Exception trong java

1. Exception là gì?

Exception là một sự kiện phá vỡ luồn chuẩn của chương trình. Nó là đối tượng được ném tai Runtime.

Một Exception được ném trong quá trình chương trình chạy là một vấn đề xảy ra trong quá trình thực hiện chương trình. Một Exception được ném có thể có nhiểu lý do:

+ Do người dùng nhập dữ liệu không chính xác.

+ Một file cần được mở nhưng không tìm thấy.

+ Kết nối mạng bị ngắt trong quá trình kết nối hoặc JVM hết bộ nhớ.

Có 3 loại Exception:

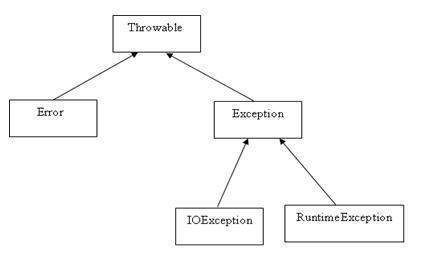
+ CheckedException: là ngoại lệ thường xảy ra do người dùng mà không lường trước đc. Exception này không thể bỏ qua trong quá trình biên dịch. CheckedException là các lớp thừa kế lớp Throwable ngoại trừ RuntimeException và Error. Ví dụ như IOException, SQLException,… CheckedException được kiểm tra tại thời điểm biên dịch.

+ UnCheckedException: là ngoại lệ xảy ra ở runtime và người lập trình có thể lường trước được. UncheckedException thừa kế từ lớp RuntimeException.

+ Error: Nó là những vấn đề ngoài tâm kiểm soát của lập trình viên cũng như người dùng. Nó được bỏ qua trong quá trình biên dịch. Những lỗi này như OutOfMemoryError, VirtualMachineError,..

2. Xử lí Exception

Xử lí Exception là 1 kĩ thuật xử lí các Runtime Error như ClassNotFound, IO, SQL, Remote,… Mục đích chính của việc xử lí Exception là để duy trì luồng chạy của chương trình không bị gián đoạn.



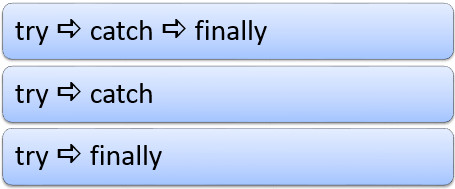
Exception thừa kế từ lớp Throwable và có 2 lớp con thừa kế từ nó là IOException và RuntimeException

3. Phương thức của lớp Exception

|  |  |
| --- | --- |
| **STT** | **Phương thức và Miêu tả** |
| 1 | **public String getMessage()**  Trả về một message cụ thể về exception đã xảy ra. Message này được khởi tạo bởi phương thức contructor của Throwable |
| 2 | **public Throwable getCause()**  Trả về nguyên nhân xảy ra exception biểu diễn bởi đối tượng Throwable |
| 3 | **public String toString()**  Trả về tên của lớp và kết hợp với kết quả từ phương thức getMessage() |
| 4 | **public void printStackTrace()**  In ra kết quả của phương thức toString cùng với stack trace đến System.err |
| 5 | **public StackTraceElement [] getStackTrace()**  Trả về một mảng chứa mỗi phần tử trên stack trace. Phần tử tại chỉ mục 0 biểu diễn phần trên cùng của Call Stack, và phần tử cuối cùng trong mảng biểu diễn phương thức tại dưới cùng của Call Stack |
| 6 | **public Throwable fillInStackTrace()**  Fills the stack trace of this Throwable object with the current stack trace, adding to any previous information in the stack trace. |

4.Cách xử lý ngoại lệ

4.1 Dùng khối try-catch



\* Khối finally luôn luôn thực hiện dù có exception hay k.

4.2 Dùng từ khóa throws

Từ khóa throws dùng trong phương thức dùng để đề xuất ra các ngoại lệ có thẻ xảy ra trong phương thức đó. Chỉ cần khai báo Exception đó trên phương thức để khi có ngoại lệ Exception sẽ ném lỗi ra cho chúng ta.

