Android Camera

**Camera** is mainly used to capture picture and video. We can control the camera by using methods of camera api.

Android provides the facility to work on camera by 2 ways:

1. By Camera Intent
2. By Camera API

Understanding basic classes of Camera Intent and API

There are mainly four classes that we are going to discuss.

Intent

By the help of 2 constants of **MediaStore** class, we can capture picture and video without using the instance of Camera class.

1. ACTION\_IMAGE\_CAPTURE
2. ACTION\_VIDEO\_CAPTURE

Camera

It is main class of camera api, that can be used to take picture and video.

SurfaceView

It represents a surface view ore preview of live camera.

MediaRecorder

It is used to record video using camera. It can also be used to record audio files as we have seen in the previous example of media framework.

### Android camera app example by camera intent

In this example, we are writing the simple code to capture image using camera and displaying the image using imageview.

#### activity\_main.xml

Drag one imageview and one button from the pallete, now the xml file will look like this:

*File: activity\_main.xml*

1. **<RelativeLayout** xmlns:androclass="http://schemas.android.com/apk/res/android"
2. xmlns:tools="http://schemas.android.com/tools"
3. android:layout\_width="match\_parent"
4. android:layout\_height="match\_parent"
5. tools:context=".MainActivity" **>**
7. **<Button**
8. android:id="@+id/button1"
9. android:layout\_width="wrap\_content"
10. android:layout\_height="wrap\_content"
11. android:layout\_alignParentBottom="true"
12. android:layout\_centerHorizontal="true"
13. android:text="Take a Photo" **>**
14. **</Button>**
16. **<ImageView**
17. android:id="@+id/imageView1"
18. android:layout\_width="fill\_parent"
19. android:layout\_height="fill\_parent"
20. android:layout\_above="@+id/button1"
21. android:layout\_alignParentTop="true"
22. android:src="@drawable/ic\_launcher" **>**
23. **</ImageView>**
24. **</RelativeLayout>**

#### Activity class

Let's write the code to capture image using camera and displaying it on the image view.

*File: MainActivity.java*

1. **package** com.example.simplecamera;
3. **import** android.app.Activity;
4. **import** android.content.Intent;
5. **import** android.graphics.Bitmap;
6. **import** android.os.Bundle;
7. **import** android.view.Menu;
8. **import** android.view.View;
9. **import** android.widget.Button;
10. **import** android.widget.ImageView;
12. **public** **class** MainActivity **extends** Activity {
13. **private** **static** **final** **int** CAMERA\_REQUEST = 1888;
14. ImageView imageView;
15. **public** **void** onCreate(Bundle savedInstanceState) {
17. **super**.onCreate(savedInstanceState);
18. setContentView(R.layout.activity\_main);
20. imageView = (ImageView) **this**.findViewById(R.id.imageView1);
21. Button photoButton = (Button) **this**.findViewById(R.id.button1);
23. photoButton.setOnClickListener(**new** View.OnClickListener() {
25. @Override
26. **public** **void** onClick(View v) {
27. Intent cameraIntent = **new** Intent(android.provider.MediaStore.ACTION\_IMAGE\_CAPTURE);
28. startActivityForResult(cameraIntent, CAMERA\_REQUEST);
29. }
30. });
31. }
33. **protected** **void** onActivityResult(**int** requestCode, **int** resultCode, Intent data) {
34. **if** (requestCode == CAMERA\_REQUEST) {
35. Bitmap photo = (Bitmap) data.getExtras().get("data");
36. imageView.setImageBitmap(photo);
37. }
38. }
40. @Override
41. **public** **boolean** onCreateOptionsMenu(Menu menu) {
42. // Inflate the menu; this adds items to the action bar if it is present.
43. getMenuInflater().inflate(R.menu.activity\_main, menu);
44. **return** **true**;
45. }
47. }