

# Intent trong lập trình Android

## Part 1

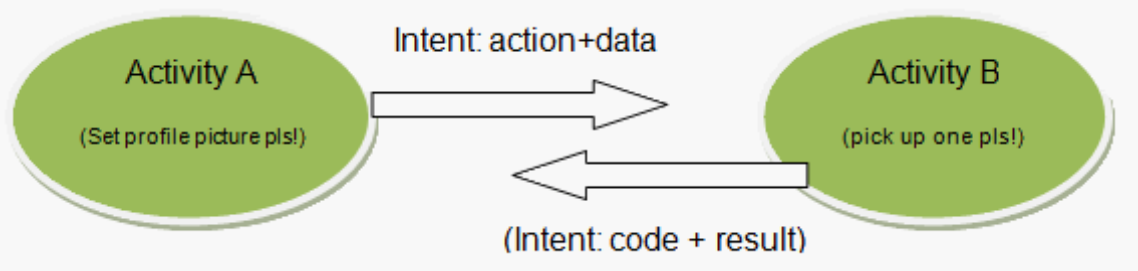
Trong bài này mình sẽ trình bày tóm tắt những kiến thức cơ bản nhất về Intent trong lập trình Android, cách truyền nhận thông tin qua Intent và minh họa bằng một ví dụ đơn giản. Hy vọng sẽ giúp các bạn mới làm quen với Android nắm bắt được một cách nhanh chóng.

### I- Intent là gì?

- Là một cấu trúc dữ liệu mô tả cách thức, đối tượng thực hiện của một Activity
- Là cầu nối giữa các Activity : ứng dụng Android thường bao gồm nhiều Activity, mỗi Activity hoạt động độc lập với nhau và thực hiện những công việc khác nhau.

Intent chính là người đưa thư, giúp chúng ta triệu gọi cũng như truyền các dữ liệu cần thiết để thực hiện một Activity từ một Activity khác. Điều này cũng giống như việc di chuyển qua lại giữa các Forms trong lập trình Windows Form

⚠ This image has been resized. Click on image to view the fullsize!



(Chú ý : trong hình vẽ trên Activity B chỉ trả về kết quả khi cần thiết. VD : giả sử Activity A nhắc người dùng chọn ảnh profile ; Activity B liệt kê các ảnh trong sdcard và cho phép người dùng chọn ảnh. Khi đó cặp “code+result” là cần thiết và có thể là “0:null” tức cancel hoặc “1:ảnh 20” tức chọn ảnh 20)


Để biết thêm về Activity [xem tại đây](#).

-Intent là một khái niệm then chốt và đặc trưng của Android Platform. Có thể nói lập trình Android chính là lập trình intent-base.

II-Intent chứa những dữ liệu gì ?

-Intent về cơ bản là một cấu trúc dữ liệu, được mô tả trong lớp android.content.Intent

-Các thuộc tính của một đối tượng Intent :

 This image has been resized. Click on image to view the fullsize!

Thuộc tính chính	Thuộc tính phụ
<b>action</b> -tên (string) của action mà Intent sẽ yêu cầu thực hiện -có thể là action được Android định nghĩa sẵn (built-in standard action) hoặc do người lập trình tự định nghĩa	<b>category</b> -thông tin về nhóm của action
	<b>type</b> -định dạng kiểu dữ liệu (chuẩn MIME) -thường được tự động xác định
<b>data</b> -dữ liệu mà Activity được gọi sẽ xử lý -định dạng Uri (thông qua hàm Uri.parse(data))	<b>component</b> -chỉ định cụ thể lớp sẽ thực thi Activity -khi được xác định, các thuộc tính khác trở thành không bắt buộc (optional)
	<b>extras</b> -chứa tất cả các cặp (key,value) do ứng dụng thêm vào để truyền qua Intent (cấu trúc Bundle)
<a href="http://developer.android.com/reference/android/content/Intent.html">http://developer.android.com/reference/android/content/Intent.html</a>	

-Các action định nghĩa sẵn :



Built-in Standard Actions	
<a href="#">ACTION_MAIN</a> <a href="#">ACTION_VIEW</a> <a href="#">ACTION_ATTACH_DATA</a> <a href="#">ACTION_EDIT</a> <a href="#">ACTION_PICK</a> <a href="#">ACTION_CHOOSER</a> <a href="#">ACTION_GET_CONTENT</a> <a href="#">ACTION_DIAL</a> <a href="#">ACTION_CALL</a> <a href="#">ACTION_SEND</a>	<a href="#">ACTION_ANSWER</a> <a href="#">ACTION_INSERT</a> <a href="#">ACTION_DELETE</a> <a href="#">ACTION_RUN</a> <a href="#">ACTION_SYNC</a> <a href="#">ACTION_PICK_ACTIVITY</a> <a href="#">ACTION_SEARCH</a> <a href="#">ACTION_WEB_SEARCH</a> <a href="#">ACTION_FACTORY_TEST</a> <a href="#">ACTION_SENDTO</a>
Built-in Standard Broadcast Actions	
<a href="#">ACTION_TIME_TICK</a> <a href="#">ACTION_TIME_CHANGED</a> <a href="#">ACTION_TIMEZONE_CHANGED</a> <a href="#">ACTION_BOOT_COMPLETED</a> <a href="#">ACTION_PACKAGE_ADDED</a> <a href="#">ACTION_PACKAGE_CHANGED</a> <a href="#">ACTION_PACKAGE_REMOVED</a>	<a href="#">ACTION_PACKAGE_RESTARTED</a> <a href="#">ACTION_PACKAGE_DATA_CLEARED</a> <a href="#">ACTION_UID_REMOVED</a> <a href="#">ACTION_BATTERY_CHANGED</a> <a href="#">ACTION_POWER_CONNECTED</a> <a href="#">ACTION_POWER_DISCONNECTED</a> <a href="#">ACTION_SHUTDOWN</a>

Đây là những hằng String đã được định nghĩa sẵn trong lớp Intent. Đi kèm với nó là các Activity hay Application được xây dựng sẵn sẽ được triệu gọi mỗi khi Intent tương ứng được gửi (tất nhiên khi được cung cấp đúng data). VD:

**+Dial một số phone:**

PHP Code:


```
Intent dialIntent = new Intent(Intent.ACTION_DIAL, Uri.  
parse("tel:123456"));  
startActivity(dialIntent);
```

**+Hiện danh bạ điện thoại:**

PHP Code:

```
Intent listContacts = new Intent(Intent.ACTION_VIEW,
                                Uri.pa
rse("content://contacts/people/"));
startActivity(listContacts);
```

Đến đây chắc bạn sẽ tự hỏi những chuỗi data trong hàm Uri.parse(data) có nghĩa là gì? Đó là định dạng dữ liệu tương ứng với mỗi action (chuẩn RFC 3986). Một khi bạn đã sử dụng built-in action thì bạn phải cung cấp data cho nó theo định dạng này. Bảng dưới đây liệt kê một số định dạng và action tương ứng đã được định nghĩa sẵn:

 This image has been resized. Click on image to view the fullsize!

Định dạng	Action	Mô tả
tel:phone_number	ACTION_VIEW	Mở Dial form (chưa gọi)
tel:phone_number	ACTION_CALL	Thực hiện gọi tới số phone
http://web_address https://web_address	ACTION_VIEW	Mở trình duyệt web với địa chỉ được cấp
"some_words" (string) http://web_address https://web_address	ACTION_WEB_SEARCH	Thực hiện search
sms://	ACTION_SENDTO	Gửi tin nhắn
geo:latitude,longitude geo:latitude,longitude?z=zoom geo:0,0?q=my+street+address geo:0,0?q=business+near+city	ACTION_VIEW	Mở ứng dụng Maps và chỉ tới vị trí được xác định

-Tự định nghĩa action

Về nguyên tắc bạn có thể đặt tên action của một intent là bất cứ thứ gì theo chuẩn đặt tên thông thường, hay thậm chí dùng luôn hằng action đã định nghĩa sẵn như ACTION\_VIEW (hay “android.intent.action.VIEW”). Cái tên VIEW thực chất chỉ là một tên gọi tả, bạn có thể dùng nó với mục đích thực hiện một activity để ... gửi mail! Tuy nhiên điều đó rõ ràng là rất “ngớ ngẩn”. Thay vào đó ta hãy

dùng ACTION\_SEND hay ACTION\_SENDTO.

Việc đặt tên action cho intent đúng tên gọi tả còn có một ý nghĩa khác đó là app của bạn có thể được triệu gọi từ một app khác. Ví dụ bạn viết một app có activity đáp ứng intent ACTION\_SEND và để chia sẻ một bức ảnh lên trang web của bạn (giống như ta làm với Facebook, Flickr etc.) Khi đó có thể app của bạn sẽ là một lựa chọn chia sẻ ảnh của người dùng điện thoại.