

# Android Application that uses GUI components, Font and Color.

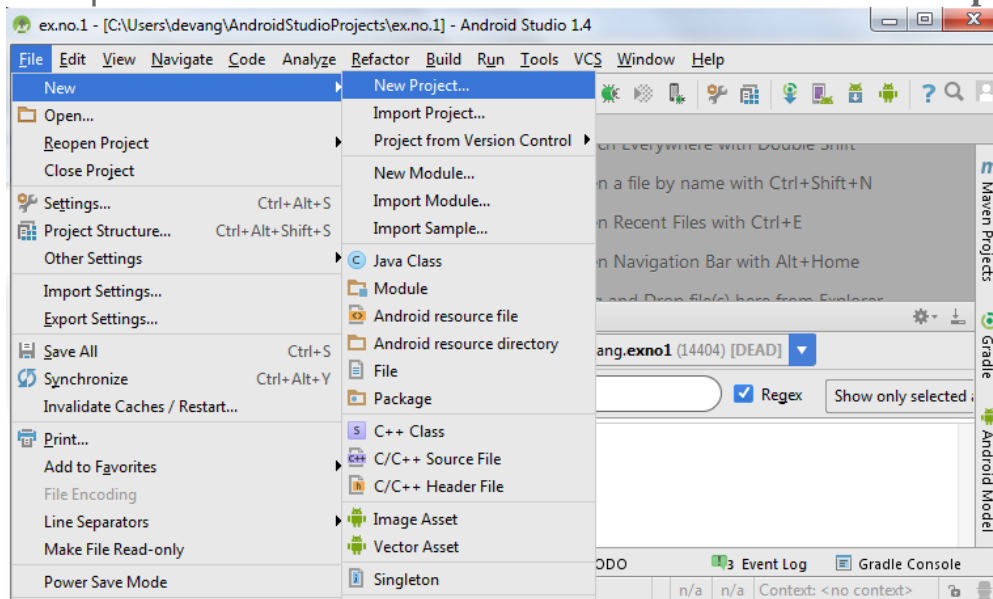
## Aim:

To develop a Simple Android [Application](#) that uses [GUI](#) components, Font and Colors.

## Procedure:

### Creating a New project:

- Open Android Studio and then click on **File -> New -> New project**.



- Then type the Application name as “**ex.no.1**” and click **Next**.

Create New Project

## New Project

Android Studio

### Configure your new project

Application name:

Company Domain:

Package name:  [Edit](#)

Project location:

The application name for most apps begins with an uppercase letter

[Previous](#) [Next](#) [Cancel](#) [Finish](#)

- Then select the **Minimum SDK** as shown below and click Next.

Create New Project

## Target Android Devices

### Select the form factors your app will run on

Different platforms may require separate SDKs

☒ Phone and Tablet  
Minimum SDK:

☐ Wear  
Minimum SDK:

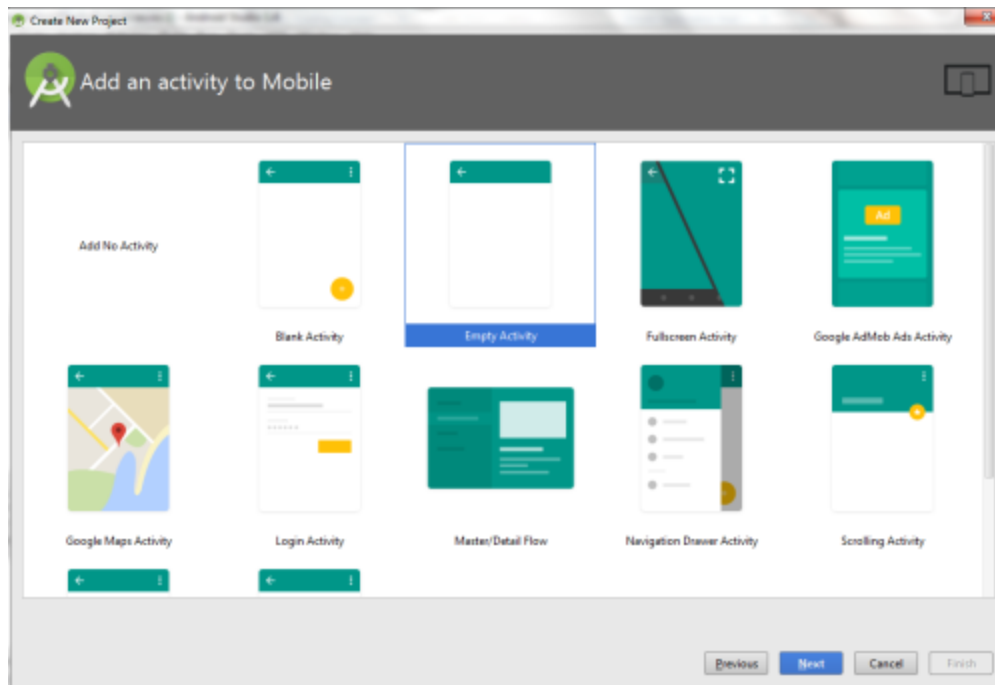
☐ TV  
Minimum SDK:

☐ Android Auto  
Minimum SDK:

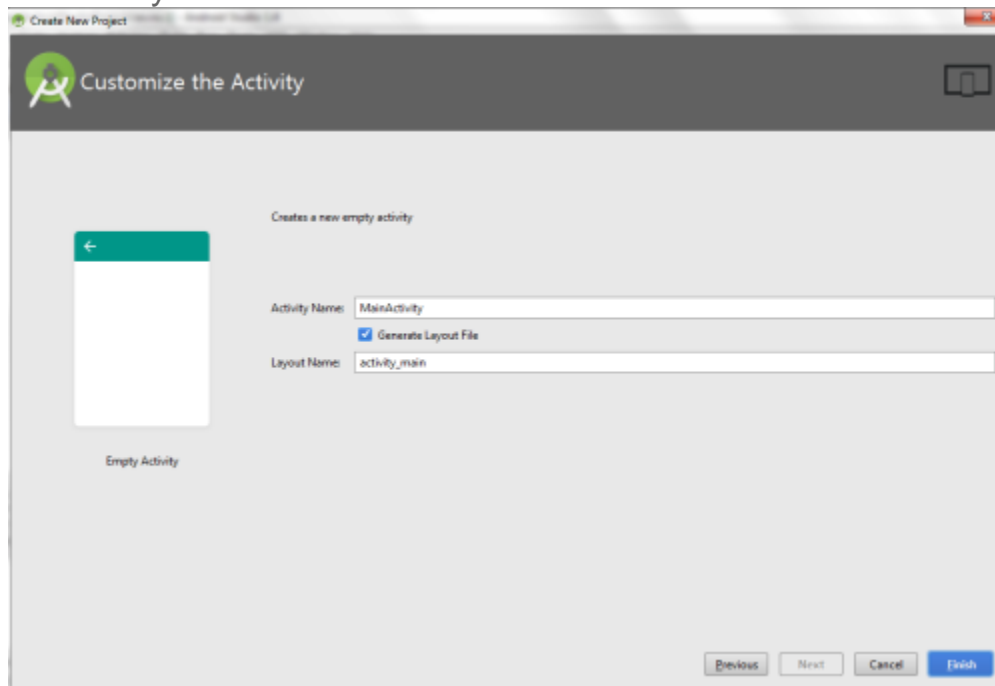
Lower API levels target more devices, but have fewer features available. By targeting API 15 and later, your app will run on approximately **94.0%** of the devices that are active on the Google Play Store.  
[Help me choose](#)

[Previous](#) [Next](#) [Cancel](#) [Finish](#)

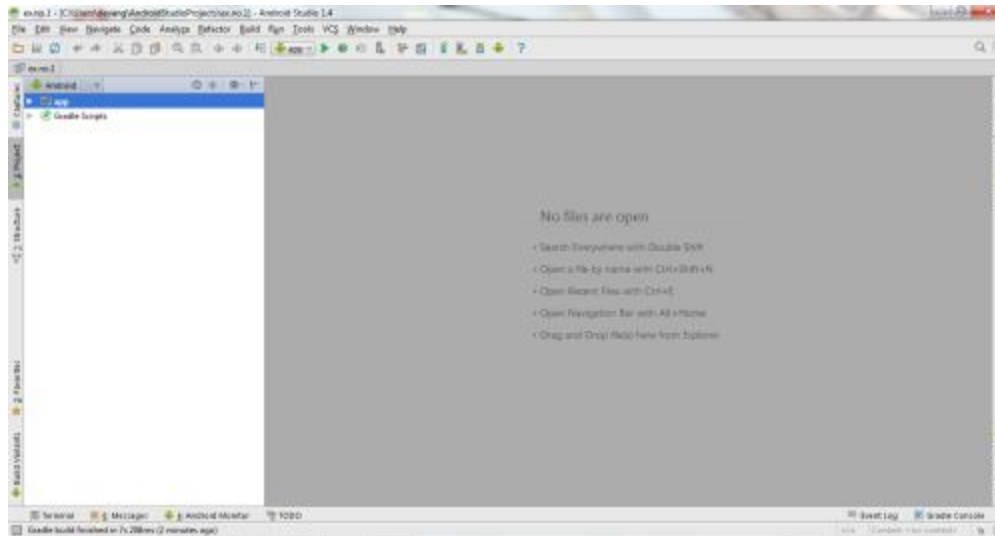
- Then select the **Empty Activity** and click Next.



- Finally click **Finish**.

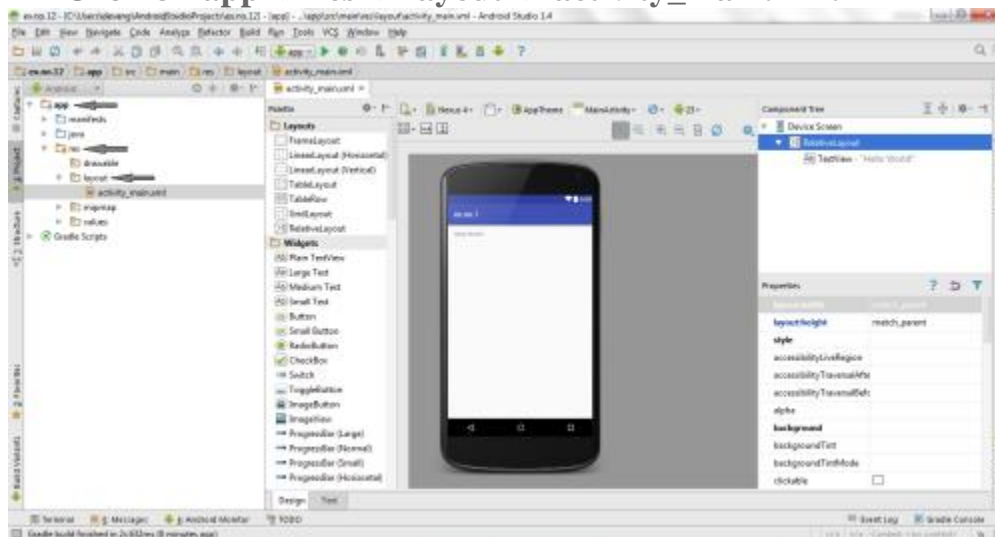


- It will take some time to build and load the project.
- After completion it will look as given below.

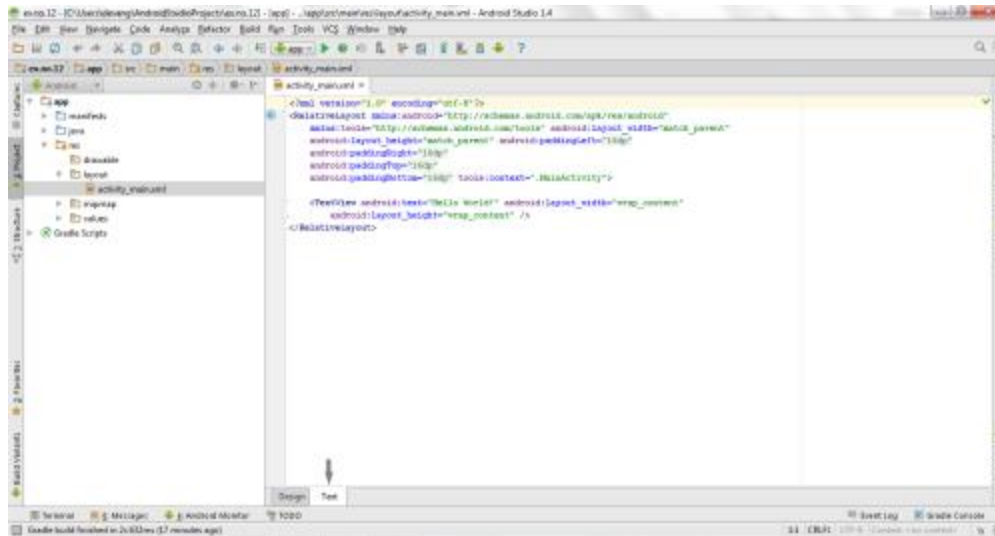


## Designing layout for the Android Application:

- Click on app -> res -> layout -> activity\_main.xml.



- Now click on Text as shown below.



- Then delete the code which is there and type the code as given below.

### Code for 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">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="30dp"
        android:gravity="center"
        android:text="Hello World!"
        android:textSize="25sp"
        android:textStyle="bold"
        tools:ignore="MissingConstraints"
        tools:layout_editor_absoluteX="187dp"
        tools:layout_editor_absoluteY="78dp" />

    <Button
        android:id="@+id/button1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20dp"
        android:gravity="center"
        android:text="Change font size"
        android:textSize="25sp"
        tools:ignore="MissingConstraints"
        tools:layout_editor_absoluteX="44dp"
```

```

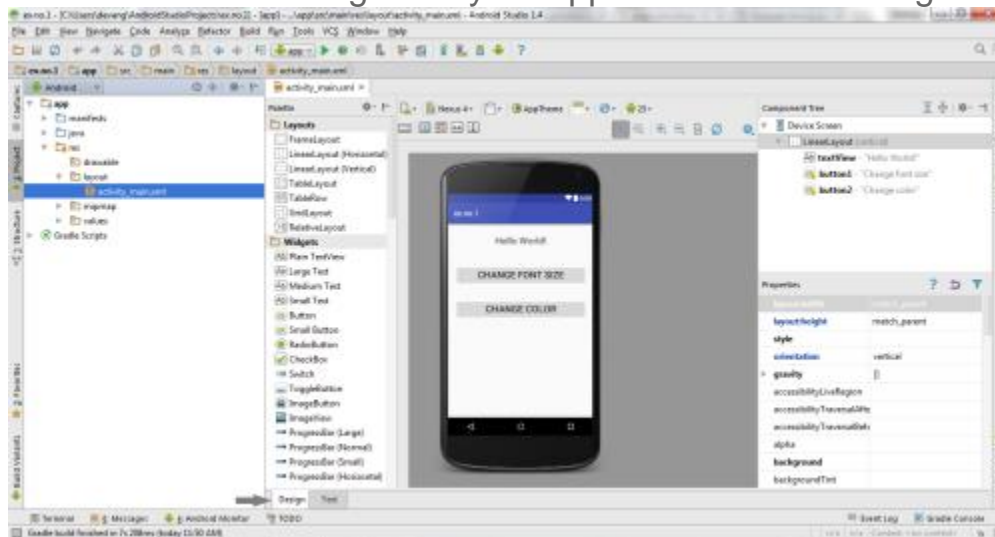
tools:layout_editor_absoluteY="353dp" />

<Button
    android:id="@+id/button2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp"
    android:gravity="center"
    android:text="Change color"
    android:textSize="25sp"
    tools:ignore="MissingConstraints"
    tools:layout_editor_absoluteX="37dp"
    tools:layout_editor_absoluteY="224dp" />

</androidx.constraintlayout.widget.ConstraintLayout>

```

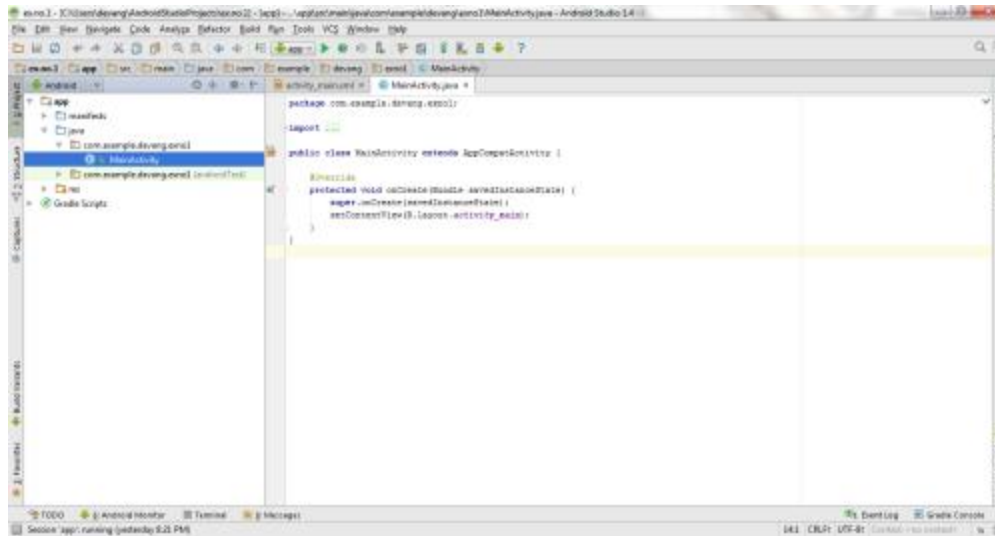
- Now click on Design and your application will look as given below.



- So now the designing part is completed.

## Java Coding for the Android Application:

- Click on **app** -> **java** -> **com.example-week\_2** -> **MainActivity**.



- Then delete the code which is there and type the code as given below.

### Code for MainActivity.java:

?

```
package com.example.week_2;

/*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;

public class MainActivity extends AppCompatActivity {

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

        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main),
        (v, insets) -> {
            Insets systemBars =
            insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top,
            systemBars.right, systemBars.bottom);
            return insets;
        });
    }
}
*/
import android.graphics.Color;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
```

```

import android.widget.TextView;

public class MainActivity extends AppCompatActivity
{
    int ch=1;
    float font=30;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final TextView t= (TextView) findViewById(R.id.textView);
        Button b1= (Button) findViewById(R.id.button1);
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                t.setTextSize(font);
                font = font + 5;
                if (font == 50)
                    font = 30;
            }
        });
        Button b2= (Button) findViewById(R.id.button2);
        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                switch (ch) {
                    case 1:
                        t.setTextColor(Color.RED);
                        break;
                    case 2:
                        t.setTextColor(Color.GREEN);
                        break;
                    case 3:
                        t.setTextColor(Color.BLUE);
                        break;
                    case 4:
                        t.setTextColor(Color.CYAN);
                        break;
                    case 5:
                        t.setTextColor(Color.YELLOW);
                        break;
                    case 6:
                        t.setTextColor(Color.MAGENTA);
                        break;
                }
                ch++;
                if (ch == 7)
                    ch = 1;
            }
        });
    }
}

```

---



- So now the Coding part is also completed.
- Now run the application to see the output.

## Output:

