

## ASSIGNMENT 3

---

# Graphical Primitives and Animation

SHASHANKA VENKATESH - 185001145

*9<sup>th</sup> August, 2021*

---

## Learning Objective

Generate the following using graphical primitives.

## Specifications

1. Draw shapes such as Line, Circle, Rectangle and Arc
2. Perform animation using any Image (Gif, Jpeg)
3. Perform transformation – Rotation, Zooming

# Home (Main Activity)

## XML (activity\_main.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".MainActivity">
8
9   <Button
10      android:id="@+id/shapesButton"
11      android:layout_width="wrap_content"
12      android:layout_height="wrap_content"
13      android:layout_marginTop="50dp"
14      android:text="@string/primitive_shapes"
15      app:layout_constraintEnd_toEndOf="parent"
16      app:layout_constraintStart_toStartOf="parent"
17      app:layout_constraintTop_toTopOf="parent" />
18
19   <Button
20      android:id="@+id/animateButton"
21      android:layout_width="wrap_content"
22      android:layout_height="wrap_content"
23      android:text="@string/animations"
24      app:layout_constraintBottom_toTopOf="@+id/transformButton"
25      app:layout_constraintEnd_toEndOf="parent"
26      app:layout_constraintStart_toStartOf="parent"
27      app:layout_constraintTop_toBottomOf="@+id/shapesButton" />
28
29   <Button
30      android:id="@+id/transformButton"
31      android:layout_width="wrap_content"
32      android:layout_height="wrap_content"
33      android:text="@string/transformations"
34      android:layout_marginBottom="50dp"
35      app:layout_constraintBottom_toBottomOf="parent"
36      app:layout_constraintEnd_toEndOf="parent"
37      app:layout_constraintStart_toStartOf="parent" />
38 </androidx.constraintlayout.widget.ConstraintLayout>
```

## Java (MainActivity.java)

```
1 package com.madlab.graphicalprimitivesandanimation;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.graphics.drawable.shapes.Shape;
7 import android.os.Bundle;
8 import android.view.View;
9 import android.widget.Button;
10
11 public class MainActivity extends AppCompatActivity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17
18         // Get references to the 3 buttons
19         final Button shapesButton = findViewById(R.id.shapesButton);
20         final Button animateButton = findViewById(R.id.animateButton);
21         final Button transformButton = findViewById(R.id.transformButton);
22
23         // Set On-Click Listeners to all of them
24         shapesButton.setOnClickListener(new View.OnClickListener() {
25             @Override
26             public void onClick(View view) {
27                 Intent shapesIntent = new Intent(MainActivity.this,
28 ShapesActivity.class);
29                 startActivity(shapesIntent);
30             }
31         });
32         animateButton.setOnClickListener(new View.OnClickListener() {
33             @Override
34             public void onClick(View view) {
35                 Intent animateIntent = new Intent(MainActivity.this,
36 AnimateActivity.class);
37                 startActivity(animateIntent);
38             }
39         });
40         transformButton.setOnClickListener(new View.OnClickListener() {
41             @Override
42             public void onClick(View view) {
43                 Intent transformIntent = new Intent(MainActivity.this,
44 TransformActivity.class);
45                 startActivity(transformIntent);
46             }
47         });
48     }
49 }
```

# Primitive Shapes

## XML (activity\_shapes.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".ShapesActivity">
8
9   <ImageView
10     android:id="@+id/canvasImage"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     app:layout_constraintBottom_toBottomOf="parent"
14     app:layout_constraintEnd_toEndOf="parent"
15     app:layout_constraintStart_toStartOf="parent"
16     app:layout_constraintTop_toTopOf="parent"
17     app:srcCompat="@color/white"/>
18
19 </androidx.constraintlayout.widget.ConstraintLayout>
```

## Java (ShapesActivity.java)

```
1 package com.madlab.graphicalprimitivesandanimation;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.graphics.*;
6 import android.widget.ImageView;
7
8 public class ShapesActivity extends AppCompatActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_shapes);
14
15         // Getting a reference to the canvas (ImageView)
16         final ImageView canvasImage = findViewById(R.id.canvasImage);
17
18         // Setting up a Bit map over which graphics can be drawn
19         Bitmap bitmap = Bitmap.createBitmap(1080, 1920, Bitmap.Config.
20             ARGB_8888);
21         canvasImage.setImageBitmap(bitmap);
22         Canvas canvas = new Canvas(bitmap);
23         Paint paint = new Paint();
24         paint.setTextSize(50);
25
26         // Painting the shapes with their names as captions
27         paint.setColor(Color.rgb(41, 22, 72));
28         canvas.drawText("Rectangle", 150, 1050, paint);
29         canvas.drawRect(50, 100, 450, 1000, paint);
30
31         paint.setColor(Color.rgb(80, 0, 40));
32         canvas.drawText("Square", 710, 700, paint);
33         canvas.drawRect(550, 150, 1050, 650, paint);
34
35         paint.setColor(Color.rgb(0, 77, 81));
36         canvas.drawText("Circle", 750, 1200, paint);
37         canvas.drawCircle(800, 1000, 150, paint);
38
39         paint.setColor(Color.rgb(165, 56, 0));
40         canvas.drawText("Arc", 250, 1450, paint);
41         RectF oval = new RectF(50, 1150, 550, 1400);
42         canvas.drawArc(oval, 0, 180, false, paint);
43
44         paint.setColor(Color.BLACK);
45         canvas.drawText("Line", 900, 1500, paint);
46         canvas.drawLine(820, 1300, 820, 1700, paint);
47     }
```

# Animation using an Image

## XML (activity\_animate.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".AnimateActivity">
8
9   <ImageView
10     android:id="@+id/canvasAnimate"
11     android:layout_width="wrap_content"
12     android:layout_height="wrap_content"
13     app:layout_constraintBottom_toBottomOf="parent"
14     app:layout_constraintEnd_toEndOf="parent"
15     app:layout_constraintStart_toStartOf="parent"
16     app:layout_constraintTop_toTopOf="parent"
17     android:src="@drawable/car"/>
18
19 </androidx.constraintlayout.widget.ConstraintLayout>
```

## Java (AnimateActivity.java)

```
1 package com.madlab.graphicalprimitivesandanimation;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.animation.*;
5 import android.os.Bundle;
6 import android.widget.ImageView;
7
8 public class AnimateActivity extends AppCompatActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_animate);
14
15         // Getting a reference to the canvas (ImageView)
16         ImageView canvasAnimate = findViewById(R.id.canvasAnimate);
17
18         ObjectAnimator moveForwards = ObjectAnimator.ofFloat(canvasAnimate
19 , "x", 150).setDuration(1000);
20         ObjectAnimator moveBackwards = ObjectAnimator.ofFloat(
21 canvasAnimate, "x", -150).setDuration(2000);
22         ObjectAnimator moveToOriginal = ObjectAnimator.ofFloat(
23 canvasAnimate, "x", 0).setDuration(2000);
24         AnimatorSet animation = new AnimatorSet();
25
26         // Reverse the direction
27         animation.play(moveBackwards).before(moveForwards);
28         animation.play(moveForwards).before(moveToOriginal);
29
30         // Loop
31         animation.addListener(new AnimatorListenerAdapter() {
32             @Override
33             public void onAnimationEnd(Animator animation) {
34                 super.onAnimationEnd(animation);
35                 animation.start();
36             }
37         });
38
39         // Start the animation
40         animation.start();
41     }
42 }
```

# Transformations

## XML (activity\_transform.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://
  schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   tools:context=".TransformActivity">
8
9   <ImageView
10     android:id="@+id/rotateImage"
11     android:layout_width="150dp"
12     android:layout_height="150dp"
13     android:layout_marginTop="50dp"
14     app:layout_constraintEnd_toEndOf="parent"
15     app:layout_constraintStart_toStartOf="parent"
16     app:layout_constraintTop_toTopOf="parent"
17     app:srcCompat="@drawable/ic_rotate"/>
18
19   <ImageView
20     android:id="@+id/profileImage"
21     android:layout_width="150dp"
22     android:layout_height="150dp"
23     app:layout_constraintBottom_toTopOf="@+id/zoomImage"
24     app:layout_constraintEnd_toEndOf="parent"
25     app:layout_constraintStart_toStartOf="parent"
26     app:layout_constraintTop_toBottomOf="@+id/rotateImage"
27     app:srcCompat="@drawable/ic_profile" />
28
29   <ImageView
30     android:id="@+id/zoomImage"
31     android:layout_width="75dp"
32     android:layout_height="75dp"
33     android:layout_marginBottom="50dp"
34     app:layout_constraintBottom_toBottomOf="parent"
35     app:layout_constraintEnd_toEndOf="parent"
36     app:layout_constraintStart_toStartOf="parent"
37     app:srcCompat="@drawable/ic_zoom"/>
38
39 </androidx.constraintlayout.widget.ConstraintLayout>
```



## Animation XML (rotate.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <set xmlns:android="http://schemas.android.com/apk/res/android"
    android:interpolator="@android:anim/cycle_interpolator">
3     <rotate android:fromDegrees="0"
4         android:toDegrees="360"
5         android:pivotX="50%"
6         android:pivotY="50%"
7         android:duration="4000"
8         android:repeatCount="infinite"
9         android:repeatMode="restart"/>
10 </set>
```

## Animation XML (fade.xml)

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <set xmlns:android="http://schemas.android.com/apk/res/android">
3     <alpha
4         android:fromAlpha="0.0"
5         android:toAlpha="1.0"
6         android:duration="2000"
7         android:repeatCount="infinite"
8         android:repeatMode="reverse"/>
9 </set>
```

## Animation XML (zoom.xml)

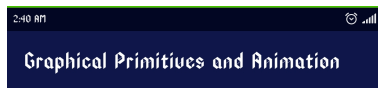
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <set xmlns:android="http://schemas.android.com/apk/res/android">
3     <scale
4         xmlns:android="http://schemas.android.com/apk/res/android"
5         android:duration="2000"
6         android:fromXScale="1"
7         android:fromYScale="1"
8         android:pivotX="50%"
9         android:pivotY="50%"
10        android:toXScale="2"
11        android:toYScale="2"
12        android:repeatCount="infinite"
13        android:repeatMode="reverse">
14    </scale>
15 </set>
```

## Java (TransformActivity.java)

```
1 package com.madlab.graphicalprimitivesandanimation;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.view.animation.Animation;
5 import android.view.animation.AnimationUtils;
6 import android.os.Bundle;
7 import android.widget.ImageView;
8
9 public class TransformActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_transform);
15
16         // Get the 3 ImageViews
17         final ImageView rotateImage = findViewById(R.id.rotateImage);
18         final ImageView profileImage = findViewById(R.id.profileImage);
19         final ImageView zoomImage = findViewById(R.id.zoomImage);
20
21         // Define the animations using the XML resources
22         Animation rotateAnimation = AnimationUtils.loadAnimation(
23             getApplicationContext(), R.anim.rotate);
24         Animation fadeAnimation = AnimationUtils.loadAnimation(
25             getApplicationContext(), R.anim.fade);
26         Animation zoomAnimation = AnimationUtils.loadAnimation(
27             getApplicationContext(), R.anim.zoom);
28
29         // Start the animations
30         rotateImage.startAnimation(rotateAnimation);
31         profileImage.startAnimation(fadeAnimation);
32         zoomImage.startAnimation(zoomAnimation);
33     }
34 }
```

# App Screenshots

## Home

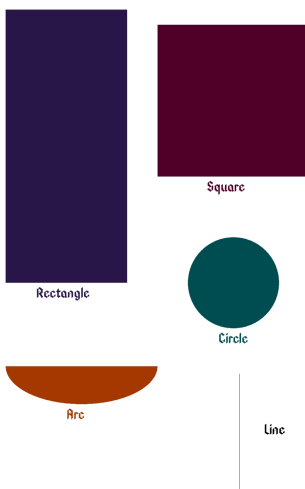
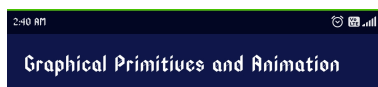


PRIMITIVE SHAPES

ANIMATIONS

TRANSFORMATIONS

## Graphical Primitives



# Animation



# Transformation

