**Bài tập 1: Sử dụng Animation từ file XML**

Bước 1:Tạo file trong app/src/main/res/values/dimens.xml

<resources>

<dimen name="margin\_base">5dp</dimen>

<dimen name="margin\_basex2">10dp</dimen>

<dimen name="text\_medium">16sp</dimen>

<dimen name="image\_base">100dp</dimen>

</resources>

Bước 2: Tạo file trong app/src/main/res/values/strings.xml

<resources>

<string name="app\_name">LAB6</string>

<string name="from\_xml">From XML</string>

<string name="from\_code">From CODE</string>

<string name="fade\_in">Fade In</string>

<string name="fade\_out">Fade Out</string>

<string name="blink">Blink</string>

<string name="zoom\_in">Zoom In</string>

<string name="zoom\_out">Zoom Out</string>

<string name="rotate">Rotate</string>

<string name="move">Move</string>

<string name="slide\_up">Slide Up</string>

<string name="bounce">Bounce</string>

<string name="combine">Combine</string>

</resources>Bước 3: Thêm logo

A computer screen shot of a logo

AI-generated content may be incorrect.

Bước 4: Tạo giao diện trong activity\_main.xml

A screenshot of a computer

AI-generated content may be incorrect.

Bước 5: Tạo thư mục anim trong thư mục res và tạo các tệp

-Tạo anim\_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:toAlpha="1.0"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_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:toAlpha="0.0"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo 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:duration="300" android:repeatMode="reverse" android:repeatCount="3"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_zoom\_in.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android" android:fillAfter="true">

<scale android:duration="1000" android:fromXScale="1" android:fromYScale="1"

android:pivotX="50%" android:pivotY="50%" android:toXScale="3" android:toYScale="3"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_zoom\_out.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android" android:fillAfter="true">

<scale 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"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_rotate.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:duration="600"

android:repeatMode="restart" android:repeatCount="2"

android:interpolator="@android:anim/cycle\_interpolator"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_move.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android" android:fillAfter="true">

<translate android:fromXDelta="0%p" android:toXDelta="75%p" android:duration="800"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_slide\_up.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android" android:fillAfter="true">

<scale android:duration="500" android:fromXScale="1.0" android:fromYScale="1.0"

android:toXScale="1.0" android:toYScale="0.0"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

-Tạo anim\_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/linear\_interpolator">

<scale android:duration="4000" android:fromXScale="1" android:fromYScale="1"

android:pivotX="50%" android:pivotY="50%" android:toXScale="3" android:toYScale="3"/>

<rotate android:duration="500" android:fromDegrees="0" android:pivotX="50%"

android:pivotY="50%" android:repeatCount="2" android:repeatMode="restart"

android:toDegrees="360"/>

</set>

A screenshot of a computer

AI-generated content may be incorrect.

Bước 6: Khai báo biến và xây dựng các hàm findViewsbyIds và InitVariables trong Mainactivity.Java

package com.example.lab6;  
  
import android.os.Bundle;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.Button;  
import android.widget.ImageView;  
import android.widget.Toast;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
 private ImageView ivXiaomiLogo;  
 private Animation.AnimationListener animationListener;  
  
 // 10 nút chỉ dùng XML trong Bài 1  
 private Button btnFadeInXml, btnFadeOutXml, btnBlinkXml, btnZoomInXml, btnZoomOutXml,  
 btnRotateXml, btnMoveXml, btnSlideUpXml, btnBounceXml, btnCombineXml;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(this);  
 setContentView(R.layout.*activity\_main*);  
 ViewCompat.*setOnApplyWindowInsetsListener*(findViewById(R.id.*main*), (v, insets) -> {  
 Insets sb = insets.getInsets(WindowInsetsCompat.Type.*systemBars*());  
 v.setPadding(sb.left, sb.top, sb.right, sb.bottom);  
 return insets;  
 });  
  
 // Ánh xạ view  
 ivXiaomiLogo = findViewById(R.id.*iv\_xiaomi\_logo*);  
 btnFadeInXml = findViewById(R.id.*btn\_fade\_in\_xml*);  
 btnFadeOutXml = findViewById(R.id.*btn\_fade\_out\_xml*);  
 btnBlinkXml = findViewById(R.id.*btn\_blink\_xml*);  
 btnZoomInXml = findViewById(R.id.*btn\_zoom\_in\_xml*);  
 btnZoomOutXml = findViewById(R.id.*btn\_zoom\_out\_xml*);  
 btnRotateXml = findViewById(R.id.*btn\_rotate\_xml*);  
 btnMoveXml = findViewById(R.id.*btn\_move\_xml*);  
 btnSlideUpXml = findViewById(R.id.*btn\_slide\_up\_xml*);  
 btnBounceXml = findViewById(R.id.*btn\_bounce\_xml*);  
 btnCombineXml = findViewById(R.id.*btn\_combine\_xml*);  
 animationListener = new Animation.AnimationListener() {  
 @Override public void onAnimationStart(Animation animation) { }  
 @Override public void onAnimationEnd(Animation animation) {  
 Toast.*makeText*(MainActivity.this, "Animation Stopped", Toast.*LENGTH\_SHORT*).show();  
 }  
 @Override public void onAnimationRepeat(Animation animation) { }  
 };  
 bindXml(btnFadeInXml, R.anim.*anim\_fade\_in*);  
 bindXml(btnFadeOutXml, R.anim.*anim\_fade\_out*);  
 bindXml(btnBlinkXml, R.anim.*anim\_blink*);  
 bindXml(btnZoomInXml, R.anim.*anim\_zoom\_in*);  
 bindXml(btnZoomOutXml, R.anim.*anim\_zoom\_out*);  
 bindXml(btnRotateXml, R.anim.*anim\_rotate*);  
 bindXml(btnMoveXml, R.anim.*anim\_move*);  
 bindXml(btnSlideUpXml, R.anim.*anim\_slide\_up*);  
 bindXml(btnBounceXml, R.anim.*anim\_bounce*);  
 bindXml(btnCombineXml, R.anim.*anim\_combine*);  
 }  
 private void bindXml(Button btn, int animRes) {  
 btn.setOnClickListener(v -> {  
 Animation a = AnimationUtils.*loadAnimation*(getApplicationContext(), animRes);  
 a.setAnimationListener(animationListener);  
 ivXiaomiLogo.startAnimation(a);  
 });  
 }  
}

**Bài tập 2: Sử dụng Animation từ code**

Bước 1: Xây dựng hàm handleClickAnimationCode và tất cả các button còn lại  
package com.example.lab6;

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.view.animation.RotateAnimation;

import android.view.animation.ScaleAnimation;

import android.view.animation.TranslateAnimation;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.Toast;

import androidx.activity.EdgeToEdge;

import androidx.appcompat.app.AppCompatActivity;

import androidx.core.graphics.Insets;

import androidx.core.view.ViewCompat;

import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

private ImageView ivXiaomiLogo;

private Animation.AnimationListener animationListener;

private Button btnFadeInXml, btnFadeOutXml, btnBlinkXml, btnZoomInXml, btnZoomOutXml,

btnRotateXml, btnMoveXml, btnSlideUpXml, btnBounceXml, btnCombineXml;

private Button btnFadeInCode, btnFadeOutCode, btnBlinkCode, btnZoomInCode, btnZoomOutCode,

btnRotateCode, btnMoveCode, btnSlideUpCode, btnBounceCode, btnCombineCode;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

EdgeToEdge.enable(this);

setContentView(R.layout.activity\_main);

View rootMaybe = findViewById(R.id.main);

if (rootMaybe != null) {

ViewCompat.setOnApplyWindowInsetsListener(rootMaybe, (v, insets) -> {

Insets sb = insets.getInsets(WindowInsetsCompat.Type.systemBars());

v.setPadding(sb.left, sb.top, sb.right, sb.bottom);

return insets;

});

}

mapViews();

initListener();

handleClickAnimationXml(btnFadeInXml, R.anim.anim\_fade\_in);

handleClickAnimationXml(btnFadeOutXml, R.anim.anim\_fade\_out);

handleClickAnimationXml(btnBlinkXml, R.anim.anim\_blink);

handleClickAnimationXml(btnZoomInXml, R.anim.anim\_zoom\_in);

handleClickAnimationXml(btnZoomOutXml, R.anim.anim\_zoom\_out);

handleClickAnimationXml(btnRotateXml, R.anim.anim\_rotate);

handleClickAnimationXml(btnMoveXml, R.anim.anim\_move);

handleClickAnimationXml(btnSlideUpXml, R.anim.anim\_slide\_up);

handleClickAnimationXml(btnBounceXml, R.anim.anim\_bounce);

handleClickAnimationXml(btnCombineXml, R.anim.anim\_combine);

handleClickAnimationCode(btnFadeInCode, buildFadeIn());

handleClickAnimationCode(btnFadeOutCode, buildFadeOut());

handleClickAnimationCode(btnBlinkCode, buildBlink());

handleClickAnimationCode(btnZoomInCode, buildZoomIn());

handleClickAnimationCode(btnZoomOutCode, buildZoomOut());

handleClickAnimationCode(btnRotateCode, buildRotate());

handleClickAnimationCode(btnMoveCode, buildMove());

handleClickAnimationCode(btnSlideUpCode, buildSlideUp());

handleClickAnimationCode(btnBounceCode, buildBounce());

handleClickAnimationCode(btnCombineCode, buildCombine());

}

private void mapViews() {

// ImageView logo (layout: android:id="@+id/iv\_xiaomi\_logo")

ivXiaomiLogo = findViewById(R.id.iv\_xiaomi\_logo);

// --- XML buttons ---

btnFadeInXml = findViewById(R.id.btn\_fade\_in\_xml);

btnFadeOutXml = findViewById(R.id.btn\_fade\_out\_xml);

btnBlinkXml = findViewById(R.id.btn\_blink\_xml);

btnZoomInXml = findViewById(R.id.btn\_zoom\_in\_xml);

btnZoomOutXml = findViewById(R.id.btn\_zoom\_out\_xml);

btnRotateXml = findViewById(R.id.btn\_rotate\_xml);

btnMoveXml = findViewById(R.id.btn\_move\_xml);

btnSlideUpXml = findViewById(R.id.btn\_slide\_up\_xml);

btnBounceXml = findViewById(R.id.btn\_bounce\_xml);

btnCombineXml = findViewById(R.id.btn\_combine\_xml);

// --- CODE buttons ---

btnFadeInCode = findViewById(R.id.btn\_fade\_in\_code);

btnFadeOutCode = findViewById(R.id.btn\_fade\_out\_code);

btnBlinkCode = findViewById(R.id.btn\_blink\_code);

btnZoomInCode = findViewById(R.id.btn\_zoom\_in\_code);

btnZoomOutCode = findViewById(R.id.btn\_zoom\_out\_code);

btnRotateCode = findViewById(R.id.btn\_rotate\_code);

btnMoveCode = findViewById(R.id.btn\_move\_code);

btnSlideUpCode = findViewById(R.id.btn\_slide\_up\_code);

btnBounceCode = findViewById(R.id.btn\_bounce\_code);

btnCombineCode = findViewById(R.id.btn\_combine\_code);

}

private void initListener() {

animationListener = new Animation.AnimationListener() {

@Override public void onAnimationStart(Animation animation) { }

@Override public void onAnimationEnd(Animation animation) {

Toast.makeText(MainActivity.this, "Animation Stopped", Toast.LENGTH\_SHORT).show();

}

@Override public void onAnimationRepeat(Animation animation) { }

};

}

private void handleClickAnimationXml(Button btn, int animRes) {

btn.setOnClickListener(v -> {

Animation a = AnimationUtils.loadAnimation(getApplicationContext(), animRes);

a.setAnimationListener(animationListener);

ivXiaomiLogo.startAnimation(a);

});

}

private void handleClickAnimationCode(Button btn, final Animation animation) {

btn.setOnClickListener(new View.OnClickListener() {

@Override public void onClick(View v) {

if (animationListener != null) animation.setAnimationListener(animationListener);

ivXiaomiLogo.startAnimation(animation);

}

});

}

private Animation buildFadeIn() {

AlphaAnimation a = new AlphaAnimation(0f, 1f);

a.setDuration(1000); a.setFillAfter(true);

return a;

}

private Animation buildFadeOut() {

AlphaAnimation a = new AlphaAnimation(1f, 0f);

a.setDuration(1000); a.setFillAfter(true);

return a;

}

private Animation buildBlink() {

AlphaAnimation a = new AlphaAnimation(0f, 1f);

a.setDuration(300);

a.setRepeatMode(Animation.REVERSE);

a.setRepeatCount(3);

return a;

}

private Animation buildZoomIn() {

ScaleAnimation a = new ScaleAnimation(1f, 3f, 1f, 3f,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, .5f);

a.setDuration(1000); a.setFillAfter(true);

return a;

}

private Animation buildZoomOut() {

ScaleAnimation a = new ScaleAnimation(1f, .5f, 1f, .5f,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, .5f);

a.setDuration(1000); a.setFillAfter(true);

return a;

}

private Animation buildRotate() {

RotateAnimation a = new RotateAnimation(0, 360,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, .5f);

a.setDuration(600);

a.setRepeatCount(2);

a.setRepeatMode(Animation.RESTART);

return a;

}

private Animation buildMove() {

TranslateAnimation a = new TranslateAnimation(

Animation.RELATIVE\_TO\_PARENT, 0f, Animation.RELATIVE\_TO\_PARENT, .75f,

Animation.RELATIVE\_TO\_PARENT, 0f, Animation.RELATIVE\_TO\_PARENT, 0f);

a.setDuration(800); a.setFillAfter(true);

return a;

}

private Animation buildSlideUp() {

ScaleAnimation a = new ScaleAnimation(1f, 1f, 1f, 0f,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, 1f);

a.setDuration(500); a.setFillAfter(true);

return a;

}

private Animation buildBounce() {

ScaleAnimation a = new ScaleAnimation(1f, 1f, 0f, 1f,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, 1f);

a.setDuration(500);

a.setInterpolator(getApplicationContext(), android.R.anim.bounce\_interpolator);

return a;

}

private Animation buildCombine() {

AnimationSet set = new AnimationSet(true);

set.setInterpolator(getApplicationContext(), android.R.anim.linear\_interpolator);

ScaleAnimation zoom = new ScaleAnimation(1f, 3f, 1f, 3f,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, .5f);

zoom.setDuration(4000);

set.addAnimation(zoom);

RotateAnimation rotate = new RotateAnimation(0, 360,

Animation.RELATIVE\_TO\_SELF, .5f, Animation.RELATIVE\_TO\_SELF, .5f);

rotate.setDuration(500);

rotate.setRepeatCount(2);

rotate.setRepeatMode(Animation.RESTART);

set.addAnimation(rotate);

set.setFillAfter(true);

return set;

}

}

**Bài tập 3: Tạo animation khi start Activity mới**

Bước 1: Tạo Activity thứ 2 trong app/src/main/res/layout/activity\_second.xml

<?xml version="1.0" encoding="utf-8"?>

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

android:id="@+id/second\_root"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:layout\_gravity="center"

android:text="Second Activity"

android:textSize="24sp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"/>

</FrameLayout>

Bước 2: bổ sung màu trong res/values/colors.xml  
<resources>

<color name="xiaomi\_orange">#FF6900</color>

<color name="xiaomi\_orange\_dark">#E85F00</color>

<color name="xiaomi\_orange\_light">#FFF3E8</color> <!-- nền nhạt -->

<color name="text\_on\_orange">#FFFFFFFF</color>

</resources>

Bước 3: Thêm style cho SecondActivity trong res/values/themes.xml

<?xml version="1.0" encoding="utf-8"?>

<resources xmlns:tools="http://schemas.android.com/tools">

<!-- Base theme do Android Studio tạo -->

<style name="Base.Theme.LAB6" parent="Theme.Material3.DayNight.NoActionBar">

<!-- để trống hoặc tuỳ chỉnh thêm -->

</style>

<style name="Theme.LAB6" parent="Base.Theme.LAB6" />

<style name="Theme.LAB6.SecondActivity" parent="Theme.Material3.Light.NoActionBar">

<!-- Màu thương hiệu -->

<item name="colorPrimary">@color/xiaomi\_orange</item>

<item name="colorSecondary">@color/xiaomi\_orange</item>

<!-- Thanh hệ thống -->

<item name="android:statusBarColor">@color/xiaomi\_orange</item>

<item name="android:navigationBarColor">@color/xiaomi\_orange</item>

<item name="android:windowLightStatusBar">true</item>

<item name="android:windowLightNavigationBar">true</item>

<!-- Nền activity -->

<item name="android:windowBackground">@color/xiaomi\_orange\_light</item>

</style>

</resources>

Bước 3: Tạo Secondactivity.Java

package com.example.lab6;  
  
import android.os.Bundle;  
  
import androidx.activity.EdgeToEdge;  
import androidx.activity.OnBackPressedCallback;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class SecondActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(this); // có thể bỏ nếu không cần  
 setContentView(R.layout.*activity\_second*);  
 getOnBackPressedDispatcher().addCallback(this, new OnBackPressedCallback(true) {  
 @Override  
 public void handleOnBackPressed() {  
 finish(); // đóng activity này  
 overridePendingTransition(R.anim.*in\_from\_left*, R.anim.*out\_to\_right*);  
 }  
 });  
 }  
}

Bước 4: tạo in\_from\_right.xml trong mục anim

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">

<translate android:fromXDelta="100%p" android:toXDelta="0" android:duration="300"/>

</set>

Bước 5: tạo out\_to\_left.xml trong anim

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">

<translate android:fromXDelta="0" android:toXDelta="-100%p" android:duration="300"/>

</set>

Bước 6: in\_from\_left.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">

<translate android:fromXDelta="-100%p" android:toXDelta="0" android:duration="300"/>

</set>

Bước 7: out\_to\_right.xml

<?xml version="1.0" encoding="utf-8"?>

<set xmlns:android="http://schemas.android.com/apk/res/android">

<translate android:fromXDelta="0" android:toXDelta="100%p" android:duration="300"/>

</set>