

Bài 58: Tổng quan về ngoại lệ

- ✓ Khái niệm và nguyên nhân
- ✓ Phân biệt ngoại lệ và lỗi
- ✓ Xử lý ngoại lệ
- ✓ Ví dụ minh họa

Khái niệm, nguyên nhân

- Ngoại lệ(exception) là sự kiện không mong muốn xảy ra trong quá trình chạy của chương trình
- Ngoại lệ xảy ra sẽ làm gián đoạn luồng thực thi của các tập lệnh do đó chương trình sẽ kết thúc bất thường
- Để chương trình không bị gián đoạn hay kết thúc đột ngột, ta phải xử lý ngoại lệ khi nó xảy ra

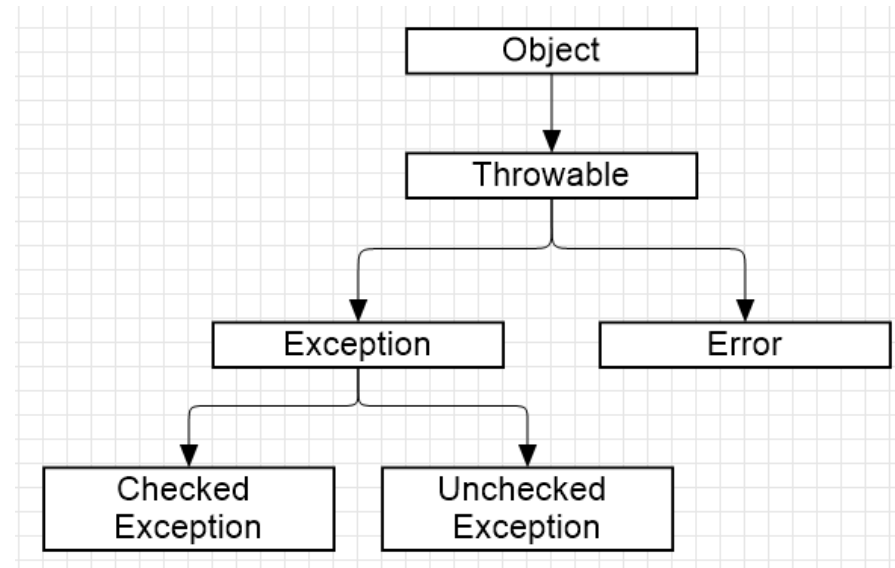
Khái niệm, nguyên nhân

Nguyên nhân xảy ra ngoại lệ có thể là chủ quan hoặc khách quan:

- Người dùng nhập dữ liệu không đúng định dạng
- File đầu vào cần đọc không tồn tại
- Lỗi logic chương trình của lập trình viên
- Bộ nhớ đầy trong quá trình chương trình chạy
- Kết nối bị ngắt khi đang thực hiện giao dịch giữa máy chủ và máy khách...

Ngoại lệ

➤ Sơ đồ phân cấp ngoại lệ:



➤ Ngoại lệ: những vấn đề nảy sinh và có thể xử lý bởi chương trình

Ví dụ

- Ví dụ: NullPointerException, ArrayIndexOutOfBoundsException
- Ta có thể phòng ngừa hoặc khắc phục những ngoại lệ này:

```
public static void main(String[] args) {  
    int[] numbers = {1, 2, 3, 6, 5, 4, 7, 8, 9};  
    int index = 100;  
    if (index >= 0 && index < numbers.length) {  
        System.out.println("Phần tử tại vị trí " +  
            index + " là: " + numbers[index]);  
    } else {  
        System.out.println("Chỉ số " + index + " không hợp lệ");  
        System.out.println("Chỉ số tối đa của mảng là: " +  
            numbers.length);  
    }  
}
```

Lỗi(Error)

- Khi ta nhắc đến error thường ám chỉ các vấn đề nghiêm trọng, vượt ngoài tầm kiểm soát, không thể xử lý bởi chương trình
- Ví dụ: lỗi hệ thống, lỗi phần cứng. Đây là các vấn đề không thể xử lý bằng chương trình.

Xử lý ngoại lệ

- Lập trình viên có thể phòng ngừa các ngoại lệ bằng cách lập luận logic code chặt chẽ, sử dụng các cấu trúc điều kiện để kiểm soát hành động
- Khi ngoại lệ xảy ra ta có hai hướng xử lý:
 - Một là xử lý triệt để với try-catch
 - Hai là trì hoãn việc xử lý ngoại lệ và chuyển việc xử lý cho nơi thực hiện lời gọi phương thức

Sử dụng try-catch

- Gói đoạn code có thể xảy ra ngoại lệ trong khối try và xử lý ngoại lệ nếu xảy ra trong catch:

```
SimpleDateFormat dateFormat =  
    new SimpleDateFormat("dd/MM/yyyy");  
var dateString = "27-05-2025";  
// chuyển đổi string sang đối tượng date:  
try {  
    // dòng code dưới đây có thể xảy ra ngoại lệ:  
    Date myBirthday = dateFormat.parse(dateString);  
    System.out.println("Đối tượng date hoàn chỉnh: "  
        + myBirthday);  
} catch (ParseException e) {  
    System.out.println("Không thể chuyển đổi String sang Date");  
    e.printStackTrace();  
}  
// các câu lệnh kế tiếp phía sau  
System.out.println("Thực hiện các câu lệnh tiếp theo...");
```


Trì hoãn ngoại lệ

- Ta có thể trì hoãn ngoại lệ bằng throws nếu như không biết cách xử lý ngoại lệ hoặc lười xử lý và muốn chuyển nhượng lại việc xử lý cho nơi thực hiện lời gọi:

```
public static void main(String[] args) throws ParseException {  
    SimpleDateFormat dateFormat =  
        new SimpleDateFormat("dd/MM/yyyy");  
    var dateString = "27-05-2025";  
    // chuyển đổi string sang đối tượng date:  
    // dòng code dưới đây có thể xảy ra ngoại lệ:  
    Date myBirthday = dateFormat.parse(dateString);  
    System.out.println("Đối tượng date hoàn chỉnh: " + myBirthday);  
    // các câu lệnh kế tiếp phía sau  
    System.out.println("Thực hiện các câu lệnh tiếp theo...");  
}
```



Minh họa

➤ Thực hiện trong công cụ lập trình

Nội dung tiếp theo

**Checked vs unchecked
exception**