Ex. No : 4

DRAW BASIC GRAPHICAL PRIMITIVES

Date:

AIM:

To develop an android application that draws basic graphical primitives on the screen using android studio and sdk.

PROCEDURE:

Step 1 : File → NewProject
Provide the application name and Click “Next”

Step 2 : Select the target android devices
Select the minimum SDK to run the application. Click “Next”.

Step 3 : Choose the activity for the application (By default choose “Blank Activity). Click “Next”.

Step 4 : Enter activity name and click "Finish".

Step 5 : Edit the program.

Step 6 : Run the application, 2-ways to run the application.

1. Running through emulator
2. Running through mobile device

SOURCE CODE:-

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.admin.ex4b">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.admin.ex4b.MainActivity">

   <com.example.admin.ex4b.TouchScreen
       android:id="@+id/drawing"
       android:layout_width="match_parent"
       android:layout_height="match_parent"
       android:layout_marginBottom="3dp"
       android:layout_marginLeft="5dp"
       android:layout_marginRight="5dp"
       android:layout_marginTop="3dp"
       android:layout_weight="1"
       android:background="#FFFFFFFF" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_gravity="center"
        android:orientation="horizontal">

        <Button
            android:text="Ellipse"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/b1"
            android:layout_weight="1" />
        <Button
            android:text="Rectangle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
            android:id="@+id/b2"
            android:layout_weight="1" />
    </LinearLayout>

</RelativeLayout>
```

MainActivity.java

```
package com.example.admin.ex4b;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    private TouchScreen touchScreen;
    Button b1,b2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        touchScreen=(TouchScreen)findViewById(R.id.drawing);

        b1=(Button)findViewById(R.id.b1);
        b2=(Button)findViewById(R.id.b2);


        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                touchScreen.setDrawing(0);
            }
        });

        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                touchScreen.setDrawing(1);
            }
        });

    }
}
```

TouchScreen.java:

```
package com.example.admin.ex4b;

import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.Path;
import android.graphics.PorterDuff;
import android.graphics.RectF;e
import android.util.AttributeSet;
import android.view.MotionEvent;
import android.view.View;

/**
 * Created by admin on 1/18/2018.
```

```
 */

public class TouchScreen extends View {

    Paint paint,canvasPaint;
    Path path;
    float X,Y,X1,Y1;
    RectF rectF;
    private int flag;
    private float width,height;
    private Bitmap canvasBitmap;
    private Canvas drawCanwas;
    public TouchScreen(Context context, AttributeSet attributeSet)
    {
        super(context,attributeSet);
        paint=new Paint();
        path=new Path();
        paint.setAntiAlias(true);
        paint.setColor(Color.BLACK);
        paint.setStrokeJoin(Paint.Join.ROUND);
        paint.setStyle(Paint.Style.STROKE);
        paint.setStrokeWidth(5f);
        paint.setStrokeCap(Paint.Cap.ROUND);
        canvasPaint=new Paint(Paint.DITHER_FLAG);
        //canvasBitmap=Bitmap.createBitmap();
    }

    @Override
    public void onSizeChanged(int w,int h,int oldw,int oldh)
    {
        super.onSizeChanged(w,h,oldw,oldh);
        canvasBitmap=Bitmap.createBitmap(w,h,Bitmap.Config.ARGB_8888);
        drawCanwas=new Canvas(canvasBitmap);
    }

    @Override
    protected void onDraw(Canvas canvas)
    {
        canvas.drawBitmap(canvasBitmap,0,0,canvasPaint);
        canvas.drawPath(path,paint);
    }

    @Override
    public boolean onTouchEvent(MotionEvent event)
    {
        //float X = event.getX();
        //float Y = event.getY();
        switch (event.getAction())
        {
            case MotionEvent.ACTION_DOWN:
                // Set a new starting point on finger touch
                X=event.getX();
                Y= event.getY();
                //path.moveTo(X, Y);
                return true;
            case MotionEvent.ACTION_MOVE:
                // Connect the points
                //path.lineTo(X, Y);
                break;
            case MotionEvent.ACTION_UP:
                //path.lineTo(X,Y);

                //Do nothing on finger lift
                X1=event.getX();
                Y1=event.getY();
                rectF=new RectF(X,Y,X1,Y1);
                if(flag==0)
                {
```

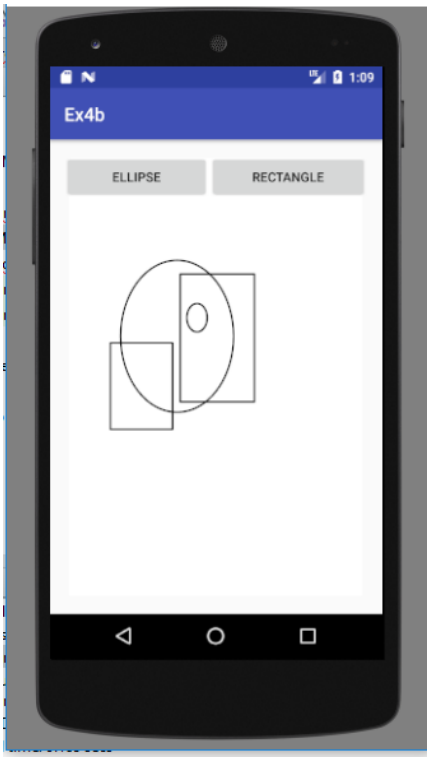
```
                    path.addOval(rectF, Path.Direction.CCW);
                }
                else
                {
                    path.addRect(rectF, Path.Direction.CCW);
                }

                drawCanwas.drawPath(path,paint);
                path.reset();
                break;
            default:
                return false;
        }
        // Makes our view repaint and call onDraw
        invalidate();
        return true;
    }
    //set Drawing
    public void setDrawing(int F){
        flag=F;
    }

    //start new
    public void startDrawing()
    {
        drawCanwas.drawColor(0, PorterDuff.Mode.CLEAR);
        invalidate();
    }

}
```

OUTPUT:-



RESULT:

Thus the android application that draws basic graphical primitives on the screen using android studio and sdk was developed successfully.