

Practical 2**Programming Resources**

Android Resources: (Color, Theme, String, Drawable, Dimension, Image),

a) Defining Color Property.

1. Create a new project and go to:
ProjectName>App>src>main>res>values>**colors.xml**
2. Defining new color properties in **colors.xml**

Go to
ProjectName->app->res->values->colors.xml

Colors.xml

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

```
<resources>
```

```
<color name="colorPrimary">#008577</color>
```

```
<color name="colorPrimaryDark">#00574B</color>
```

```
<color name="colorAccent">#D81B60</color>
```

Default Content

```
// Define new colors here using hexadecimal color codes.
```

```
<color name="Red">#FF0000</color>
```

```
<color name="Blue">#0000FF</color>
```

```
<color name="Pink">#FF00FF</color>
```

```
<color name="Cyan">#00FFFF</color>
```

```
<color name="Grey">#AABBFF</color>
```

Add this code in the
colors.xml

```
</resources>
```

Now go to **activity_main.xml** and type the following code:

- Drag and drop LinearLayout(Vertical) From Layout tab in Palette window
- Drag and drop four TextView from Text tab in palette window
- The default Code of one TextView is as follows

```
<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="56dp"
    android:text="TextView" />
```

Now call the colors from the colors.xml as

```
android:background="@color/Red"
```

Now the TextView Code looks like

```
<TextView
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="56dp"
    android:text="TextView"

    android:background="@color/Red"
```

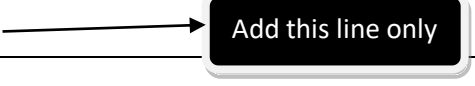
Now the final Code of **activity_main.xml** is

```
<?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">

    <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/textView"
            android:layout_width="match_parent"
            android:layout_height="56dp"
            android:text="TextView"

            android:background="@color/Red"
```



Add this line only

```
</>

<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="52dp"
    android:text="TextView"

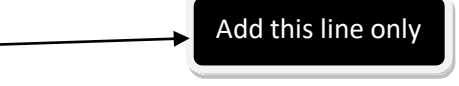
    android:background="@color/Grey"
/>

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="59dp"
    android:text="TextView"


    android:background="@color/Blue"
/>

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="51dp"
    android:text="TextView"

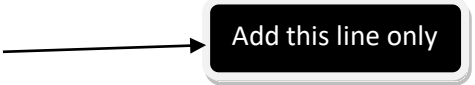
    android:background="@color/Cyan"
/>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```



Add this line only

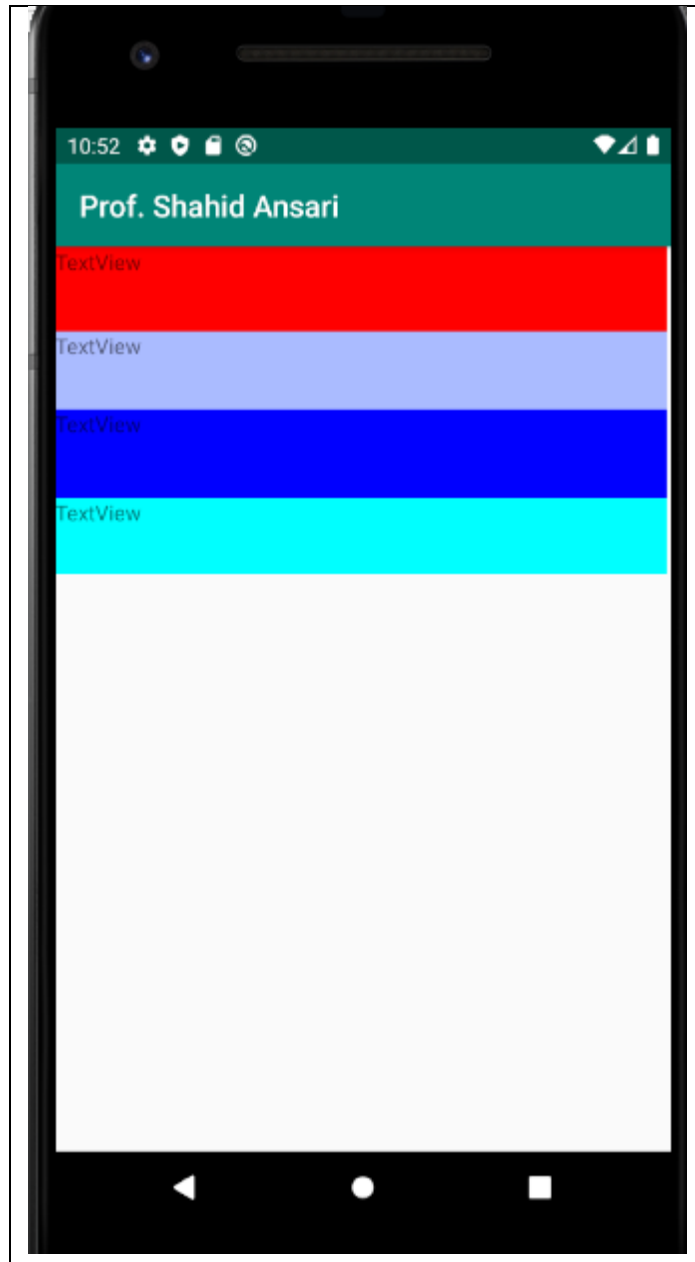


Add this line only



Add this line only

Output



We are done with Colors

b) Defining Theme Property.

- Defining new theme properties in **styles.xml**
- Create a new project and go to:
ProjectName>App>src>main>res>values>**styles.xml**

styles.xml

<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">#000000</item>

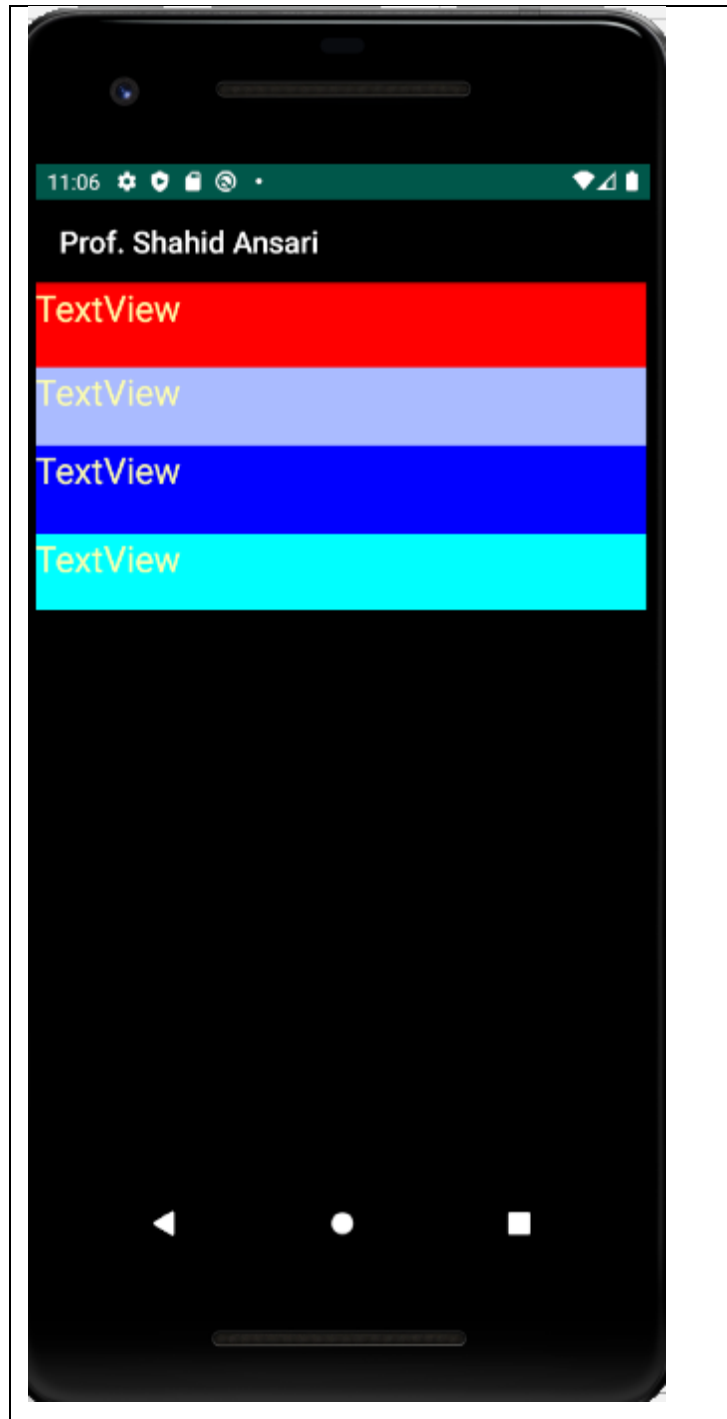
<item name="android:textColor">#FFFFAA</item>

Add these lines in styles.xml

</style>

</resources>

Output



We are done with the styles

c) Defining string property

- Defining paragraph and header property in **strings.xml**
- Create a new project and go to:
ProjectName>App>src>main>res>values>**strings.xml**

strings.xml

<resources>

<string name="app_name">Prof. Shahid Ansari</string>

<string name="Heading">Programming Resources</string>

<string name="Description">

Android is a mobile operating system developed by Google.
It is based on a modified version of the Linux
kernel and other open source software, and is
designed primarily for touch screen mobile devices
such as smart phones and tablets

</string>

</resources>

Now go to **activity_main.xml** and type the following code

<TextView

android:id="@+id/textView"

android:layout_width="match_parent"

android:layout_height="56dp"

android:background="@color/Red"

android:text="@string/Heading" →

Change this attribute

android:textSize="24sp" />

<TextView

android:id="@+id/textView2"

android:layout_width="match_parent"

android:layout_height="wrap_content"

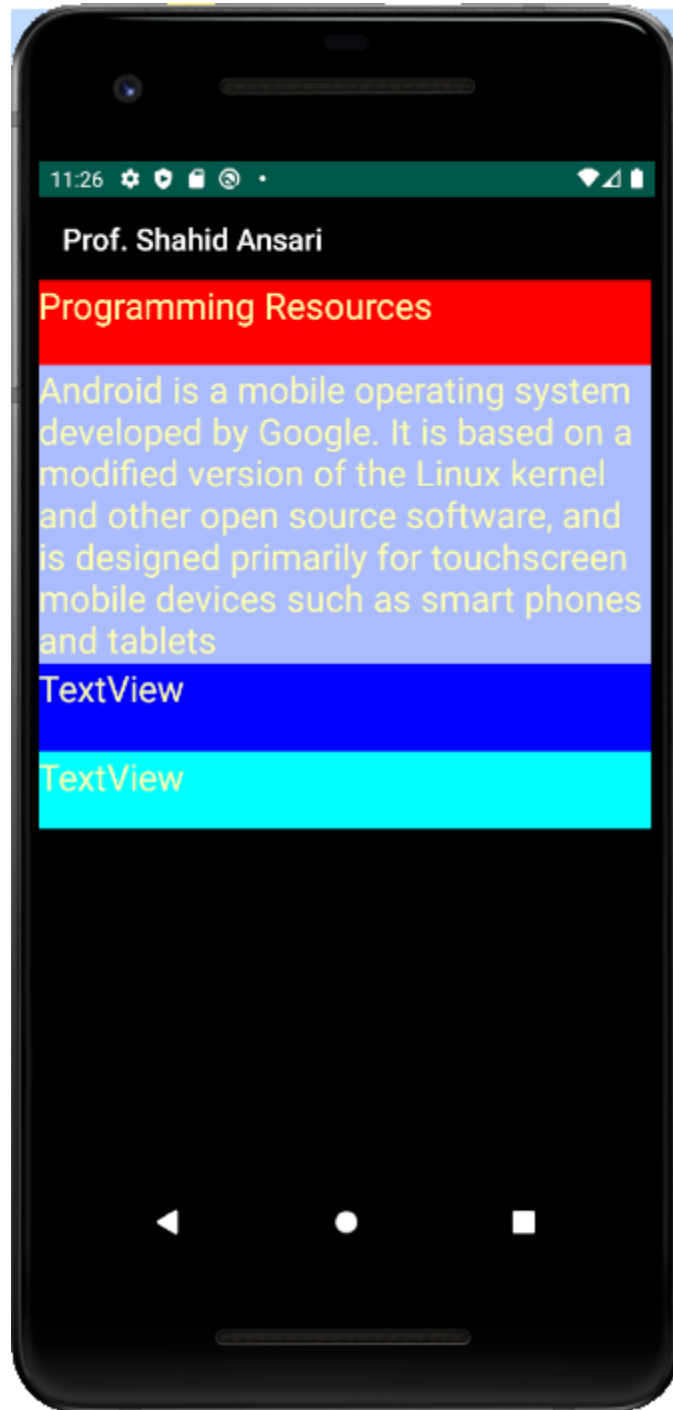
android:background="@color/Grey" →

Change this attribute

android:text="@string/Description"

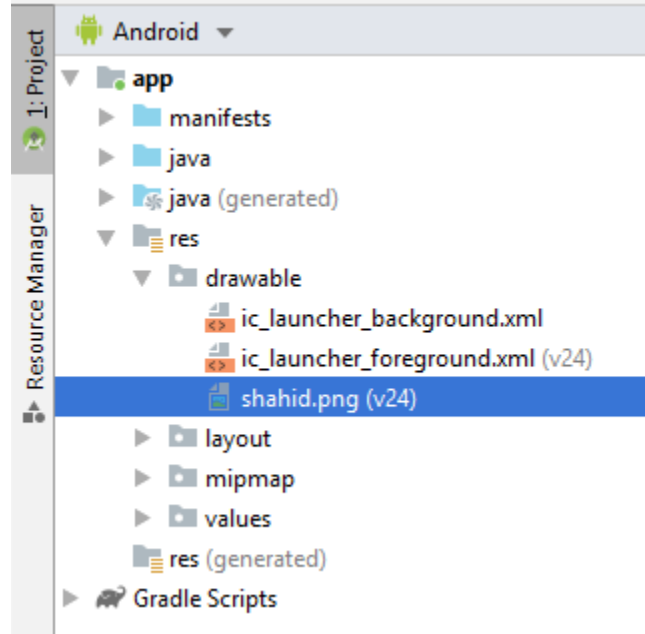
android:textSize="24sp" />

Output



d) Adding images and dimensions

- Adding Images to Application created
- For adding a new image files, do the following:
- ProjectName>App>src>main>res>drawable>Right-click and paste the images that are copied



- Add a new **dim.xml** file and write the following code in **dimens.xml**

```
<?xml version="1.0" encoding="utf-8" ?>
<resources>
  <dimen name="textview_height">35dp</dimen>
  <dimen name="textview_width">150dp</dimen>
  <dimen name="font_size">26sp</dimen>
</resources>
```

Now go to **activity_main.xml** and type the following code:

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

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

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

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

Output

