

```
<!-------#RainbowTextView.java-----!>
```

```
package com.example.app;
```

```
import android.content.Context;
import android.content.Context;
import android.graphics.*;
import android.graphics.BitmapShader;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Canvas;
import android.graphics.LinearGradient;
import android.graphics.Matrix;
import android.graphics.Paint;
import android.graphics.Shader;
import android.util.AttributeSet;
import android.view.animation.LinearInterpolator;
import android.widget.TextView;
import android.animation.ValueAnimator;
```

```
public class RainbowTextView extends TextView {
```

```
    private Paint paint;
    private Bitmap rainbowBitmap;
    private Shader shader;
    private Matrix matrix;
    private float gradientX;
    private ValueAnimator animator;
```

```
    public RainbowTextView(Context context) {
        super(context);
        init();
    }
```

```
    public RainbowTextView(Context context, AttributeSet attrs) {
        super(context, attrs);
        init();
    }
```

```
    public RainbowTextView(Context context, AttributeSet attrs, int defStyleAttr) {
        super(context, attrs, defStyleAttr);
        init();
    }
```

```
    private void init() {
        // Load the rainbow image from resources
        rainbowBitmap = BitmapFactory.decodeResource(getResources(), R.drawable.rainbow);

        // Create a shader from the rainbow bitmap
        shader = new BitmapShader(rainbowBitmap, Shader.TileMode.REPEAT, Shader.TileMode.REPEAT);

        // Create a paint object and set the shader
        paint = new Paint();
        paint.setShader(shader);

        // Create a matrix to translate the shader
        matrix = new Matrix();

        // Initialize the animator
        animator = ValueAnimator.ofFloat(0, rainbowBitmap.getWidth());
        animator.setDuration(7000); // Duration of the animation
        animator.setRepeatCount(ValueAnimator.INFINITE); // Repeat indefinitely
        animator.setInterpolator(new LinearInterpolator());
        animator.addUpdateListener(new ValueAnimator.AnimatorUpdateListener() {
            @Override
            public void onAnimationUpdate(ValueAnimator animation) {
                gradientX = (float) animation.getAnimatedValue();
                invalidate(); // Redraw the view
            }
        });
        animator.start();
    }
```

```
    @Override
    protected void onDraw(Canvas canvas) {
        // Translate the shader matrix to create the animation effect
        matrix.setTranslate(gradientX, 0);
        shader.setLocalMatrix(matrix);

        // Set the shader to the paint
        paint.setShader(shader);

        // Draw the text with the shader
        getPaint().setShader(shader);
        super.onDraw(canvas);
    }
```

```
    @Override
    protected void onDetachedFromWindow() {
        super.onDetachedFromWindow();
        if (animator != null) {
            animator.cancel();
        }
    }
}
```

```
}
```

```
<!-------#activity_main.xml-----!>
```

```
Use RainbowTextView In 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">

        <com.example.application.RainbowTextView
            android:id="@+id/rainbow_view"
            android:layout_width="match_parent"
            android:layout_height="match_parent"/>

    </RelativeLayout>

    <!-------#MainActivity-----!>

    Use RainbowTextView In MainActivity

    package com.example.RainbowTextView;

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

    public class MainActivity extends AppCompatActivity {
    private RainbowTextView RainbowTextView;

        @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_main);
            RainbowTextView = findViewById(R.id.rainbow_view); // From activity_main.xml Id

            // Without Any Xml Id By Only Java Source
            RainbowTextView rainP = new RainbowTextView(this);
            layout_name.addView(rainP); // Replace With Actual Layout Id | Layout Names($$layout_name$$)-----!

        }
    }
```