

# Animation

# Creating animations with androids graphics API

Animation is a method in which pictures are manipulated to appear as moving images.

Android allows changing object properties over a certain time interval via the properties animation API

1. Basic Animation
2. Animate Drawable Graphics(rect,circle..)
3. Animation between Activities(screen)

# Basic Animation using XML

Important XML attributes that can be used in your application to have a perfect Animation :

1. **android:duration:** It is used to specify the duration of animation i.e. how long a particular Animation will continue.
2. **android:interpolator:** It is used to define the rate of change of Animation i.e. how fast a particular UI will change to another UI.
3. **android:startOffset:** When you are having a number of animations one after the other, then you have to specify some time to a particular animation and that animation will wait for that duration and after that, it will start.
4. **android:repeatMode:** This is used to repeat a particular animation.
5. **Android:repeatCount:** This is used to specify the repeat count of a particular animation.

You can set the repeat count of an animation to be infinite, if you want infinite repetition.

## Parameters

**fromXDelta**

**float:** Change in X coordinate to apply at the start of the animation

**toXDelta**

**float:** Change in X coordinate to apply at the end of the animation

**fromYDelta**

**float:** Change in Y coordinate to apply at the start of the animation

**toYDelta**

**float:** Change in Y coordinate to apply at the end of the animation

## Fade\_in.xml : res/anim

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true">
    <alpha
        android:duration="5000"
        android:fromAlpha="0.0"
        android:interpolator="@android:anim/accelerate_interpolator"
        android:toAlpha="1.0" />
</set>
```

# Fade\_Out.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<set xmlns:android="http://schemas.android.com/apk/res/android"  
    android:fillAfter="true" >  
  
    <alpha  
        android:duration="5000"  
        android:fromAlpha="1.0"  
        android:interpolator="@android:anim/accelerate_interpolator"  
        android:toAlpha="0.0" />  
  
    </set>
```

# startAnimation()

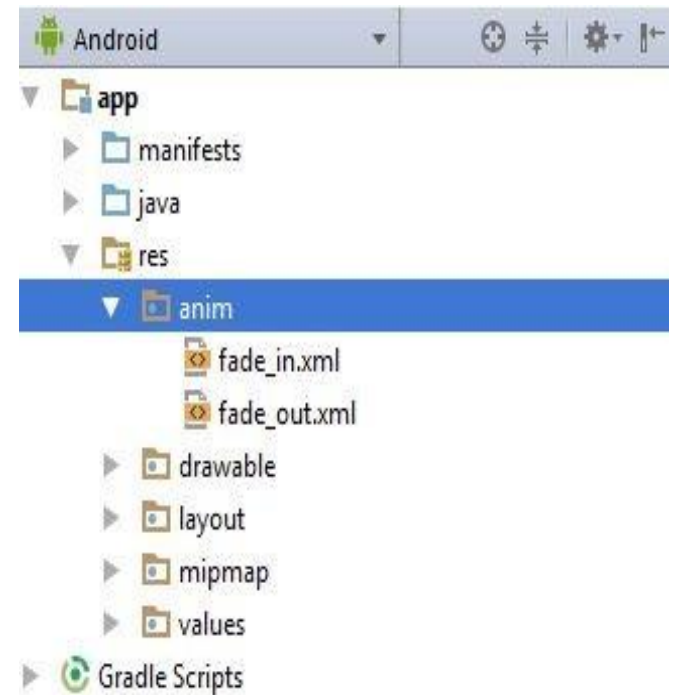
**startAnimation()** method is used to start an animation on a particular component. If you want to perform some operations before, after or during the Animation, then you can implement the AnimationListener in your call and then override the below function for the same:

1. **onAnimationStart()**: Will be called when the Animation will start.
2. **onAnimationEnd()**: Will be called when the Animation will end.
3. **onAnimationRepeat()**: Will be called when you repeat an Animation.

All you need to do is implement the AnimationListener and then set the animation listener.

# Code Snippet :

```
ImageView img = findViewById(R.id.imgvw);  
  
Animation aniFade =  
AnimationUtils.loadAnimation(getApplicationContext(), R.anim.fade_in);  
  
img.startAnimation(aniFade);
```





# zoom\_in.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true" >

    <scale
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:duration="1000"
        android:fromXScale="1"
        android:fromYScale="1"
        android:pivotX="50%"
        android:pivotY="50%"
        android:toXScale="3"
        android:toYScale="3" >

    </scale>
</set>
```

# zoom\_out.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true" >
    <scale
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:duration="1000"
        android:fromXScale="1.0"
        android:fromYScale="1.0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:toXScale="0.5"
        android:toYScale="0.5" >
    </scale>
</set>
```

## Animation for Move :To move one component from one position to other

```
<set
```

```
    xmlns:android="http://schemas.android.com/apk/res/android"
```

```
    android:interpolator="@android:anim/linear_interpolator"
```

```
    android:fillAfter="true">
```

```
        <translate
```

```
            android:fromXDelta="0%p"
```

```
            android:toXDelta="75%p"
```

```
            android:duration="800" />
```

```
</set>
```

## hyperspace\_jump.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android" android:shareInterpolator="false">
  <scale
    android:interpolator="@android:anim/accelerate_decelerate_interpolator"
    android:fromXScale="1.0"
    android:toXScale="1.4"
    android:fromYScale="1.0"
    android:toYScale="0.6"
    android:pivotX="50%"
    android:pivotY="50%"
    android:fillAfter="false"
    android:duration="700" />
  <set
    android:interpolator="@android:anim/accelerate_interpolator" android:startOffset="700">
    <scale
      android:fromXScale="1.4"
      android:toXScale="0.0"
      android:fromYScale="0.6"
      android:toYScale="0.0"
      android:pivotX="50%"
      android:pivotY="50%"
      android:duration="400" />
    <rotate
      android:fromDegrees="0"
      android:toDegrees="-45"
      android:toYScale="0.0"
      android:pivotX="50%"
      android:pivotY="50%"
      android:duration="400" />
    </set>
  </set>
```

# Example

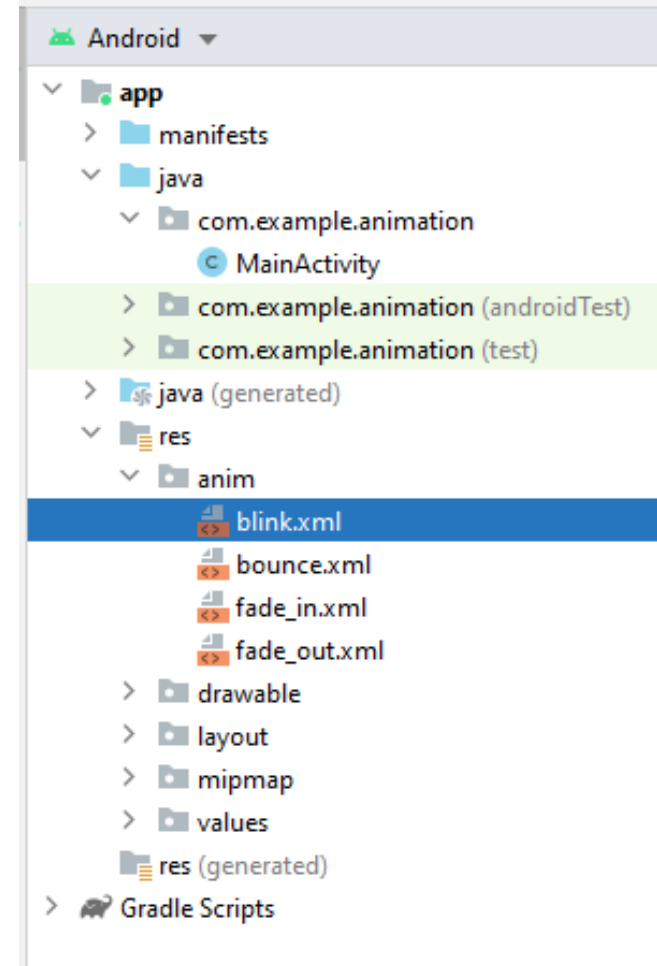
- Create an application to implement simple animation
  - Create **anim** directory in res folder
  - Create **blink.xml** in anim

*blink.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<set
xmlns:android="http://schemas.android.
com/apk/res/android">
    <alpha android:fromAlpha="0.0"
        android:toAlpha="1.0"

android:interpolator="@android:anim/ac
celerate_interpolator"
        android:duration="600"
        android:repeatMode="reverse"

android:repeatCount="infinite"/>
</set>
```



### ***bounce.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true"
        android:interpolator="@android:anim/bounce_interpolator">

    <scale
        android:duration="500"
        android:fromXScale="1.0"
        android:fromYScale="0.0"
        android:toXScale="1.0"
        android:toYScale="1.0" />

</set>
```

### ***fade\_in.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true">
    <alpha
        android:duration="1000"
        android:fromAlpha="0.0"
        android:interpolator="@android:anim/accelerate_interpolator"
        android:toAlpha="1.0" />
</set>
```

*fade\_out.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:fillAfter="true">
    <alpha
        android:duration="1000"
        android:fromAlpha="1.0"

        android:interpolator="@android:anim/accelerate_interpolator"
        android:toAlpha="0.0" />
    </set>
```

## ***activity\_main.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    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">

    <Button
        android:id="@+id/animBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="Animation Demo" />
    <ImageView
        android:id="@+id/animImage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_launcher_foreground"></ImageView>

</RelativeLayout>
```



## MainActivity.java

```
package com.example.animation;

import androidx.appcompat.app.AppCompatActivity;

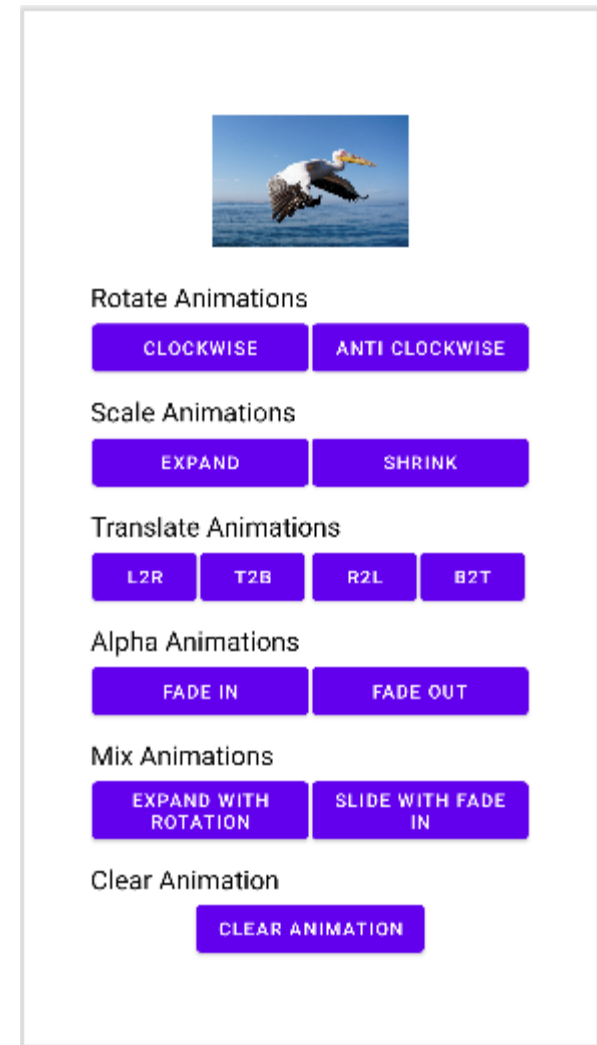
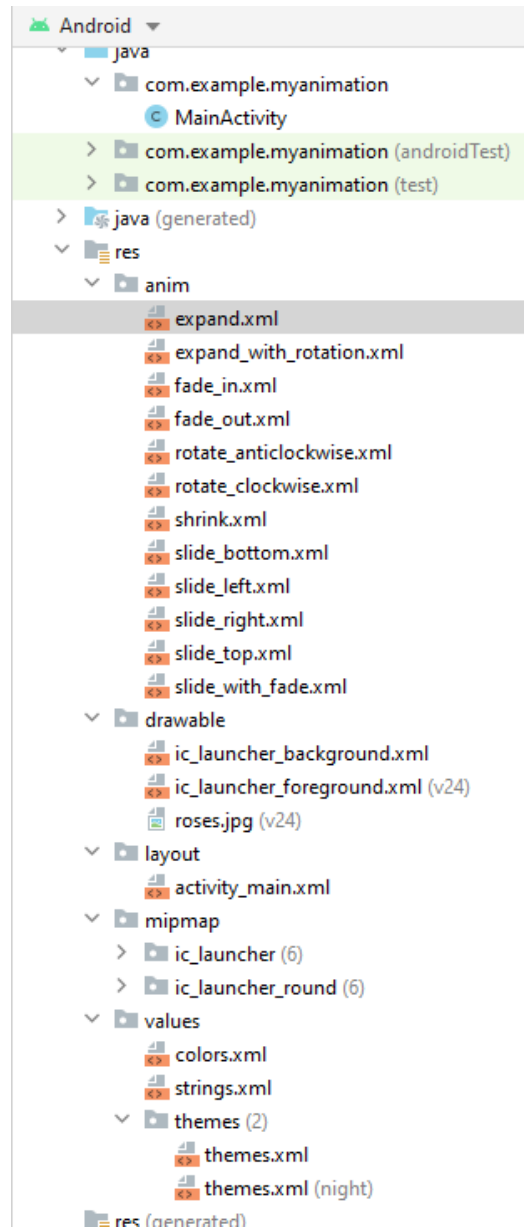
import android.os.Bundle;
import android.view.View;
import android.view.animation.AlphaAnimation;
import android.view.animation.Animation;
import android.view.animation.AnimationSet;
import android.view.animation.AnimationUtils;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {
    Button b;
    ImageView image;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        b=findViewById(R.id.animBtn);
        image=findViewById(R.id.animImage);
        Animation anim=
        AnimationUtils.makeInAnimation(getApplicationContext(),true);
        Animation out=AnimationUtils.makeOutAnimation(this,true);
```

## MainActivity.java contd...

```
b.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Animation animation =  
AnimationUtils.loadAnimation(MainActivity.this, R.ani  
m.bounce);  
        b.startAnimation(animation);  
        image.startAnimation(anim);  
        image.startAnimation(out);  
    }  
});  
}
```

- Create an android application that applies different animations on an image



## Add these styles to themes.xml

```
<!--      animation title properties      -->
<style name="animation_title_props">
    <item name="android:layout_columnSpan">4</item>
    <item name="android:textColor">@color/black</item>
    <item name="android:textSize">20sp</item>
    <item name="android:paddingTop">10dp</item>
</style>

<!--      small button properties      -->
<style name="btn_min_props">
    <item name="android:layout_columnSpan">1</item>
    <item name="android:layout_width">80dp</item>
    <item name="android:layout_margin">1dp</item>
</style>

<!--      medium button properties      -->
<style name="btn_medium_props">
    <item name="android:layout_columnSpan">2</item>
    <item name="android:layout_width">165dp</item>
    <item name="android:layout_margin">1dp</item>
</style>

<!--      large button properties      -->
<style name="btn_max_props">
    <item name="android:layout_columnSpan">4</item>
    <item name="android:layout_width">wrap_content</item>
    <item name="android:layout_gravity">center</item>
</style>
```

### ***expand.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:fromXScale="0"
        android:toXScale="2"
        android:fromYScale="0"
        android:toYScale="2"
        android:pivotX="70%"
        android:pivotY="70%"
        android:repeatCount="infinite"
        android:duration="2000" />
</set>
```

### ***expand\_with\_rotation.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:fromXScale="0"
        android:toXScale="1"
        android:fromYScale="0"
        android:toYScale="1"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="2000" />
    <rotate
        android:fromDegrees="0"
        android:toDegrees="360"
        android:pivotX="50%"
        android:pivotY="50%"
        android:repeatCount="0"
        android:duration="2500" />
</set>
```

### ***fade-in.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha
        android:fromAlpha="0"
        android:toAlpha="1"
        android:duration="2000" />
</set>
```

### ***fade\_out.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha
        android:fromAlpha="1"
        android:toAlpha="0"
        android:duration="2000" />
</set>
```

### ***rotate\_anticlockwise.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <rotate
        android:fromDegrees="360"
        android:toDegrees="0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:repeatCount="infinite"
        android:duration="3000" />
</set>
```

### ***rotate\_clockwise.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <rotate
        android:fromDegrees="0"
        android:toDegrees="360"
        android:pivotX="50%"
        android:pivotY="50%"
        android:repeatCount="infinite"
        android:duration="3000" />
</set>
```

### ***shrink.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:fromXScale="1"
        android:toXScale="0"
        android:fromYScale="1"
        android:toYScale="0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="2000" />
</set>
```

### ***slide\_bottom.xml***

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromYDelta="-100%"
        android:toYDelta="0%"
        android:duration="2000" />
</set>
```

### **slide\_left.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="100%"
        android:toXDelta="0%"
        android:duration="2000" />
</set>
```

### **slide\_right.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set
xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="-100%"
        android:toXDelta="0%"
        android:duration="2000" />
</set>
```

### **slide\_top.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromYDelta="100%"
        android:toYDelta="0%"
        android:duration="2000" />
</set>
```

### **slide\_with\_fade**

```
<?xml version="1.0" encoding="utf-8"?>
<set
xmlns:android="http://schemas.android.com/apk/res/
android">
    <translate
        android:fromXDelta="-150%"
        android:toXDelta="0%"
        android:duration="2000" />
    <alpha
        android:fromAlpha="0"
        android:toAlpha="1"
        android:duration="3000" />
</set>
```



## **activity\_main.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:columnCount="4"
    android:layout_gravity="center"
    tools:context=".MainActivity">

    <!--      this layout holds image which won't
    overflow out of this frame during animations      -->
    <FrameLayout
        android:padding="5dp"
        android:layout_gravity="center"
        android:layout_columnSpan="4">

        <ImageView
            android:layout_width="150dp"
            android:layout_height="120dp"
            android:id="@+id/bird"
            android:src="@drawable/bird" />

    </FrameLayout>
```

```
<!--      button to perform different animation      -->
```

```
<TextView  
    style="@style/animation_title_props"  
    android:text="Rotate Animations" />
```

```
<Button  
    style="@style/btn_medium_props"  
    android:text="Clockwise"  
    android:onClick="clockwise"/>
```

```
<Button  
    style="@style/btn_medium_props"  
    android:text="Anti Clockwise"  
    android:onClick="antiClockwise"  
    android:layout_width="170dp"/>
```

```
<TextView  
    style="@style/animation_title_props"  
    android:text="Scale Animations" />
```

```
<Button  
    style="@style/btn_medium_props"  
    android:text="Expand"  
    android:onClick="expand"/>
```

```
<Button  
    style="@style/btn_medium_props"  
    android:text="Shrink"  
    android:onClick="shrink"/>
```

```
<TextView
    style="@style/animation_title_props"
    android:text="Translate Animations" />

<Button
    style="@style/btn_min_props"
    android:text="L2R"
    android:onClick="slideL2R"/>

<Button
    style="@style/btn_min_props"
    android:text="T2B"
    android:onClick="slideT2B"/>

<Button
    style="@style/btn_min_props"
    android:text="R2L"
    android:onClick="slideR2L"/>

<Button
    style="@style/btn_min_props"
    android:text="B2T"
    android:onClick="slideB2T"/>

<TextView
    style="@style/animation_title_props"
    android:text="Alpha Animations" />

<Button
    style="@style/btn_medium_props"
    android:text="Fade In"
    android:onClick="fadeIn"/>
```

```
<Button
    style="@style/btn_medium_props"
    android:text="Fade Out"
    android:onClick="fadeOut"/>

<TextView
    style="@style/animation_title_props"
    android:text="Mix Animations" />

<Button
    style="@style/btn_medium_props"
    android:text="Expand with Rotation"
    android:onClick="expandWithRotation"/>

<Button
    style="@style/btn_medium_props"
    android:text="Slide with Fade In"
    android:onClick="slideWithFadeIn"/>

<TextView
    style="@style/animation_title_props"
    android:text="Clear Animation" />

<Button
    style="@style/btn_max_props"
    android:text="Clear Animation"
    android:onClick="clearAnimation"/>

</GridLayout>
```

## MainActivity.java

```
package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ImageView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    ImageView bird;
    Animation animation;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        bird=findViewById(R.id.bird);
    }
    public void clockwise(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.rotate_clockwise);
        bird.startAnimation(animation);
        Toast.makeText(this, "Rotating Clockwise...",
Toast.LENGTH_SHORT).show();
    }

    public void antiClockwise(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.rotate_anticlockwise);
        bird.startAnimation(animation);
        Toast.makeText(this, "Rotating Anti-Clockwise...",
Toast.LENGTH_SHORT).show();
    }
}
```

## MainActivity.java contd..

```
public void expand(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.expand);
    bird.startAnimation(animation);
    Toast.makeText(this, "Expanding...", Toast.LENGTH_SHORT).show();
}

public void shrink(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.shrink);
    bird.startAnimation(animation);
    Toast.makeText(this, "Shrinking Scaling...", Toast.LENGTH_SHORT).show();
}

public void slideL2R(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.slide_right);
    bird.startAnimation(animation);
    Toast.makeText(this, "Sliding Left to Right...",
Toast.LENGTH_SHORT).show();
}

public void slideT2B(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.slide_bottom);
    bird.startAnimation(animation);
    Toast.makeText(this, "Sliding Top to Bottom...",
Toast.LENGTH_SHORT).show();
}
```

## MainActivity.java contd..

```
public void slideR2L(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.slide_left);
    bird.startAnimation(animation);
    Toast.makeText(this, "Sliding Right to Left...",
Toast.LENGTH_SHORT).show();
}

    public void slideB2T(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.slide_top);
        bird.startAnimation(animation);
        Toast.makeText(this, "Sliding Bottom to Top...",
Toast.LENGTH_SHORT).show();
    }

    public void fadeIn(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.fade_in);
        bird.startAnimation(animation);
        Toast.makeText(this, "Fading In...", Toast.LENGTH_SHORT).show();
    }

    public void fadeOut(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.fade_out);
        bird.startAnimation(animation);
        Toast.makeText(this, "Fading Out...", Toast.LENGTH_SHORT).show();
    }
}
```

## MainActivity.java contd..

```
public void expandWithRotation(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.expand_with_rotation);
    bird.startAnimation(animation);
    Toast.makeText(this, "Expanding with Rotation...",
Toast.LENGTH_SHORT).show();
}

    public void slideWithFadeIn(View view) {
        animation = AnimationUtils.loadAnimation(MainActivity.this,
R.anim.slide_with_fade);
        bird.startAnimation(animation);
        Toast.makeText(this, "Sliding with Fade In...",
Toast.LENGTH_SHORT).show();
    }

    public void clearAnimation(View view) {
        bird.clearAnimation();
        Toast.makeText(this, "Animation Cleared...",
Toast.LENGTH_SHORT).show();
    }
}
```



# Reference

<https://developer.android.com/guide/topics/resources/animation-resource>