

MOBILE APPLICATION DEVELOPMENT LAB

Program 4

Develop a canvas to draw different shapes and to fill the shapes with different colors.

Java Code

```
package com.example.drawingshapes;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new com.example.drawingshapes.CustomView(this));
    }
}
```

XML code

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    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="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

CustomView.java

```
package com.example.drawingshapes;
```

```
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Rect;
import android.view.View;
```

```
public class CustomView extends View {
```

```
    private Rect rectangle;
    private Paint paint, p1;
```

```
    public CustomView(Context context) {
        super(context);
        int x = 200;
        int y = 50;
        int width = 800;
        int height = 300;
```

```
        // create a rectangle that we'll draw later
        rectangle = new Rect(x, y, width, height);
```

```
        // create the Paint and set its color
        paint = new Paint();
        paint.setColor(Color.GRAY);
```

```
        p1 = new Paint();
        p1.setColor(Color.RED);
    }
```

```
    @Override
    protected void onDraw(Canvas canvas) {
        canvas.drawColor(Color.BLUE);
        canvas.drawRect(rectangle, paint);
    }
```

```
}
```

Output

