# Practical No. 7

## Title: Android program to work with graphics and animation

### Aim: Create an application to demonstrate graphics and animation

### Introduction

#### Create Drawing Objects

The android.graphics framework divides drawing into two areas:

1. *What* to draw, handled by Canvas
2. *How* to draw, handled by Paint.

For instance, Canvas provides a method to draw a line, while Paint provides methods to define that line's color. Canvas has a method to draw a rectangle, while Paint defines whether to fill that rectangle with a color or leave it empty. Simply put, Canvas defines shapes that you can draw on the screen, while Paint defines the color, style, font, and so forth of each shape you draw.

#### Draw!

Once you have your object creation and measuring code defined, you can implement onDraw(). Every view implements onDraw() differently, but there are some common operations that most views share:

- Draw text using drawText(). Specify the typeface by calling setTypeface(), and the text color by calling setColor().
- Draw primitive shapes using drawRect(), drawOval(), and drawArc(). Change whether the shapes are filled, outlined, or both by calling setStyle().
- Draw more complex shapes using the Path class. Define a shape by adding lines and curves to a Path object, then draw the shape using drawPath(). Just as with primitive shapes, paths can be outlined, filled, or both, depending on the setStyle().
- Define gradient fills by creating LinearGradient objects. Call setShader() to use your LinearGradient on filled shapes.
- Draw bitmaps using drawBitmap().

## Exercise - Create android application to demonstrate graphics and animation

**Implementation:**
**Program:**

**MainActivity.java**

```
import android.app.Activity;
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends Activity
{

     @Override
     public void onCreate(Bundle savedInstanceState)
     {
         super.onCreate(savedInstanceState);
         setContentView(new MyView(this));
     }

     public class MyView extends View
     {
         Paint paint = null;
         public MyView(Context context)
         {
              super(context);
              paint = new Paint();
         }

         @Override
         protected void onDraw(Canvas canvas)
         {
            super.onDraw(canvas);
            int x = getWidth();
            int y = getHeight();
            int radius;
            radius = 100;
            paint.setStyle(Paint.Style.FILL);
            paint.setColor(Color.WHITE);
            canvas.drawPaint(paint);
            // Use Color.parseColor to define HTML colors
```

```
            paint.setColor(Color.parseColor("#CD5C5C"));
            canvas.drawCircle(x / 2, y / 2, radius, paint);
        }
    }
}
```

**Output:**