Course Description

Add Course

Google I the my

q w e r t y u i o p

a s d f g h j k l

↑ z x c v b n m

?123 , 😊 . ↵

10:10 AM 0.3KB/s

### GFG App

Java

30 days

20 Tracks

Java self paced course

Add Course

10:10 AM 0.3KB/s

### GFG App

Enter course Name

Enter Course Duration

Enter Course Tracks

Enter Course Description

Add Course

Course has been added.