



FPT POLYTECHNIC



www.poly.edu.vn

LẬP TRÌNH ANDROID 1

TOOLBAR VÀ MENU

- ❑ Cách sử dụng và thiết lập một số thuộc tính trong Toolbar
- ❑ Cách tạo ra Menu trong ứng dụng Android
 - ❖ Options Menu
 - ❖ Context Menu
 - ❖ Popup Menu



FPT POLYTECHNIC



www.poly.edu.vn

LẬP TRÌNH ANDROID 1

BÀI 6.1: TOOLBAR

MỤC TIÊU

- CÁCH SỬ DỤNG VÀ THIẾT LẬP MỘT SỐ THUỘC TÍNH TRONG TOOLBAR
- CÁCH TẠO RA MENU TRONG ỨNG DỤNG ANDROID
- OPTIONS MENU
- CONTEXT MENU
- POPUP MENU





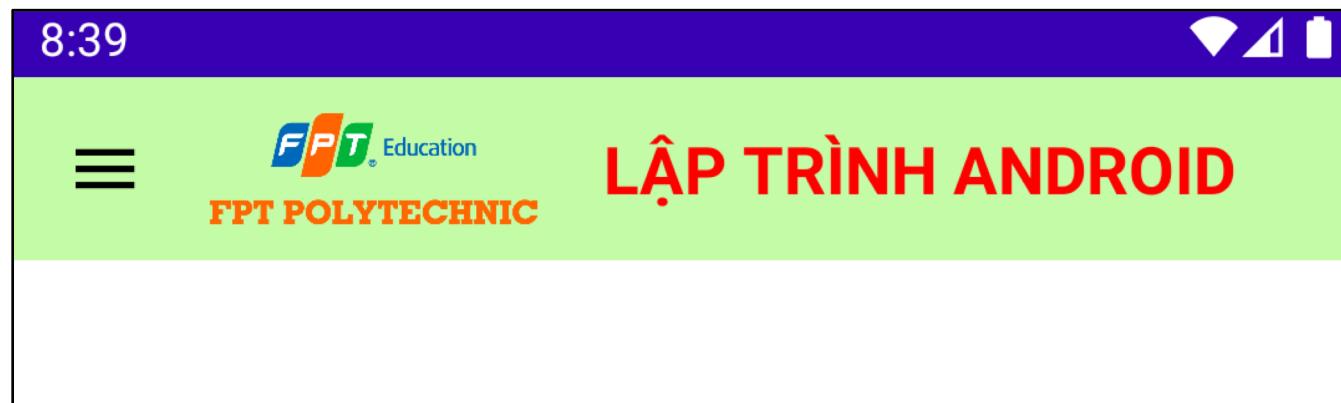
ACTIONBAR & TOOLBAR

❑ **ActionBar** được hiểu là App Bar theo thiết kế Material Design, nó chứa các thành phần với mục đích điều hướng ứng dụng như:

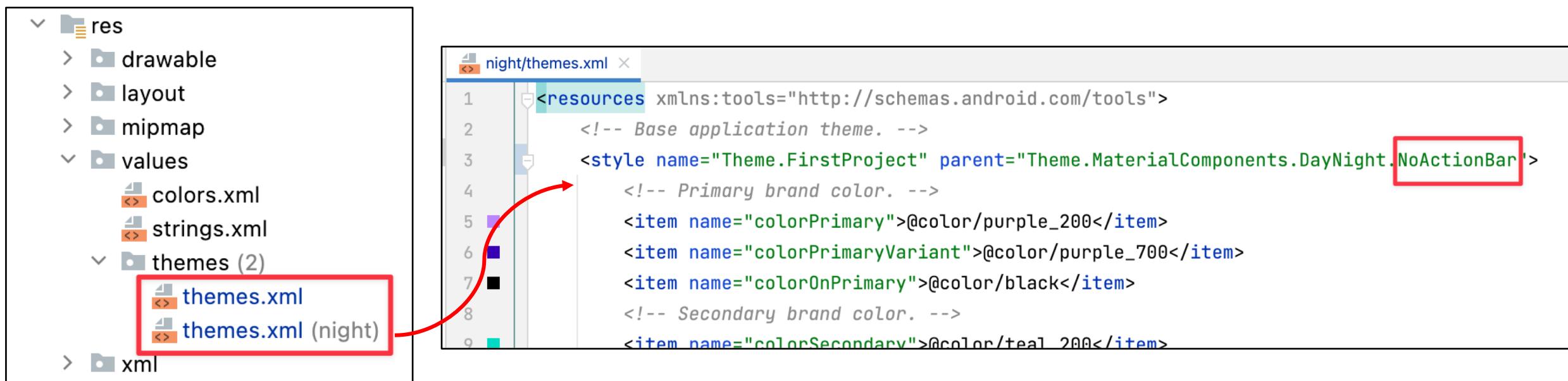
- ❖ Icon ứng dụng
- ❖ Nút mũi tên để quay lại cửa sổ cha
- ❖ Tiêu đề
- ❖ Những nút bấm chính cho Activity
- ❖ Các thành phần điều hướng (ví dụ Navigation Drawer)



- ❑ **Toolbar** được giới thiệu từ API 21, nó như là phần mở rộng của ActionBar, giúp cho việc tùy biến ActionBar dễ dàng hơn
- ❑ Khác biệt giữa **Toolbar** và **ActionBar** đó là với **Toolbar** nó là một View (*ViewGroup*), nên nó có thể khai báo trong Layout, nó có thể chứa các View khác để bạn thoải mái tùy biến, nó có thể nằm ở vị trí tùy biến khác nhau trong Layout, có thể thực hiện các Animate dễ dàng, có thể chèn nhiều Toolbar trong một Activity



- ❑ Để sử dụng Toolbar, thay đổi theme trong file *themes.xml* từ **DarkActionBar** thành **NoActionBar** (thao tác này sẽ làm cho Activity mất ActionBar mặc định)



- ❑ Khai báo Toolbar trong file Layout

```
<androidx.appcompat.widget.Toolbar  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:background="#C4FBA6"/>
```

- ❑ Toolbar như một ViewGroup nên có thể bỏ bất kỳ View nào vào trong nó (TextView, ImageView, Button, Spinner,...)
- ❑ Trong hàm **onCreate**, ánh xạ và sử dụng hàm **setSupportActionBar** để gán ToolBar thay thế cho ActionBar

```
Toolbar toolbar = findViewById(R.id.toolbar);  
setSupportActionBar(toolbar);
```

❑ Một số cài đặt cho Toolbar

- ❖ Tiêu đề (Title)

```
getSupportActionBar().setTitle("Android 01");
```

- ❖ Tiêu đề phụ (subTitle)

```
getSupportActionBar().setSubtitle("FPT Polytechnic");
```

- ❖ Logo

```
getSupportActionBar().setLogo(R.drawable.ic_favorite_24);
```

❑ Một số cài đặt cho Toolbar

❖ Nút Back trở về Activity trước

```
getSupportActionBar().setDisplayHomeAsUpEnabled(true);
```

Bắt sự kiện khi nhấn vào nút mũi tên quay lại

```
@Override  
public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
    if (item.getItemId() == android.R.id.home) {  
        onBackPressed();  
    }  
    return super.onOptionsItemSelected(item);  
}
```

❖ Thay đổi icon Back thành biểu tượng khác

```
getSupportActionBar().setHomeAsUpIndicator(R.drawable.shopping_cart_24);
```



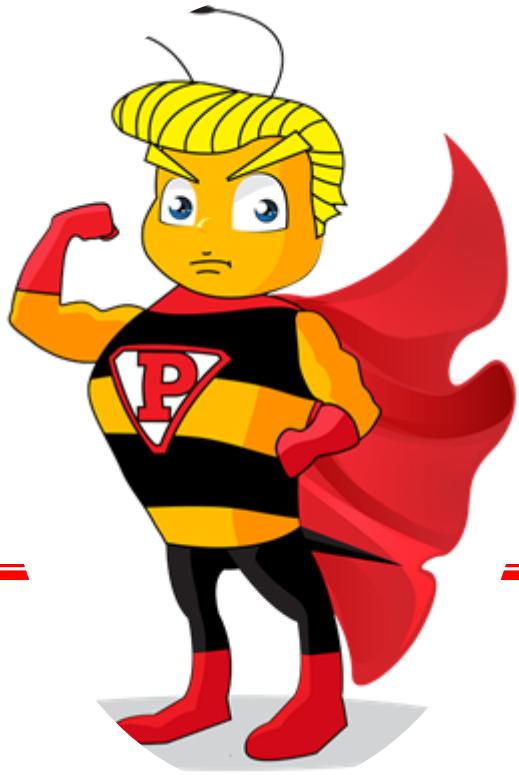
FPT POLYTECHNIC



www.poly.edu.vn

LẬP TRÌNH ANDROID 1

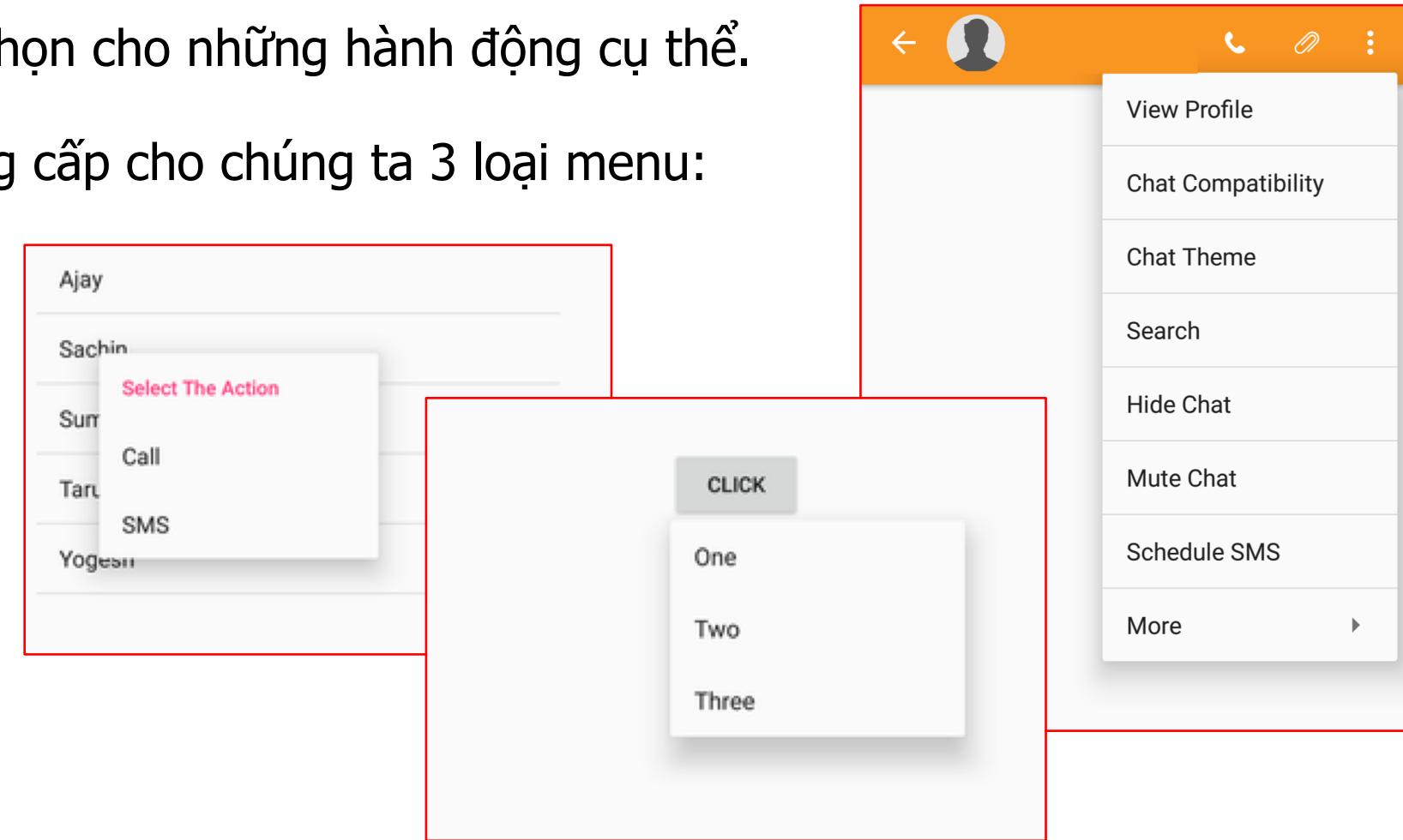
BÀI 6.2: MENU



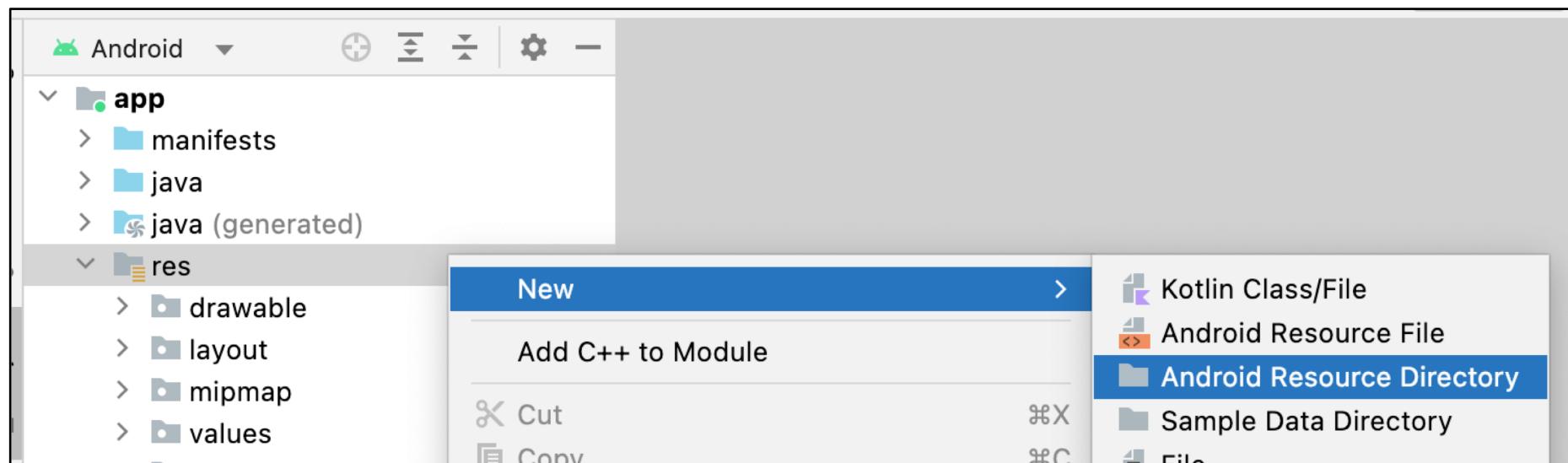
MENU

- ❑ **Menu** là một trong những thành phần giao diện quan trọng, cung cấp cho người dùng một vài lựa chọn cho những hành động cụ thể.
- ❑ Trong Android cung cấp cho chúng ta 3 loại menu:

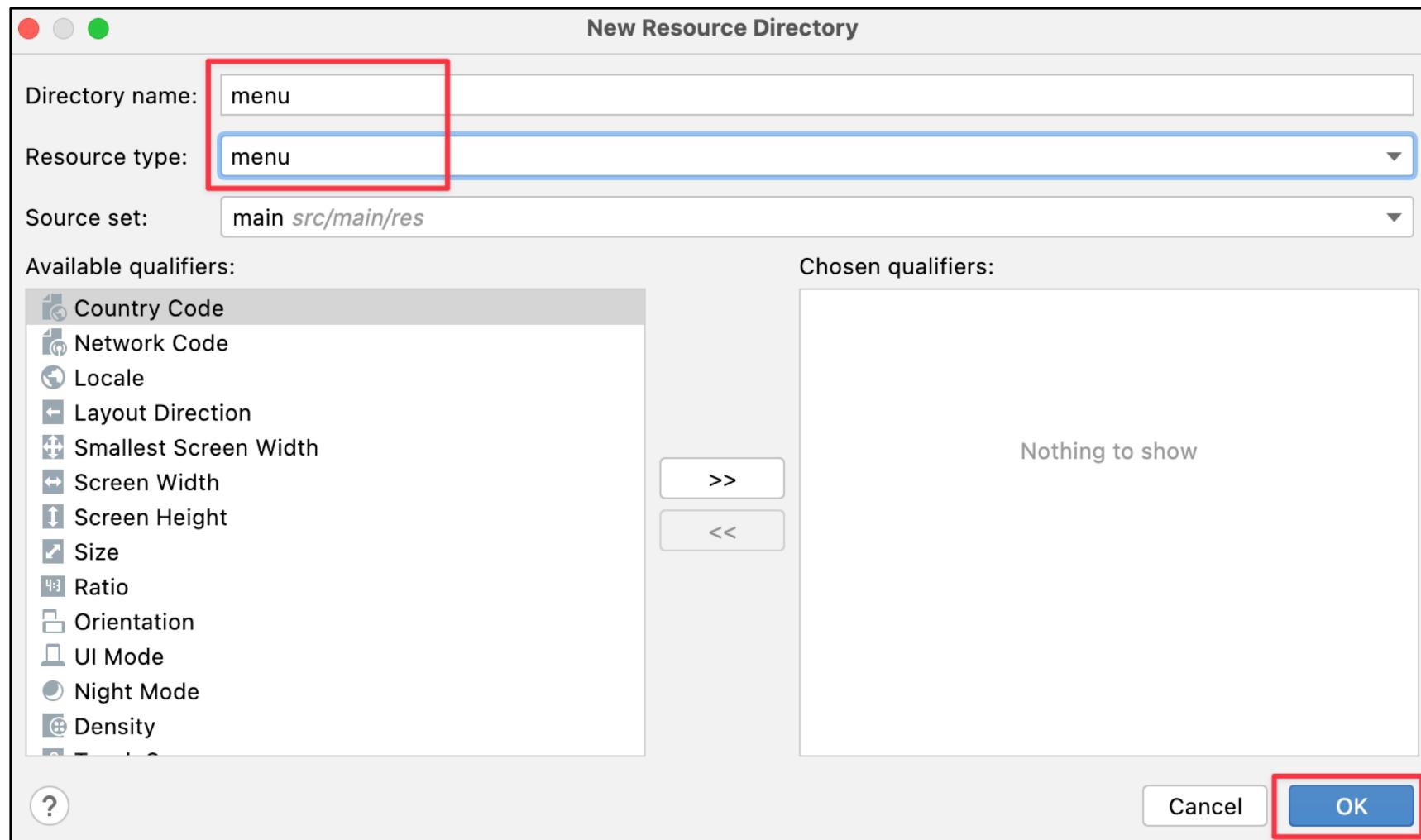
- ❖ **OptionsMenu**
- ❖ **ContextMenu**
- ❖ **PopupMenu**



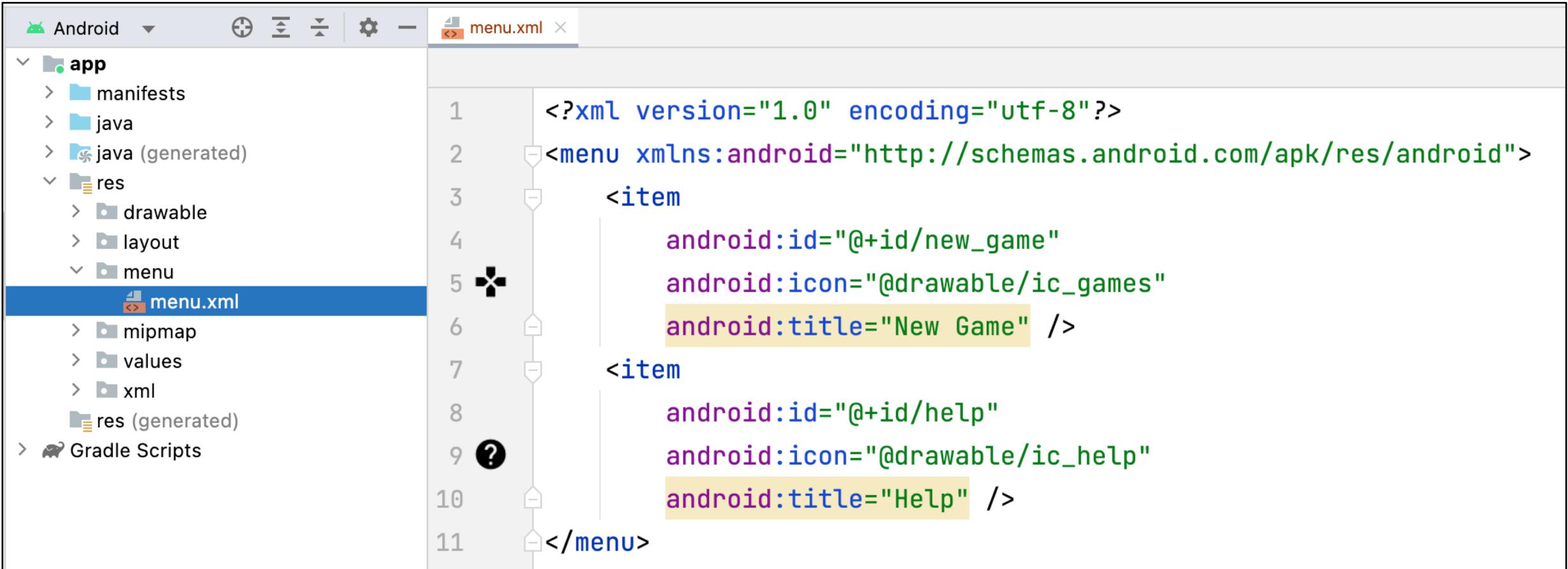
- ❑ Android cung cấp một định dạng XML chuẩn để định nghĩa menu (bao gồm options menu, context menu và popup menu).
- ❑ Để định nghĩa menu, chúng ta tạo tập tin XML trong thư mục **menu** (*nếu thư mục này chưa tồn tại, bạn phải tạo nó trong thư mục res*)
 - ❖ Chuột phải vào **res** → chọn **New** → chọn **Android Resource Directory**



❖ **Resource type** chọn **Menu**, sau đó nhấn **OK**



- ❑ Sử dụng các thành phần sau để xây dựng menu:
 - ❖ **<menu>** - Định nghĩa một Menu, dùng để chứa menu items. Thành phần **<menu>** phải là root của tập tin XML và có thể chứa một hoặc nhiều **<item>** và **<group>**.
 - ❖ **<item>** - Tạo một MenuItem, cái mà sẽ hiển thị trong menu. Thành phần này có thể chứa **<menu>** bên trong để tạo ra *submenu*.
 - ❖ **<group>** - Không bắt buộc, được sử dụng để nhóm các menu
- ❑ Có thể cho **<menu>** lồng vào **<item>** để tạo thành menu đa cấp

Ví dụ:

The screenshot shows the Android Studio interface with the project structure on the left and the code editor on the right.

Project Structure:

- app
 - manifests
 - java
 - java (generated)
 - res
 - drawable
 - layout
 - menu
 - menu.xml
 - mipmap
 - values
 - xml
 - res (generated)
- Gradle Scripts

Code Editor (menu.xml):

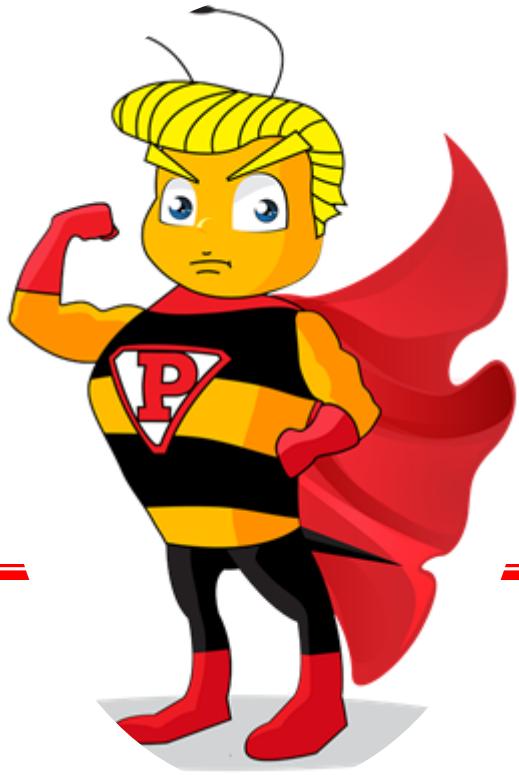
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/new_game"
        android:icon="@drawable/ic_games"
        android:title="New Game" />
    <item
        android:id="@+id/help"
        android:icon="@drawable/ic_help"
        android:title="Help" />
</menu>
```

The code defines two menu items: "New Game" and "Help". The "New Game" item has an icon from the ic_games.drawable resource and the title "New Game". The "Help" item also has an icon from the ic_help.drawable resource and the title "Help".

❑ Một số thuộc tính cần chú ý:

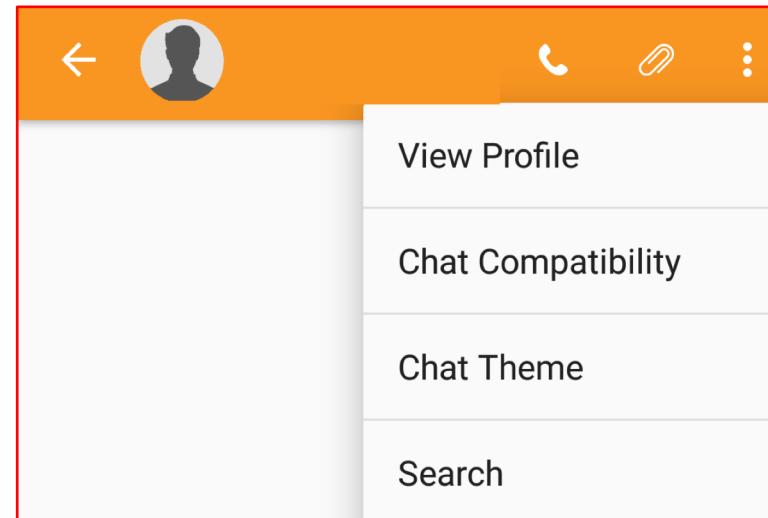
- ❖ **id** – xác định item trong menu để xử lý chức năng tương ứng trong java
- ❖ **icon** – icon của menu item
- ❖ **title** – tiêu đề hiển thị của từng menu item
- ❖ **showAsAction**
 - ifRoom*** – hiển thị menu item lên ActionBar/Toolbar nếu đủ chỗ, không đủ chỗ sẽ đưa vào Overflow icon
 - always*** – luôn luôn hiển thị trên ActionBar/Toolbar
 - never*** – luôn hiển thị trong Overflow icon

...



OPTIONSMENU

- ❑ **OptionsMenu** là nơi bao gồm các hành động và các tùy chọn khác có liên quan đến screen hiện tại, ví dụ như: tìm kiếm, cài đặt,...
- ❑ Đối với Android 2.3.x trở về trước, options menu sẽ hiển thị ở bên dưới màn hình khi người dùng nhấn nút Menu.
- ❑ Từ Android 3.0 trở lên, vị trí menu sẽ xuất hiện là ở app bar của ứng dụng.

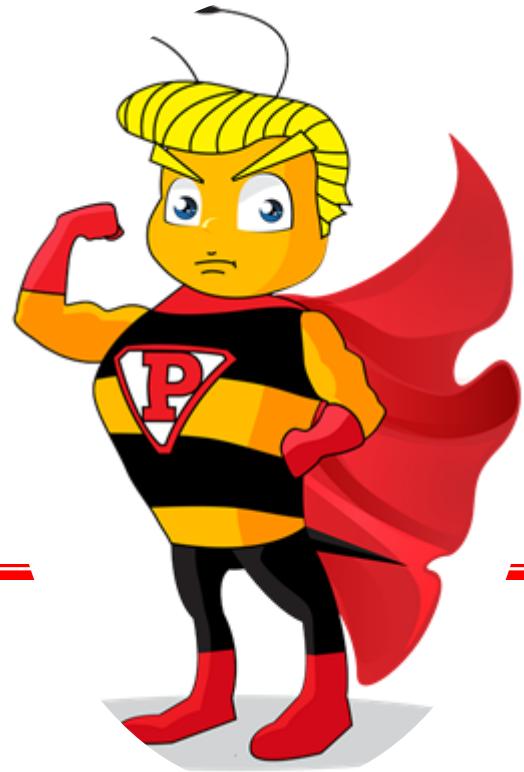


- Override hàm **onCreateOptionsMenu** để gắn **OptionsMenu** lên ActionBar/Toolbar

```
@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.menu, menu);  
    return super.onCreateOptionsMenu(menu);  
}
```

- ❑ Gọi hàm **onOptionsItemSelected** để xử lý sự kiện khi nhấn vào item bất kỳ trong menu

```
@Override  
public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.new_game:  
            Toast.makeText(context: this, text: "New Game", Toast.LENGTH_SHORT).show();  
            break;  
        case R.id.help:  
            Toast.makeText(context: this, text: "Help", Toast.LENGTH_SHORT).show();  
            break;  
        default:  
            break;  
    }  
    return super.onOptionsItemSelected(item);  
}
```



CONTEXTMENU

- ❑ **ContextMenu** xuất hiện khi người dùng tác động lên một view nào đó.
- ❑ Ví dụ như khi người dùng long click (nhấn giữ) vào Listview hoặc một vài thao tác khác để người dùng lựa chọn (context menu)



- ❑ Trong **onCreate** đăng ký view được nhận menu

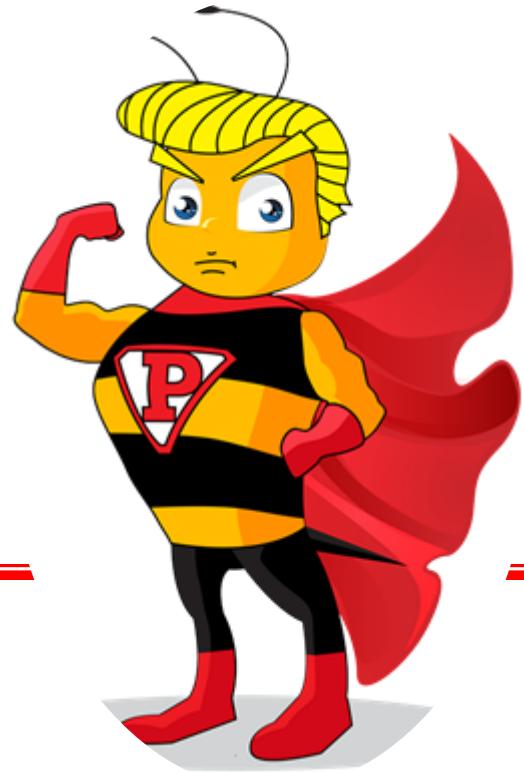
```
registerForContextMenu(lvFood);
```

- ❑ Override hàm **onCreateContextMenu** để tạo **ContextMenu**

```
@Override  
public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuItemInfo menuInfo){  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.context_menu, menu);  
    super.onCreateContextMenu(menu, v, menuInfo);  
}
```

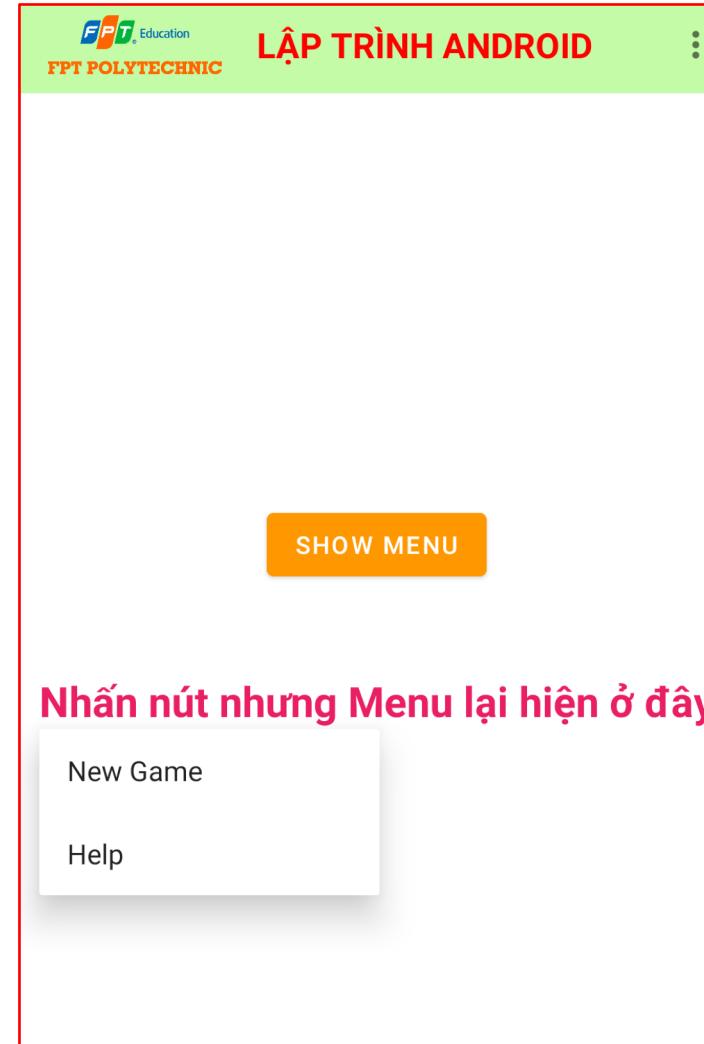
- ❑ Gọi hàm **onContextItemSelected** để xử lý sự kiện khi nhấp vào item bất kỳ trong menu

```
@Override  
public boolean onContextItemSelected(@NonNull MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.edit:  
            Toast.makeText(context: this, text: "Edit", Toast.LENGTH_SHORT).show();  
            break;  
        case R.id.delete:  
            Toast.makeText(context: this, text: "Delete", Toast.LENGTH_SHORT).show();  
            break;  
        default:  
            break;  
    }  
    return super.onContextItemSelected(item);  
}
```



POPUPMENU

- ❑ **PopupMenu** là menu có thể hiển thị ở trên bất kỳ View nào



☐ Tạo PopupMenu

```
PopupMenu popupMenu = new PopupMenu( context: MainActivity.this, txtShowHere);
popupMenu.getMenuInflater().inflate(R.menu.menu, popupMenu.getMenu());
new *
popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
    new *
    @Override
    public boolean onMenuItemClick(MenuItem menuItem) {
        switch (menuItem.getItemId()) {
            case R.id.new_game:
                Toast.makeText( context: MainActivity.this, text: "New Game", Toast.LENGTH_SHORT).show();
                break;
            case R.id.help:
                Toast.makeText( context: MainActivity.this, text: "Help", Toast.LENGTH_SHORT).show();
                break;
            default:
                break;
        }
        return false;
    }
});
```

popupMenu.show();

- ❑ Đổi số thứ 2 trong PopupMenu chính là vị trí mong muốn hiển thị Menu

```
PopupMenu popupMenu = new PopupMenu( context: MainActivity.this, txtShowHere);
```

- ❑ **popupMenu.setOnMenuItemClickListener** là hàm xử lý sự kiện khi nhấp vào item bất kỳ trong menu



FPT Education

FPT POLYTECHNIC

Thank you