

Lab Manual

Lab (4): Image View

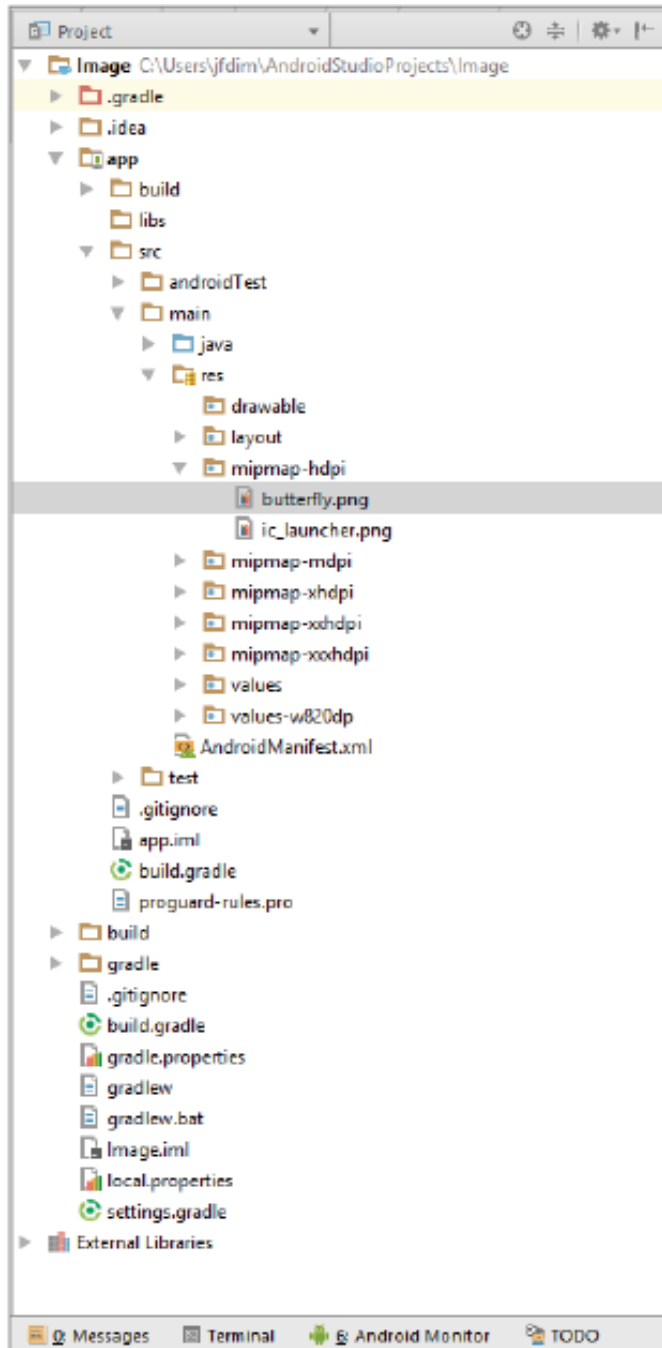


Visual Programming for Smart Devices lab

❖ Objectives:

create an android application using Image View

Goal: using Image view to display pictures.



ImageView View

The `ImageView` is a view that shows images on the device screen. The following Try It Out shows you how to use the `ImageView` view to display an image.

TRY IT OUT Using the Image View (Gallery.zip)

1. Using Android Studio, create a new Android project and name it `Image`.
2. Add an image to your project under the `res/mipmap` folder as shown in Figure 6-1. (The image I used in this example is `butterfly.png` from <http://jfdimarzio.com/butterfly.png>.) Please note that you must be in project view to drag and drop images into the `res/mipmap` folder.
3. Modify the `activity_main.xml` file as shown in bold:

```
<?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/activity_main"
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    tools:context="com.jfdimarzio.image.MainActivity">

    <android.support.v7.widget.AppCompatImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:src="@mipmap/butterfly" />

</LinearLayout>
```

4. Press Shift+F9 to debug the application on the Android emulator. Figure 6-2 shows the image displayed in the Image view.



FIGURE 6-2

Task 1: create an application have one button to replace your image to another image.

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

<GridLayout 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">

    <ImageView

        android:layout_height="wrap_content"

        android:layout_width="fill_parent"

        android:id="@+id/imag1"

        android:src="@drawable/rain"

        android:layout_column="1"

        android:layout_row="0"/>

    <Button

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:id="@+id/bt1"

        android:text="click me"

        android:layout_row="1"

        android:layout_column="1"

        android:onClick="changeImg"

    />

</GridLayout>
```

```
package com.example.user.image;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

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

    public void changeImg(View view) {
        ImageView iv=(ImageView)findViewById(R.id.imag1);
        Button bt=(Button)findViewById(R.id.bt1);

        iv.setImageResource(R.drawable.rain2);
    }
}
```

Task 2: Write your code to design the following application:

