Practical 10: Playing Video

Steps: Create "raw" folder in res - > put a video in it (with .mp4 extension) -> then write the following codes

Activity_main.xml

```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">


    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <VideoView
            android:id="@+id/videoView"
            android:layout_width="match_parent"
            android:layout_height="258dp" />

        <Button
            android:id="@+id/button"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Play Video" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```java
package com.example.test;

import android.net.Uri;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.MediaController;
import android.widget.VideoView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    VideoView videoView;
    Button btn;

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

        videoView = findViewById(R.id.videoView);
        btn = findViewById(R.id.button); // Ensure button is initialized

        setupVideoView();

        // Set button click listener
        btn.setOnClickListener(this::playVideo);
    }

    private void setupVideoView() {
        MediaController mediaController = new MediaController(this);
        mediaController.setAnchorView(videoView);
        videoView.setMediaController(mediaController);
        videoView.setOnPreparedListener(mp -> mediaController.show(0)); // Force show controls
    }

    public void playVideo(View view) {
        try {
            Uri uri = Uri.parse("android.resource://" + getPackageName() + "/" + R.raw.myvideo);
```
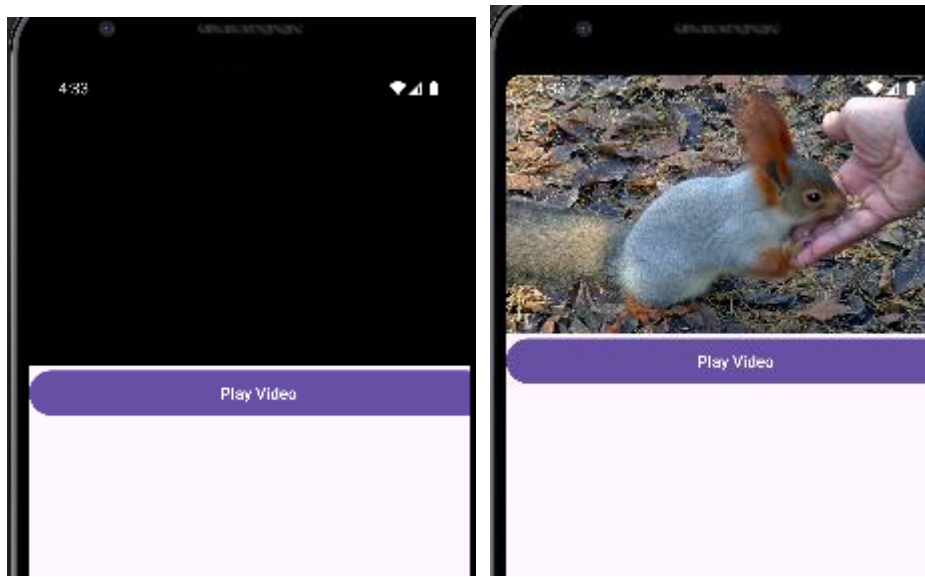
```
            videoView.setVideoURI(uri);
            videoView.requestFocus();
            videoView.setOnPreparedListener(mp -> videoView.start());
        } catch (Exception e) {
            Log.e("Error", "Video playback failed", e);
        }
    }
} // Ensure this closing brace is present for the class
```

Output:

Practical 9: Login Example

Create an android application that displays a login form with text for username and password and button for submit and reset. On submitting, toast should be displayed accordingly i.e. "Correct username and password" if username and password match and "Incorrect username/password" if username and password do not match.

(Note: Can change design accordingly.)

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"
```

```xml
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">


<RelativeLayout
    android:layout_width="409dp"
    android:layout_height="729dp"
    tools:layout_editor_absoluteX="1dp"
    tools:layout_editor_absoluteY="28dp">

    <TextView
        android:id="@+id/LoginExample"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#ED0101"
        android:fontFamily="sans-serif-condensed"
        android:text="Login Example"
        android:textAlignment="center"
        android:textColor="#FFFFFF"
        android:textSize="34sp" />

    <EditText
        android:id="@+id/Username"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/LoginExample"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:ems="10"
        android:focusable="true"
        android:hint="Enter Username"
        android:inputType="text" />

    <EditText
        android:id="@+id/Password"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/Username"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
```

```xml
        android:layout_alignParentEnd="@+id/Username"
        android:layout_alignParentRight="@+id/Username"
        android:ems="10"
        android:hint="Password"
        android:inputType="textPassword" />

    <TextView
        android:id="@+id/AttemptsRemaining"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/Password"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:maxHeight="35dp"
        android:text="Attempts Remaining"
        android:textSize="20sp" />

    <TextView
        android:id="@+id/num"

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@id/AttemptsRemaining"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_marginStart="97dp"
        android:layout_marginLeft="97dp"
        android:layout_marginEnd="-281dp"
        android:layout_marginRight="-281dp"
        android:layout_marginBottom="-59dp"
        android:textSize="34sp" />

    <Button
        android:id="@+id/submit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentBottom="true"
        android:text="Submit" />

    <Button
        android:id="@+id/cancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
```

```
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:text="Cancel" />
    </RelativeLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```java
package com.example.p9;

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

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 {
    Button b1, b2;
    EditText e1, e2;
    TextView tx1;

    int counter = 3;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);

        e1 = (EditText)findViewById(R.id.Username);
        e2 = (EditText)findViewById(R.id.Password);
        b1 = (Button)findViewById(R.id.submit);
        b2 = (Button)findViewById(R.id.cancel);
        tx1 = (TextView)findViewById(R.id.num);
```

```java
    b1.setOnClickListener(v -> {
        if (e1.getText().toString().equals("admin") && e2.getText().toString().equals("admin")) {
            Toast.makeText(getApplicationContext(), " Correct username and password.",
Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(getApplicationContext(), "Wrong Credentials",
Toast.LENGTH_SHORT).show();
            tx1.setVisibility(View.VISIBLE);
            tx1.setBackgroundColor(Color.RED);
            tx1.setText(String.valueOf(--counter));

            if (counter == 0) {
                b1.setEnabled(false);
            }
        }
    });
  }
}
```

Output:
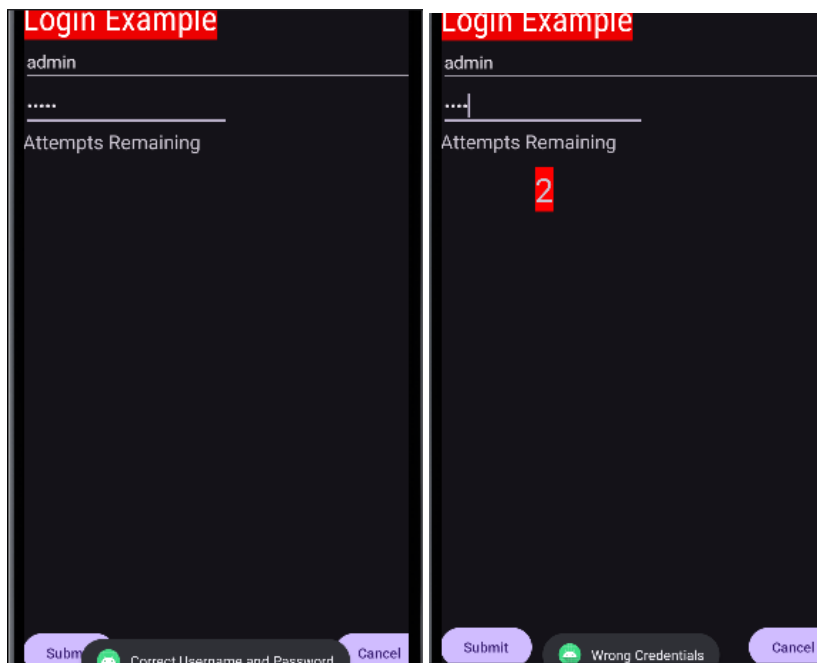
Create an android application that opens the website www.google.com in the browser on the click of a button using intents.

Activity_main.xml

```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">


    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <EditText
            android:id="@+id/enterurl"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="40dp"
            android:ems="10"
            android:hint="enter url"
            android:inputType="text" />

        <Button
            android:id="@+id/submit"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="GO"
            android:textAppearance="@style/TextAppearance.AppCompat.Large" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```java
package com.example.practical8;

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

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 {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
        Button runWebsite = findViewById(R.id.submit);
        EditText website = findViewById(R.id.enterurl);

        runWebsite.setOnClickListener(view -> {
            String url = website.getText().toString().trim();

            if (url.isEmpty()) {
                Toast.makeText(this, "Please enter a URL", Toast.LENGTH_SHORT).show();
                return;
            }

            if (!url.startsWith("http://") && !url.startsWith("https://")) {
                url = "https://" + url;
            }

            startActivity(new Intent(Intent.ACTION_VIEW, Uri.parse(url)));
```

```
        });

    }
}
```

Output:
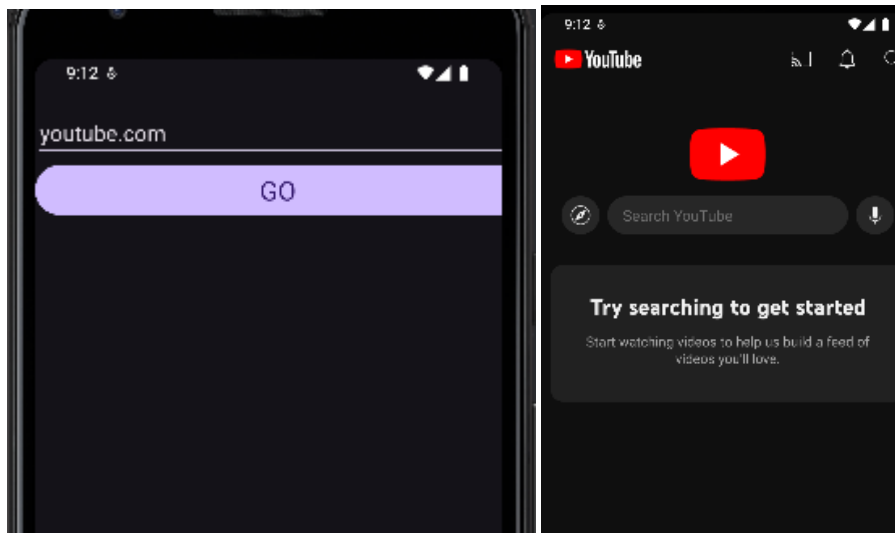
Practical 7: Create an android application to generate notification

Note: Add this permission inside the <manifest> tag to handle notifications for Android 13+.

AndroidManifest.xml

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

MainActivity.java

```java
package com.example.practical7;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.content.Intent;
import android.content.pm.PackageManager;
```

```java
import android.os.Build;
import android.os.Bundle;
import android.provider.Settings;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationManagerCompat;

public class MainActivity extends AppCompatActivity {
    private static final String CHANNEL_ID = "MyNotification";
    private static final int NOTIFICATION_ID = 999, PERMISSION_REQUEST = 101;

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

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.TIRAMISU &&
                checkSelfPermission(android.Manifest.permission.POST_NOTIFICATIONS) !=
PackageManager.PERMISSION_GRANTED) {
            requestPermissions(new String[]{android.Manifest.permission.POST_NOTIFICATIONS},
PERMISSION_REQUEST);
        } else {
            sendNotification();
        }
    }

    private void sendNotification() {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O)
            getSystemService(NotificationManager.class).createNotificationChannel(
                    new NotificationChannel(CHANNEL_ID, "Simple Notification",
NotificationManager.IMPORTANCE_HIGH));

        NotificationManagerCompat manager = NotificationManagerCompat.from(this);
        if (manager.areNotificationsEnabled()) {
            manager.notify(NOTIFICATION_ID, new NotificationCompat.Builder(this, CHANNEL_ID)
                    .setContentTitle("Notification from AMP")
                    .setSmallIcon(android.R.drawable.ic_dialog_info)
                    .setAutoCancel(true)
                    .setPriority(NotificationCompat.PRIORITY_HIGH)
                    .setContentText("Small Notification").build());
        } else {
            Toast.makeText(this, "Enable notifications in settings!", Toast.LENGTH_SHORT).show();
            startActivity(new Intent(Settings.ACTION_APP_NOTIFICATION_SETTINGS)
                    .putExtra(Settings.EXTRA_APP_PACKAGE, getPackageName()));
```
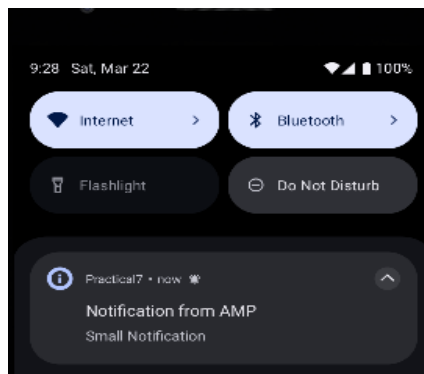
```
    }
  }

  @Override
  public void onRequestPermissionsResult(int requestCode, String[] permissions, int[]
grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == PERMISSION_REQUEST && grantResults.length > 0 && grantResults[0]
== 0) {
      sendNotification();
    } else {
      Toast.makeText(this, "Permission denied!", Toast.LENGTH_SHORT).show();
    }
  }
}
```

Output:

Practical 6: Create an android application for the following menu items ,the appropriate toast should appear by clicking on the item:

• Settings

• Search

• Compose Email

• FeedBack

(make Compose Email item disabled)

Note: Create a new folder in "res" named "menu" -> Add a new xml file in menu folder named "menu.xml"

Menu.xml

```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">

  <item
    android:id="@+id/settings"
    android:title="Settings" />

  <item
    android:id="@+id/search"
    android:title="Search" />

  <item
    android:id="@+id/email"
    android:title="Compose Email"
    android:enabled="false" />

  <item
    android:id="@+id/feedback"
    android:title="Feedback" />
</menu>
```

Activity_main.xml

```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:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
    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="match_parent"
            android:layout_height="?attr/actionBarSize"
            android:background="?attr/colorPrimary"
            app:title="My App"
            app:layout_constraintTop_toTopOf="parent"/>

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

MainActivity.java

```java
package com.example.practical6;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;
import androidx.appcompat.widget.Toolbar;
import androidx.activity.EdgeToEdge;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity
{

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

    @Override
    public boolean onCreateOptionsMenu(Menu menu)
    {

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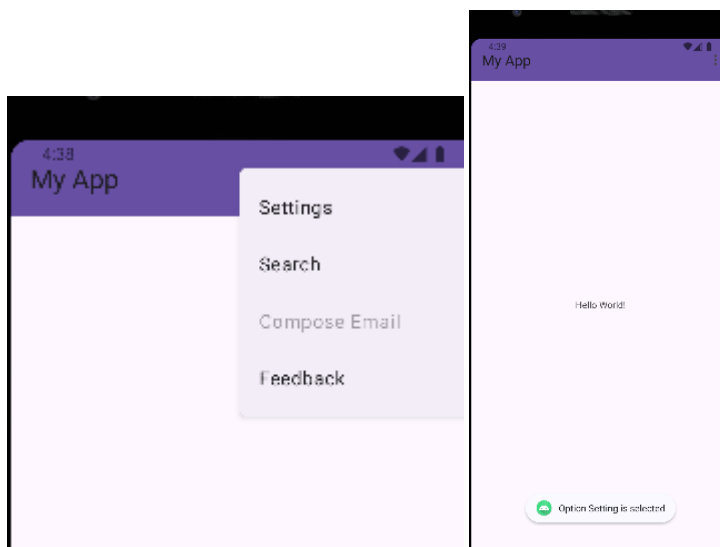
```
    }

    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.settings) {
            Toast.makeText(this, "Option Setting is selected", Toast.LENGTH_SHORT).show();
        } else if (id == R.id.email) {
            Toast.makeText(this, "Option Compose Email is selected", Toast.LENGTH_SHORT).show();
        } else {
            return super.onOptionsItemSelected(item);
        }
        return true;
    }
}
```

Output:

a) Demonstration of Application Bar

Note: make toolbar.xml in layout folder in res

toolbar.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.Toolbar
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/black"
    android:elevation="4dp"
    >
</androidx.appcompat.widget.Toolbar>
```

activity_main.xml

```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">

    <RelativeLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

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

```xml
    <include android:id="@+id/toolbar"
      layout="@layout/toolbar" />

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

MainActivity.java

```java
package com.example.pract5a;

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 androidx.appcompat.widget.Toolbar;

public class MainActivity extends AppCompatActivity {

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

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

  }
}
```
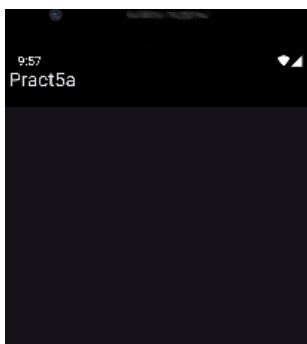
Output:

Practical 5b:

Demonstration of UI Component (TextView, EditText, Button)

Activity_main.xml

```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">

    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <EditText
            android:id="@+id/et1"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="40dp"
            android:ems="10"
            android:inputType="text"
            android:text="Number 1" />

        <EditText
            android:id="@+id/et2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="text"
            android:text="Number 2" />

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

```xml
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Addition" />

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

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

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

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

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="" />
  </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```java
package com.example.practical5b;

import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

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 {

    Button b1, b2, b3, b4, b5;
    EditText e1, e2;
    TextView tx1;

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

        e1 = (EditText)findViewById(R.id.et1);
        e2 = (EditText)findViewById(R.id.et2);
        b1 = (Button)findViewById(R.id.add);
        b2 = (Button)findViewById(R.id.sub);
        b3 = (Button)findViewById(R.id.mult);
        b4 = (Button)findViewById(R.id.div);
        b5 = (Button)findViewById(R.id.clear);

        tx1 = (TextView)findViewById(R.id.result);


        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v)
```

```java
        {
            String s1 = e1.getText().toString();
            String s2 = e2.getText().toString();

            int n1 = Integer.parseInt(s1);
            int n2 = Integer.parseInt(s2);

            int sum = n1 + n2;

            tx1.setText("Addition:" + sum);
        }
    }
);


b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v)
            {
                String s1 = e1.getText().toString();
                String s2 = e2.getText().toString();

                int n1 = Integer.parseInt(s1);
                int n2 = Integer.parseInt(s2);

                int sub = n1 - n2;

                tx1.setText("Subtraction: " + sub);
            }
        }
);


b3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v)
            {
                String s1 = e1.getText().toString();
                String s2 = e2.getText().toString();

                int n1 = Integer.parseInt(s1);
                int n2 = Integer.parseInt(s2);

                int mul = n1 * n2;

                tx1.setText("Multiplication:" + mul);
```

```java
                }
            }
        );

        b4.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View v)
                {
                    String s1 = e1.getText().toString();
                    String s2 = e2.getText().toString();

                    int n1 = Integer.parseInt(s1);
                    int n2 = Integer.parseInt(s2);

                    int div = n1 / n2;

                    tx1.setText("Division: " + div);
                }
            }
        );

        b5.setOnClickListener(new View.OnClickListener() { @Override
        public void onClick(View v) {

            e1.setText(" ");
            e2.setText(" ");
            tx1.setText(" ");

        }
        });
    }
}
```

Output:
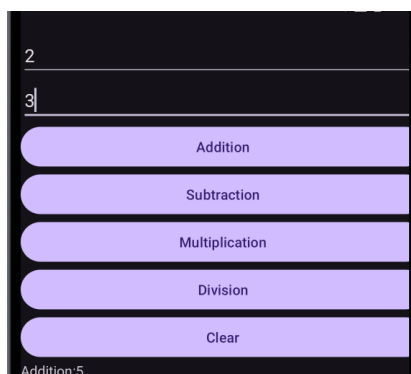
Practical 4:

Programs related to different Layouts Linear, Relative, Table

a) Linear Layout (Vertical)
b) Linear Layout (Horizontal)

Note:

The Same Code is used for both Linear Layout Vertical and Linear Layout Horizontal. Just Change One Word in the below line. Horizontal to Vertical and Vice Versa
android:orientation="horizontal"

Note 2: Make Sure all layouts are wrap_content, like given below

android:layout_width="wrap_content"
android:layout_height="wrap_content"

activity_main.xml

```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">

    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="horizontal"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <EditText
```

```
        android:id="@+id/editTextText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="text"
        android:text="Enter Name" />

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="1" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="2" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="3" />
  </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```
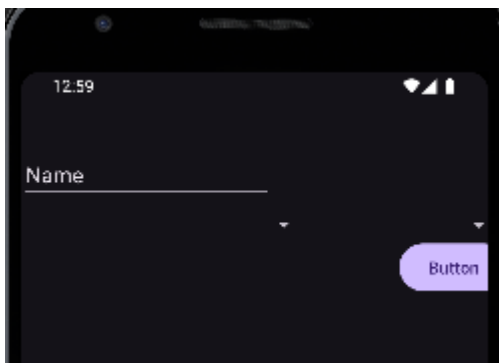
Output:

c) Relative Layout

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

    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="horizontal"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <EditText
            android:id="@+id/editTextText"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="text"
            android:text="Enter Name" />

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

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

```
        android:layout_height="wrap_content"
        android:text="1" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="2" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="3" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:

d) Table Layout

activity_main.xml

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

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


    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="50dp">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Row1, Column1"
            android:textSize="24sp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Row1, Column2"
            android:textSize="24sp" />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="70dp">

        <TextView
            android:layout_width="231dp"
            android:layout_height="wrap_content"
            android:text="Row2, Column1"
            android:textSize="24sp" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Row2, Column2"
            android:textSize="24sp" />
    </TableRow>


</TableLayout>
```
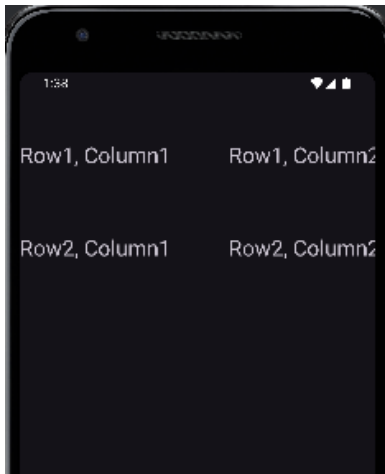
==**Programming Activities and fragments Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.**==

a) **Activity Life Cycle**

MainActivity.java

```java
package com.example.practical3a;
import android.os.Bundle;
import android.util.Log;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;


public class MainActivity extends AppCompatActivity {

  String tag= "Android Lifecycle Demonstration";

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_main);
    Log.d(tag, "In onCreate() Event");

  }
```

```java
    public void onStart()
    {
        super.onStart();
        Log.d(tag, "In onStart() Event");
    }

    public void onRestart()
    {
        super.onRestart();
        Log.d(tag, "In onRestart() Event");
    }

public void onPause()
{
    super.onPause();
    Log.d(tag, "In onPause()Event");
}

public void onStop()
{
    super.onStop();
    Log.d(tag, "In onStop() Event");
}

public void onDestroy()
{
    super.onDestroy();
    Log.d(tag, "In onDestroy() Event");
}

}
```

Output:

b) Life Cycle of fragments

Note: The Following Steps are required: Right Click on Java -> New -> Fragment ->
Fragment (Blank)
Give Fragment Name "LifeCycleOfFragment"
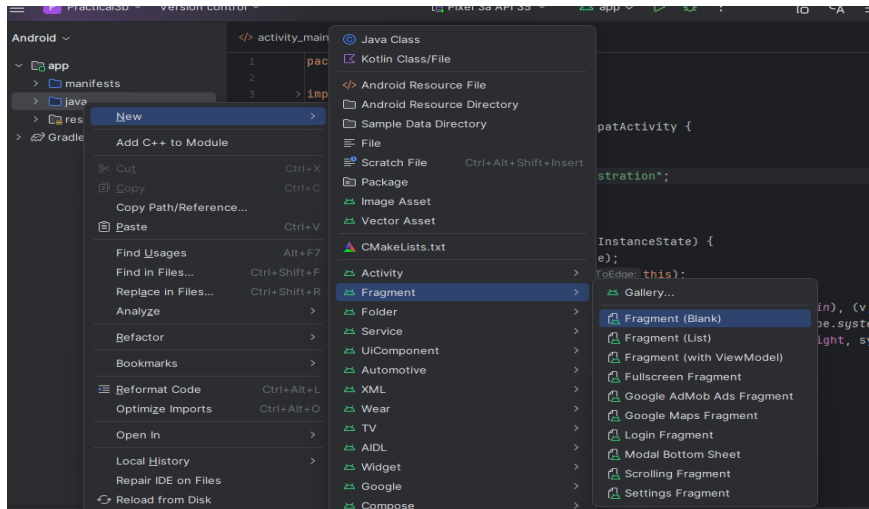
LifeCycleOfFragment.java

package com.example.practical3b;

import android.os.Bundle;
import androidx.fragment.app.Fragment;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

/**
 * A simple {@link Fragment} subclass.
 * Use the {@link LifeCycleOfFragment#newInstance} factory method to
 * create an instance of this fragment.
 */
public class LifeCycleOfFragment extends Fragment {

    private static final String TAG = "FragmentLifecycle";

//   private static final String ARG_PARAM1 = "param1";
//   private static final String ARG_PARAM2 = "param2";
//

```java
//    // TODO: Rename and change types of parameters
//    private String mParam1;
//    private String mParam2;


    public LifeCycleOfFragment() {
        // Required empty public constructor
    }

//    public static LifeCycleOfFragment newInstance(String param1, String param2) {
//        LifeCycleOfFragment fragment = new LifeCycleOfFragment();
//        Bundle args = new Bundle();
//        args.putString(ARG_PARAM1, param1);
//        args.putString(ARG_PARAM2, param2);
//        fragment.setArguments(args);
//        return fragment;
//    }

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        if (getArguments() != null) {
            Log.d(TAG, "onCreate() called");
        }
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                 Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        Log.d(TAG, "onCreateView() called");
        return inflater.inflate(R.layout.fragment_life_cycle_of, container, false);
    }

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

    @Override
    public void onPause()
    {
        super.onPause();
```

```java
      Log.d(TAG, "onPause() called");
    }

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

    @Override
    public void onStop() {
      super.onStop();
      Log.d(TAG, "onStop() called");
    }
    @Override
    public void onDestroyView() {
      super.onDestroyView();
      Log.d(TAG, "onDestroyView() called");
    }
    @Override
    public void onDestroy() {
      super.onDestroy();
      Log.d(TAG, "onDestroy() called");
    }

}
```

activity_main.xml

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

```xml
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <FrameLayout
        android:id="@+id/fragment_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

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

MainActivity.java

```java
package com.example.practical3b;

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 androidx.fragment.app.FragmentManager;
import androidx.fragment.app.FragmentTransaction;

public class MainActivity extends AppCompatActivity {

    String tag= "Android Lifecycle Demonstration";


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

        if(savedInstanceState == null)
        {
            FragmentManager fragmentManager = getSupportFragmentManager();
            FragmentTransaction fragmentTransaction = fragmentManager.beginTransaction();
            fragmentTransaction.replace(R.id.fragment_container, new LifeCycleOfFragment());
            fragmentTransaction.commit();
```
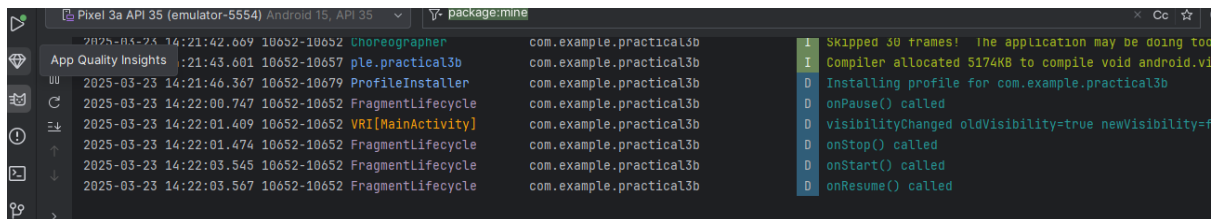
```
      }
    }
}
```

Output:

<mark>Practical 2</mark>

<mark>Programming Resources Android Resources: (Color, Theme, String, Drawable, Dimension, Image)</mark>

a) Defining Color Property

Color.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="red">#FB0808</color>
    <color name="blue">#2196F3</color>
    <color name="green">#8BC34A</color>
    <color name="yellow">#E5FF00</color>

</resources>
```

activity_main.xml

```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">


    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:orientation="vertical"
        tools:layout_editor_absoluteX="1dp"
        tools:layout_editor_absoluteY="1dp">

        <TextView
            android:id="@+id/red"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/red"
            android:text="Red" />

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

        <TextView
            android:id="@+id/yellow"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/yellow"
            android:text="yellow" />
```
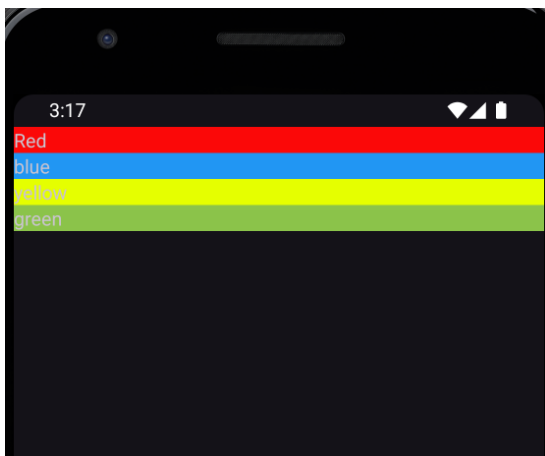
```xml
    <TextView
        android:id="@+id/green"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/green"
        android:text="green" />
  </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:

b) Defining Theme Property

Note: Go to res -> values -> create Styles.xml -> Put Below Code

Styles.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<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>
    <item name="android:background">#FFFFFF</item>
    <item name="android:textColor">#000000</item>
```
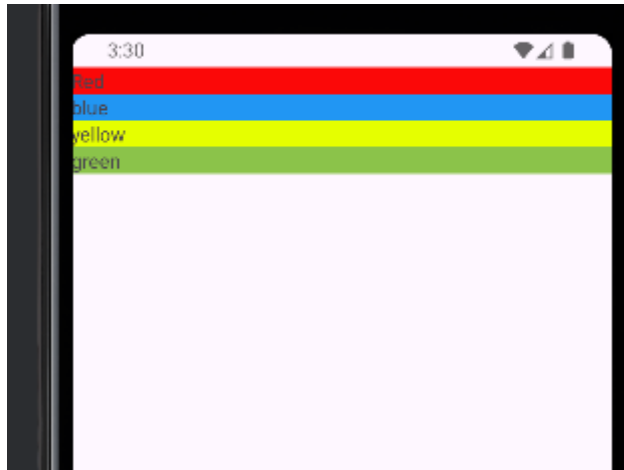
```
    </style>
</resources>
```

Activity_main.xml

(keep it same as above one [2a])

Output:

c) Defining string property

String.xml

```
<resources>
    <string name="app_name">Practical2b</string>
    <string name="Heading">Life SUCKS!</string>
    <string name="description">Performing Practicals for Exam. I am extremely bored and tired.
    I need my holidays. When will I get my vacation? I am tired once again. I am tired x 1000000
    Please someone rescue me!!!! HAHAHHAHAHAHAHAHHA</string>
</resources>
```

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

```xml
<LinearLayout
  android:layout_width="409dp"
  android:layout_height="729dp"
  android:orientation="vertical"
  tools:layout_editor_absoluteX="1dp"
  tools:layout_editor_absoluteY="1dp">

  <TextView
    android:id="@+id/red"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/red"
    android:text="Red" />

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

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

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

  <TextView
    android:id="@+id/name"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/purple"
    android:text="@string/app_name" />

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

```
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/purple"
            android:text="@string/Heading" />

        <TextView
            android:id="@+id/description"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@color/purple"
            android:text="@string/description" />

    </LinearLayout>

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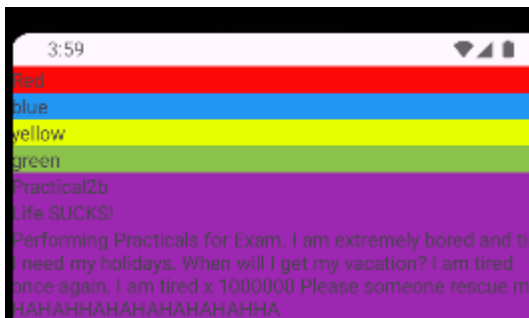
Output:

d) Adding images and dimensions

Note: For dimensions of Image, make changes using attributes of Image View

Activity_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">

    <LinearLayout
        android:layout_width="409dp"
        android:layout_height="729dp"
```

```xml
    android:orientation="vertical"
    tools:layout_editor_absoluteX="1dp"
    tools:layout_editor_absoluteY="1dp">

    <TextView
        android:id="@+id/red"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/red"
        android:text="Red" />

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

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

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

    <TextView
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/purple"
        android:text="@string/app_name" />

    <TextView
        android:id="@+id/heading"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/purple"
        android:text="@string/Heading" />
```

```xml
    <TextView
      android:id="@+id/description"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:background="@color/purple"
      android:text="@string/description" />

    <ImageView
      android:id="@+id/imageView"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      app:srcCompat="@drawable/cat" />
  </LinearLayout>

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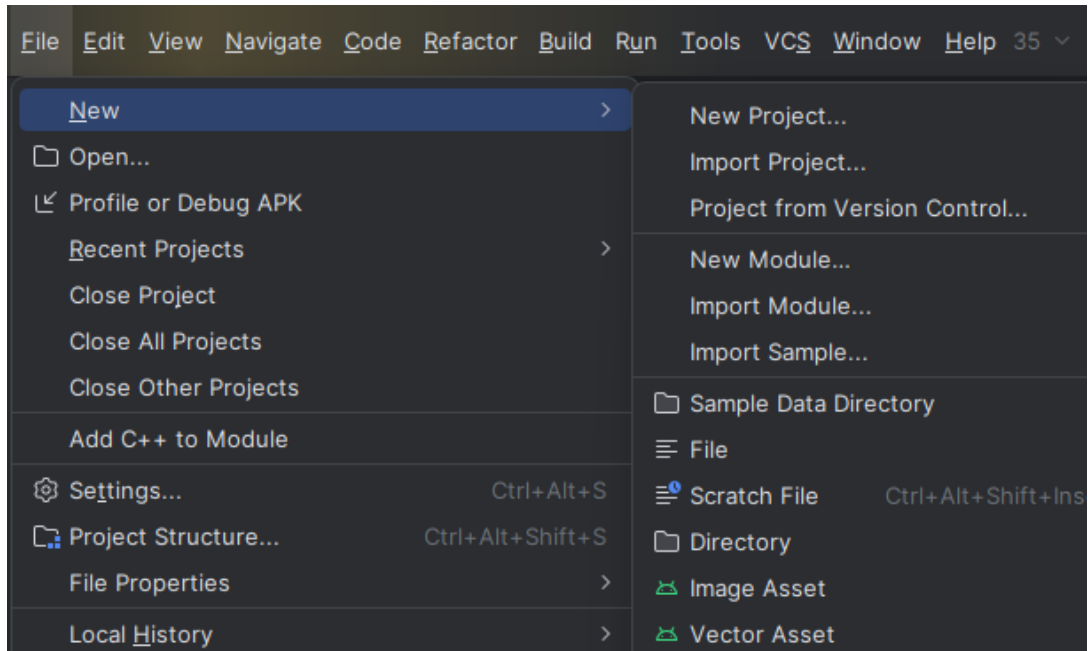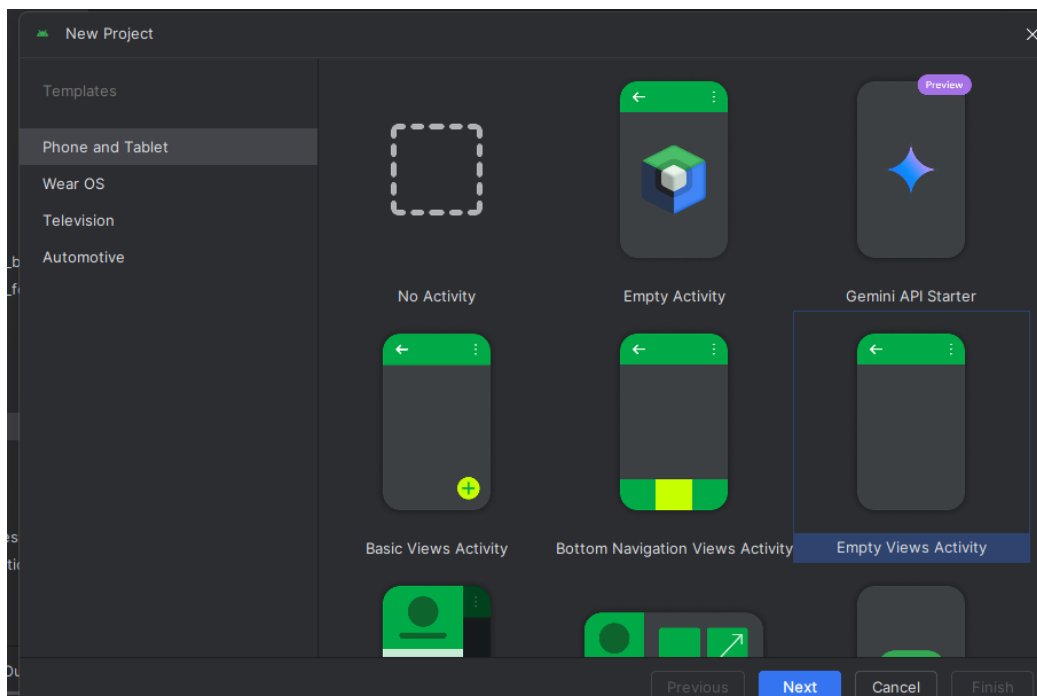
Output:

HOW TO PERFORM PRACTICALSSS!!!!!!!

File -> New -> New Project

Select [Empty Views Activity] -> Next

Give Name -> Select Minimum SDK as Kitkat Android 4.4 -> Rest all as it is -> Finish
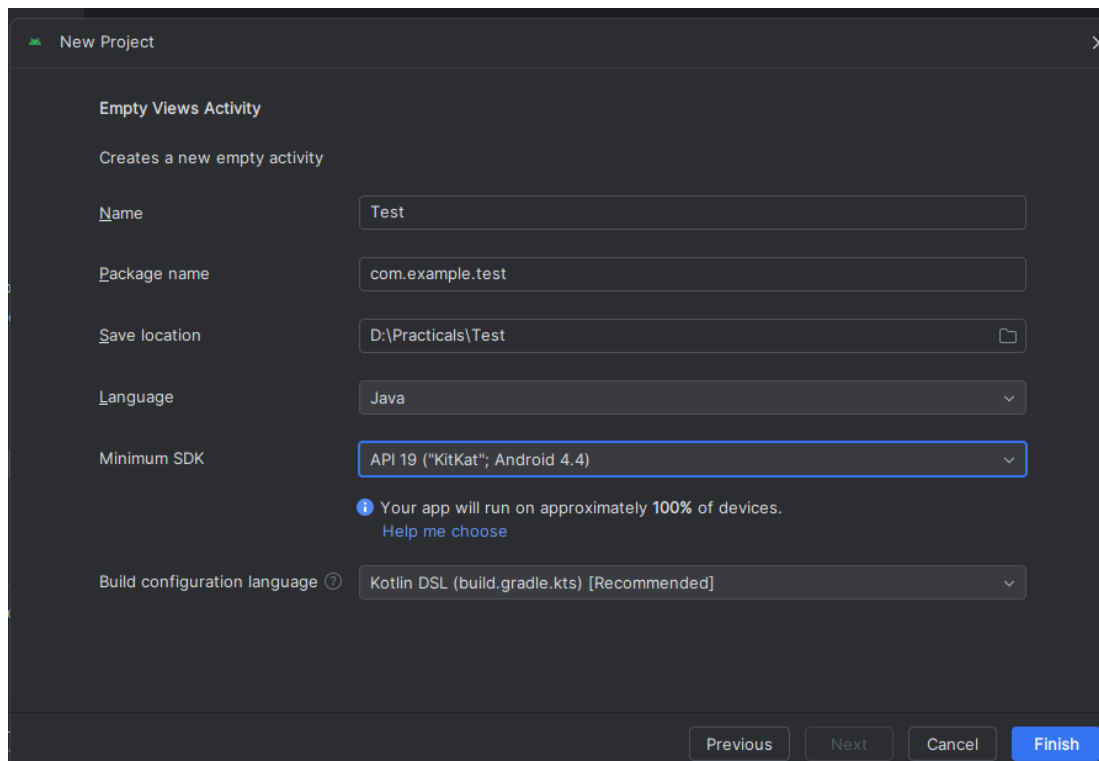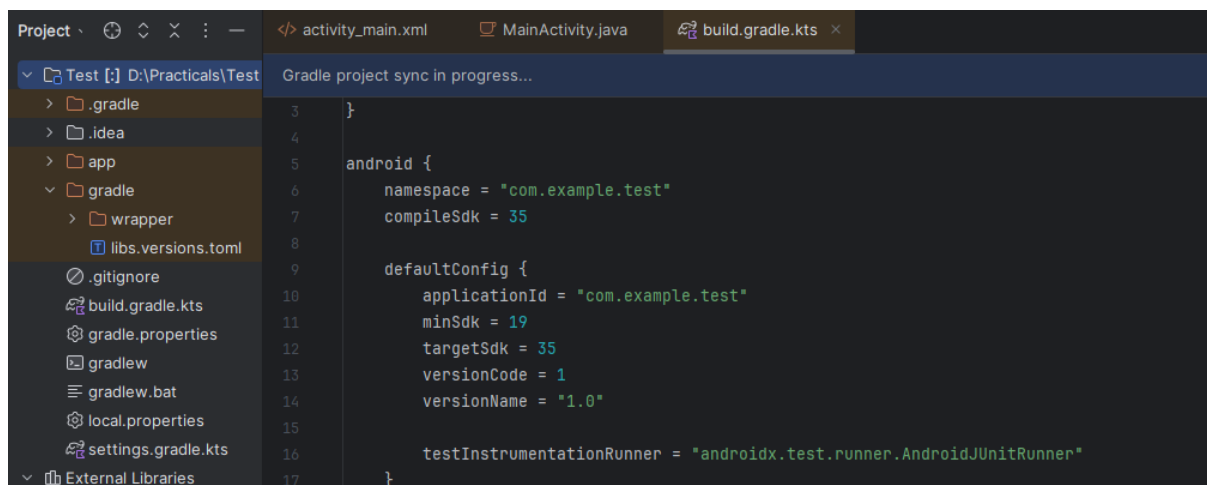

Also Note that:

In build.gradle.kts ->

Change minSdk -> 21

CompileSdk, targetSdk -> 35 (if its not already 35, sometimes it comes as 34 so change to 35)

After Making Changes -> Click on Sync Now