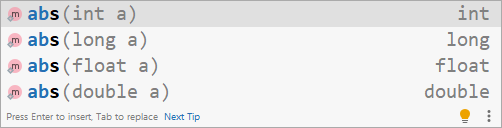
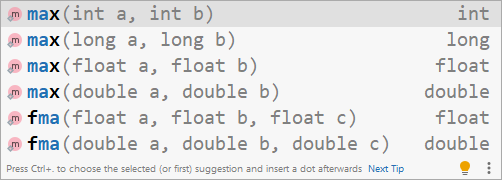
**Khái niệm**

* Nạp chồng phương thức là kĩ thuật cho phép nhiều phương thức có cùng tên nhưng khác tham số.
* Nạp chồng phương thức không được chấp nhận nếu hai hay nhiều phương thức chỉ khác nhau kiểu trả về.
* Ví dụ về phương thức nạp chồng:

[](https://braniumacademy.net/wp-content/uploads/2020/08/v%C3%AD-d%E1%BB%A5-overloading2.png)[](https://braniumacademy.net/wp-content/uploads/2020/08/v%C3%AD-d%E1%BB%A5-overloading1.png)

* Giải thích: các phương thức max cùng thực hiện chức năng tìm giá trị lớn nhất trong hai số. Tuy nhiên chúng có kiểu tham số khác nhau: int-int, long-long, double-double… và kiểu trả về tương ứng kiểu của tham số. Điều này giúp cho chương trình dễ dàng nhận các tham số ở các kiểu khác nhau. Do vậy có thể tìm được giá trị lớn nhất của các kiểu số thay vì chỉ 1 kiểu số.
* Tương tự như thế, phương thức abs() cũng nhận vào các tham số khác nhau và thực hiện lấy trị tuyệt đối cho các kiểu số khác nhau.

**Mục đích sử dụng**

* Nạp chồng phương thức giúp cho nhiều phương thức thực hiện cùng một chức năng có thể chia sẻ với nhau một cái tên chung.
* Việc này làm cho tên của phương thức được thống nhất và có thể áp dụng với nhiều kiểu dữ liệu khác nhau.
* Thay vì mỗi phương thức lại phải đặt một cái tên riêng gây khó khăn và không nhất quán.

**Yêu cầu của nạp chồng**

* Các phương thức nạp chồng phải cùng tên, thực hiện chức năng giống nhau.
* Các phương thức nạp chồng phải khác nhau về số lượng, thứ tự và kiểu của tham số.
* Các phương thức nạp chồng có thể khác nhau kiểu trả về.
* Các phương thức nạp chồng không thể chỉ khác nhau mỗi kiểu trả về, còn các thành phần khác giống nhau.
* Các phương thức nạp chồng thường nằm trong 1 lớp hoặc trong 1 chuỗi kế thừa(học sau).

**Ví dụ minh họa**

Sau đây là ví dụ minh họa nạp chồng phương thức:

* Hai phương thức nạp chồng sau không hợp lệ vì chỉ khác nhau kiểu trả về

*// tính và trả về tổng hai số nguyên int*

public static **void** **add(int a, int b)** { *// kiểu void*

var sum = a + b;

    System.*out*.println(sum);

}

*// tính và trả về tổng hai số nguyên int*

public static **int** **add(int a, int b)** { *// kiểu int*

var sum = a + b;

    return sum;

}

* Các phương thức nạp chồng sau là hợp lệ:

*// tính và trả về tổng hai số nguyên int*

public static int add(int a, int b) { *// kiểu int*

return a + b;

}

*// tính và trả về tổng hai số thực float*

public static float add(float a, float b) { *// kiểu float*

return a + b;

}

*// tính và trả về tổng hai số thực double*

public static double add(double a, double b) { *// kiểu double*

return a + b;

}

*// tính và trả về tổng hai số nguyên long*

public static long add(long a, long b) { *// kiểu long*

return a + b;

}

*// tính và trả về tổng hai số 3 nguyên int, int, long*

public static long add(int a, int b, long c) { *// kiểu long*

return a + b + c;

}