

BÀI 2: ACTIVITY

I. Tóm tắt nội dung thực hành

1. Yêu cầu lý thuyết

Activity là một thành phần cơ bản và quan trọng trong phát triển ứng dụng di động Android. Nó đại diện cho một màn hình cụ thể mà người dùng có thể tương tác, chẳng hạn như một màn hình đăng nhập, trang chính của ứng dụng, hoặc cài đặt.

- **Đặc điểm:** Mỗi Activity đều có một lớp Java tương ứng được viết để quản lý hành vi của nó. Lớp này thường kế thừa từ lớp “Activity” trong Android SDK.
- **Chu kỳ sống:** Mỗi Activity trong ứng dụng Android có một chu kỳ sống riêng. Các trạng thái chính bao gồm: `onCreate()`, `onStart()`, `onResume()`, `onPause()`, `onStop()`, và `onDestroy()`. Các phương thức này cho phép thực hiện các tác vụ như khởi tạo, hiển thị, ẩn hoặc giải phóng tài nguyên của Activity.
- **Giao diện người dùng:** Mỗi Activity thường được kết hợp với một giao diện người dùng được định nghĩa bằng XML thông qua tập tin layout. Giao diện này có thể chứa các thành phần như nút nhấn, ô nhập liệu, danh sách và hình ảnh.
- **Tương tác với người dùng:** Activity cho phép người dùng tương tác trực tiếp thông qua các hành động như nhấn nút, di chuyển, chạm hoặc đơn giản là nhập liệu.
- **Quản lý thông qua Intent:** Activity thường được khởi động thông qua Intent, một cơ chế giao tiếp giữa các thành phần khác nhau của ứng dụng. Intent có thể chứa dữ liệu như thông tin cần thiết để khởi động một Activity mới hoặc chuyển dữ liệu giữa các Activity.

Nhìn chung, Activity là một phần không thể thiếu của một ứng dụng Android, điều khiển các màn hình và tương tác chính với người dùng. Khái niệm này là cực kỳ quan trọng để xây dựng ứng dụng Android hoàn chỉnh và dễ sử dụng.

2. Nội dung

Thực hiện ứng dụng Android cơ bản để tạo và dùng Activity theo từng bước của hướng dẫn.

II. Bài tập yêu cầu

Bài 1: Viết chương trình sử dụng ImageView sử dụng 2 cách là code java và xml



Hình 28 Giao diện bài tập 2.1

Đầu tiên chọn 1 file ảnh VD: android.png. Chép file này vào thư mục con có tên drawable của thư mục res

	Name	Date modified	Type	Size
> src				
> androidTest	ic_launcher_background	3/21/2024 10:07 AM	Microsoft Edge HTM...	6 KB
< main	ic_launcher_foreground	3/21/2024 10:07 AM	Microsoft Edge HTM...	2 KB
> java	android	3/24/2024 10:14 PM	PNG File	38 KB
< res				
drawable				
layout				
mipmap-anydpi				
mipmap-hdpi				
mipmap-mdpi				
mipmap-xhdpi				
mipmap-xxhdpi				
mipmap-xxxhdpi				

Hình 29 Thư mục hình ảnh trong project

Cách 1: Hiển thị ảnh bằng ImageView tạo bằng file java

```

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        ImageView myImg = new ImageView( context: this );
        myImg.setImageResource(R.drawable.android);

        setContentView(myImg);
    }
}

```

Chạy và xem kết quả

Cách 2: Xóa các code tạo ImageView trong file .java

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

Trong file main.xml, thêm code tạo ImageView như hình dưới

```
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res-auto"  
    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">  
  
    <ImageView  
        android:id="@+id/imageView2"  
        android:layout_width="376dp"  
        android:layout_height="702dp"  
        app:layout_constraintBottom_toBottomOf="parent"  
        app:layout_constraintEnd_toEndOf="parent"  
        app:layout_constraintHorizontal_bias="1.0"  
        app:layout_constraintStart_toStartOf="parent"  
        app:layout_constraintTop_toTopOf="parent"  
        app:srcCompat="@drawable/android" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>
```

Chạy chương trình cho kết quả là



Hình 30 Kết quả bài tập 2.1

Bài 2: Viết chương trình hiển thị 1 hình ảnh kèm tên của hình và edit text cho viết comment



Hình 31 Giao diện bài tập 2.2

Gợi ý: sử dụng các thành phần TextView, ImageView và EditText.

Bài 3: Xây dựng giao diện ứng dụng như hình dưới



Hình 32 Giao diện bài tập 2.3

-Tạo một project mới (xem lại bài cũ)

-Ứng dụng gồm 1 layout lớn, trong nó sẽ chứa 2 layout con

+Layout con thứ nhất sẽ chứa 1 ImageView hiển thị hình ảnh

+Layout con thứ hai sẽ chứa 4 button

-Phân cấp trong activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    >

    <LinearLayout...>
    <LinearLayout...>
</LinearLayout>
```

-Các thuộc tính cho LinearLayout

orientation: sắp xếp theo ngang hoặc dọc

layout_width: chiều rộng

layout_height: chiều cao

Trong đó, layout đầu, bổ sung các thuộc tính cho ImageView

id: đặt tên cho imageview

src: đường dẫn đến file ảnh

```
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:layout_weight="1"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/imageAndroid"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:contentDescription="android"
        android:src="@drawable/android">

    </ImageView>
</LinearLayout>
```

Với layout còn lại, bổ sung các thuộc tính cho Button

```
<LinearLayout  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent"  
    android:orientation="vertical"  
    android:layout_weight="1"  
>  
  
<Button...>  
<Button...>  
<Button...>  
<Button...>  
</LinearLayout>
```

Và tương tự cho 3 button còn lại



III. Câu hỏi ôn tập:

1. Định nghĩa Activity trong Android là gì?
 - A. Một thành phần UI trong ứng dụng Android
 - B. Một thành phần xử lý logic trong ứng dụng Android
 - C. Cả A và B đều đúng
2. Mục đích chính của một Activity là gì?
 - A. Cung cấp giao diện người dùng (UI) cho ứng dụng
 - B. Xử lý logic nghiệp vụ của ứng dụng
 - C. Cả A và B đều đúng
3. Một ứng dụng Android có thể có bao nhiêu Activity?
 - A. Chỉ 1 Activity

B. Nhiều Activity

4. Mỗi Activity trong ứng dụng Android được định nghĩa bằng cách nào?

- A. Sử dụng thẻ <activity> trong file AndroidManifest.xml
- B. Tạo một lớp Java kế thừa từ lớp Activity
- C. Cả A và B đều đúng

5. Câu hỏi: Mỗi Activity trong ứng dụng Android có một vòng đời (lifecycle) riêng. Các trạng thái chính của vòng đời Activity là gì?

- A. onCreate(), onStart(), onResume(), onPause(), onStop(), onDestroy()
- B. Chỉ có onCreate(), onStart(), onResume(), onPause(), onStop()
- C. onCreate(), onStart(), onResume(), onPause(), onStop(), onRestart(), onDestroy()

6. Phương thức nào được gọi khi một Activity bắt đầu?

- A. onCreate()
- B. onStart()
- C. onResume()

7. Phương thức nào được gọi khi một Activity trở thành không hiển thị?

- A. onPause()
- B. onStop()
- C. onDestroy()

8. Phương thức nào được gọi khi một Activity bị hủy?

- A. onPause()
- B. onStop()
- C. onDestroy()