
Like & *Share* & *Subscribe*
thuydung **ProIT4All**

Khối lệnh finally

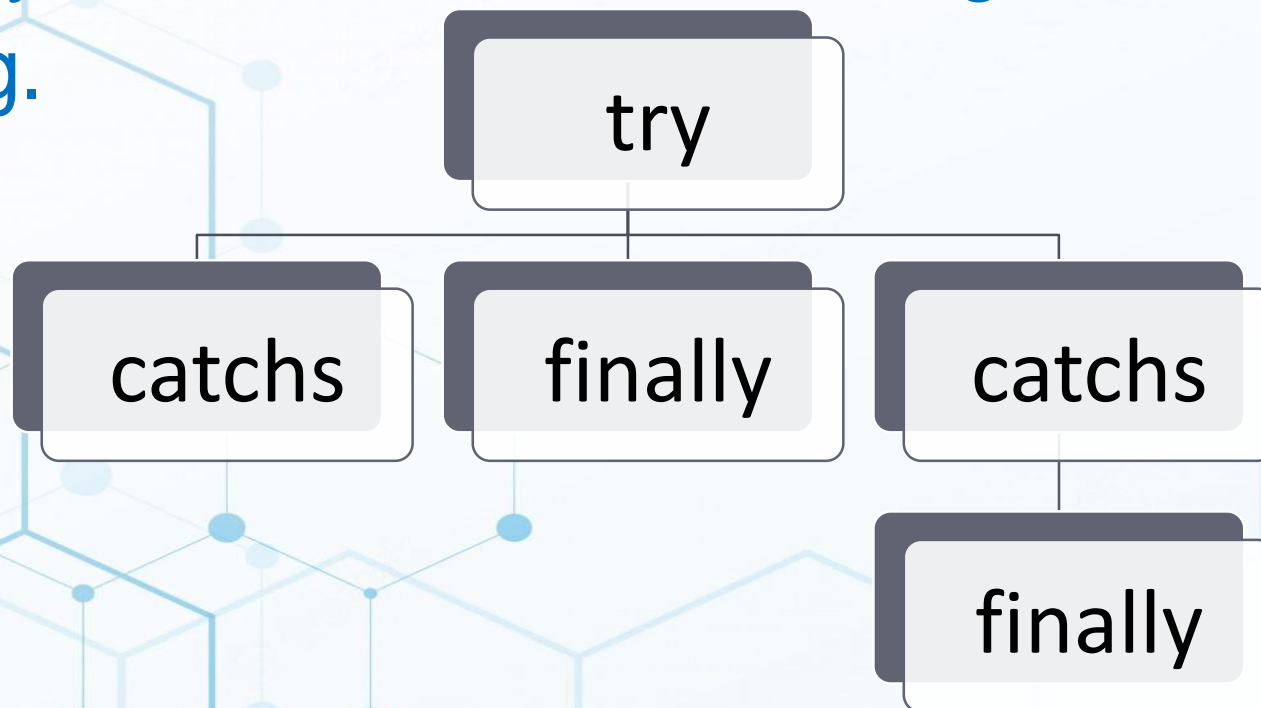


Fan page: <http://facebook.com/Proit4All>



Sử dụng finally

- ❑ Mỗi khối **try** yêu cầu có ít nhất một khối **catch** hoặc/và duy nhất một khối **finally**.
- ❑ Khối **finally** sẽ được thực hiện dù ngoại lệ có xuất hiện hay không.





Tại sao phải sử dụng finally?

□ Khối finally được sử dụng để

❖ Thực thi các “cleanup” code_code dọn dẹp như là:

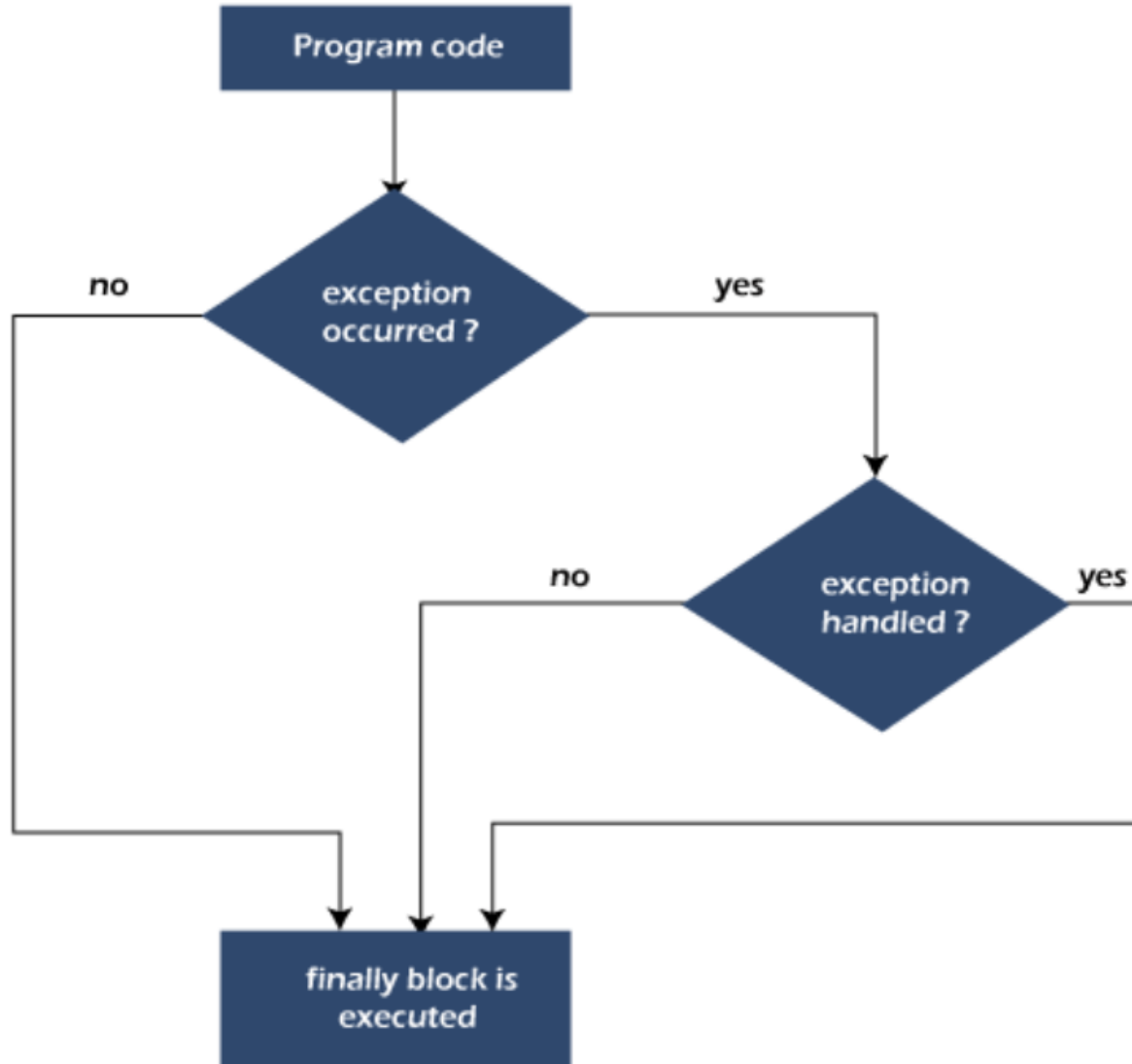
➤ đóng kết nối,

➤ đóng file,...

❖ Thực thi những câu lệnh quan trọng cần in ra



Các trường hợp sử dụng finally





Các trường hợp sử dụng finally

- ❑ Do đó, nó chứa tất cả các câu lệnh cần thiết cần được thực hiện bất kể ngoại lệ có xảy ra hay không.
- ❑ Khi ngoại lệ không xảy ra → Chương trình không ném ra ngoại lệ → và Khối finally được thực hiện sau khối try
- ❑ Có phát sinh ngoại lệ → Khối catch ko bắt được ngoại lệ → và Khối finally được thực hiện sau khối try → và chương trình kết thúc lỗi
- ❑ Có phát sinh ngoại lệ → Khối catch xử lý ngoại lệ → và Khối finally được thực hiện sau khối try → và phần còn lại của chương trình được thực hiện



Ví dụ 1

```
public static void main(String args[]) {  
    try {  
        //câu lệnh sau không ném ra ngoại lệ  
        int data = 50 / 5;  
        System.out.println(data);  
    } //catch sẽ không được thực hiện  
    catch (NullPointerException e) {  
        System.out.println(e);  
    } //Khối finally được thực hiện bất kể ngoại lệ có xảy ra hay không  
    finally {  
        System.out.println("finally block is always executed");  
    }  
  
    System.out.println("rest of the code...");  
}
```

output

10

finally block is always executed

rest of the code...



Ví dụ 2

```
public static void main(String args[]) {  
    try {  
        //câu lệnh sau ném ra ngoại lệ  
        int data = 50 / 0;  
        System.out.println(data);  
    }  
    //catch không xử lý ngoại lệ Arithmetic,  
    //chỉ chấp nhận ngoại lệ NullPointerException  
    catch (NullPointerException e) {  
        System.out.println(e);  
    } //Khối finally được thực hiện bất kể ngoại lệ có xảy ra hay không  
    finally {  
        System.out.println("finally block is always executed");  
    }  
  
    System.out.println("rest of the code...");  
}
```

output

finally block is always executed

Exception in thread "main" java.lang.ArithmeticException: / by zero



Ví dụ 3

```
public static void main(String args[]) {  
    try {  
        //câu lệnh sau ném ra ngoại lệ  
        int data = 50 / 0;  
        System.out.println(data);  
    }  
    //catch xử lý ngoại lệ Arithmetic  
    catch (ArithmeticException e) {  
        System.out.println(e);  
    } //Khởi finally được thực hiện bất kể ngoại lệ có xảy ra hay không  
    finally {  
        System.out.println("finally block is always executed");  
    }  
  
    System.out.println("rest of the code...");  
}
```

output

```
java.lang.ArithmeticException: / by zero  
finally block is always executed  
rest of the code...
```




Like & Share & Subscribe

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

DEMO

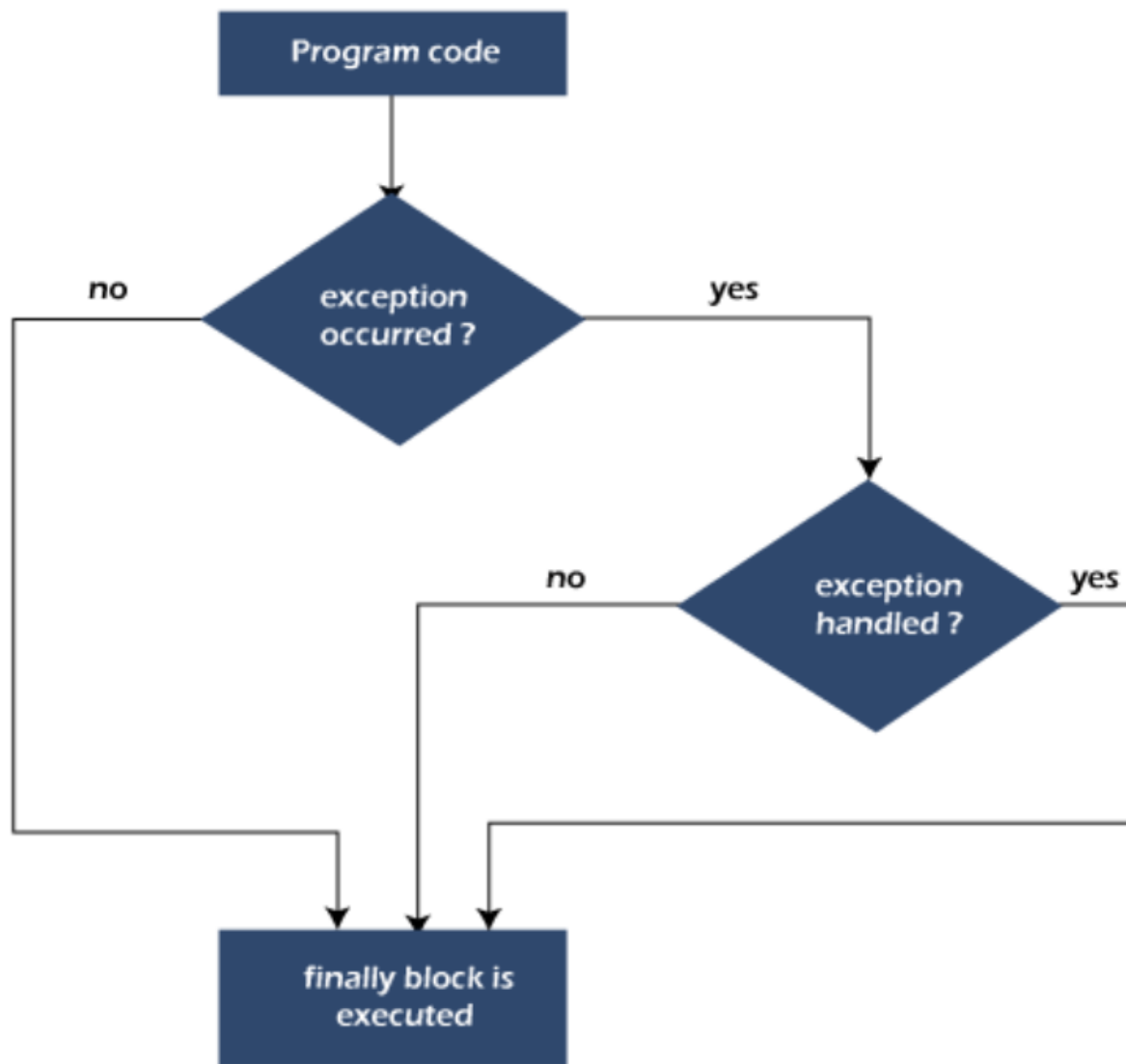


Java™





Các trường hợp sử dụng finally



thuydung

ProIT4All

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



THANK YOU

<http://youtube.com/@AnhNguyenNgoc>



Java™

