Name: Prashant Bhosale
Roll No: 804

EXPERIMENT 11

WORKING WITH IMAGES

AIM: To create an application which captures images and apply various animations on the image.

THEORY:

Capture an image:

You will use MediaStore.ACTION_IMAGE_CAPTURE to launch an existing camera application installed on your phone. Its syntax is given below

```
Intent intent = new Intent(android.provider.MediaStore.ACTION_IMAGE_CAPTURE);
```

Now you will use the function startActivityForResult() to launch this activity and wait for its result. Its syntax is given below

```
startActivityForResult(intent,0)
```

Display an image:

- Add a picture from your computer in drawable folder
- Create an **ImageView** in activity_main.xml and display the image

Change one image to another:

Example:

```
setImageResource(R.drawable.image_name);
```

Sets a drawable as the content of this ImageView. Image name should be the name of the image which you have placed in drawable folder.

Rotate an image:

Rotation is done using getrotation( ) and setrotation( ) methods. An example for rotating an image by 90 degree is given below

```
img.setRotation(img.getRotation() + 90);
```

Zoom an image:

In order to perform animation in android , we are going to call a static function loadAnimation() of the class AnimationUtils. We are going to receive the result in an instance of Animation Object. Its syntax is as follows –

```
Animation animation = AnimationUtils.loadAnimation(getApplicationContext(),
   R.anim.myanimation);
```

Name: Prashant Bhosale
Roll No: 804

Note the second parameter. It is the name of the our animation xml file. You have to create a new folder called anim under res directory and make an xml file under anim folder.

Move an image:

The **ObjectAnimator** API provides an easy way to change the properties of a view with a specified duration.

Here is an **example of an ObjectAnimator** that moves the image to 100 pixels from the left of the screen in 2 seconds:

```
ObjectAnimator   animation   =   ObjectAnimator.ofFloat(image,   "x",   100f);
animation.setDuration(2000);
animation.start();
```

This example uses the ObjectAnimator.ofFloat() method since the translation values have to be floats.

```
ObjectAnimator.ofFloat(image, "x", 100f);
```
- The first parameter is the image you want to animate.

- The second parameter is the property you are animating. Since the view needs to be moved horizontally, the "x" property is used.

- The last parameter is the end value of the animation. Since this value is 100, it will be that many pixels from the left of the screen.

```
animation.setDuration(2000);
```
The next method specifies how long the animation should take in milliseconds. In this example the animation will run for 2 seconds (2000 milliseconds).

```
animation.start();
```
The last method causes the animation to run which will update the view's position on screen.

ASSIGNMENT

1. Create an application in android that will capture an image.

   CODE

   activity_main.xml

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

<RelativeLayout 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"
```

Name: Prashant Bhosale

Roll No: 804

```xml
        tools:context=".MainActivity">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Image View"
            android:textSize="30dp"
            android:textStyle="bold"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="10dp"
            android:id="@+id/textV1" />
        <ImageView
            android:layout_width="200dp"
            android:layout_height="200dp"
            android:text="Image View"
            android:textSize="30dp"
            android:textStyle="bold"
            android:layout_below="@+id/textV1"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"
            android:id="@+id/imgLaunch_Camera"
            />
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Image View"
            android:textSize="30dp"
            android:textStyle="bold"
            android:layout_below="@+id/imgLaunch_Camera"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"
            android:id="@+id/btnTackPic"
            />
</RelativeLayout>
```

MainActivity.java

```java
package com.example.mca1634.imageview;

import android.content.Intent;
import android.graphics.Bitmap;
import android.provider.MediaStore;
import android.support.annotation.Nullable;
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 {
    ImageView imgLaunch_Camera;
    Button btnTackPic;
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable
Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
```

```java
            if(requestCode==1)
            {
                Bitmap photo=(Bitmap) data.getExtras().get("data");
                imgLaunch_Camera.setImageBitmap(photo);
            }
    }
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imgLaunch_Camera=(ImageView)findViewById(R.id.imgLaunch_Camera);
        btnTackPic=(Button)findViewById(R.id.btnTackPic);
        btnTackPic.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent=new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                //Take a picture and pass results along to onActivityResult
                startActivityForResult(intent,1);
            }
        });
    }
}
```

OUTPUT

2. Create an application that will perform the following animations

    1. Display image

    2. Change one image to another

    3. Rotate an image

    4. Move an image

    5. Expand or zoom an image

CODE

activity_main.xml

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

<RelativeLayout 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">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Image View"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:id="@+id/textV1" />
    <ImageView
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:text="Image View"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_below="@+id/textV1"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:id="@+id/imgLaunch_Camera"
        android:src="@drawable/cr7_1"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Move"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="40dp"
        android:layout_below="@+id/imgLaunch_Camera"
        android:layout_marginTop="10dp"
```

```xml
        android:id="@+id/btnMove"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Rotate"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="205dp"
        android:layout_below="@+id/imgLaunch_Camera"
        android:layout_marginTop="10dp"
        android:id="@+id/btnRotate"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Zoom"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="40dp"
        android:layout_below="@+id/btnMove"
        android:layout_marginTop="10dp"
        android:id="@+id/btnZoom"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="CHANGE IMAGE"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="205dp"
        android:layout_below="@+id/btnRotate"
        android:layout_marginTop="10dp"
        android:id="@+id/btnChangeImage"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Zoom In"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="40dp"
        android:layout_below="@+id/btnZoom"
        android:layout_marginTop="45dp"
        android:id="@+id/btnZoomIn"
        />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Zoom Out"
        android:textSize="30dp"
        android:textStyle="bold"
        android:layout_marginLeft="205dp"
```

```xml
        android:layout_below="@+id/btnChangeImage"
        android:layout_marginTop="10dp"
        android:id="@+id/btnZoomOut"
        />
</RelativeLayout>
```

zoom.xml

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

<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:duration="3000"
        android:fromXScale="0.2"
        android:fromYScale="0.2"
        android:pivotY="50%"
        android:pivotX="50%"
        android:toXScale="2.0"
        android:toYScale="2.0"
        android:repeatMode="reverse"
        android:repeatCount="2"
        />
</set>
```

zoom_in.xml

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

<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:duration="3000"
        android:fromXScale="0.2"
        android:fromYScale="0.2"
        android:pivotY="50%"
        android:pivotX="50%"
        android:toXScale="2.0"
        android:toYScale="2.0"
        android:repeatCount="2"
        />
</set>
```

zoom_out.xml

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

<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:duration="3000"
        android:fromXScale="2.0"
        android:fromYScale="2.0"
        android:pivotY="50%"
        android:pivotX="50%"
        android:toXScale="0.2"
        android:toYScale="0.2"
```

```xml
        android:repeatCount="2"
        />
</set>
```

MainActivity.java

package com.example.mca1634.imageview_rot_zoom;

```java
import android.animation.ObjectAnimator;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;
public class MainActivity extends AppCompatActivity {
    ImageView imgLaunch_Camera;
    Button btnMove,btnRotate,btnZoom,btnChangeImage,btnZoomIn,btnZoomOut;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imgLaunch_Camera=(ImageView)findViewById(R.id.imgLaunch_Camera);
        btnMove=(Button)findViewById(R.id.btnMove);
        btnRotate=(Button)findViewById(R.id.btnRotate);
        btnZoom=(Button)findViewById(R.id.btnZoom);
        btnChangeImage=(Button)findViewById(R.id.btnChangeImage);
        btnZoomIn=(Button)findViewById(R.id.btnZoomIn);
        btnZoomOut=(Button)findViewById(R.id.btnZoomOut);
        btnMove.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                ObjectAnimator
animatorY=ObjectAnimator.ofFloat(imgLaunch_Camera,"y",500f);
                ObjectAnimator
animatorX=ObjectAnimator.ofFloat(imgLaunch_Camera,"x",500f);
                animatorY.setDuration(3000);
                animatorX.setDuration(3000);
                animatorY.start();
                animatorX.start();
            }
        });
        btnRotate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                    imgLaunch_Camera.setRotation(imgLaunch_Camera.getRotation()
+90); //anti-clockwise
            }
        });
        btnZoom.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```java
                Animation zoomAnimation =
AnimationUtils.loadAnimation(MainActivity.this,R.anim.zoom);
                imgLaunch_Camera.startAnimation(zoomAnimation);
            }
        });
        btnChangeImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                    imgLaunch_Camera.setImageResource(R.drawable.cr7_2);
            }
        });
        btnZoomIn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Animation zoomAnimation =
AnimationUtils.loadAnimation(MainActivity.this,R.anim.zoom_in);
                imgLaunch_Camera.startAnimation(zoomAnimation);
            }
        });
        btnZoomOut.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Animation zoomAnimation =
AnimationUtils.loadAnimation(MainActivity.this,R.anim.zoom_out);
                imgLaunch_Camera.startAnimation(zoomAnimation);
            }
        });
    }
}
```
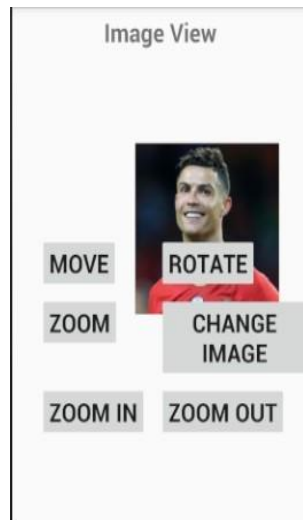
OUTPUT
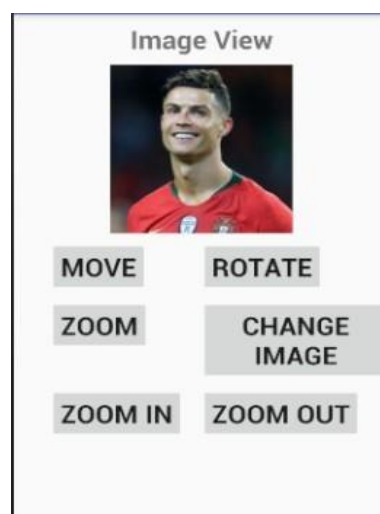
Name: Prashant Bhosale
Roll No: 804

CONCLUSION: Successfully Understand Image View in Android Studio.