

Try-Catch-finally

Nguyễn Anh Tuấn

KTECH
COLLEGE



Nội dung bài giảng

- 1 Try-catch-finally
- 2 Throw - Throws
- 3 Cách Debug ứng dụng
- 4 Hướng dẫn assignment

Try-catch- finally

Try-catch-finally

❖ Định nghĩa:

- Khối **try-catch** trong Java là một công cụ để xử lý ngoại lệ, lỗi runtime.
- Khi một ngoại lệ xảy ra trong khối **try**, nó sẽ được bắt bởi khối **catch** tương ứng, giúp chương trình không bị dừng đột ngột.
- Khối **finally** sẽ luôn được thực thi dù có xảy ra ngoại lệ hay không.

❖ Lưu ý:

- Khối **catch** trong java được sử dụng để xử lý nếu xảy ra Exception, nếu không thì nó bị bỏ qua.
- Khối lệnh **try** trong java được sử dụng để chứa một đoạn code thực thi mà có thể trong quá trình thực thi nó sẽ xảy ra một ngoại lệ.
- Khối **catch** bắt buộc phải sử dụng ở sau khối **try**.
- Trong khối **try-catch** có thể triển khai nhiều khối **catch**, nhưng chỉ có thể triển khai 1 khối **try**.
- Có thể có hoặc không có khối **finally** sau **try-catch**.

Try-catch-finally

❖ Một số trường hợp có thể xảy ra ngoại lệ:

- Trường hợp lỗi logic
- Trường hợp sai kiểu dữ liệu
- Trường hợp lỗi khi làm việc với cơ sở dữ liệu
- Trường hợp lỗi kết nối mạng
- Trường hợp lỗi xử lý file
- Trường hợp không thể lường trước
- ...

Try-catch-finally

❖ **Cú pháp try-catch:**

```
try {  
    // Khối try dùng để bọc đoạn code có thể xảy ra ngoại lệ  
  
    // Truy cập vào phần tử nằm ngoài mảng  
    int[] numbers = {1, 2, 3};  
    System.out.println(numbers[10]);  
  
} catch (ArrayIndexOutOfBoundsException e) {  
    // Khối catch dùng để bắt ngoại lệ có thể xảy ra trong khối try  
  
    // Xử lý ngoại lệ  
    System.out.println("Lỗi: Truy cập chỉ số mảng không hợp lệ.");  
}
```

Try-catch-finally

❖ Một số ngoại lệ thường gặp:

- **Unchecked exceptions:** Những ngoại lệ do code sai, truyền tham số không đúng, ...
 - NullPointerException
 - ArrayIndexOutOfBoundsException
 - NumberFormatException
 - ArithmeticException
- **Checked exceptions:** Những ngoại lệ bắt buộc phải khai báo
 - IOException
 - SQLException
 - ClassNotFoundException

Throw - Throws

Throw và throws

Nội dung	Throw	Throws
Định nghĩa	Được sử dụng để ném một ngoại lệ cụ thể từ một phương thức hoặc trong khối try catch	Được sử dụng trong khai báo phương thức. Khi phương thức này được gọi, các ngoại lệ của phương thức đó sẽ được liệt kê
Vị trí sử dụng	Bên trong phần thân phương thức hoặc trong khối try catch	Khi khai báo phương thức
Loại ngoại lệ	Bao gồm cả Unchecked và Checked Exceptions	Bao gồm cả Unchecked và Checked Exceptions. Tuy nhiên nếu là Unchecked thì sẽ không có thông báo lỗi khi phương thức được gọi

Cách Debug ứng dụng

Cách Debug ứng dụng

❖ Một số keyword cần nhớ đối với **IntelliJ IDE**:

- **Debug point:** Điểm debug. Có thể đặt nhiều điểm debug ở mọi nơi trong chương trình. Khi khởi chạy debug, chương trình vẫn thực thi bình thường cho tới **debug point** sẽ tạm dừng lại.
- **F8:** Chạy đến dòng code tiếp theo
- **F7:** Chạy vào trong một phương thức
- **F9:** Chạy đến debug point tiếp theo. Nếu không có debug point tiếp theo, chương trình sẽ chạy đến khi kết thúc toàn bộ xử lý.
- **Add to watches:** Thêm 1 biến, 1 đối tượng hoặc giá trị trả về của một phương thức vào Watches để theo dõi giá trị của chúng trong quá trình thực thi chương trình.

Hướng dẫn assignment

ROAD TO KOREA

Nếu có bất kỳ thắc mắc nào, hãy đặt câu hỏi qua

mail@mail.com hoặc Zalo 0xxx xxx xxx