

PHÁT TRIỂN ỨNG DỤNG TRÊN THIẾT BỊ DI ĐỘNG

Hồ Văn Tú

Bộ môn Tin học ứng dụng

Khoa CNTT và truyền thông

hvtu@ctu.edu.vn

Chương 2

Ứng dụng và Activity

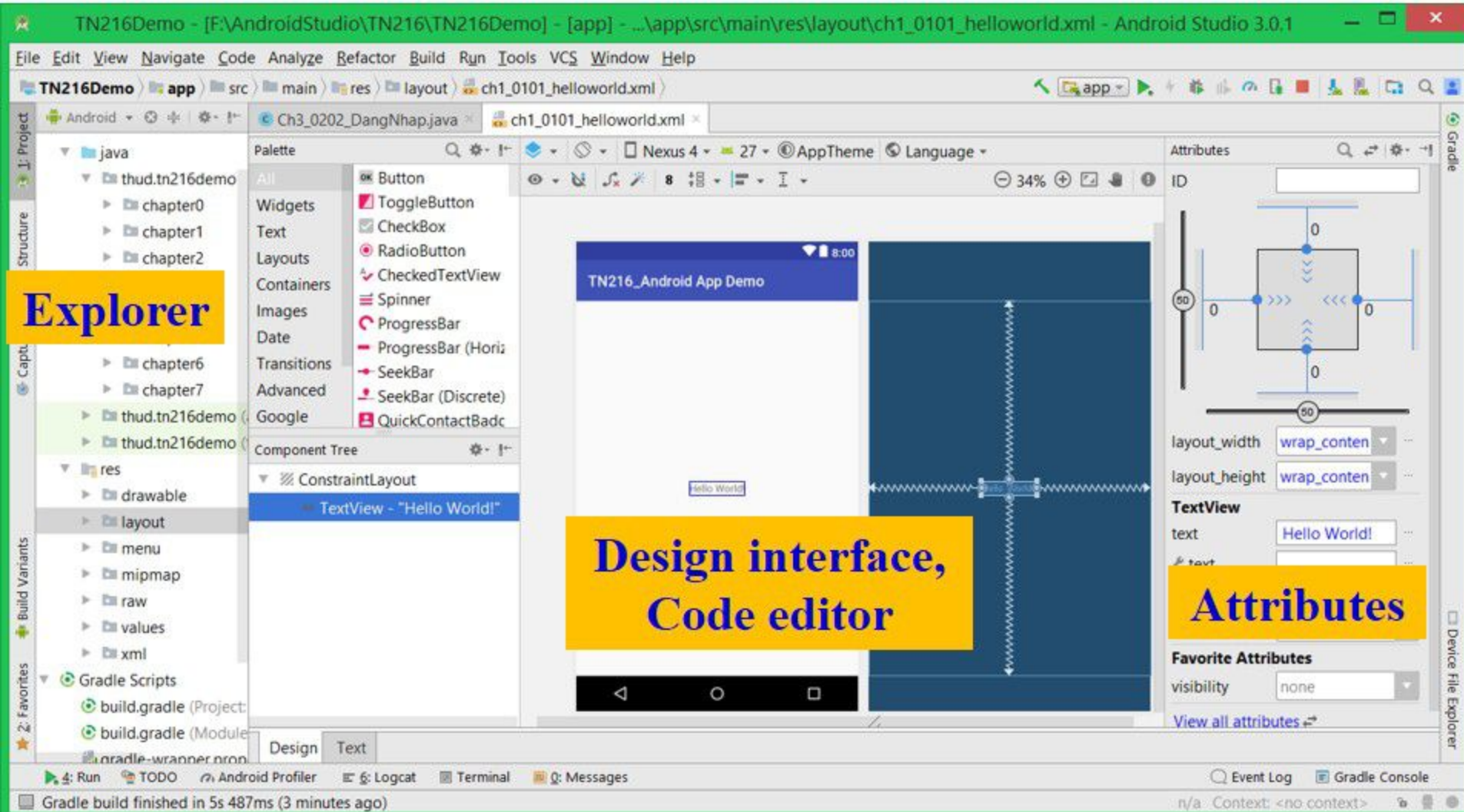
NỘI DUNG (1)

- Ứng dụng Android
 - Cấu trúc của ứng dụng
 - Các thành phần trong ứng dụng
 - Kích hoạt các thành phần trong ứng dụng Android
- Android Manifest
 - Giới thiệu
 - Cấu trúc của Android Manifest

NỘI DUNG (2)

- Quản lý và sử dụng tập tin tài nguyên
 - Giới thiệu
 - Các tài nguyên thông dụng
- Activity
 - Giới thiệu
 - Vòng đời của Activity
 - Tạo và thực thi Activity

Ứng dụng Android_Cửa sổ làm việc



Ứng dụng Android_Cấu trúc (1)

The screenshot displays the Android Studio interface with the 'Project' tool window on the left and the 'Build Variants' tool window on the right.

Project Structure (Left Panel):

- File Edit View Navigate Code Analyze Refactor
- TN216Demo > app > src > main > java
- Android
- app
 - manifests
 - java
 - thud.tn216demo
 - chapter0
 - chapter1
 - Ch1_0101_HelloWorld
 - Ch1_0205_WelcomeAndroid
 - chapter2
 - chapter3
 - chapter4
 - chapter5
 - chapter6
 - chapter7
 - thud.tn216demo (androidTest)
 - thud.tn216demo (test)
 - res
 - drawable
 - layout

Build Variants (Right Panel):

- res
 - drawable
 - layout
 - menu
 - mipmap
 - raw
 - values
 - colors.xml
 - dimens.xml (2)
 - drawables.xml
 - strings.xml
 - styles.xml (2)
 - xml
- Gradle Scripts
 - build.gradle (Project: TN216Demo)
 - build.gradle (Module: app)
 - gradle-wrapper.properties (Gradle Version)
 - proguard-rules.pro (ProGuard Rules for app)
 - gradle.properties (Project Properties)
 - settings.gradle (Project Settings)
 - local.properties (SDK Location)

Bottom Bar:

- 4: Run
- TODO
- Android Profiler
- Logcat

Ứng dụng Android_Cấu trúc (2)

- ▼ res
 - ▼ drawable
 - ic_launcher_background.xml
 - ic_launcher_foreground.xml (v24)
 - ic_menu_camera.xml (v21)
 - ic_menu_gallery.xml (v21)
 - ic_menu_manage.xml (v21)
 - ic_menu_send.xml (v21)
 - ic_menu_share.xml (v21)
 - ic_menu_slideshow.xml (v21)
 - side_nav_bar.xml
 - ▼ layout
 - app_bar_ch3_0130_du_lich_da_lat.xml
 - app_bar_ch3_0211_du_lich_da_lat_drawer.xml
 - ch0_00_change_orientation.xml
 - ch0_00_mainactivity.xml
 - ch0_00_mainactivity_land.xml
 - ch0_01_list_chapter.xml
 - ch1_0101_helloworld.xml

- ▼ res
 - ▶ drawable
 - ▶ layout
 - ▶ menu
 - ▶ mipmap
 - ▼ raw
 - huongluamiennam.mp3
 - nucuoixuan.txt
 - ▼ values
 - colors.xml
 - ▶ dims.xml (2)
 - drawables.xml
 - strings.xml**
 - ▶ styles.xml (2)
 - ▼ xml
 - settings_layout.xml
- ▼ Gradle Scripts
 - build.gradle (Project: TN216Demo)
 - build.gradle (Module: app)

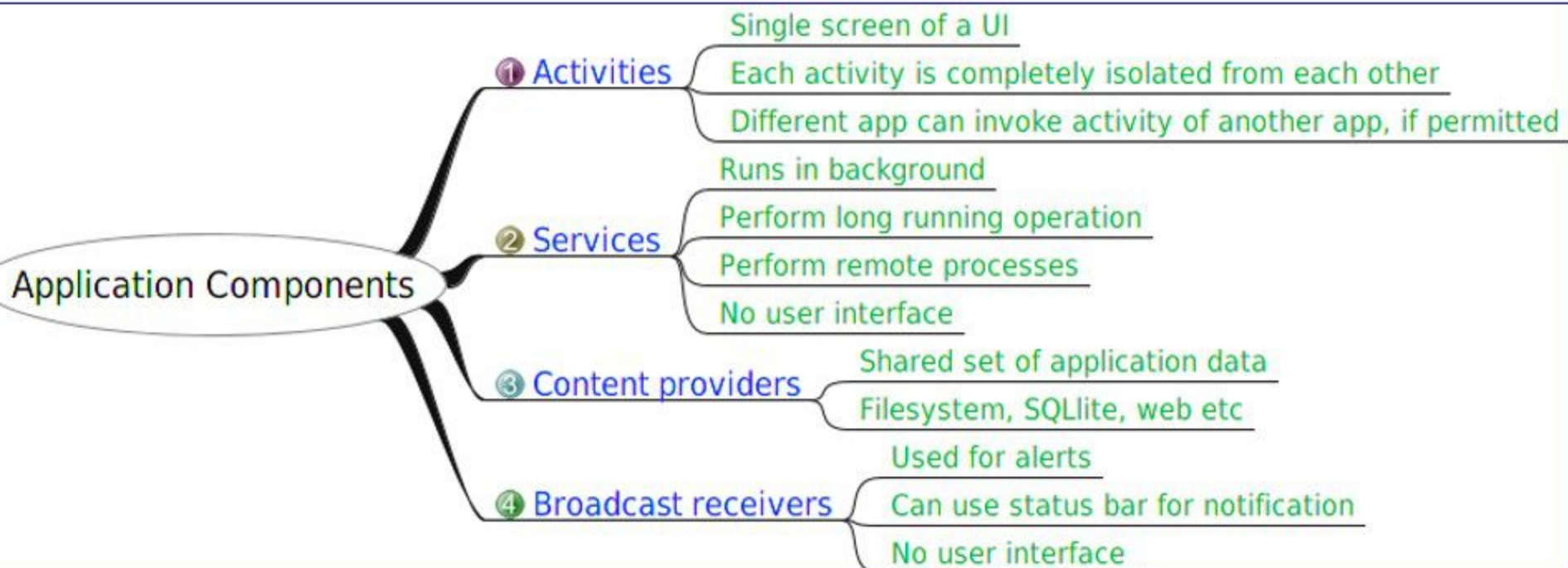
Ứng dụng Android_Cấu trúc (3)

- app: mã nguồn, tài nguyên và cài đặt cấu hình
 - manifests: thông tin thiết yếu về ứng dụng
 - java (source code): các files .java trong các gói (package)
 - res (resources): lưu trữ tài nguyên của ứng dụng
 - anim: files cho đối tượng động (.xml)
 - drawable/ mipmap: files hình ảnh
 - layout: files giao diện ứng dụng (.xml)
 - menu: files thực đơn (.xml)
 - raw: các files dữ liệu thô
 - values: files lưu giá trị các biến (.xml)

Ứng dụng Android_Cấu trúc (4)

- app
 - res (resources)
 - values: files lưu giá trị các biến (.xml)
 - colors: giá trị biến màu sắc
 - dims: giá trị biến kích thước (Dimension)
 - strings: giá trị biến chuỗi
 - styles: giá trị biến định dạng style
- Gradle Scripts: lưu trữ các tập tin hỗ trợ biên dịch và đóng gói ứng dụng

Ứng dụng Android_Các thành phần (1)



Ứng dụng Android_Các thành phần (2)

- Activity
 - Là nền của 1 ứng dụng, cung cấp giao diện cho phép người dùng tương tác để thực hiện hành động
 - Mỗi Activity là độc lập nhau
 - Có thể được gọi từ ứng dụng khác
- Service
 - Thành phần chạy ngầm thực hiện các công việc không cần giao diện
 - Có thể được gọi từ thành phần khác

Ứng dụng Android_Các thành phần (3)

- Content Provider
 - Quản lý và chia sẻ dữ liệu trong các ứng dụng
 - Sử dụng tập tin hệ thống, SQLite, Web, thiết bị lưu trữ khác
- Broadcast Receiver
 - Thu nhận các thông báo từ hệ điều hành hay ứng dụng khác (pin yếu, có sms, có cuộc gọi đến, ...)
 - Phát thông báo cho ứng dụng khác

Ứng dụng Android_Các thành phần (4)

- Intent
 - Cung cấp thông tin, dữ liệu giúp xác định công việc cần thực hiện
 - Là cơ cấu cho phép truyền thông điệp giữa các thành phần của ứng dụng và giữa các ứng dụng
 - Được sử dụng để gửi yêu cầu khởi tạo Activity, Service, hay Broadcast Receiver
- Notification
 - Đưa ra các thông báo mà không làm cho các Activity phải ngừng hoạt động

Ứng dụng Android_Kích hoạt các thành phần

- Activity, Service, Broadcast Receiver

Được kích hoạt bằng Intent

- Content Provider

Được kích hoạt bằng ContentResolver (bộ giải quyết nội dung)

Android Manifest (1)

- Điều khiển ứng dụng: mô tả các thành phần và điều kiện/quyền để thực thi, chia sẻ ứng dụng
 - Khai báo tên Package, mức Android API sử dụng
 - Mô tả các thành phần của ứng dụng: Activity, Service, Broadcast Receiver, Content Provider
 - Khai báo quyền cần sử dụng để truy cập các protected API và tương tác với các ứng dụng khác
 - Thông báo những quyền mà các ứng dụng khác cần có để tương tác với ứng dụng hiện thời
 - Thư viện sử dụng

Android Manifest (2)

- Thẻ manifest: mức cao nhất, gồm các thẻ con như
 - Thẻ application: mô tả các thành phần của ứng dụng

```
<application
    android:allowBackup="true"
    android:icon="@drawable/water01"
    android:label="@string/app_name"
    android:theme="@android:style/Theme.Holo.Light" >
    <activity
        android:name=".DangNhap"
        android:label="@string/app_name" >
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
        android:name=".MainActivity"
        android:label="@string/app_name" >
    </activity>
```


Android Manifest (3)

- Thẻ uses-sdk: mức Android API sử dụng
- Thẻ uses-feature: chức năng sử dụng
- Thẻ uses-permission: khai báo quyền truy cập

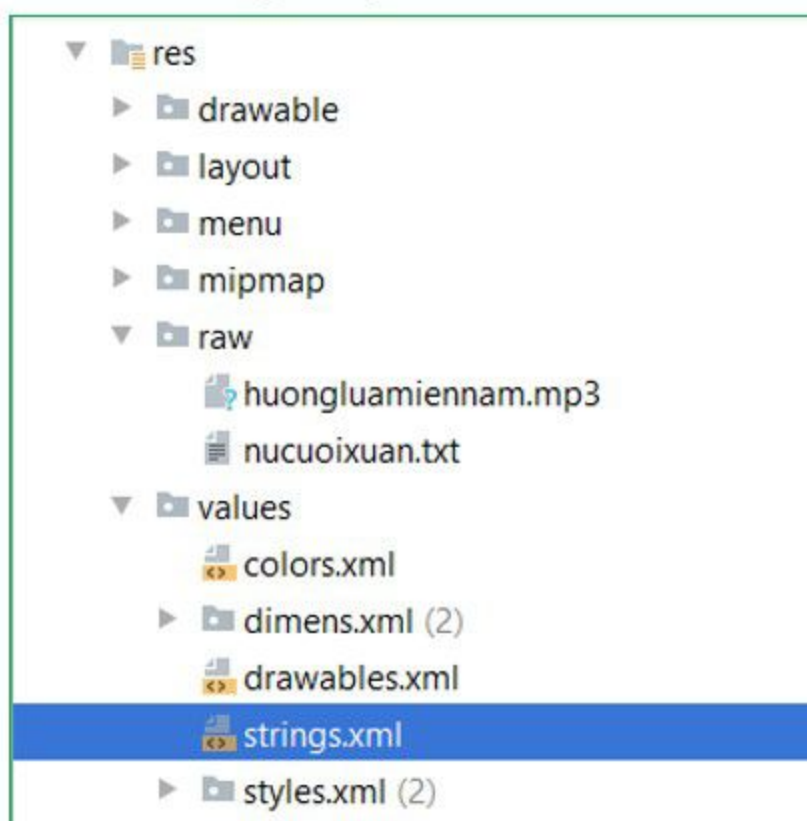
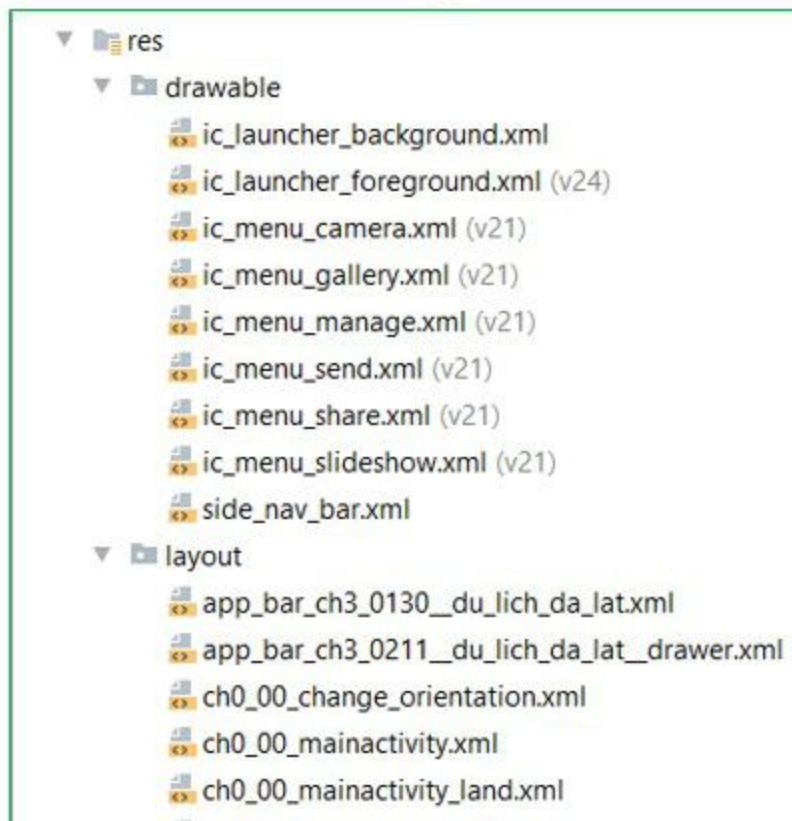
```
<uses-sdk  
    android:minSdkVersion="14"  
    android:targetSdkVersion="19" />
```

```
<uses-feature  
    android:glEsVersion="0x00020000"  
    android:required="true" />
```

```
<uses-permission android:name="android.permission.INTERNET"/>  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
```

Quản lý và sử dụng tập tin tài nguyên (1)

- Tập tin tài nguyên (res: Resources): sử dụng trong ứng dụng/ mã nguồn như hình ảnh, giao diện, thực đơn, tập tin thô, khai báo giá trị các biến/ định dạng styles



Quản lý và sử dụng tập tin tài nguyên (2)

- `res/layout/activity_main.xml`: tập tin giao diện

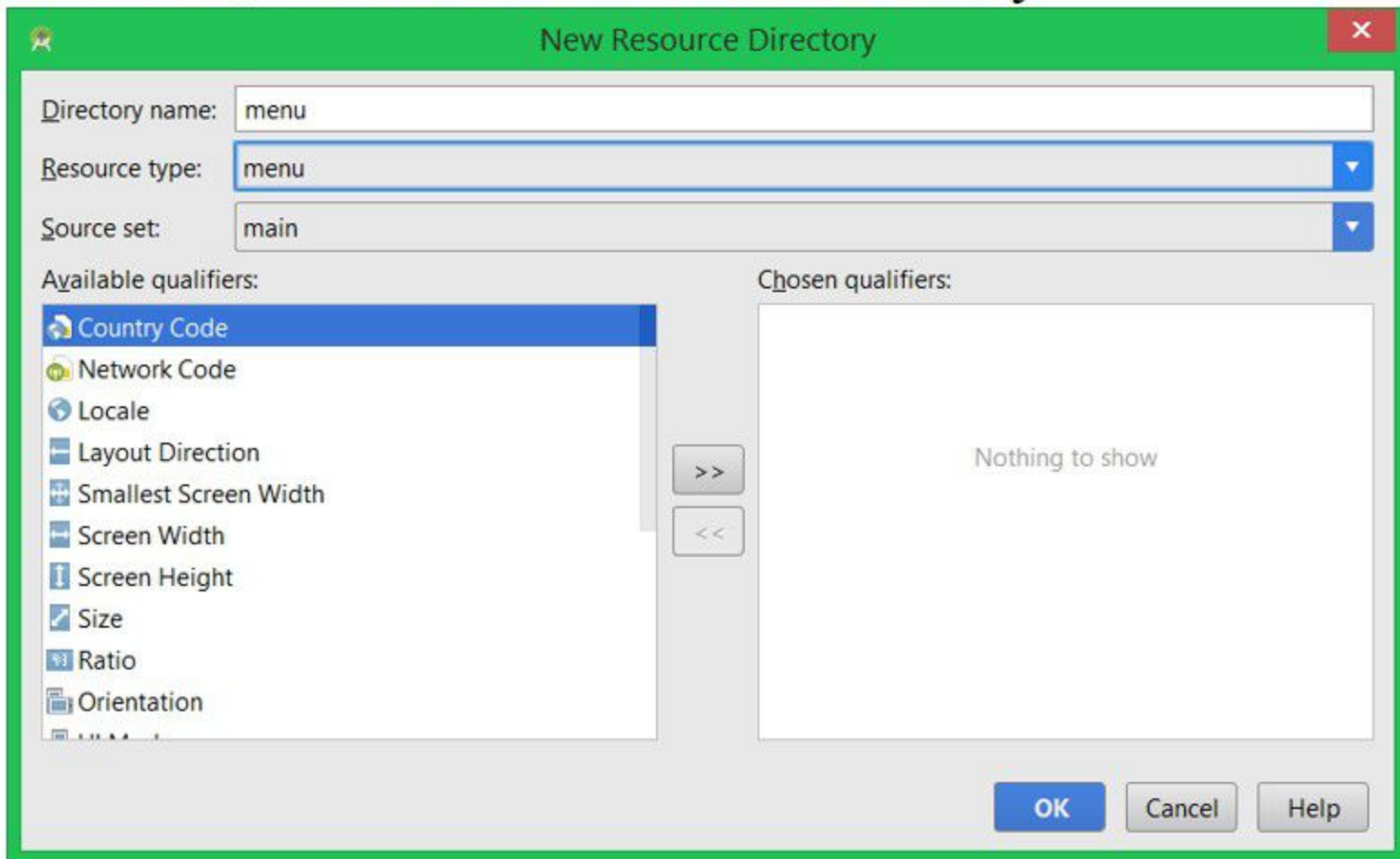
```
public class MainActivity extends AppCompatActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
}
```

- `res/mipmap/globeicon.png`: tập tin hình

```
myActionBar.setIcon(R.mipmap.globeicon);
```


Quản lý và sử dụng tập tin tài nguyên (3)

- Tạo thư mục tài nguyên: R_Click lên thư mục res → chọn New → chọn Android resource directory



Quản lý và sử dụng tập tin tài nguyên (4)

- Tạo tập tin tài nguyên: R_Click lên thư mục res → chọn New → chọn Android resouce file

New Resource File

File name: settings ↑↓

Resource type: XML ▼

Root element: PreferenceScreen

Source set: main ▼

Directory name: xml

Available qualifiers:

- Country Code
- Network Code
- Locale
- Layout Direction

Chosen qualifiers:

Nothing to show

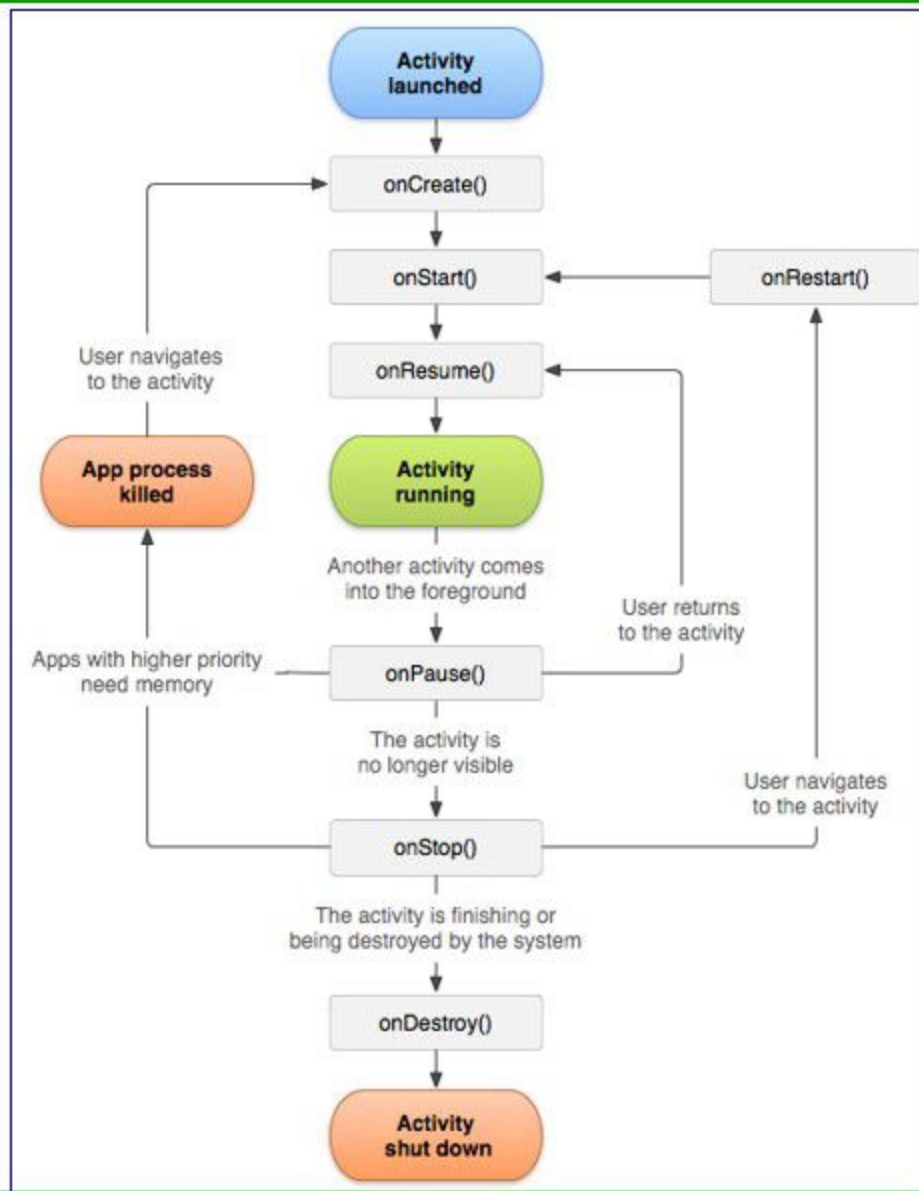
>> <<

OK Cancel Help

Activity_Giới thiệu

- Thành phần ứng dụng có giao diện tương tác
- Mỗi ứng dụng có một hoặc nhiều Activity
- Đối tượng thuộc lớp `android.app.Activity` (hoặc sử dụng thư viện `android.support.v7.app.AppCompatActivity`)
- Mỗi một Activity muốn được gọi trong ứng dụng thì bắt buộc phải được khai báo trong Manifest
- Activity thực thi khi mở ứng dụng: “main” Activity
- Mỗi một Activity có một vòng đời độc lập hoàn toàn với các Activity khác

Activity_Vòng đời (1)



Activity_Vòng đời (2)

- Các sự kiện thực thi trong vòng đời của Activity
 - onCreate(): được gọi khi Activity được khởi tạo
 - onStart(): được gọi khi Activity được hiển thị
 - onResume(): được gọi khi Activity có thể tương tác
 - onPause(): được gọi khi Activity tạm ngưng do Activity khác onResume()
 - onStop(): được gọi khi Activity không còn hiển thị
 - onRestart(): được gọi khi Activity khởi động lại từ onStop()
 - onDestroy(): được gọi khi Activity bị hủy theo lệnh hoặc do hệ thống thiếu bộ nhớ

Activity_Vòng đời (3)

- Android quản lý Activity theo dạng Stack
 - Khi Activity mới khởi tạo được xếp lên đỉnh Stack và trở thành running Activity
 - Các Activity trước đó sẽ bị tạm dừng và chỉ hoạt động trở lại khi Activity mới hơn được giải phóng
 - Entire lifetime: từ onCreate() tới onDestroy()
 - Visible lifetime: từ sau onStart() tới trước onStop()
 - Foreground lifetime: từ sau onResume() tới trước onPause()

Activity_Vòng đời (4)

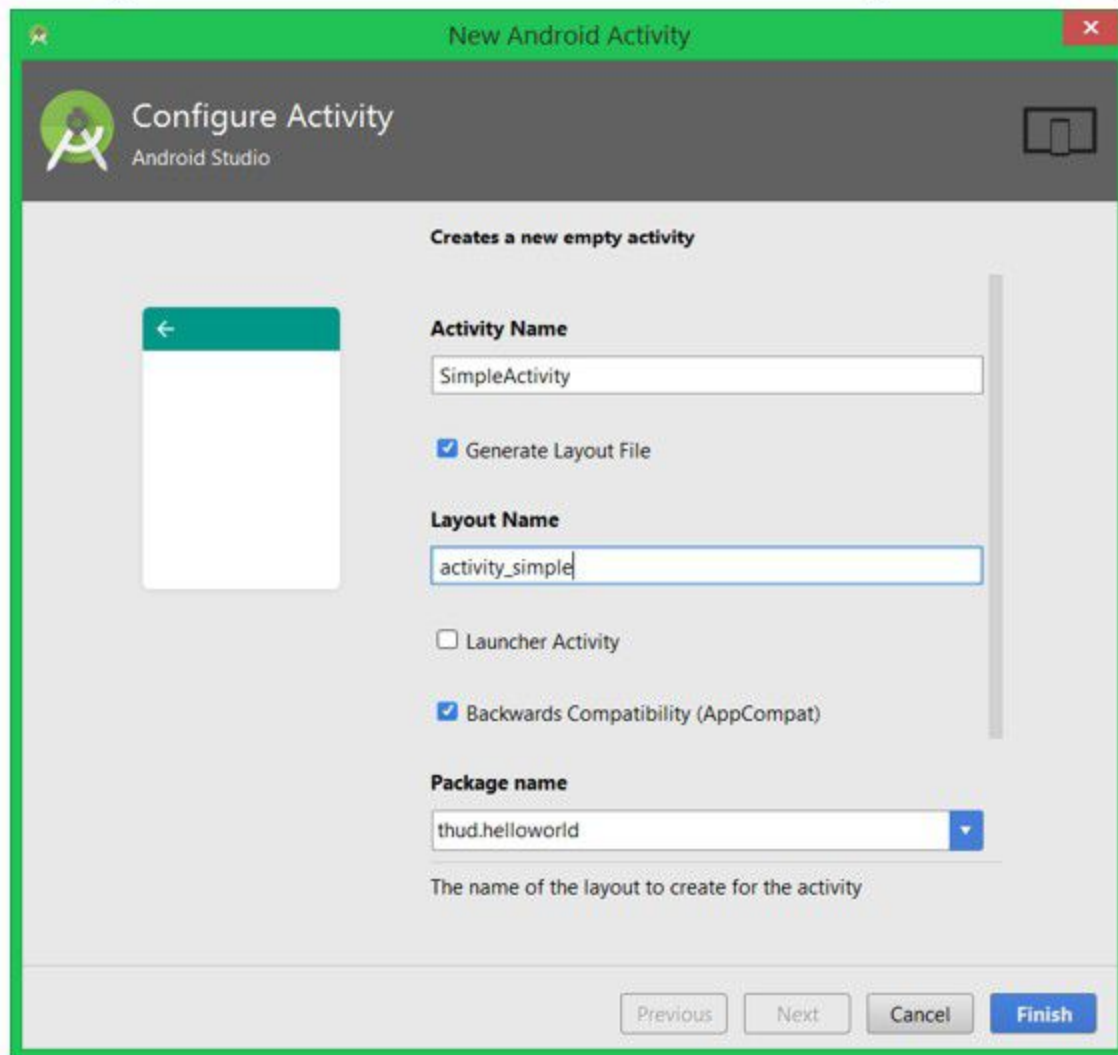
- Các trạng thái của Activity
 - Active (running): Activity đang hiển thị trên màn hình (foreground)
 - Paused: Activity vẫn hiển thị (visible) nhưng không thể tương tác (lost focus)
 - Stop: Activity bị thay thế hoàn toàn bởi Activity mới sẽ tiến đến trạng thái stop
 - Killed: khi hệ thống bị thiếu bộ nhớ, nó sẽ giải phóng các tiến trình theo nguyên tắc ưu tiên

Activity_Tạo và thực thi (1)

- Nội dung công việc
 - Mở ứng dụng HelloWorld, tạo Activity mới SimpleActivity
 - Trong activity_main.xml, tạo nút lệnh “*Thực thi*”
 - Trong MainActivity.java, viết mã lệnh cho phép thực thi SimpleActivity khi Click chọn nút lệnh “*Thực thi*”
 - Trong activity_simple.xml, tạo nút lệnh “*Đóng Activity*”
 - Trong SimpleActivity.java, viết mã lệnh cho phép kết thúc SimpleActivity khi Click chọn nút lệnh “*Đóng Activity*”
 - Trong strings.xml, khai báo và gán giá trị cho các biến
 - Khai báo SimpleActivity trong AndroidManifest.xml
 - Thực thi, kiểm tra kết quả

Activity_Tạo và thực thi (2)

- Tạo Activity: File → New → Activity → Empty Activity



Activity_Tạo và thực thi (3)

- Tạo nút lệnh trong activity_main.xml

```
10      <TextView
11          android:id="@+id/txt_tieude"
12          android:layout_width="wrap_content"
13          android:layout_height="wrap_content"
14          android:text="@string/hello_world" />
15
16      <Button
17          android:id="@+id/btn_thucthi"
18          android:layout_width="wrap_content"
19          android:layout_height="wrap_content"
20          android:layout_alignLeft="@+id/txt_tieude"
21          android:layout_below="@+id/txt_tieude"
22          android:text="@string/thucthi" />
```

Activity_Tạo và thực thi (4)

- Viết mã lệnh trong MainActivity.java

```
public class MainActivity extends AppCompatActivity {  
    private static String TAG = "Vòng đời Activity";  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        Log.i(TAG, "onCreate thực thi");  
        Toast.makeText(this, "onCreate thực thi",  
                        Toast.LENGTH_SHORT).show();  
    }  
}
```


Activity_Tạo và thực thi (5)

- Viết mã lệnh trong MainActivity.java

```
@Override
```

```
protected void onResume() {
```

```
    super.onResume();
```

```
    Log.i(TAG, "onResume thực thi");
```

```
    Toast.makeText(this, "onResume thực thi",  
                    Toast.LENGTH_SHORT).show();
```

```
}
```

```
public void ThucThiActivity(View view) {
```

```
    Intent intent = new Intent(this, SimpleActivity.class);
```

```
    startActivity(intent);
```

```
}
```

```
}
```

Activity_Tạo và thực thi (6)

- Tạo nút lệnh trong activity_simple.xml

```
11      <TextView
12          android:id="@+id/txt_tieude"
13          android:layout_width="wrap_content"
14          android:layout_height="wrap_content"
15          android:text="@string/new_activity" />
16
17      <Button
18          android:id="@+id/btn_dong"
19          android:layout_width="wrap_content"
20          android:layout_height="wrap_content"
21          android:layout_alignLeft="@+id/txt_tieude"
22          android:layout_below="@+id/txt_tieude"
23          android:onClick="DongActivity"
24          android:text="@string/dong" />
```

Activity_Tạo và thực thi (7)

- Viết mã lệnh trong SimpleActivity.java

```
7 public class SimpleActivity extends Activity {  
8  
9     @Override  
10    protected void onCreate(Bundle savedInstanceState) {  
11        super.onCreate(savedInstanceState);  
12        setContentView(R.layout.activity_simple);  
13    }  
14  
15    public void DongActivity(View view){  
16        finish();  
17    }  
18  
19 }
```


Activity_Tạo và thực thi (8)

- Khai báo và gán giá trị cho các biến trong strings.xml

```
3 <resources>
4
5     <string name="app_name">HelloWorld</string>
6     <string name="hello_world">Hello world!</string>
7     <string name="action_settings">Settings</string>
8     <string name="simple_activity">New Simple Activity</string>
9     <string name="thucthi">Thực thi</string>
10    <string name="dong">Đóng Activity</string>
11
12 </resources>
```

Activity_Tạo và thực thi (8)

- Khai báo SimpleActivity trong AndroidManifest.xml

```
11< application
12    android:allowBackup="true"
13    android:icon="@drawable/ic_launcher"
14    android:label="@string/app_name"
15    android:theme="@android:style/Theme.Holo.Light" >
16    <activity
17        android:name=".MainActivity"
18        android:label="@string/app_name" >
19        <intent-filter>
20            <action android:name="android.intent.action.MAIN" />
21            <category android:name="android.intent.category.LAUNCHER" />
22        </intent-filter>
23    </activity>
24    <activity
25        android:name=".SimpleActivity"
26        android:label="@string/simple_activity" >
27    </activity>
28</application>
```

Activity_Tạo và thực thi (9)

- Thực thi

