

Like



Share



Subscribe

AnhNguyenNgoc

ProIT4All



Giới thiệu

Phương thức (Methods)



Fan page: <http://facebook.com/Proit4All>



Phương thức là gì?

☐ Phương thức (Method)

- ❖ Là một khối mã thực hiện một nhiệm vụ cụ thể
- ❖ Là một đoạn mã có thể tái sử dụng, được gọi từ các phần khác nhau trong chương trình
- ❖ Các Hàm được định nghĩa trong các lớp
- ❖ Được sử dụng để đóng gói chức năng và giúp mã dễ đọc và dễ bảo trì hơn.

☐ Có hai loại phương thức:

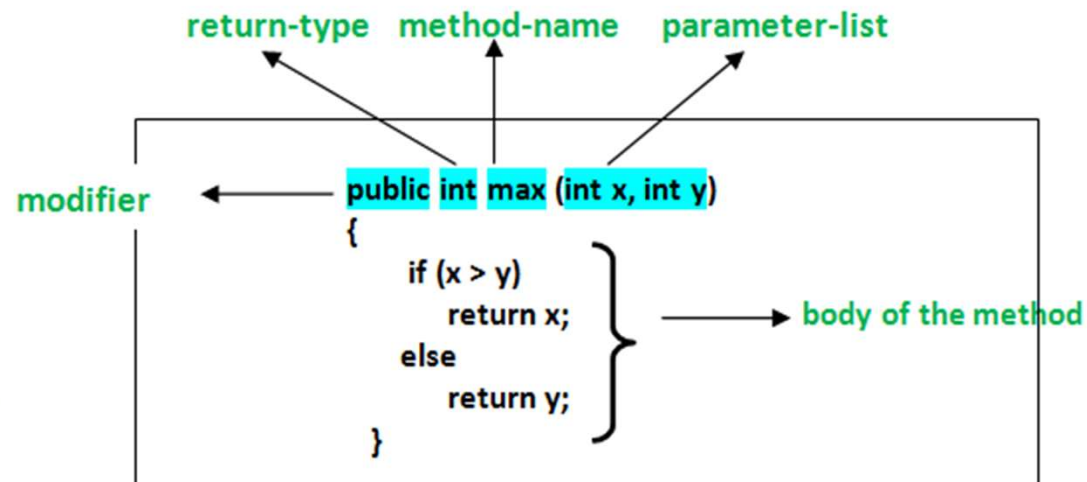
- ❖ Các phương thức do người dùng định nghĩa là các phương thức do người lập trình tự tạo
- ❖ Các phương thức thư viện là những phương thức được cung cấp bởi Java.
 - *Có thể sử dụng và không cần phải định nghĩa lại*

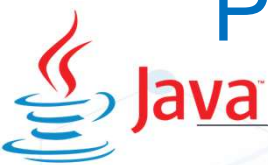


Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```





Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□Định nghĩa phương thức bao gồm:

- ❖access specifier (bộ xác định truy cập)
- ❖return type: kiểu trả về
- ❖method name: tên của phương thức
- ❖parameter list: danh sách tham số
- ❖method body: phần thân, hay nội dung xử lý của phương thức



Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□Access Specifier/Modifier (bộ xác định truy cập):

- ❖ Xác định phạm vi truy cập của phương thức
- ❖ Java cung cấp bốn loại chỉ định truy cập:
 - *public*
 - *private*
 - *protected*
 - *Default* (Mặc định)



Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□return type:

- ❖ Là kiểu của dữ liệu mà phương thức trả về.
- ❖ Có thể là kiểu dữ liệu nguyên thủy, đối tượng, kiểu tập hợp, **void**, v.v.
 - Nếu phương thức không trả về bất cứ thứ gì, thì sử dụng từ khóa **void**.



Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□method name:

- ❖ Là định danh (**Identifier**) duy nhất dùng để nhận biết, gọi phương thức.
- ❖ Tên phải có ý nghĩa và mô tả được chức năng xử lý của phương thức.



Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□Đặc tên cho phương thức:

- ❖ Tên của phương thức bắt đầu bằng một động từ và theo sau là danh từ hay tính từ
- ❖ Từ đầu tiên của tên được viết thường, các từ tiếp theo viết hoa ký tự đầu tiên
- ❖ Ví dụ: `sum()`, `area()`, `areaOfCircle()`, `stringComparison()`



Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□parameter list (Danh sách tham số):

- ❖ Là danh sách các tham số được phân tách bằng dấu phẩy (,) và được đặt trong cặp dấu ngoặc đơn.
- ❖ Mỗi tham số bao gồm: kiểu dữ liệu và tên biến tham số.
- ❖ Nếu phương thức không có tham số, để trống dấu ngoặc đơn.



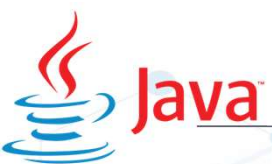
Phương thức là gì?

□Cú pháp:

```
[access specifier] <return type> <method name> ([Parameter list]){  
    //method body  
}
```

□method body (Phần thân phương thức):

- ❖chứa các mã xử lý của phương thức.
- ❖Đặt trong cặp dấu ngoặc nhọn {}.



Like & Share & Subscribe

Fan page: <http://facebook.com/Proit4All>



DEMO

Định nghĩa phương thức



Java™

