

**Android Basic**

**(Part 8)**

Họ và tên : Trần Đức Lâm

Trường : Học viện công nghệ Bưu chính Viễn thông

Staff-code: INS046

Khóa: Open Education 12.

Đơn vị : Framgia

Hà Nội – 2018

1. Menu

* Định nghĩa :

Đối với tất cả các loại menu, Android cung cấp một định dạng XML chuẩn để định nghĩa các mục menu. Thay vì xây dựng một menu trong mã của hoạt động của bạn, bạn nên định nghĩa một menu và tất cả các mục của nó trong một [tài nguyên menu](https://developer.android.com/guide/topics/resources/menu-resource.html?hl=vi) XML. Khi đó, bạn có thể bung tài nguyên menu (tải nó như một đối tượng [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=vi)) trong hoạt động hoặc phân đoạn của mình.

Sử dụng một tài nguyên menu là một cách làm hay vì một vài lý do:

* Nó dễ trực quan hóa cấu trúc menu trong XML hơn.
* Nó tách riêng nội dung cho menu với mã hành vi của ứng dụng của bạn.
* Nó cho phép bạn tạo các cấu hình menu phái sinh cho các phiên bản nền tảng, kích cỡ màn hình khác nhau và các cấu hình khác bằng cách tận dụng khuôn khổ [tài nguyên ứng dụng](https://developer.android.com/guide/topics/resources/index.html?hl=vi).

Để định nghĩa menu, hãy tạo một tệp XML bên trong thư mục res/menu/ dự án của bạn và xây dựng menu với các phần tử sau:

<menu>

Định nghĩa một [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=vi), đó là một bộ chứa các mục menu. Phần tử <menu> phải là một nút gốc cho tệp và có thể giữ một hoặc nhiều phần tử <item> và <group>.

<item>

Tạo một [MenuItem](https://developer.android.com/reference/android/view/MenuItem.html?hl=vi), nó biểu diễn một mục đơn trong một menu. Phần tử này có thể chứa một phần tử <menu>được lồng nhau để tạo một menu con.

<group>

Một bộ chứa tùy chọn, vô hình cho các phần tử &lt;item&gt;. Nó cho phép bạn phân loại các mục menu sao cho chúng chia sẻ các tính chất như trạng thái hiện hoạt và khả năng hiển thị. Để biết thêm thông tin, hãy xem phần nói về [Tạo Nhóm Menu](https://developer.android.com/guide/topics/ui/menus?hl=vi#groups).

Sau đây là một menu ví dụ có tên là game\_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\_new\_game"  
          android:title="@string/new\_game"  
          android:showAsAction="ifRoom"/>  
    <item android:id="@+id/help"  
          android:icon="@drawable/ic\_help"  
          android:title="@string/help" />  
</menu>

Phần tử <item> hỗ trợ một vài thuộc tính bạn có thể sử dụng để định nghĩa biểu hiện bên ngoài và hành vi của một mục. Các mục trong menu trên bao gồm những thuộc tính sau:

android:id

Một ID tài nguyên duy nhất đối với mục, nó cho phép ứng dụng có thể nhận ra mục đó khi người dùng chọn nó.

android:icon

Một tham chiếu tới một nội dung vẽ được để dùng làm biểu tượng của mục.

android:title

Một tham chiếu tới một xâu để dùng làm tiêu đề của mục.

android:showAsAction

Quy định thời điểm và cách thức mục này nên xuất hiện như một mục hành động trong [thanh hành động](https://developer.android.com/guide/topics/ui/actionbar.html?hl=vi).

Đây là những thuộc tính quan trọng nhất bạn nên sử dụng, nhưng còn nhiều thuộc tính sẵn có khác. Để biết thông tin về tất cả thuộc tính được hỗ trợ, hãy xem tài liệu [Tài nguyên Menu](https://developer.android.com/guide/topics/resources/menu-resource.html?hl=vi).

Bạn có thể thêm một menu con vào một mục trong bất kỳ menu nào (ngoại trừ menu con) bằng cách thêm một phần tử &lt;menu&gt; làm con của &lt;item&gt;. Các menu con thường hữu ích khi ứng dụng của bạn có nhiều chức năng mà có thể được tổ chức thành các chủ đề, như các mục trong thanh menu của một ứng dụng PC (Tệp, Chỉnh sửa, Dạng xem, v.v.). Ví dụ:

<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
    <item android:id="@+id/file"  
          android:title="@string/file" >  
        <!-- "file" submenu -->  
        <menu>  
            <item android:id="@+id/create\_new"  
                  android:title="@string/create\_new" />  
            <item android:id="@+id/open"  
                  android:title="@string/open" />  
        </menu>  
    </item>  
</menu>

Để sử dụng menu trong hoạt động của mình, bạn cần bung tài nguyên menu (chuyển tài nguyên XML thành một đối tượng có thể lập trình) bằng cách sử dụng [MenuInflater.inflate()](https://developer.android.com/reference/android/view/MenuInflater.html?hl=vi#inflate(int,%20android.view.Menu)). Trong những phần sau, bạn sẽ biết cách bung một menu đối với mỗi loại menu.

1. Tạo menu tùy chọn

Bạn có thể khai báo các mục cho menu tùy chọn từ lớp con [Activity](https://developer.android.com/reference/android/app/Activity.html?hl=vi) của bạn hoặc một lớp con [Fragment](https://developer.android.com/reference/android/app/Fragment.html?hl=vi). Nếu cả hoạt động của bạn và (các) phân đoạn đều khai báo các mục cho menu tùy chọn, chúng sẽ được kết hợp lại trong UI. Các mục của hoạt động xuất hiện trước, sau đó là các mục của từng phân đoạn theo thứ tự phân đoạn được thêm vào hoạt động. Nếu cần, bạn có thể sắp xếp lại các mục menu bằng thuộc tính android:orderInCategory trong mỗi &lt;item&gt; mà bạn cần di chuyển.

Để quy định menu tùy chọn cho một hoạt động, hãy khống chế [onCreateOptionsMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onCreateOptionsMenu(android.view.Menu)) (các phân đoạn cung cấp phương pháp gọi lại [onCreateOptionsMenu()](https://developer.android.com/reference/android/app/Fragment.html?hl=vi#onCreateOptionsMenu(android.view.Menu,%20android.view.MenuInflater)) của chính mình). Trong phương pháp này, bạn có thể bung tài nguyên menu của mình ([được định nghĩa trong XML](https://developer.android.com/guide/topics/ui/menus?hl=vi#xml)) vào [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=vi) được cung cấp trong phương pháp gọi lại.

Ví dụ :

@Override  
public boolean onCreateOptionsMenu(Menu menu) {  
    MenuInflater inflater = [**getMenuInflater()**](https://developer.android.com/reference/android/app/Activity.html?hl=vi#getMenuInflater());  
    inflater.inflate(R.menu.game\_menu, menu);  
    return true;  
}

1. Sự kiện menu

Khi người dùng chọn một mục từ menu tùy chọn (bao gồm các mục hành động trong thanh hành động), hệ thống sẽ gọi phương pháp [onOptionsItemSelected()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onOptionsItemSelected(android.view.MenuItem)) của hoạt động của bạn. Phương pháp này thông qua [MenuItem](https://developer.android.com/reference/android/view/MenuItem.html?hl=vi) được chọn. Bạn có thể nhận biết mục bằng cách gọi [getItemId()](https://developer.android.com/reference/android/view/MenuItem.html?hl=vi#getItemId()), nó trả về ID duy nhất cho mục menu (được định nghĩa bởi thuộc tính android:id trong tài nguyên menu hoặc bằng một số nguyên được cấp cho phương pháp [add()](https://developer.android.com/reference/android/view/Menu.html?hl=vi#add(int,%20int,%20int,%20int))). Bạn có thể khớp ID này với các mục menu đã biết để thực hiện hành động phù hợp.   
Ví dụ :

@Override  
public boolean onOptionsItemSelected(MenuItem item) {  
    // Handle item selection  
    switch (item.getItemId()) {  
        case R.id.new\_game:  
            newGame();  
            return true;  
        case R.id.help:  
            showHelp();  
            return true;  
        default:  
            return super.onOptionsItemSelected(item);  
    }  
}

1. Thay đổi các mục menu

Sau khi hệ thống gọi [onCreateOptionsMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onCreateOptionsMenu(android.view.Menu)), nó sẽ giữ lại một thực thể của [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=vi) mà bạn đưa vào và sẽ không gọi lại [onCreateOptionsMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onCreateOptionsMenu(android.view.Menu)) trừ khi menu bị vô hiệu hóa vì lý do nào đó. Tuy nhiên, bạn chỉ nên sử dụng [onCreateOptionsMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onCreateOptionsMenu(android.view.Menu)) để tạo trạng thái menu ban đầu chứ không phải để thực hiện thay đổi trong vòng đời của hoạt động.

Nếu bạn muốn sửa đổi menu tùy chọn dựa trên các sự kiện xảy ra trong vòng đời của hoạt động, bạn có thể làm vậy trong phương pháp [onPrepareOptionsMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onPrepareOptionsMenu(android.view.Menu)). Phương pháp này chuyển cho bạn đối tượng [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=vi) như hiện đang có để bạn có thể sửa đổi nó, chẳng hạn như thêm, xóa bỏ, hoặc vô hiệu hóa các mục. (Phân đoạn cũng cung cấp lệnh gọi lại [onPrepareOptionsMenu()](https://developer.android.com/reference/android/app/Fragment.html?hl=vi#onPrepareOptionsMenu(android.view.Menu)).)

1. Tạo một menu ngữ cảnh nổi.

Để cung cấp một menu ngữ cảnh nổi:

Đăng ký [View](https://developer.android.com/reference/android/view/View.html?hl=vi) mà menu ngữ cảnh nên được liên kết với bằng cách gọi [registerForContextMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#registerForContextMenu(android.view.View)) và chuyển cho nó [View](https://developer.android.com/reference/android/view/View.html?hl=vi).

Nếu hoạt động của bạn sử dụng một [ListView](https://developer.android.com/reference/android/widget/ListView.html?hl=vi) hoặc [GridView](https://developer.android.com/reference/android/widget/GridView.html?hl=vi) và bạn muốn từng mục cung cấp cùng menu ngữ cảnh, hãy đăng ký tất cả mục cho một menu ngữ cảnh bằng cách chuyển [ListView](https://developer.android.com/reference/android/widget/ListView.html?hl=vi) hoặc [GridView](https://developer.android.com/reference/android/widget/GridView.html?hl=vi) cho [registerForContextMenu()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#registerForContextMenu(android.view.View)).

Triển khai phương pháp [onCreateContextMenu()](https://developer.android.com/reference/android/view/View.OnCreateContextMenuListener.html?hl=vi#onCreateContextMenu(android.view.ContextMenu,%20android.view.View,%20android.view.ContextMenu.ContextMenuInfo)) trong [Activity](https://developer.android.com/reference/android/app/Activity.html?hl=vi) hoặc [Fragment](https://developer.android.com/reference/android/app/Fragment.html?hl=vi) của bạn.

Khi dạng xem được đăng ký nhận được một sự kiện nhấp giữ, hệ thống sẽ gọi phương pháp [onCreateContextMenu()](https://developer.android.com/reference/android/view/View.OnCreateContextMenuListener.html?hl=vi#onCreateContextMenu(android.view.ContextMenu,%20android.view.View,%20android.view.ContextMenu.ContextMenuInfo)) của bạn. Đây là nơi bạn định nghĩa các mục menu, thường bằng cách bung một tài nguyên menu. Ví dụ:

@Override  
public void onCreateContextMenu(ContextMenu menu, View v,  
                                ContextMenuInfo menuInfo) {  
    super.onCreateContextMenu(menu, v, menuInfo);  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.context\_menu, menu);  
}

[MenuInflater](https://developer.android.com/reference/android/view/MenuInflater.html?hl=vi) cho phép bạn bung menu ngữ cảnh từ một [tài nguyên menu](https://developer.android.com/guide/topics/resources/menu-resource.html?hl=vi). Các tham số của phương pháp gọi lại bao gồm [View](https://developer.android.com/reference/android/view/View.html?hl=vi) mà người dùng đã chọn và một đối tượng [ContextMenu.ContextMenuInfo](https://developer.android.com/reference/android/view/ContextMenu.ContextMenuInfo.html?hl=vi) cung cấp thông tin bổ sung về mục được chọn. Nếu hoạt động của bạn có một vài dạng xem mà mỗi dạng cung cấp một menu ngữ cảnh khác nhau, bạn có thể sử dụng những tham số này để xác định menu ngữ cảnh nào cần bung.

Triển khai [onContextItemSelected()](https://developer.android.com/reference/android/app/Activity.html?hl=vi#onContextItemSelected(android.view.MenuItem)).

Khi người dùng chọn một mục menu, hệ thống sẽ gọi phương pháp này để bạn có thể thực hiện hành động phù hợp. Ví dụ:

@Override  
public boolean onContextItemSelected(MenuItem item) {  
    AdapterContextMenuInfo info = (AdapterContextMenuInfo) item.getMenuInfo();  
    switch (item.getItemId()) {  
        case R.id.edit:  
            editNote(info.id);  
            return true;  
        case R.id.delete:  
            deleteNote(info.id);  
            return true;  
        default:  
            return super.onContextItemSelected(item);  
    }  
}

Phương pháp [getItemId()](https://developer.android.com/reference/android/view/MenuItem.html?hl=vi#getItemId()) sẽ truy vấn ID cho mục menu được chọn, bạn nên gán mục này cho từng mục menu trong XML bằng cách sử dụng thuộc tính android:id như trình bày trong phần về [Định nghĩa một Menu trong XML](https://developer.android.com/guide/topics/ui/menus?hl=vi#xml).

1. Preferences
2. Tổng quan

Thay vì sử dụng các đối tượng [View](https://developer.android.com/reference/android/view/View.html) để xây dựng giao diện người dùng, thiết đặt được xây dựng bằng cách sử dụng các lớp con khác nhau của lớp [Preference](https://developer.android.com/reference/android/preference/Preference.html) mà bạn khai báo trong một tệp XML.

Đối tượng [Preference](https://developer.android.com/reference/android/preference/Preference.html) là một khối dựng cho một thiết đặt đơn lẻ. Mỗi [Preference](https://developer.android.com/reference/android/preference/Preference.html) xuất hiện như một mục trong một danh sách và cung cấp UI phù hợp để người dùng sửa đổi thiết đặt. Ví dụ, một [CheckBoxPreference](https://developer.android.com/reference/android/preference/CheckBoxPreference.html) tạo một mục danh sách hiển thị một hộp kiểm, và một [ListPreference](https://developer.android.com/reference/android/preference/ListPreference.html) tạo một mục mở ra một hộp thoại với danh sách lựa chọn.

Mỗi [Preference](https://developer.android.com/reference/android/preference/Preference.html) mà bạn thêm có một cặp khóa-giá trị tương ứng mà hệ thống sử dụng để lưu thiết đặt trong một tệp [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences.html) mặc định cho thiết đặt của ứng dụng của bạn. Khi người dùng thay đổi một thiết đặt, hệ thống sẽ cập nhật giá trị tương ứng trong tệp [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences.html) cho bạn. Lần duy nhất mà bạn nên trực tiếp tương tác với tệp [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences.html) được liên kết đó là khi bạn cần đọc giá trị để xác định xem hành vi ứng dụng của mình có được dựa trên thiết đặt của người dùng không.

Giá trị được lưu trong [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences.html) cho từng thiết đặt có thể là một trong các kiểu dữ liệu sau:

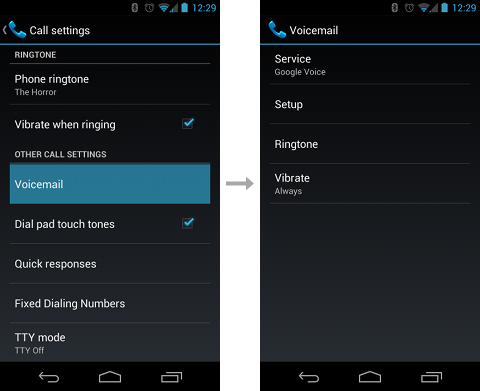
* Boolean
* Float
* Int
* Long
* String
* String [Set](https://developer.android.com/reference/java/util/Set.html)

Vì thiết đặt của ứng dụng của bạn được xây dựng bằng cách sử dụng các đối tượng [Preference](https://developer.android.com/reference/android/preference/Preference.html) thay vì đối tượng[View](https://developer.android.com/reference/android/view/View.html), bạn nên sử dụng một lớp con [Activity](https://developer.android.com/reference/android/app/Activity.html) hoặc [Fragment](https://developer.android.com/reference/android/app/Fragment.html) chuyên dụng để hiển thị thiết đặt danh sách:

* Nếu ứng dụng của bạn hỗ trợ các phiên bản Android cũ hơn 3.0 (API mức 10 và thấp hơn), bạn phải xây dựng hoạt động như một phần mở rộng của lớp [PreferenceActivity](https://developer.android.com/reference/android/preference/PreferenceActivity.html).
* Trên phiên bản Android 3.0 trở lên, thay vào đó, bạn nên sử dụng một [Activity](https://developer.android.com/reference/android/app/Activity.html) truyền thống nơi lưu giữ [PreferenceFragment](https://developer.android.com/reference/android/preference/PreferenceFragment.html) để hiển thị thiết đặt ứng dụng của bạn. Tuy nhiên, bạn cũng có thể sử dụng [PreferenceActivity](https://developer.android.com/reference/android/preference/PreferenceActivity.html) để tạo một bố trí hai bảng cho màn hình lớn khi bạn có nhiều nhóm thiết đặt.

Cách thiết đặt [PreferenceActivity](https://developer.android.com/reference/android/preference/PreferenceActivity.html) của bạn và các thực thể của [PreferenceFragment](https://developer.android.com/reference/android/preference/PreferenceFragment.html) được trình bày trong các phần về [Tạo một Hoạt động Tùy chọn](https://developer.android.com/guide/topics/ui/settings#Activity) và [Sử dụng Phân đoạn Tùy chọn](https://developer.android.com/guide/topics/ui/settings#Fragment).

1. PreferenceScreen

Demo :

<PreferenceScreen  xmlns:android="http://schemas.android.com/apk/res/android">  
    <!-- opens a subscreen of settings -->  
    <PreferenceScreen  
        android:key="button\_voicemail\_category\_key"  
        android:title="@string/voicemail"  
        android:persistent="false">  
        <ListPreference  
            android:key="button\_voicemail\_provider\_key"  
            android:title="@string/voicemail\_provider" ... />  
        <!-- opens another nested subscreen -->  
        <PreferenceScreen  
            android:key="button\_voicemail\_setting\_key"  
            android:title="@string/voicemail\_settings"  
            android:persistent="false">  
            ...  
        </PreferenceScreen>  
        <RingtonePreference  
            android:key="button\_voicemail\_ringtone\_key"  
            android:title="@string/voicemail\_ringtone\_title"  
            android:ringtoneType="notification" ... />  
        ...  
    </PreferenceScreen>  
    ...  
</PreferenceScreen>

1. Sử dụng

Bạn có thể tạo cả ý định biểu thị và không biểu thị bằng cách sử dụng các thuộc tính sau:

android:action

Hành động cần gán, theo mỗi phương pháp [setAction()](https://developer.android.com/reference/android/content/Intent.html#setAction(java.lang.String)) .

android:data

Dữ liệu cần gán, theo mỗi phương pháp [setData()](https://developer.android.com/reference/android/content/Intent.html#setData(android.net.Uri)).

android:mimeType

Kiểu MIME cần gán, theo mỗi phương pháp [setType()](https://developer.android.com/reference/android/content/Intent.html#setType(java.lang.String)) .

android:targetClass

Phần lớp của tên thành phần, theo mỗi phương pháp [setComponent()](https://developer.android.com/reference/android/content/Intent.html#setComponent(android.content.ComponentName)).

android:targetPackage

Phần gói của tên thành phần, theo mỗi phương pháp [setComponent()](https://developer.android.com/reference/android/content/Intent.html#setComponent(android.content.ComponentName)).

1. Shared Preferences

Bạn tạo có thể định nghĩa một số hành vi quan trọng cho ứng dụng của bạn, vì thế bạn cần phải khởi tạo tệp [SharedPreferences](https://developer.android.com/reference/android/content/SharedPreferences.html) kèm theo với các giá trị mặc định cho từng [Preference](https://developer.android.com/reference/android/preference/Preference.html) khi người dùng lần đầu mở ứng dụng của bạn.

Điều đầu tiên bạn phải làm đó là quy định một giá trị mặc định cho từng đối tượng [Preference](https://developer.android.com/reference/android/preference/Preference.html) trong tệp XML của bạn bằng cách sử dụng thuộc tính android:defaultValue. Giá trị đó có thể là bất kỳ kiểu dữ liệu nào mà phù hợp với đối tượng [Preference](https://developer.android.com/reference/android/preference/Preference.html) tương ứng. Ví dụ:

<!-- default value is a boolean -->  
<CheckBoxPreference  
    android:defaultValue="true"  
    ... />  
  
<!-- default value is a string -->  
<ListPreference  
    android:defaultValue="@string/pref\_syncConnectionTypes\_default"  
    ... />

Khi đó, từ phương pháp [onCreate()](https://developer.android.com/reference/android/app/Activity.html#onCreate(android.os.Bundle)) trong hoạt động chính —của ứng dụng của bạn và trong bất kỳ hoạt động nào khác mà thông qua đó người dùng có thể vào ứng dụng của bạn lần đầu tiên—hãy gọi [setDefaultValues()](https://developer.android.com/reference/android/preference/PreferenceManager.html#setDefaultValues(android.content.Context,%20int,%20boolean)):

PreferenceManager.setDefaultValues(this, R.xml.advanced\_preferences, false);

Việc gọi này trong khi [onCreate()](https://developer.android.com/reference/android/app/Activity.html#onCreate(android.os.Bundle)) sẽ đảm bảo rằng ứng dụng của bạn được khởi tạo phù hợp với các thiết đặt mặc định mà ứng dụng của bạn có thể cần đọc để xác định một số hành vi (chẳng hạn như có tải xuống dữ liệu trong khi đang trên mạng di động hay không).

Phương pháp này dùng ba tham đối:

* [Context](https://developer.android.com/reference/android/content/Context.html) ứng dụng của bạn.
* ID tài nguyên cho tệp XML tùy chọn mà bạn muốn đặt các giá trị mặc định cho.
* Một boolean cho biết các giá trị mặc định có nên được đặt nhiều hơn một lần hay không.

Khi tham đối này là false, hệ thống sẽ đặt các giá trị mặc định chỉ khi phương pháp này chưa từng được gọi trước đây (hoặc [KEY\_HAS\_SET\_DEFAULT\_VALUES](https://developer.android.com/reference/android/preference/PreferenceManager.html#KEY_HAS_SET_DEFAULT_VALUES) trong tệp tùy chọn được chia sẻ giá trị mặc định là sai).

Miễn là bạn đặt tham đối thứ ba này thành false, bạn có thể gọi phương pháp này một cách an toàn mỗi khi hoạt động của bạn bắt đầu mà không khống chế các tùy chọn đã lưu của người dùng bằng cách đặt lại chúng thành mặc định. Tuy nhiên, nếu bạn đặt nó thành true, bạn sẽ khống chế mọi giá trị trước đó bằng các giá trị mặc định.