

Name of Student: Pushkar Sane		
Roll Number: 45		Lab Assignment Number: 6
Title of Lab Assignment: To perform the animation on an image and to apply various filters on an image.		
DOP: 10-09-2024		DOS: 11-10-2024
CO Mapped: CO2, CO4	PO Mapped: PO2, PO3, PO5, PS01, PS02	Signature:

Practical No. 6

Aim: To perform the animation on an image and to apply various filters on an image.

1. Perform the following animation on the image:
 - a. Move.
 - b. Rotate.
 - c. Expand.
2. Apply the following effects on the image:
 - a. Brightness.
 - b. Darkness.
 - c. Grayscale.

Theory:**Steps to Add an Image to the Drawable Folder in Android Studio**

1. Switch to Project View:
 - a. In Android Studio, look for the Project panel on the top-left.
 - b. Change the view from **Android** to **Project**.
2. Navigate to Drawable Folder:
 - a. In the Project View, expand the folder structure as follows:
app > src > main > res > drawable
3. Open Drawable Folder in File Explorer:
 - a. Right-click on the drawable folder.
 - b. Select Open in **Explorer** (Windows) or Reveal in **Finder** (macOS).
4. Add the Image: Once the drawable folder opens in your file explorer, copy your flower.jpg image file into this folder.

Code:**MainActivity.java**

```
package com.example.practical6;
import android.graphics.ColorMatrix;
import android.graphics.ColorMatrixColorFilter;
import android.os.Bundle;
import android.view.animation.ScaleAnimation;
import android.view.animation.TranslateAnimation;
import android.view.animation.RotateAnimation;
import android.view.View;
```

```
import android.widget.Button;
import android.widget.ImageView;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    private ImageView imageView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        imageView = findViewById(R.id.imageView);
        Button btnMove = findViewById(R.id.btnMove);
        Button btnRotate = findViewById(R.id.btnRotate);
        Button btnExpand = findViewById(R.id.btnExpand);
        Button btnBrightness = findViewById(R.id.btnBrightness);
        Button btnDarkness = findViewById(R.id.btnDarkness);
        Button btnGrayscale = findViewById(R.id.btnGrayscale);

        // Animation: Move
        btnMove.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                TranslateAnimation translate = new TranslateAnimation(0, 300, 0, 300);
                translate.setDuration(1000);
                translate.setFillAfter(true);
                imageView.startAnimation(translate);
            }
        });

        // Animation: Rotate
        btnRotate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                RotateAnimation rotate = new RotateAnimation(0, 360,
                    imageView.getWidth() / 2, imageView.getHeight() / 2);
                rotate.setDuration(1000);
```

```
        rotate.setFillAfter(true);
        imageView.startAnimation(rotate);
    }
});
```

// Animation: Expand

```
btnExpand.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        ScaleAnimation scale = new ScaleAnimation(1f, 2f, 1f, 2f,
            imageView.getWidth() / 2, imageView.getHeight() / 2);
        scale.setDuration(1000);
        scale.setFillAfter(true);
        imageView.startAnimation(scale);
    }
});
```

// Filter: Brightness

```
btnBrightness.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        ColorMatrix matrix = new ColorMatrix();
        matrix.set(new float[]{
            1.5f, 0, 0, 0, 0,
            0, 1.5f, 0, 0, 0,
            0, 0, 1.5f, 0, 0,
            0, 0, 0, 1, 0
        });
        imageView.setColorFilter(new ColorMatrixColorFilter(matrix));
    }
});
```

// Filter: Darkness

```
btnDarkness.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
```

```
        ColorMatrix matrix = new ColorMatrix();
        matrix.set(new float[]{
            0.5f, 0, 0, 0, 0,
            0, 0.5f, 0, 0, 0,
            0, 0, 0.5f, 0, 0,
            0, 0, 0, 1, 0
        });
        imageView.setColorFilter(new ColorMatrixColorFilter(matrix));
    }
});

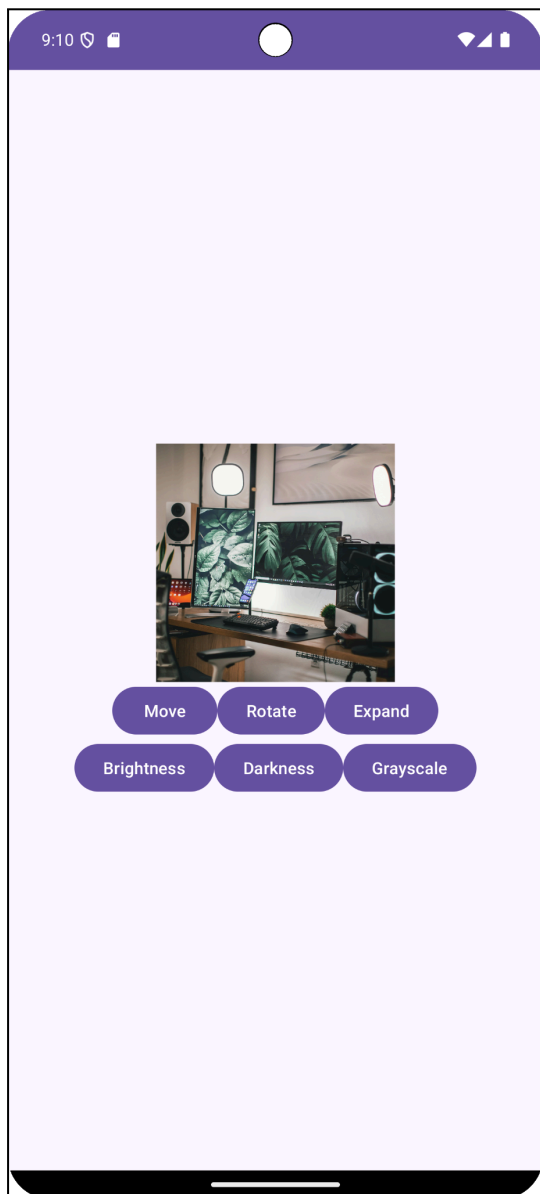
// Filter: Grayscale
btnGrayscale.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        ColorMatrix matrix = new ColorMatrix();
        matrix.setSaturation(0);
        imageView.setColorFilter(new ColorMatrixColorFilter(matrix));
    }
});
}
```

main_activity.xml

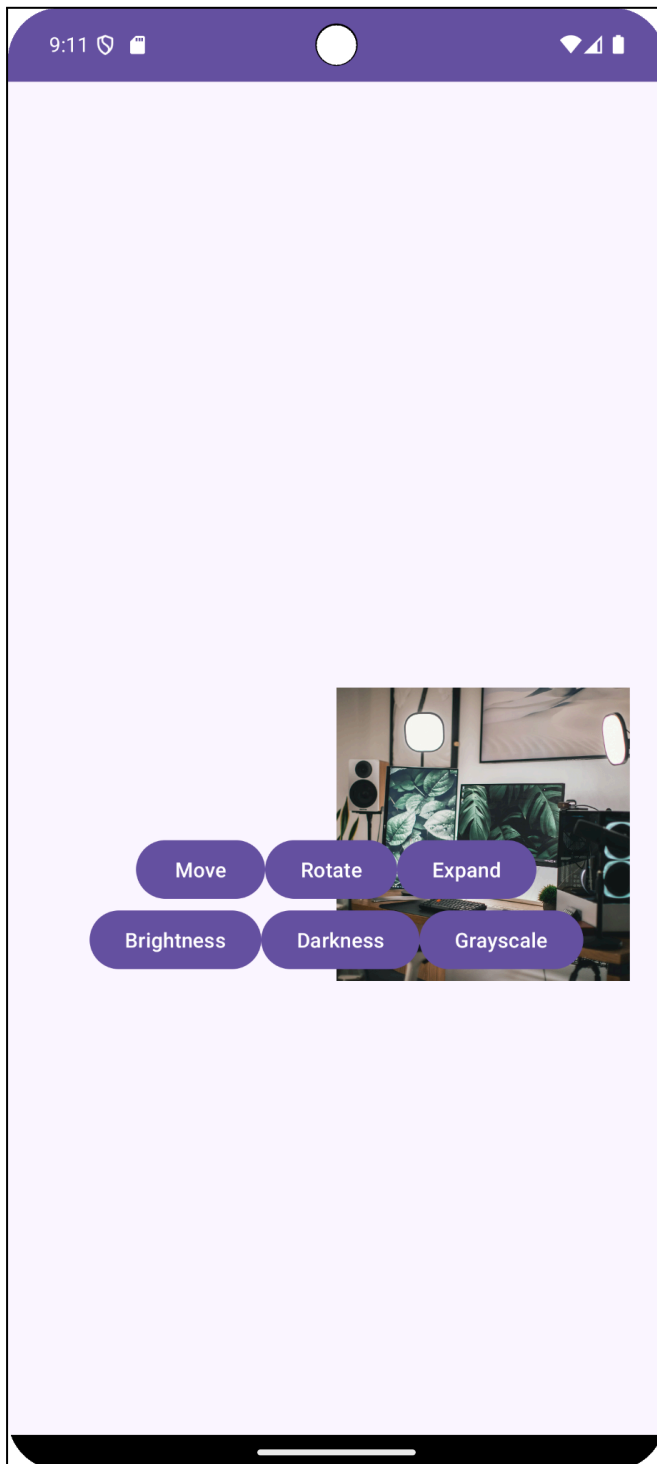
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center">
    <ImageView
        android:id="@+id/imageView"
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:src="@drawable/image"
```

```
        android:scaleType="centerCrop"/>
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <!-- Buttons for animations -->
    <Button
        android:id="@+id/btnMove"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Move"/>
    <Button
        android:id="@+id/btnRotate"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Rotate"/>
    <Button
        android:id="@+id/btnExpand"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Expand"/>
</LinearLayout>
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <!-- Buttons for filters -->
    <Button
        android:id="@+id/btnBrightness"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Brightness"/>
    <Button
        android:id="@+id/btnDarkness"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

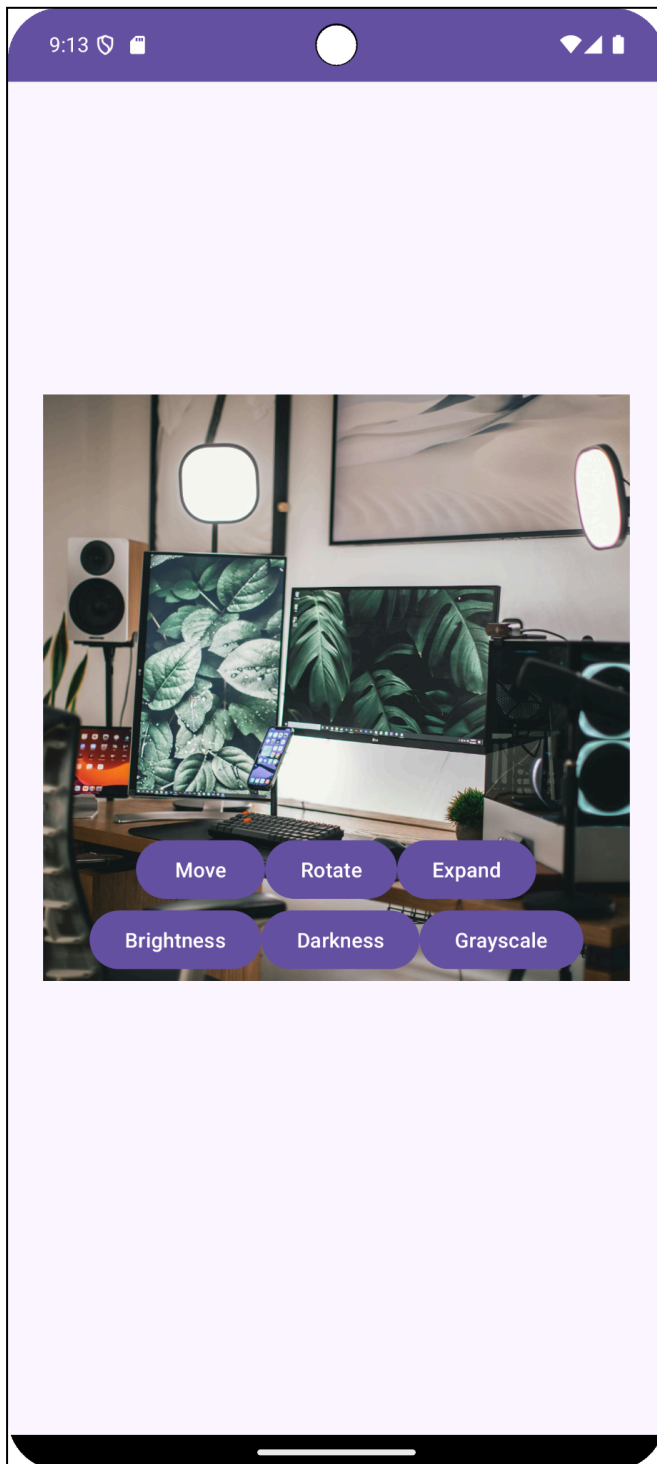
```
        android:text="Darkness"/>
    <Button
        android:id="@+id/btnGrayscale"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Grayscale"/>
    </LinearLayout>
</LinearLayout>
```

Output:

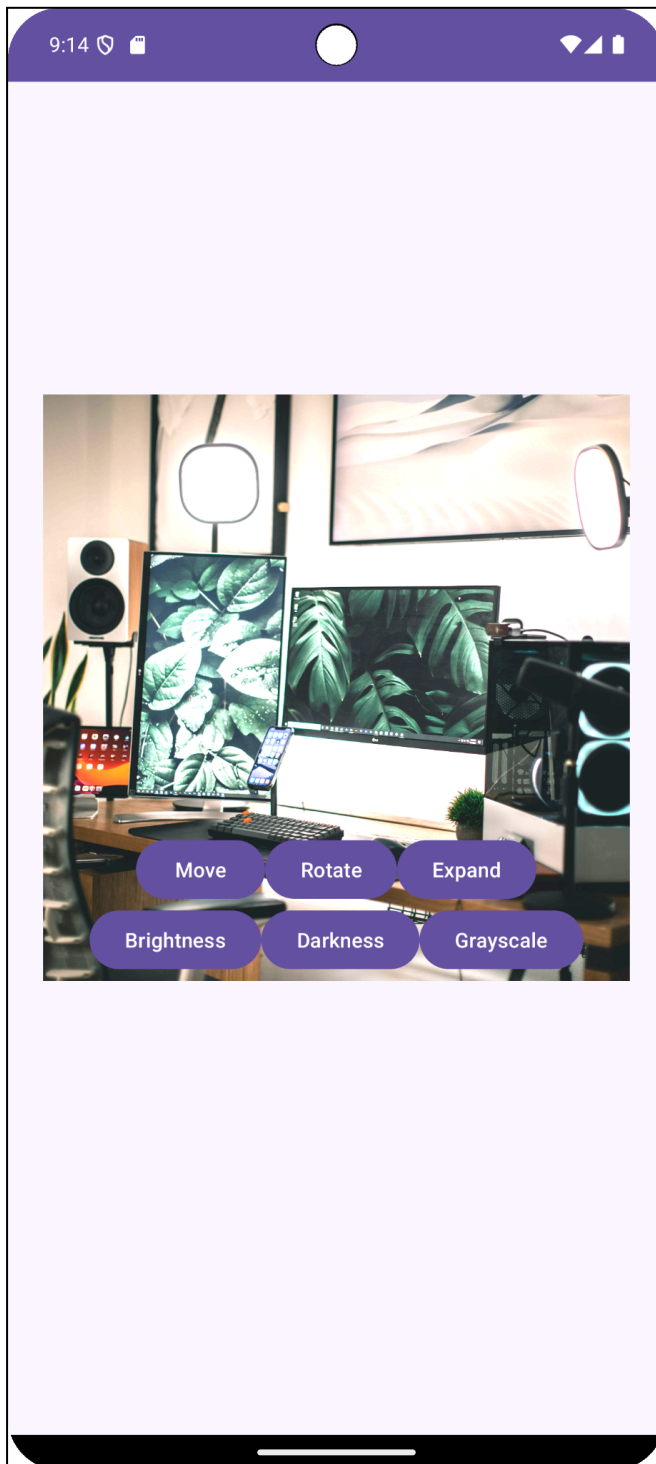
Move:



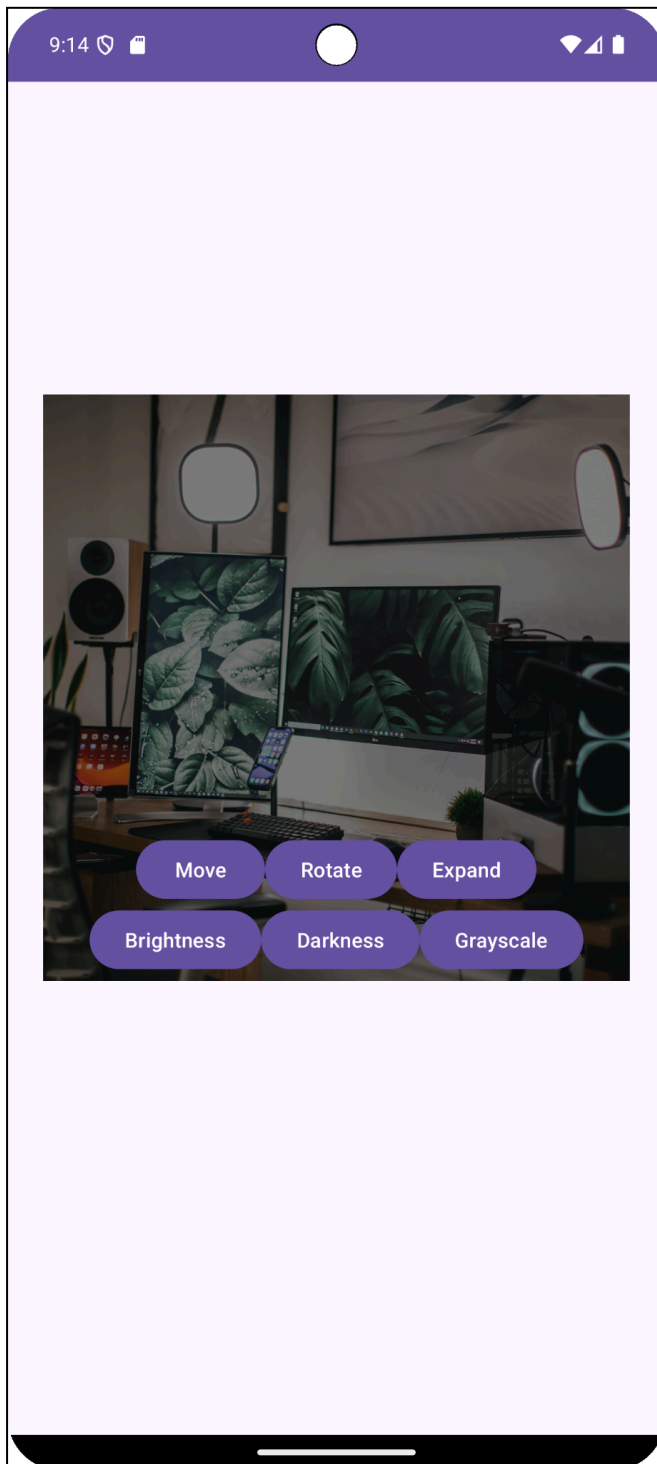
Expand:

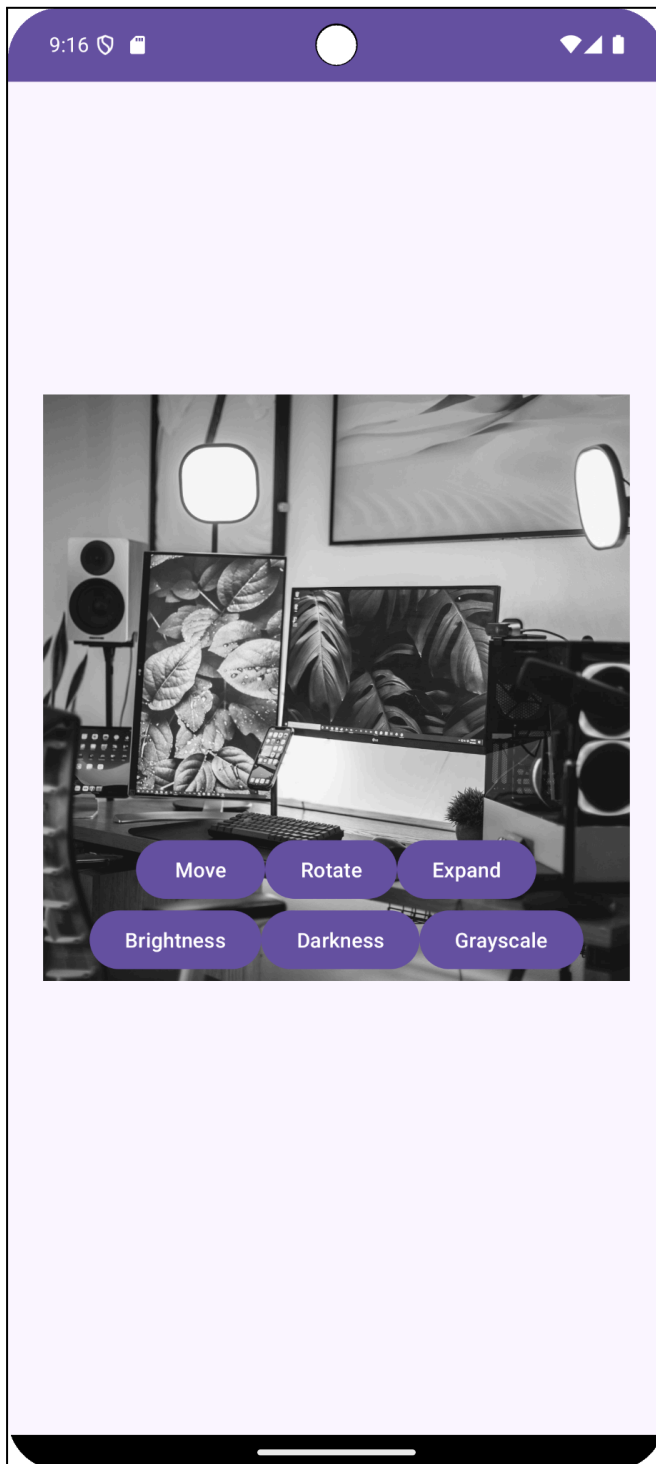


Brightness:



Darkness:



GreyScale:**Conclusion:**

Successfully demonstrated implementation of animation on the image and applying effects on the image.