

## Experiment No: 1

Aim: Write an android application to draw basic 2D graphical primitives

Theory: An application based on Java API

Android studio and android manifest

- Android Studio

⇒ Android studio is the official Integrated Development Environment (IDE) for android application development.

- Android studio provides more features that enhances our productivity while building android apps.

- The features of android studio are :-

- 1) It has a flexible gradle-based build system
- 2) It has a fast and feature-rich emulator for app testing.
- 3) It supports C++ and NDK
- 4) It provides built-in supports for Google cloud platform.

- 2D primitives

⇒ In mobile computing, 2D primitives refers to basic graphical elements that are used to create 2D graphics and user interfaces on mobile devices.

- These primitives are essential for creating visually appealing and interactive mobile applications.

- Some of the common 2D primitives are :-

1) Points

⇒ The most basic primitive, representing a single pixel on the screen. Points are the building blocks for other shapes.

2) Lines

⇒ A sequence of connected points forming a straight path.

3) Rectangles

⇒ defined by four points, representing a four-sided polygon with equal angles at each corner.

4) Circles and Ellipses

⇒ Represented by their center point and radius or two radii.

5) Polygons

⇒ Multi-sided shapes formed by connecting multiple points.

## - XML

- ⇒ XML stands for the extensible Markup Language
- XML is a markup language.
- XML is designed to store and transport the data. XML is designed to be self-descriptive.
- XML is a platform independent and language independent.
- XML tags are not predefined, we must define our own tags.
- XML is designed to carry data, not to display data, XML is not a replacement for HTML.

## - Functions

### 1) Bitmap

- ⇒ A 'Bitmap' is a representation of graphic image as a pixel array. It is often used to load, manipulate and display images in Android apps.
- Bitmap can be created from various sources, including resources, files or dynamically generated.

### 2) Canvas

- ⇒ The 'Canvas' class is used to draw graphics on an Android device. It provides methods to draw various shapes, text and images onto a Bitmap or the screen.

### 3) ImageView

⇒ An "ImageView" is a UI widget in Android used to display images. It can be used to load and display Bitmaps, drawables or other images from sources.

### 4) Paint

⇒ The 'Paint' class is used to define how to draw graphical elements, such as color, style, stroke width, and text attributes.

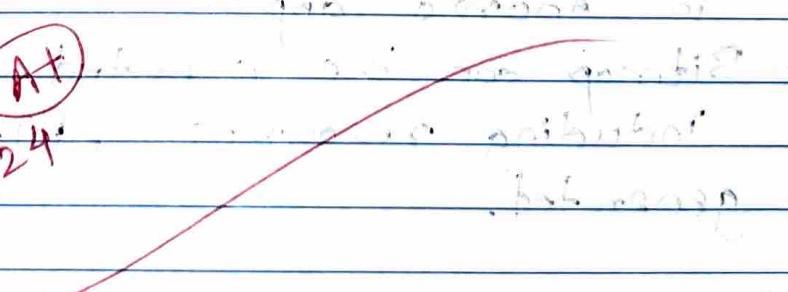
### 5) Color

⇒ The 'Color' class in Android provides methods to work with colors.

## Conclusion :

In this experiment, we learnt about the tool android studio and how to draw 2D primitives in android studio

Paint  
23/11/24



Code:

MainActivity.java

```
package com.example.expl_mcc;

import android.app.Activity;
import android.graphics.Bitmap;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.drawable.BitmapDrawable;
import android.os.Bundle;
import android.widget.ImageView;

public class MainActivity extends Activity
{
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        //Creating a Bitmap
        Bitmap bg = Bitmap.createBitmap(720, 1280, Bitmap.Config.RGB_565);

        //Setting the Bitmap as background for the ImageView

        ImageView i = (ImageView) findViewById(R.id.imageView);
        i.setBackgroundDrawable(new BitmapDrawable(bg));

        //Creating the Canvas Object
        Canvas canvas = new Canvas(bg);

        //Creating the Paint Object and set its color & TextSize
        Paint paint = new Paint();
        paint.setColor(Color.RED);
        paint.setTextSize(50);

        //To draw a Rectangle
        canvas.drawText("Rectangle", 420, 150, paint);
        canvas.drawRect(400, 200, 650, 700, paint);

        paint.setColor(Color.BLUE);

        //To draw a Circle
        canvas.drawText("Circle", 120, 150, paint);
        canvas.drawCircle(200, 350, 150, paint);

        paint.setColor(Color.YELLOW);

        //To draw a Square
```

```
    canvas.drawText("Square", 120, 800, paint);
    canvas.drawRect(50, 850, 350, 1150, paint);

    paint.setColor(Color.RED);

    //To draw a Line
//    canvas.drawText("Line", 480, 800, paint);
//    canvas.drawLine(520, 850, 520, 1150, paint);

    //To draw a Line
    canvas.drawText("Line", 480, 800, paint);
    canvas.drawLine(520, 850, 460, 1150, paint);
    canvas.drawLine(520, 850, 580, 1150, paint);
    canvas.drawLine(460, 1150, 580, 1150, paint);

}
}
```

### activity\_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/imageView" />
</RelativeLayout>
```

Output:

