

Lập trình thiết bị di động  
**ASSETS & SHARED  
PREFERENCES**

ThS. Nguyễn Quang Phúc  
[phucnq.edu@gmail.com](mailto:phucnq.edu@gmail.com)

# NỘI DUNG

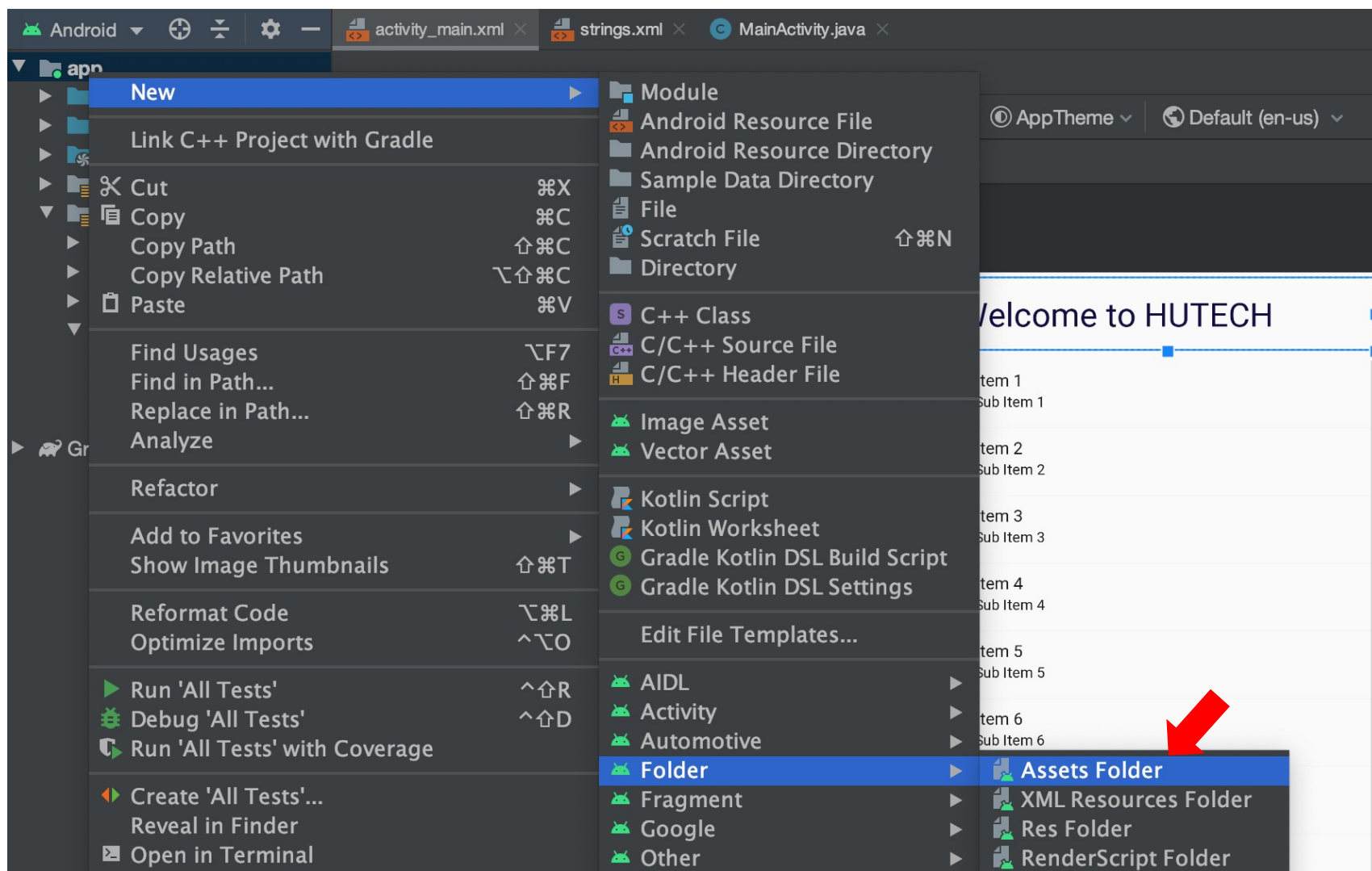
1. Giới thiệu về Assets trong Android
2. Cách thức truy xuất dữ liệu từ Assets
3. Giới thiệu về Shared Preferences
4. Sử dụng Shared Preferences

# 1. Giới thiệu về Assets trong Android

- **Assets** là thư mục chứa các tệp dữ liệu đầu vào cho ứng dụng: *âm thanh, hình ảnh, văn bản, tệp csdl, ....* Những tệp này sẽ không được biên dịch khi ứng dụng được đóng gói.
- Các tệp trong **assets/** không được cấp ID tài nguyên → chỉ có thể đọc chúng bằng cách sử dụng **AssetManager**.

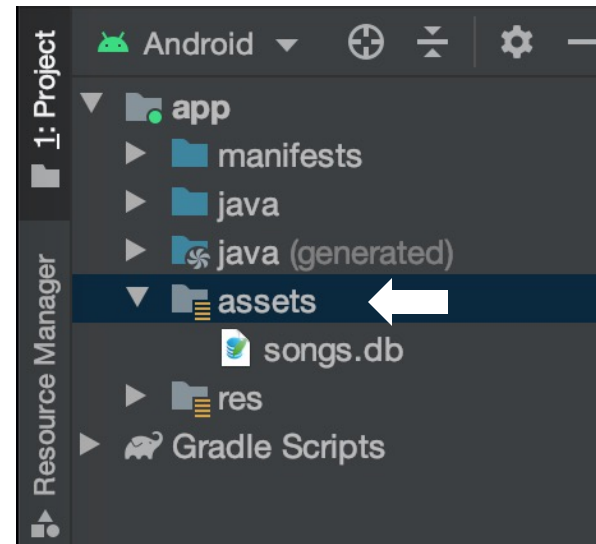
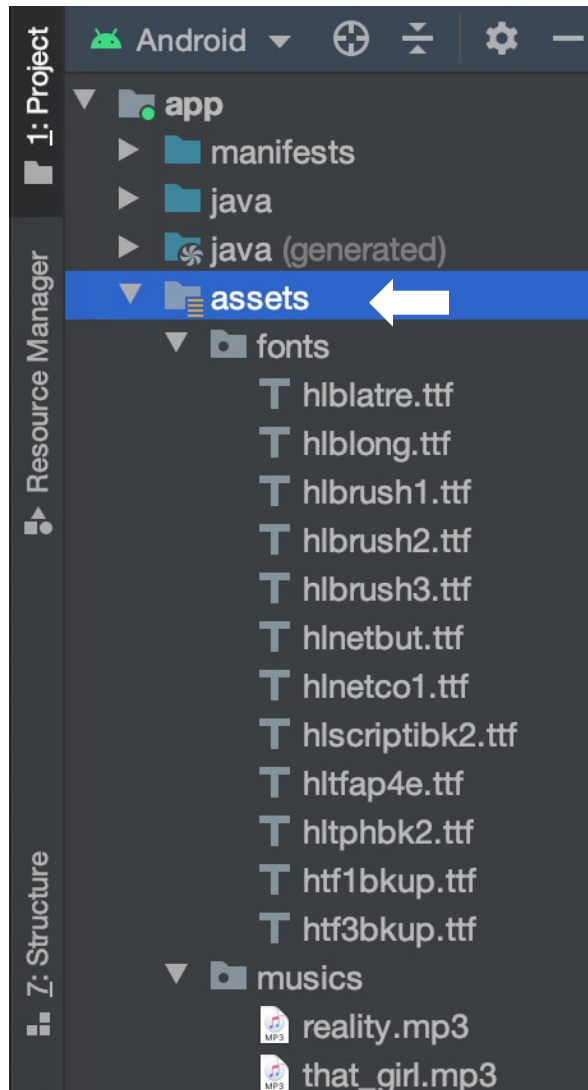
# 1. Giới thiệu về Assets trong Android

## »» Tạo thư mục Assets



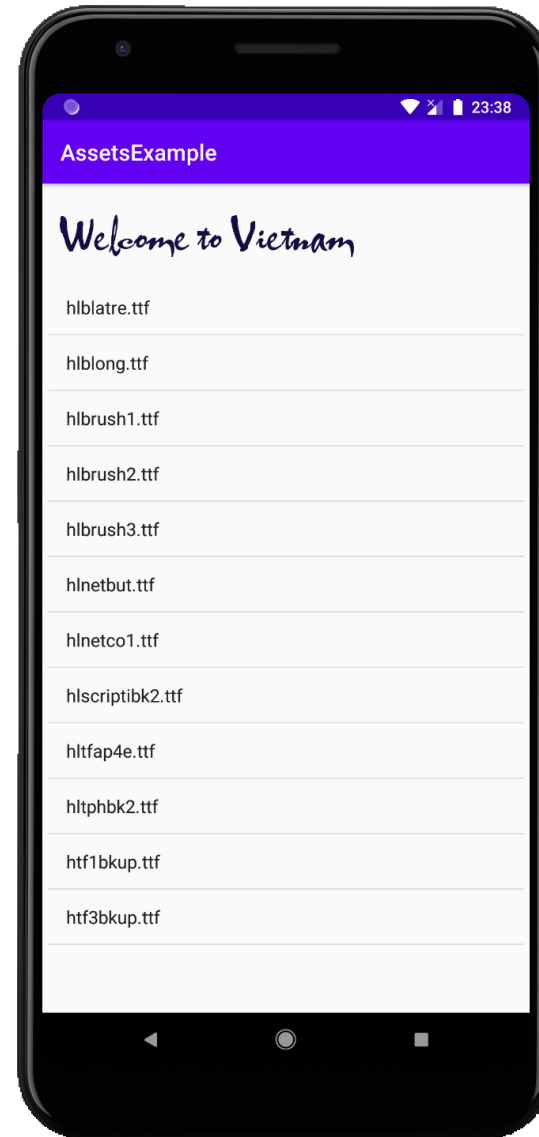
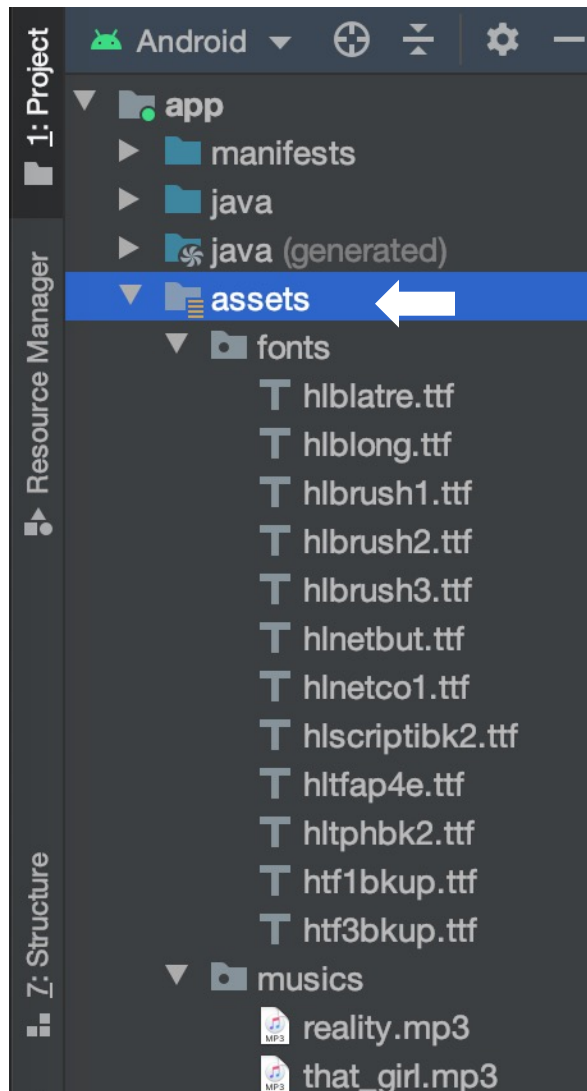
# 1. Giới thiệu về Assets trong Android

## »» Tạo thư mục Assets



## 2. Cách thức truy xuất dữ liệu từ Assets

»» Vd: Truy xuất Font chữ



## 2. Cách thức truy xuất dữ liệu từ Assets

### »» Vd: Truy xuất Font chữ

```
lvFonts = findViewById(R.id.lvFonts);  
fontAdapter = new ArrayAdapter<String>(MainActivity.this,  
    android.R.layout.simple_list_item_1);  
lvFonts.setAdapter(fontAdapter);  
try{  
    AssetManager assetManager = getAssets();  
    String[] arrFonts = assetManager.list("fonts");  
    fontAdapter.addAll(arrFonts);  
}catch (Exception e){  
    Log.e("Error: ", e.toString());  
}
```

```
Typeface typeface = Typeface.createFromAsset(getAssets(),  
    "fonts/" + fontAdapter.getItem(i));  
textContent.setTypeface(typeface);
```

## 2. Cách thức truy xuất dữ liệu từ Assets

»» Vd: Truy xuất tệp âm thanh

```
private void playAudio() {  
    try{  
        AssetFileDescriptor assetFileDescriptor =  
            getAssets().openFd("musics/ting.mp3");  
        MediaPlayer mediaPlayer = new MediaPlayer();  
        mediaPlayer.setDataSource(  
            assetFileDescriptor.getFileDescriptor(),  
            assetFileDescriptor.getStartOffset(),  
            assetFileDescriptor.getLength());  
        assetFileDescriptor.close();  
        mediaPlayer.prepare();  
        mediaPlayer.start();  
    }catch (Exception e){  
        Log.e("Error: ", e.toString());  
    }  
}
```



### 3. Giới thiệu về Shared Preferences

- **Shared Preferences** cho phép lưu trữ dữ liệu ứng dụng bằng file XML.
- Dữ liệu được lưu trữ và truy xuất theo cặp khóa **key-value** với các kiểu dữ liệu như: *Boolean, Float, Int, Long, String*.
- Dữ liệu vẫn được bảo toàn ngay cả khi ứng dụng đóng hoàn toàn.

## 4. Sử dụng Shared Preferences

### »» Lưu trữ dữ liệu

- **B1:** Khởi tạo đối tượng Shared Preferences

```
SharedPreferences preferences =  
    getSharedPreferences(PREFERENCES_NAME, MODE);
```

- **B2:** Tạo đối tượng Editor từ phương thức **edit()** của preferences

```
SharedPreferences.Editor editor = preferences.edit();
```

- **B3:** Thêm dữ liệu với phương thức **put<Type>** tùy theo kiểu dữ liệu


```
editor.putInt("int", 8);  
editor.putFloat("float", 6.8f);  
editor.putLong("long", 36);  
editor.putBoolean("boolean", true);  
editor.putString("string", "Welcome to Vietnam");
```

- **B4:** Lưu Shared Preferences với phương thức **apply()** hoặc **commit()**

```
editor.apply();
```

## 4. Sử dụng Shared Preferences

### »» Lưu trữ dữ liệu

▼	com.example.sharedpreferencesexample	drwx-----	2020-05-24 23:40	4 KB
▶	cache	drwxrws--x	2020-05-24 23:40	4 KB
▶	code_cache	drwxrws--x	2020-05-24 23:40	4 KB
▼	shared_prefs	drwxrwx--x	2020-05-25 20:46	4 KB
	 my_data.xml	-rw-rw----	2020-05-25 20:46	406 B

## 4. Sử dụng Shared Preferences

### » Truy xuất dữ liệu

- Dữ liệu được truy xuất với phương thức **get<Type>** tùy theo kiểu dữ liệu:

```
get<Type>(Key, DefValue);
```

**DefValue**: là giá trị mặc định được trả về khi **get<Type>** không trả về được *value* ứng với *key* truyền vào.

- Ví dụ:

```
SharedPreferences preferences =  
    getSharedPreferences(PREFERENCES_NAME, MODE);  
int i = preferences.getInt("int", 0);  
float f = preferences.getFloat("float", 0.0f);  
long l = preferences.getLong("long", 0);  
boolean b = preferences.getBoolean("boolean", false);  
String s = preferences.getString("string", "");
```

## 4. Sử dụng Shared Preferences

**Nên lưu và phục hồi dữ liệu ở phương thức nào theo vòng đời ứng dụng?**

- Lưu: onPause;
- Phục hồi: onResume

**Q & A**