Android graphics

- Android provides a huge set of 2D-drawing APIs that allow you to create graphics.
- Android has got visually appealing graphics and mind blowing animations.
- The Android framework provides a rich set of powerful APIS for applying animation to UI elements and graphics as well as drawing custom 2D and 3D graphics.

Android Graphics

When you want to **draw shapes or text into a view** on Android, you need:

- A Canvas object.
    - Very simplified, a Canvas is a logical 2D drawing surface that provides methods for drawing onto a bitmap.
- An instance of the Bitmap class
    - which represents the physical drawing surface and gets pushed to the display by the GPU.
- A View instance associated with the bitmap.
- A Paint object
    - that holds the style and color information about how to draw geometries, text, and on bitmap.
- The Canvas class also provides methods for clipping views.
- Clipping is the action of defining geometrically what portion of the canvas the user sees in the view.
- This visible portion is called the viewport in graphics terminology.

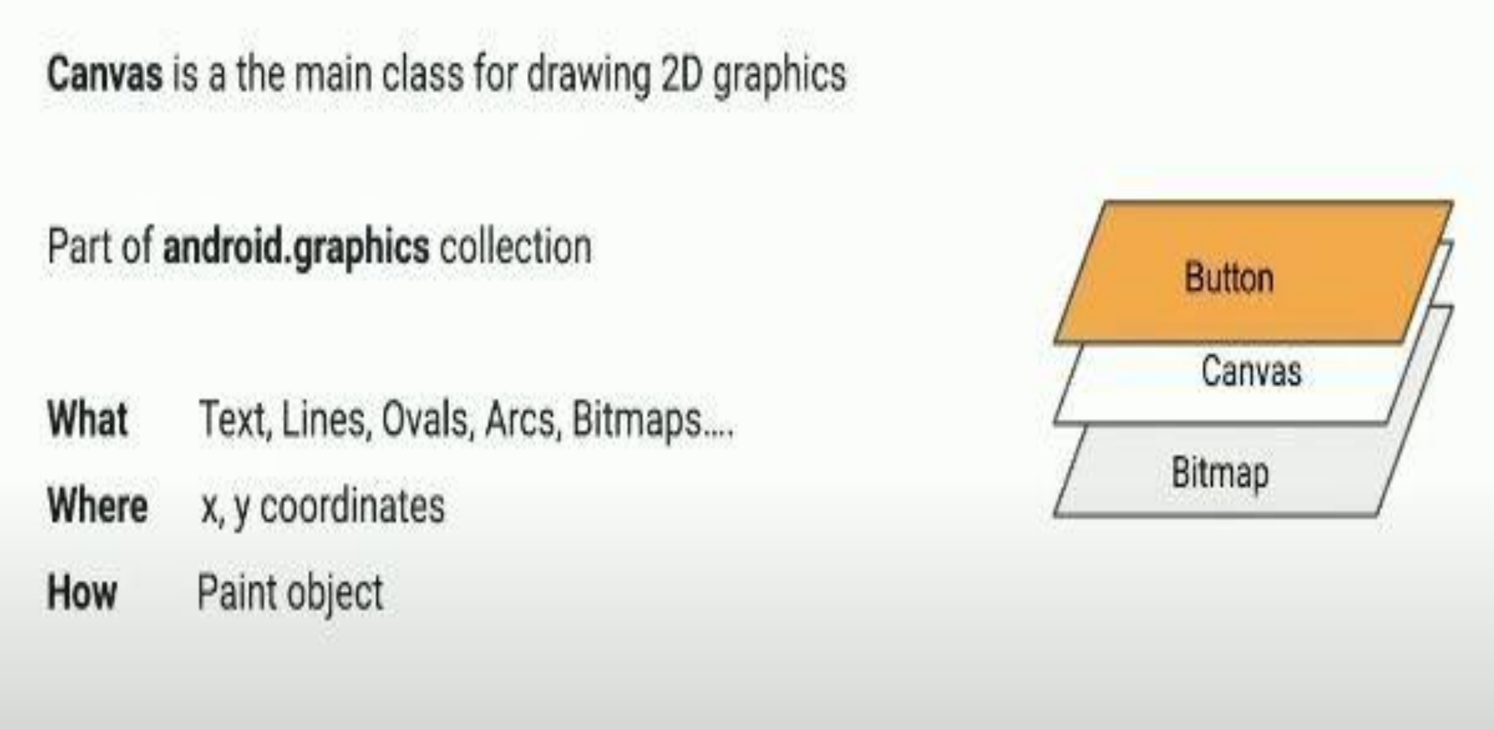
Android graphics

Make simple 2D shapes by using android.graphics package

Android.graphics : Provides low level graphics tools such as canvases, color filters, points, and rectangles that let you handle drawing to the screen directly.

- The **android.graphics.Canvas** can be used to draw graphics in android. It provides methods to draw oval, rectangle, picture, text, line etc.
- The **android.graphics.Paint** class is used with canvas to draw objects. It holds the information of color and style.

Android Graphics : Canvas



Canvas is a the main class for drawing 2D graphics

Part of **android.graphics** collection

What	Text, Lines, Ovals, Arcs, Bitmaps....
Where	x, y coordinates
How	Paint object

Every view has canvas function as **onDraw(Canvas canvas)**
To redraw : **invalidate()**

Android Graphics : Canvas

Canvas :

- Android graphics provides low level graphics tools such as canvases, color, filters, points and rectangles which handle drawing to the screen directly.
- The Android framework provides a set of 2D-DRAWING APIs which allows user to provide own custom graphics onto a canvas or to modify existing views to customize their look and feel.
- **There are two ways to draw 2D graphics**
  1. Draw your animation into a View object from your layout.
  2. Draw your animation directly to a Canvas.

Android Graphics : Canvas

Some of the important methods of Canvas Class are as follows

i) drawText()
ii) drawRoundRect()
iii) drawCircle()
iv) drawRect()
v) drawBitmap()
vi) drawARGB()

- You can use these methods in onDraw() method to create your own custom user interface.
- Drawing an animation with a View is the best option to draw simple graphics that do not need to change dynamically and are not a part of a performance-intensive game. It is used when user wants to display a static graphic or predefined animation.
- Drawing an animation with a Canvas is better option when your application needs to re-draw itself regularly. For example video games should be drawing to the Canvas on its own

Simple example

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/a
ndroid"
    xmlns:app="http://schemas.android.com/apk/res-
auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
android:layout_height="match_parent"
        android:layout_width="match_parent"
        android:orientation="vertical"
        android:id="@+id/rect"></LinearLayout>
</RelativeLayout>
```

MainActivity.java

```java
package com.example.drawingrectangle;

import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.PorterDuff;
import android.graphics.drawable.BitmapDrawable;
import android.os.Bundle;
import android.widget.LinearLayout;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Paint paint=new Paint();
        paint.setColor(Color.parseColor("#da4747"));
        Paint circle=new Paint();
        circle.setColor(getResources().getColor(R.color.circle));
        //Configuration-Each pixel is stored on 4 bytes.
        Bitmap bg=Bitmap.createBitmap(480,800, Bitmap.Config.ARGB_8888);
        Canvas canvas=new Canvas(bg);
        paint.setColor(Color.RED);
        paint.setStrokeWidth(10);
canvas.drawLine(50, 200, 200, 300, paint);
        canvas.drawLine(50, 250, 200, 200, paint);
        canvas.drawRect(50,50,100,100,paint);
       canvas.drawCircle(200,80,30,circle);
        Bitmap bitmap =
BitmapFactory.decodeResource(getResources(),
R.drawable.roses);

        canvas.drawBitmap(bitmap,null, new

Rect(50,320,450,700), null);
       paint.setColor(Color.BLUE);
       paint.setTextSize(30);
       canvas.rotate(-15);
       canvas.drawText("Moblie computing
lab",80,200,paint);
        LinearLayout ll=findViewById(R.id.rect);
        ll.setBackground(new BitmapDrawable(bg));
    }
}
```

colors.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="circle">#fa9696</color>
</resources>
```

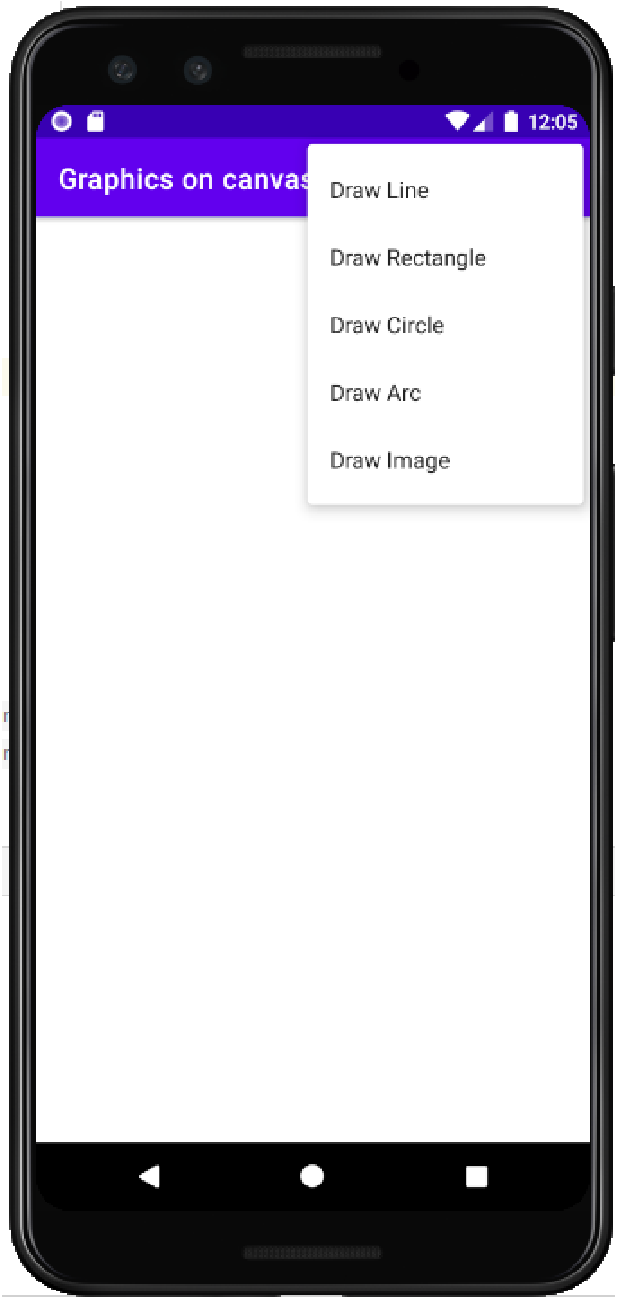
Drawing Graphics directly to a Canvas

MainActivity.java

```java
package com.example.graphics;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.graphics.RectF;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new MyView(this));
    }
    public class MyView extends View {
        public MyView(Context context)
        {
            super(context);
        }
```

```
 @Override
        protected void onDraw(Canvas canvas) {
            super.onDraw(canvas);
            int x = getWidth();
            int y = getHeight();
            int radius;
            radius = 200;
            Paint paint = new Paint();
            paint.setStyle(Paint.Style.FILL);
            paint.setColor(Color.GRAY);
            paint.setTextSize(80);
            canvas.drawPaint(paint);
            paint.setColor(Color.parseColor("#ffffff"));
            canvas.drawCircle(x / 2, y / 2, radius, paint);
            canvas.drawArc(50,200,300,300,30,100,true,paint);
            canvas.drawRoundRect(new RectF(50, 50, 150, 150), 15, 15, paint);
            Bitmap bitmap = BitmapFactory.decodeResource(getResources(),
R.drawable.roses);
            canvas.drawBitmap(bitmap,50,350,paint);
            paint.setStrokeWidth(10);
            canvas.drawLine(50, 800, 200, 900, paint);
            canvas.drawLine(50, 850, 200, 800, paint);
            canvas.rotate(-10);
            canvas.drawText("Mobile Computing Lab", 20, 450, paint);
        }
        }
    }
```

- Create an android application to draw graphics. Include option menu to display various graphics options.



- **Create Menu XML**
  - Create **menu** folder in **res** folder. Create **main.xml** file

main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/drawLine"
        android:title="Draw Line"></item>

    <item
        android:id="@+id/drawRectangle"
        android:title="Draw Rectangle"></item>

    <item
        android:id="@+id/drawCircle"
        android:title="Draw Circle"></item>

    <item
        android:id="@+id/drawArc"
        android:title="Draw Arc"></item>

    <item
        android:id="@+id/drawImage"
        android:title="Draw Image"></item>
</menu>
```

- **Draw Line**
  - Create new java class named **DrawLine.java** in package

```
DrawLine.java

package com.example.graphicsoncanvas;

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

public class DrawLine extends View {
    Paint paint = new Paint();

    public DrawLine(Context context) {
        super(context);
    }

    @Override
    public void onDraw(Canvas canvas) {
        // Draw Line
        paint.setColor(Color.RED);
        paint.setStrokeWidth(20);
        canvas.drawLine(50, 100, 600, 600, paint);
        canvas.drawLine(50, 550, 770, 0, paint);
    }
}
```

- **Draw Rectangle**
  - Create new java class named **DrawRectangle.java** in package

DrawRectangle.java

```java
package com.example.graphicsoncanvas;

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

public class DrawRectangle extends View {
    Paint paint = new Paint();

    public DrawRectangle(Context context) {
        super(context);
    }

    @Override
    public void onDraw(Canvas canvas) {
        // Draw Rectangle
        paint.setColor(Color.RED);
        paint.setStrokeWidth(5);
        canvas.drawRect(30, 30, 500, 200, paint);

    }
}
```

- **Draw Circle**
  - Create new java class named **DrawCircle.java** in package

DrawCircle.java

```java
package com.example.graphicsoncanvas;
import android.content.Context;
import android.graphics.Canvas;
import android.graphics.Color;
import android.graphics.Paint;
import android.view.View;

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

    @Override
    public void onDraw(Canvas canvas) {
        // Draw Circle
        paint.setColor(Color.RED);
        paint.setStrokeWidth(5);
        canvas.drawCircle(200, 200, 150, paint);
    }
}
```

- **Draw Arc**
  - Create new java class named **DrawArc.java** in package

DrawArc.java

```java
package com.example.graphicsoncanvas;

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

public class DrawArc extends View {
    Paint paint=new Paint();

    public DrawArc(Context context) {
        super(context);
    }

    @Override
    protected void onDraw(Canvas canvas) {
        super.onDraw(canvas);
            paint.setColor(Color.RED);
            canvas.drawArc(50,200,400,400,30,100,true,paint);
            canvas.drawArc(50,450,900,900,30,100,false,paint);

        }
}
```

- **Draw Image**
  - Create new java class named **DrawImage.java** in package

DrawImage.java

```
package com.example.graphicsoncanvas;
import android.content.Context;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.Paint;
import android.view.View;

public class DrawImage extends View {
    Paint paint = new Paint();
    public DrawImage(Context context) {
        super(context);
    }
    @Override
    public void onDraw(Canvas canvas) {
        // Draw Image
        Bitmap bitmap =
BitmapFactory.decodeResource(getResources(), R.drawable.roses);
        canvas.drawBitmap(bitmap, 100, 100, paint);
    }
}
```

- **Main Activity Class**
  - Add code to **MainActivity.java**

MainActivity.java

```
package com.example.graphicsoncanvas;

import androidx.appcompat.app.AppCompatActivity;
import android.graphics.Paint;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.main, menu);
        return super.onCreateOptionsMenu(menu);
    }
```

```
 @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        if (item.getItemId() == R.id.drawLine) {
            DrawLine drawLine = new DrawLine(this);
            setContentView(drawLine);
        } else if(item.getItemId() == R.id.drawRectangle) {
            DrawRectangle drawRectangle = new DrawRectangle(this);
            setContentView(drawRectangle);
        } else if(item.getItemId() == R.id.drawCircle) {
            setContentView(new DrawCircle(this));

        } else if(item.getItemId() == R.id.drawArc) {
           setContentView(new DrawArc(this));

        } else if(item.getItemId() == R.id.drawImage) {
             setContentView(new DrawImage(this));
        }
        return super.onOptionsItemSelected(item);
    }

}
```

References

- Bitmap.Config class
- https://docs.microsoft.com/en-us/dotnet/api/android.graphics.bitmap.config?view=xamarin-android-sdk-9