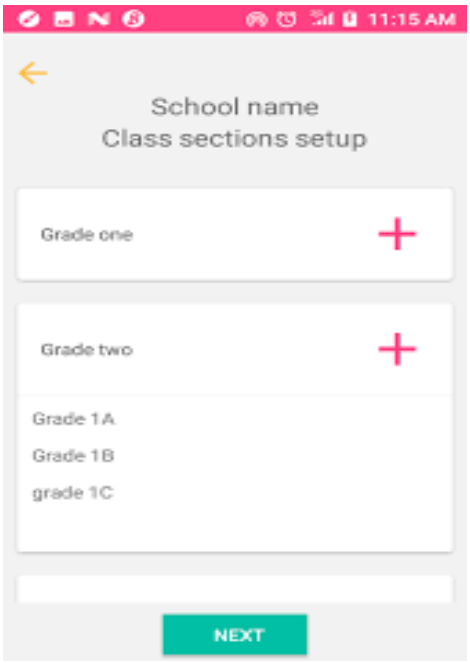
1) Implement an Android project having activity similar to given below



Activity_main.xml

2)
```
<?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:background="#F5F5F5"
        android:padding="16dp">

        <TextView
            android:id="@+id/school_name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="School name"
            android:textSize="24sp"
            android:textColor="#000"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="16dp"/>

        <TextView
            android:id="@+id/class_sections_setup"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Class sections setup"
            android:textSize="18sp"
            android:textColor="#666"
            android:layout_below="@id/school_name"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="8dp"/>

        <LinearLayout
```

```
            android:id="@+id/grade_one_layout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@id/class_sections_setup"
            android:layout_marginTop="24dp"
            android:background="#FFF"
            android:elevation="2dp"
            android:minWidth="48dp"
            android:orientation="vertical"
            android:padding="16dp">

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

                <TextView
                    android:id="@+id/grade_one"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Grade one"
                    android:textColor="#000"
                    android:textSize="18sp" />

                <ImageButton
                    android:id="@+id/grade_one_button"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_alignParentEnd="true"
                    android:layout_centerVertical="true"

    android:background="?attr/selectableItemBackgroundBorderless"
                    android:minWidth="48dp"
                    android:minHeight="48dp"

    android:contentDescription="grade_one_button_description"

                    android:src="@android:drawable/ic_input_add" />
            </RelativeLayout>

            <LinearLayout
                android:id="@+id/grade_one_sections"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="8dp"
                android:orientation="vertical"
                android:visibility="gone">

                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="4dp"
                    android:text="Grade 1A"
                    android:textColor="#000"
                    android:textSize="16sp" />

                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="4dp"
                    android:text="Grade 1B"
                    android:textColor="#000"
```

```
                android:textSize="16sp" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginTop="4dp"
                android:text="Grade 1C"
                android:textColor="#000"
                android:textSize="16sp" />
        </LinearLayout>
    </LinearLayout>

    <LinearLayout
        android:id="@+id/grade_two_layout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:background="#FFF"
        android:elevation="2dp"
        android:padding="16dp"
        android:layout_below="@id/grade_one_layout"
        android:layout_marginTop="16dp">

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

            <TextView
                android:id="@+id/grade_two"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Grade two"
                android:textSize="18sp"
                android:textColor="#000"/>

            <ImageButton
                android:id="@+id/grade_two_button"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@android:drawable/ic_input_add"

android:background="?attr/selectableItemBackgroundBorderless"
                android:layout_alignParentEnd="true"
                android:minWidth="48dp"
                android:minHeight="48dp"

android:contentDescription="grade_two_button_description"

                android:layout_centerVertical="true"/>
        </RelativeLayout>

        <LinearLayout
            android:id="@+id/grade_two_sections"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:visibility="gone"
            android:layout_marginTop="8dp">

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

```
                    android:layout_height="wrap_content"
                    android:text="Grade 2A"
                    android:textSize="16sp"
                    android:textColor="#000"
                    android:layout_marginTop="4dp"/>

                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Grade 2B"
                    android:textSize="16sp"
                    android:textColor="#000"
                    android:layout_marginTop="4dp"/>

                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Grade 2C"
                    android:textSize="16sp"
                    android:textColor="#000"
                    android:layout_marginTop="4dp"/>
            </LinearLayout>
        </LinearLayout>

        <Button
            android:id="@+id/next_button"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="NEXT"

            android:layout_alignParentBottom="true"
            android:layout_marginTop="16dp"/>
    </RelativeLayout>
```

Mainactivity.java

```
package com.example.myfirst;

import android.os.Bundle;
import android.view.View;
import android.widget.ImageButton;
import android.widget.LinearLayout;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private LinearLayout gradeOneSections;
    private LinearLayout gradeTwoSections;

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

        gradeOneSections = findViewById(R.id.grade_one_sections);
        gradeTwoSections = findViewById(R.id.grade_two_sections);

        ImageButton gradeOneButton = findViewById(R.id.grade_one_button);
```

```
        ImageButton gradeTwoButton = findViewById(R.id.grade_two_button);

        gradeOneButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (gradeOneSections.getVisibility() == View.GONE) {
                    gradeOneSections.setVisibility(View.VISIBLE);
                } else {
                    gradeOneSections.setVisibility(View.GONE);
                }
            }
        });

        gradeTwoButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (gradeTwoSections.getVisibility() == View.GONE) {
                    gradeTwoSections.setVisibility(View.VISIBLE);
                } else {
                    gradeTwoSections.setVisibility(View.GONE);
                }
            }
        });
    }
}
```

2) Write an Android project to create 3 activities with explicit Intents and implicit intent to open camera.

Mainactivity.java

```
package com.example.camopen;


import android.annotation.SuppressLint;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.provider.MediaStore;
import android.widget.Button;
import android.widget.ImageView;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {
    private static final int pic_id = 123;
    // Define the button and imageview type variable
    Button camera_open_id;
    ImageView click_image_id;
```

```
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        camera_open_id = findViewById(R.id.camera_button);
        click_image_id = findViewById(R.id.click_image);
        camera_open_id.setOnClickListener(v -> {
            // Create the camera_intent ACTION_IMAGE_CAPTURE it will open
the camera for capture the image
            Intent camera_intent = new
Intent(MediaStore.ACTION_IMAGE_CAPTURE);
            // Start the activity with camera_intent, and request pic id
            startActivityForResult(camera_intent, pic_id);
        });
    }
    protected void onActivityResult(int requestCode, int resultCode, Intent
data) {
        super.onActivityResult(requestCode, resultCode, data);
        // Match the request 'pic id with requestCode
        if (requestCode == pic_id) {
            // BitMap is data structure of image file which store the image
in memory
            Bitmap photo = (Bitmap) data.getExtras().get("data");
            // Set the image in imageview for display
            click_image_id.setImageBitmap(photo);
        }
    }
}
```

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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/camera_button"
        android:layout_width="100dp"
        android:layout_height="50dp"
        android:layout_marginStart="150dp"
        android:text="Camera" />

    <!-- add ImageView to display the captured image -->
    <ImageView
        android:id="@+id/click_image"
        android:layout_width="350dp"
        android:layout_height="450dp"
        android:layout_marginStart="30dp"
        android:layout_marginTop="70dp"
        android:layout_marginBottom="10dp" />

</LinearLayout>
```

3) Write an Android project to create 2 activities and Implicit Intents to open a web page of javapoint.com

Activity_main.xml

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

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="60dp"
        android:minHeight="48dp"
        android:hint="enter"
        android:ems="10"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.575"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginRight="8dp"
        android:layout_marginLeft="156dp"
        android:layout_marginTop="172dp"
        android:text="Visit"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.java

```
package com.example.myapplication;

import androidx.constraintlayout.widget.ConstraintLayout;
```

```
import android.content.Intent;
import android.net.Uri;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    Button button;
    EditText editText;

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

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

        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String url=editText.getText().toString();
                Intent intent=new Intent(Intent.ACTION_VIEW,
Uri.parse(url));
                startActivity(intent);
            }
        });
    }
}
```

4. Write an Android project to create app bar and create option menus and context menu.

Mainactivity.xml

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

   <TextView
      android:id="@+id/textView"
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Long press me for context menu"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

  <androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbar"
    android:layout_width="409dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="1dp"
    android:layout_marginBottom="288dp"
    android:background="?attr/colorPrimary"
    android:minHeight="?attr/actionBarSize"
    android:theme="?attr/actionBarTheme"
    app:layout_constraintBottom_toTopOf="@+id/textView"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

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

Main menu(res->menu->main_menu.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<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/home"
    android:title="home"
    app:showAsAction="always"
    android:icon="@drawable/home"/>
  <item  android:id="@+id/share"
    android:title="Share"
    app:showAsAction="always"
    android:icon="@drawable/share"/>
  <item  android:id="@+id/star"
    android:title="star"
```

```
    app:showAsAction="ifRoom"
    android:icon="@drawable/star"/>
  <item  android:id="@+id/setting"
    android:title="settings"
    app:showAsAction="never"
    android:icon="@drawable/settings"/>
  <item  android:id="@+id/exit"
    android:title="exit"
    app:showAsAction="always"
    android:icon="@drawable/close"/>
  <item  android:id="@+id/refresh"
    android:title="refresh"
    app:showAsAction="always"
    android:icon="@drawable/refresh"/>
</menu>
```

Contextmenu.xml(res->menu->context _menu.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
  <item
    android:id="@+id/copy"
    android:title="copy" />
  <item
    android:id="@+id/cut"
    android:title="cut" />
  <item
    android:id="@+id/paste"
    android:title="Paste" />
  <item
    android:id="@+id/share"
    android:title="Share" />
  <item
    android:id="@+id/delete"
    android:title="Delete" />
</menu>
```

Mainactivity.java

```
package com.example.option_context;

    import android.app.ActionBar;
    import android.graphics.Color;
    import android.graphics.drawable.ColorDrawable;
    import android.os.Bundle;
    import android.view.ContextMenu;
    import android.view.Menu;
    import android.view.MenuInflater;
    import android.view.MenuItem;
    import android.view.View;
    import android.widget.TextView;
    import android.widget.Toast;

    import androidx.activity.EdgeToEdge;
    import androidx.appcompat.app.AppCompatActivity;
    import androidx.appcompat.widget.Toolbar;
    import androidx.core.graphics.Insets;
    import androidx.core.view.ViewCompat;
    import androidx.core.view.WindowInsetsCompat;

    public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Toolbar toolbar = findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);
    ActionBar actionBar = getActionBar();
    if (actionBar != null) {
    actionBar.setBackgroundDrawable(new ColorDrawable(Color.BLUE));
    actionBar.setTitle("Menu");
    actionBar.show();
    }


    //        for context menu
    TextView textView=findViewById(R.id.textView);
    registerForContextMenu(textView);
    }
    //option menu
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main_menu, menu);
    return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
    int itemId = item.getItemId();
    if (itemId == R.id.home) {
    Toast.makeText(getApplicationContext(), "home icon",
Toast.LENGTH_SHORT).show();

    return true;
    } else if (itemId == R.id.share) {
    Toast.makeText(getApplicationContext(), "Share icon",
Toast.LENGTH_SHORT).show();
    return (true);
    } else if (itemId == R.id.star) {
```

```
    Toast.makeText(getApplicationContext(), "star icon",
Toast.LENGTH_SHORT).show();
    return (true);
    } else if (itemId == R.id.setting) {
    Toast.makeText(getApplicationContext(), "setting menu",
Toast.LENGTH_SHORT).show();
    return (true);
    }
    else if (itemId == R.id.refresh) {
    Toast.makeText(getApplicationContext(), "Refresh menu",
Toast.LENGTH_SHORT).show();
    return (true);
    }else if (itemId == R.id.exit) {
    finish();
    return (true);
    }
    return super.onOptionsItemSelected(item);

    }

    //    context menu
    @Override
    public void onCreateContextMenu(ContextMenu menu, View view,
ContextMenu.ContextMenuInfo menuInfo){
    super.onCreateContextMenu(menu,view,menuInfo);
    MenuInflater inflater=getMenuInflater();
    inflater.inflate(R.menu.context_menu,menu);

    }

    @Override
    public boolean onContextItemSelected(MenuItem item){
    int itemId = item.getItemId();
    if (itemId == R.id.copy) {
    Toast.makeText(getApplicationContext(), " copied",
Toast.LENGTH_SHORT).show();
    return true;
    } else if (itemId == R.id.cut) {
    Toast.makeText(getApplicationContext(), "cut",
Toast.LENGTH_SHORT).show();
    return (true);
    } else if (itemId == R.id.paste) {
    Toast.makeText(getApplicationContext(), "paste",
Toast.LENGTH_SHORT).show();
    return (true);
    } else if (itemId == R.id.share) {
    Toast.makeText(getApplicationContext(), "share",
Toast.LENGTH_SHORT).show();
    return (true);
    }
    else if (itemId == R.id.delete) {
    Toast.makeText(getApplicationContext(), "deleted",
Toast.LENGTH_SHORT).show();
    return (true);
    }
    return super.onOptionsItemSelected(item);

    }

    }
```

5. Write an Android project to change the Color theme of the App and create a custom theme for views.

Mainactivity.java

```
package com.example.theme;


import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.app.AppCompatDelegate;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private static final String PREFS_NAME = "prefs";
    private static final String PREF_DARK_THEME = "dark_theme";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Check if dark theme is enabled
        SharedPreferences preferences = getSharedPreferences(PREFS_NAME,
MODE_PRIVATE);
        boolean useDarkTheme = preferences.getBoolean(PREF_DARK_THEME,
false);

        if (useDarkTheme) {
            setTheme(R.style.Theme_ThemeChangeApp_Dark);
        } else {
            setTheme(R.style.Theme_ThemeChangeApp);
        }

        setContentView(R.layout.activity_main);

        Button toggleButton = findViewById(R.id.toggle);
        toggleButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                toggleTheme();
            }
        });
    }
```

```
    private void toggleTheme() {
        SharedPreferences.Editor editor = getSharedPreferences(PREFS_NAME,
MODE_PRIVATE).edit();
        boolean useDarkTheme = getSharedPreferences(PREFS_NAME,
MODE_PRIVATE)
                .getBoolean(PREF_DARK_THEME, false);

        if (useDarkTheme) {
            editor.putBoolean(PREF_DARK_THEME, false);
        } else {
            editor.putBoolean(PREF_DARK_THEME, true);
        }

        editor.apply();

        // Restart activity to apply theme change
        Intent intent = getIntent();
        finish();
        startActivity(intent);
    }
}
```

activity.xml

```
<?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"
    android:padding="16dp"
    >

    <Button
        android:id="@+id/toggle"
        style="@style/CustomButtonTheme"
        android:layout_width="371dp"
        android:layout_height="48dp"
        android:text="Toggle Theme"
        android:textSize="20dp" />

</LinearLayout>
```

Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="purple_500">#6200EE</color>
    <color name="purple_700">#3700B3</color>
    <color name="teal_200">#03DAC5</color>
    <color name="teal_700">#018786</color>
</resources>
```

Themes.xml

```
<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Base.Theme.ChangeTheme"
```

```
parent="Theme.Material3.DayNight.NoActionBar">
        <!-- Customize your light theme here. -->
        <!-- <item name="colorPrimary">@color/my_light_primary</item> -->
    </style>


    <!-- Base application theme. -->
    <style name="Theme.ThemeChangeApp"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/purple_500</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/white</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_700</item>
        <item
name="colorOnSecondary">@color/design_default_color_error</item>
        <!-- Default background color. -->
        <item name="android:colorBackground">@color/white</item>
        <item
name="android:textColor">@color/design_default_color_error</item>
    </style>

    <!-- Custom theme for a specific view -->
    <style name="CustomButtonTheme"
parent="Widget.MaterialComponents.Button">
        <item name="android:backgroundTint">@color/teal_200</item>
        <item name="android:textColor">@color/white</item>
    </style>

    <!-- Dark theme -->
    <style name="Theme.ThemeChangeApp.Dark"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <item name="colorPrimary">@color/design_default_color_error</item>
        <item
name="colorPrimaryVariant">@color/design_default_color_error</item>
        <item name="colorOnPrimary">@color/white</item>
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_700</item>
        <item
name="colorOnSecondary">@color/design_default_color_error</item>
        <item
name="android:colorBackground">@color/design_default_color_error</item>
        <item name="android:textColor">@color/white</item>
    </style>


</resources>
```

Manifest.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">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
```

```
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.ThemeChangeApp"
        tools:targetApi="31">
        <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>
    </application>

</manifest>
```

6. Write an Android project to create Background Task using AsyncTask to download a file/play music.

7. Write an Android project for Intent to pass data from one activity to another using extra and data

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/send_text_id"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="20dp"
        android:hint="Input"
        android:textSize="25dp"
        android:textStyle="bold" />

    <Button
        android:id="@+id/send_button_id"
```

```
        android:layout_width="wrap_content"
        android:layout_height="48dp"
        android:layout_marginLeft="150dp"
        android:layout_marginTop="150dp"
        android:text="send"
        android:textStyle="bold" />
</RelativeLayout>
```

Mainactivity.java

```
package com.example.my7;

import androidx.activity.EdgeToEdge;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import android.content.Intent;
import android.widget.Button;
import android.widget.EditText;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    // define the variable
    Button send_button;
    EditText send_text;

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

        send_button = findViewById(R.id.send_button_id);
        send_text = findViewById(R.id.send_text_id);

        // add the OnClickListener in sender button after clicked this
button following Instruction will run
        send_button.setOnClickListener(v -> {
            // get the value which input by user in EditText and convert it
to string
            String str = send_text.getText().toString();
            // Create the Intent object of this class Context() to
Second_activity class
            Intent intent = new Intent(getApplicationContext(),
SecondActivity.class);
            // now by putExtra method put the value in key, value pair key
is
            // message_key by this key we will receive the value, and put
the string
            intent.putExtra("message_key", str);
            // start the Intent
            startActivity(intent);
        });
    }
}
```

activity_second.xml

```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=".SecondActivity">

    <TextView
        android:id="@+id/received"
        android:layout_width="300dp"
        android:layout_height="50dp"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="20dp"
        android:textSize="40sp"
        android:textStyle="bold"
        android:layout_marginStart="40dp" />
</RelativeLayout>
```

Secondactivity.main

```java
package com.example.my7;

import android.annotation.SuppressLint;
import android.os.Bundle;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {

    TextView receiver_msg;

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

        receiver_msg = findViewById(R.id.received);
        // create the get Intent object
        Intent intent = getIntent();
        // receive the value by getStringExtra() method and
        // key must be same which is send by first activity
        String str = intent.getStringExtra("message_key");
        // display the string into textView
        receiver_msg.setText(str);
    }
}
```

manifest.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">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.My7"
        tools:targetApi="31">
        <activity
            android:name=".SecondActivity"
            android:exported="false" />
        <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>
    </application>

</manifest>
```

8.. Write an Android project for creating Notification

Android Manifest

```
<?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.POST_NOTIFICATIONS"
/>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Notification"
        tools:targetApi="31">

        <activity
```

```
            android:name="com.example.notification.MainActivity"
            android:label="@string/app_name" android:exported="true">

            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>

        </activity>

        <activity android:name=".Notification_View"
            android:label="Details of notification"
            android:parentActivityName=".MainActivity">
            <meta-data
                android:name="android.support.PARENT_ACTIVITY"
                android:value=".MainActivity"/>
        </activity>

    </application>
</manifest>
```

MainActivity.java

```
package com.example.notification;

import static android.Manifest.permission.POST_NOTIFICATIONS;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationManagerCompat;

import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
```

```
import android.content.pm.PackageManager;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    Button b1;

    private final String CHANNEL_ID = "example_channel_id";

    private final int NOTIFICATION_ID = 0;

    private final int REQUEST_CODE = 1;



    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);



        b1 = (Button) findViewById(R.id.button);

        b1.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {

                createNotificationChannel();

                if
(ActivityCompat.checkSelfPermission(MainActivity.this,POST_NOTIFICATIONS)
!= PackageManager.PERMISSION_GRANTED) {

                    // Request the missing permission

                    ActivityCompat.requestPermissions(MainActivity.this,
new String[]{POST_NOTIFICATIONS}, REQUEST_CODE);

                } else {

                    // Permission is already granted, proceed with
notification

                    addNotification();

                }

            }

        });

    }
```

```
    private void createNotificationChannel() {

        // Create the NotificationChannel, but only on API 26+ because

        // the NotificationChannel class is new and not in the support
library

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {

            CharSequence name = "Example Channel";

            String description = "This is an example notification channel";

            int importance = NotificationManager.IMPORTANCE_DEFAULT;

            NotificationChannel channel = new
NotificationChannel(CHANNEL_ID, name, importance);

            channel.setDescription(description);



            // Register the channel with the system

            NotificationManager notificationManager =
getSystemService(NotificationManager.class);

            notificationManager.createNotificationChannel(channel);

        }

    }



    private void addNotification() {

        NotificationCompat.Builder builder = new
NotificationCompat.Builder(this, CHANNEL_ID)

                .setSmallIcon(R.drawable.ic_launcher_foreground)  //
replace with your actual icon

                .setContentTitle("Notifications Example")

                .setContentText("This is a test notification")

                .setPriority(NotificationCompat.PRIORITY_DEFAULT);



        Intent notificationIntent = new Intent(this, MainActivity.class);

        PendingIntent contentIntent = PendingIntent.getActivity(this, 0,
notificationIntent, PendingIntent.FLAG_UPDATE_CURRENT |
PendingIntent.FLAG_IMMUTABLE);

        builder.setContentIntent(contentIntent);
```

```
        // Show the notification

        NotificationManagerCompat notificationManager =
NotificationManagerCompat.from(this);

        if (ActivityCompat.checkSelfPermission(this,POST_NOTIFICATIONS) !=
PackageManager.PERMISSION_GRANTED) {

            // TODO: Consider calling

            //    ActivityCompat#requestPermissions

            // here to request the missing permissions, and then overriding

            //   public void onRequestPermissionsResult(int requestCode,
String[] permissions,

            //                                          int[] grantResults)

            // to handle the case where the user grants the permission. See
the documentation

            // for ActivityCompat#requestPermissions for more details.

            return;

        }

        notificationManager.notify(NOTIFICATION_ID, builder.build());

    }



    @Override

    public void onRequestPermissionsResult(int requestCode, String[]
permissions,  int[] grantResults) {

        super.onRequestPermissionsResult(requestCode, permissions,
grantResults);

        if (requestCode == REQUEST_CODE) {

            // Check if the permission is granted

            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {

                // Permission is granted, proceed with notification

                addNotification();

            } else {

                // Permission is denied, handle accordingly (e.g., show a
message)

            }
```

```
        }

    }

}
```

Activity_Main.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=".MainActivity">

    <TextView

        android:id="@+id/textView1"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_alignParentTop="true"

        android:layout_centerHorizontal="true"

        android:text="Notification Example"

        android:textSize="30dp" />


    <TextView

        android:id="@+id/textView2"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_below="@+id/textView1"

        android:layout_centerHorizontal="true"

        android:layout_marginTop="48dp"

        android:text="Tutorials point"

        android:textColor="#ff87ff09"

        android:textSize="30dp" />


    <ImageButton

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

```
        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_below="@+id/textView2"

        android:layout_centerHorizontal="true"

        android:layout_marginTop="48dp"

        android:src="@drawable/ic_launcher_foreground" />


    <Button

        android:id="@+id/button"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_below="@+id/imageButton"

        android:layout_centerHorizontal="true"

        android:layout_marginTop="-164dp"

        android:text="Notification" />


</RelativeLayout>
```

Notification_View.java

```
package com.example.notification;

import androidx.appcompat.app.AppCompatActivity;

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

public class Notification_View extends AppCompatActivity {


        TextView textView;
        @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_notification_view);
            textView = findViewById(R.id.textView);
            //getting the notification message
            String message=getIntent().getStringExtra("message");
            textView.setText(message);
        }

}
```

activity_Notification_View.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=".Notification_View">

    <TextView

        android:id="@+id/textView"

        android:layout_width="fill_parent"

        android:layout_height="400dp"
        android:text="Hi, Your Detailed notification view goes here...." />


</LinearLayout>
```

9. Write an Android project to create shared preferences.

MainActivity.java

```
package com.example.sharedpreference;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

    private EditText name, age;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = findViewById(R.id.edit1);
        age = findViewById(R.id.edit2);
    }

    // Fetch the stored data in onResume() Because this is what will be
called when the app opens again
```

```
    @Override
    protected void onResume() {
        super.onResume();
        // Fetching the stored data from the SharedPreference
        SharedPreferences sh = getSharedPreferences("MySharedPref",
MODE_PRIVATE);
        String s1 = sh.getString("name", "");
        int a = sh.getInt("age", 0);

        // Setting the fetched data in the EditTexts
        name.setText(s1);
        age.setText(String.valueOf(a));
    }

    // Store the data in the SharedPreference in the onPause() method
    // When the user closes the application onPause() will be called and
data will be stored
    @Override
    protected void onPause() {
        super.onPause();
        // Creating a shared pref object with a file name "MySharedPref" in
private mode
        SharedPreferences sharedPreferences =
getSharedPreferences("MySharedPref", MODE_PRIVATE);
        SharedPreferences.Editor myEdit = sharedPreferences.edit();

        // write all the data entered by the user in SharedPreference and
apply
        myEdit.putString("name", name.getText().toString());
        myEdit.putInt("age", Integer.parseInt(age.getText().toString()));
        myEdit.apply();
    }

}
```

activity_main.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=".MainActivity">

    <TextView
        android:id="@+id/textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="32dp"
        android:text="Shared Preferences Demo"
        android:textColor="@android:color/black"
        android:textSize="24sp" />

    <!--EditText to take the data from the user and save the data in
SharedPreferences-->
```

```
    <EditText
        android:id="@+id/edit1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textview"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Name"
        android:padding="10dp" />

    <!--EditText to take the data from the user and save the data in
SharedPreferences-->
    <EditText
        android:id="@+id/edit2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/edit1"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Age"
        android:inputType="number"
        android:padding="10dp" />


</RelativeLayout>
```

10. Write an Android project to store data in SQLite database.

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">

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

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

    <Button
```

```
        android:id="@+id/insertButton"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Insert" />

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

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


</LinearLayout>
```

MainActivity.java

```
package com.example.sqlitedatabase;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    private DatabaseManager dbManager;
    private EditText nameEditText;
    private EditText emailEditText;
    private TextView resultTextView;
    private Button insertButton;
    private Button fetchButton;

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

        dbManager = new DatabaseManager(this);
        dbManager.open();

        nameEditText = findViewById(R.id.nameEditText);
        emailEditText = findViewById(R.id.emailEditText);
        resultTextView = findViewById(R.id.resultTextView);
        insertButton = findViewById(R.id.insertButton);
        fetchButton = findViewById(R.id.fetchButton);

        insertButton.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
            public void onClick(View v) {
                String name = nameEditText.getText().toString();
                String email = emailEditText.getText().toString();
                dbManager.insert(name, email);
            }
        });

        fetchButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Cursor cursor = dbManager.fetch();
                if (cursor.moveToFirst()) {
                    StringBuilder result = new StringBuilder();
                    do {
                        result.append("ID:
").append(cursor.getInt(0)).append(", ")
                                .append("Name:
").append(cursor.getString(1)).append(", ")
                                .append("Email:
").append(cursor.getString(2)).append("\n");
                    } while (cursor.moveToNext());
                    resultTextView.setText(result.toString());
                }
            }
        });
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        dbManager.close();
    }

}
```

DatabaseHelper.java

```
package com.example.sqlitedatabase;

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 = "example.db";
    private static final int DATABASE_VERSION = 1;

    public static final String TABLE_NAME = "users";
    public static final String COLUMN_ID = "_id";
    public static final String COLUMN_NAME = "name";
    public static final String COLUMN_EMAIL = "email";

    private static final String TABLE_CREATE =
            "CREATE TABLE " + TABLE_NAME + " (" +
                    COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
                    COLUMN_NAME + " TEXT, " +
```

```
                    COLUMN_EMAIL + " TEXT);";

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

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(TABLE_CREATE);
    }

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

}
```

DatabaseManager.java

```
package com.example.sqlitedatabase;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;

public class DatabaseManager {
    private DatabaseHelper dbHelper;
    private Context context;
    private SQLiteDatabase database;

    public DatabaseManager(Context context) {
        this.context = context;
    }

    public DatabaseManager open() throws SQLException {
        dbHelper = new DatabaseHelper(context);
        database = dbHelper.getWritableDatabase();
        return this;
    }

    public void close() {
        dbHelper.close();
    }

    public void insert(String name, String email) {
        ContentValues contentValue = new ContentValues();
        contentValue.put(DatabaseHelper.COLUMN_NAME, name);
        contentValue.put(DatabaseHelper.COLUMN_EMAIL, email);
        database.insert(DatabaseHelper.TABLE_NAME, null, contentValue);
    }

    public Cursor fetch() {
```

```
        String[] columns = new String[] {
                DatabaseHelper.COLUMN_ID,
                DatabaseHelper.COLUMN_NAME,
                DatabaseHelper.COLUMN_EMAIL
        };
        Cursor cursor = database.query(DatabaseHelper.TABLE_NAME, columns,
null, null, null, null, null);
        if (cursor != null) {
            cursor.moveToFirst();
        }
        return cursor;
    }

    public int update(long _id, String name, String email) {
        ContentValues contentValues = new ContentValues();
        contentValues.put(DatabaseHelper.COLUMN_NAME, name);
        contentValues.put(DatabaseHelper.COLUMN_EMAIL, email);
        return database.update(DatabaseHelper.TABLE_NAME, contentValues,
DatabaseHelper.COLUMN_ID + " = " + _id, null);
    }

    public void delete(long _id) {
        database.delete(DatabaseHelper.TABLE_NAME, DatabaseHelper.COLUMN_ID
+ "=" + _id, null);
    }

}
```

11. Write an Android project to create 3 activities (registration, login, profile)using explicit Intent and back navigation using back arrows.

MainActivity.java

```
package com.example.intent;

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

public class MainActivity extends AppCompatActivity {

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

```
        Button btnOpenActivity2 = findViewById(R.id.btnOpenActivity2);
        btnOpenActivity2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this,
Activity2.class);
                startActivity(intent);
            }
        });
    }
}
```

Main2.java

```
package com.example.intent;

import android.content.Intent;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;

public class Activity2 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_2);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);

        Button btnOpenActivity3 = findViewById(R.id.btnOpenActivity3);
        btnOpenActivity3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Activity2.this, Activity3.class);
                startActivity(intent);
            }
        });
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == android.R.id.home) {
            finish();
            return true;
        }
        return super.onOptionsItemSelected(item);
```

```
    }
}
```

Mainact3.java

```
package com.example.intent;

import android.content.Intent;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.MenuItem;

public class Activity3 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_3);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == android.R.id.home) {
            finish();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
}
```

Activitymain.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="center">

    <Button
```

```
            android:id="@+id/btnOpenActivity2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Open Activity 2" />

</LinearLayout>
```

Main2.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="center">

    <Button
            android:id="@+id/btnOpenActivity3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Open Activity 3" />

</LinearLayout>
```

Main3.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="center">

    <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Activity 3"
            android:textSize="24sp" />

</LinearLayout>
```

Manifest

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

    <application
            android:allowBackup="true"
            android:dataExtractionRules="@xml/data_extraction_rules"
            android:fullBackupContent="@xml/backup_rules"
            android:icon="@mipmap/ic_launcher"
            android:label="@string/app_name"
            android:supportsRtl="true"
            android:theme="@style/Theme.Intent"
            tools:targetApi="31">
        <activity
                android:name=".Activity3"
                android:exported="false"/>
        <activity
                android:name=".Activity2"
                android:exported="false"/>
        <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>
    </application>

</manifest>
```

12. Implement an Android project. Use Recyclerview like below

Mainactivity.java

```java
package com.example.ass1;

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

import android.os.Bundle;

import java.util.ArrayList;
import java.util.List;

public class MainActivity extends AppCompatActivity {

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

        List<String> list=new ArrayList<>();
        int
images[]={R.drawable.settings,R.drawable.settings,R.drawable.setting
s,R.drawable.settings,R.drawable.settings,R.drawable.settings,R.draw
able.settings,R.drawable.settings};


        list.add("C++");
        list.add("C#");
        list.add("Java");
        list.add("Android");
        list.add("Kotlin");
        list.add("HTML");
        list.add("CSS");
        list.add("JavaScript");

        RecyclerView rc = findViewById(R.id.rc);
        rc.setLayoutManager(new LinearLayoutManager(this));

        Adapter ad = new Adapter(list,images);

        rc.setAdapter(ad);
    }
}
```

Adapter.java

```java
package com.example.ass1;


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

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

import java.util.List;

public class Adapter  extends
RecyclerView.Adapter<Adapter.Viewholder> {
   List<String>list;
   int images[];
   public Adapter(List<String> list,int images[]){
      this.list=list;
      this.images=images;

   }
   @NonNull
   @Override

   public Viewholder onCreateViewHolder(@NonNull ViewGroup
parent, int viewType) {

      View view =
LayoutInflater.from(parent.getContext()).inflate(R.layout.item,parent,fa
lse);

      return new Viewholder(view);
   }

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

      holder.textView.setText(list.get(position));
      holder.imageView.setImageResource(images[position]);
```

```
    }

    @Override
    public int getItemCount() {
        return list.size();
    }

    public class Viewholder extends RecyclerView.ViewHolder {

        TextView textView;
        ImageView imageView;

        public Viewholder(@NonNull View itemView) {
            super(itemView);
            textView=itemView.findViewById(R.id.textview);
            imageView=itemView.findViewById(R.id.imageView);

        }
    }
}
```

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">

    <androidx.recyclerview.widget.RecyclerView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/rc"/>

</LinearLayout>
```

item.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="wrap_content"
   android:gravity="center_vertical"
   android:orientation="vertical"
   tools:context=".MainActivity">

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="100dp"
      android:layout_gravity="center_vertical"
      android:layout_margin="10dp"
      android:layout_marginStart="10dp"
      android:layout_marginLeft="10dp"
      android:layout_marginTop="10dp"
      android:layout_marginEnd="10dp"
      android:layout_marginRight="10dp"
      android:layout_marginBottom="10dp"
      android:elevation="50dp">

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

         <ImageView
            android:id="@+id/imageView"
            android:layout_width="50dp"
            android:layout_height="50dp"

            android:layout_marginLeft="20dp" />

         <TextView
            android:id="@+id/textview"
            android:layout_width="200dp"
            android:layout_height="50dp"
            android:layout_gravity="center"

            android:gravity="center"
            android:textSize="30dp" />
```

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

</LinearLayout>
```

13. Implement an Android project to play audio and video using AsyncTask

AsyncTask Audio vdo

```
<?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"
   android:padding="16dp"
   tools:context=".MainActivity">

   <androidx.appcompat.widget.Toolbar
      android:id="@+id/toolbar"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:background="?attr/colorPrimary"
      android:minHeight="?attr/actionBarSize"
      android:theme="?attr/actionBarTheme" />

   <Button
      android:id="@+id/play_audio"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="Play Audio" />

   <Button
      android:id="@+id/play_video"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:text="Play Video"
      android:layout_marginTop="16dp" />
```

```
    <VideoView
        android:id="@+id/video_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp" />
</LinearLayout>
```

Mainactivity.java

```
package com.example.asyncniranjan;
// src/main/java/com/example/mediaplaybackexample/MainActivity.java
import android.content.Context;
import android.media.MediaPlayer;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.PopupMenu;
import android.widget.VideoView;
import android.net.Uri;
import android.widget.MediaController;

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

public class MainActivity extends AppCompatActivity {

    private Button playAudioButton;
    private Button playVideoButton;
    private VideoView videoView;



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

        playAudioButton = findViewById(R.id.play_audio);
        playVideoButton = findViewById(R.id.play_video);
        videoView = findViewById(R.id.video_view);
```

```
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        playAudioButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                new PlayAudioTask(getApplicationContext()).execute();
                popup(v);

            }
        });

        playVideoButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                playVideo(Uri.parse("android.resource://" +
getPackageName() + "/" + R.raw.sample));

            }
        });
    }

    private class PlayAudioTask extends AsyncTask<String, Void, Void> {

        private MediaPlayer mediaPlayer;

        Context context;

        public  PlayAudioTask(Context context){
            this.context=context;


        }
        @Override
        protected Void doInBackground(String... strings) {

            mediaPlayer = MediaPlayer.create(context,R.raw.samp);
            mediaPlayer.setLooping(true);
            mediaPlayer.start();
            return null;
        }

        @Override
        protected void onPostExecute(Void aVoid) {
            if (mediaPlayer != null) {
                mediaPlayer.start();
```

```
        }
    }

    @Override
    protected void onCancelled() {
        if (mediaPlayer != null) {
            mediaPlayer.release();
        }
    }
}

private void playVideo(Uri videoUrl) {


    videoView.setVideoURI(videoUrl);

    MediaController mediaController = new MediaController(this);
    videoView.setMediaController(mediaController);
    mediaController.setAnchorView(videoView);

    videoView.start();
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {

    getMenuInflater().inflate(R.menu.menuitem,menu);
    return super.onCreateOptionsMenu(menu);
}

void popup(View view)
{
    PopupMenu pm =new PopupMenu(getApplicationContext(),view);
    pm.inflate(R.menu.menuitem);
    pm.show();
}
}
```

menuitem.xml

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

</menu>
```

Menuitem1.xml

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

</menu>
```

15. Write an Android project to create app bar and create option menus and popup menu.

Main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/button"
        android:text="Click"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Main.java

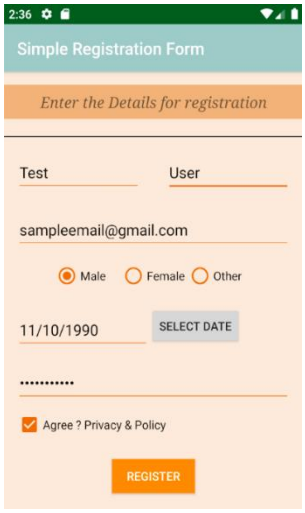
```
    package com.example.mad15;
    import androidx.appcompat.app.AppCompatActivity;
    import android.os.Bundle;
    import android.view.MenuItem;
    import android.view.View;
    import android.widget.Button;
```

```
    import android.widget.PopupMenu;
    import android.widget.Toast;
    public class MainActivity extends AppCompatActivity {
    Button button;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    button = (Button) findViewById(R.id.button);
    button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
    //Creating the instance of PopupMenu
    PopupMenu popup = new PopupMenu(MainActivity.this, button);
    //Inflating the Popup using xml file
    popup.getMenuInflater().inflate(R.menu.popup_menu,
    popup.getMenu());
    //registering popup with OnMenuItemClickListener
    popup.setOnMenuItemClickListener(new
    PopupMenu.OnMenuItemClickListener() {
    public boolean onMenuItemClick(MenuItem item) {
    Toast.makeText(MainActivity.this,"You Clicked : " +
    item.getTitle(), Toast.LENGTH_SHORT).show();
    return true;
    }
    });
    popup.show();//showing popup menu
    }
    });//closing the setOnClickListener method
    }
    }
```

popup_menu.xml(res/Menu/popup_menu)

```
    <?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/one"
        android:title="One" />
    <item
        android:id="@+id/two"
        android:title="Two"/>
    <item
        android:id="@+id/three"
        android:title="Three"/>
</menu>
```

18 Implement an android project for registration form use even handling for display data of registration form



Activity_main.xml

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

    <EditText
        android:id="@+id/et_name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Name"
        android:minHeight="48dp" />

    <EditText
        android:id="@+id/et_username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"
        android:minHeight="48dp" />

    <EditText
        android:id="@+id/et_email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"
        android:minHeight="48dp" />

    <RadioGroup
        android:id="@+id/rg_gender"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <RadioButton
            android:id="@+id/rb_male"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="Male" />

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

        <RadioButton
            android:id="@+id/rb_other"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Other" />

    </RadioGroup>

    <EditText
        android:id="@+id/et_dob"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Date of Birth (DD/MM/YYYY)"
        android:inputType="date"
        android:minHeight="48dp" />

    <Button
        android:id="@+id/btn_register"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:text="Register" />

    <TextView
        android:id="@+id/tv_register_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="16dp"
        android:text="" />

</LinearLayout>
```

Mainactivity.java

```
package com.example.valid;

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

public class MainActivity extends AppCompatActivity {
    private EditText etName;
    private EditText etUsername;
    private EditText etEmail;
    private RadioGroup rgGender;
```

```
    private EditText etDob;
    private Button btnRegister;
    private TextView tvRegisterResult;

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

        etName = findViewById(R.id.et_name);
        etUsername = findViewById(R.id.et_username);
        etEmail = findViewById(R.id.et_email);
        rgGender = findViewById(R.id.rg_gender);
        etDob = findViewById(R.id.et_dob);
        btnRegister = findViewById(R.id.btn_register);
        tvRegisterResult = findViewById(R.id.tv_register_result);

        btnRegister.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String name = etName.getText().toString();
                String username = etUsername.getText().toString();
                String email = etEmail.getText().toString();

                String gender = "";
                int checkedRadioButtonId =
rgGender.getCheckedRadioButtonId();
                if (checkedRadioButtonId == R.id.rb_male) {
                    gender = "Male";
                } else if (checkedRadioButtonId == R.id.rb_female) {
                    gender = "Female";
                } else if (checkedRadioButtonId == R.id.rb_other) {
                    gender = "Other";
                }

                String dob = etDob.getText().toString();

                // Perform registration logic here
                // For example, you can use the following code to display
the input data:
                tvRegisterResult.setText("Name: " + name + "\nUsername: " +
username + "\nEmail: " + email + "\nGender: " + gender + "\nDate of Birth:
" + dob);
            }
        });
    }
}
```