

예외와 예외 클래스

■ 예외와 예외 클래스

- 에러(error) : 컴퓨터 하드웨어의 고장으로 인해 응용프로그램 실행 오류가 발생하는 것 (외부적 요인)
- 예외(exception) : 잘못된 사용 또는 코딩으로 인한 오류를 말한다.
- 예외가 발생되면 프로그램은 곧바로 종료된다는 점에서는 에러와 동일하지만,
- 예외 처리를 통해 계속 실행 상태를 유지할 수 있다. 예외는 다음 두 가지가 있다.

- 일반 예외(Exception)

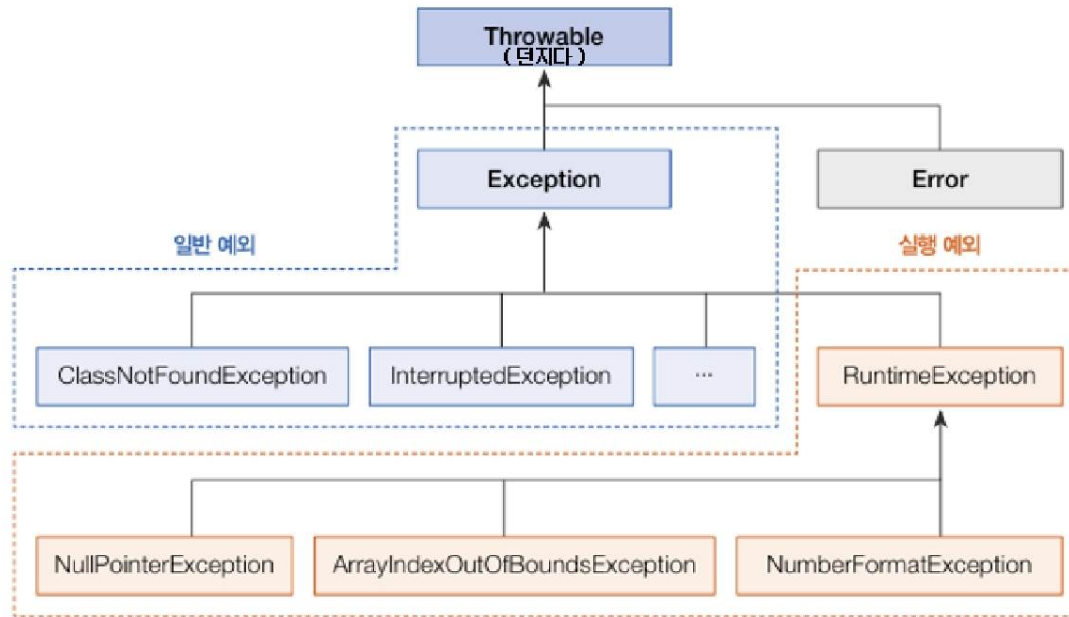
컴파일러가 예외 처리 코드 여부를 검사하는 예외를 말한다. (컴파일러가 체크 O)

- 실행 예외(Runtime Exception)

컴파일러가 예외 처리 코드 여부를 검사하지 않는 예외를 말한다. (컴파일러가 체크 X, 실행하고 나서 발생 -> 경험이 필요)

자바는 예외가 발생하면 예외 클래스로부터 객체를 생성한다. 이 객체는 예외 처리 시 사용된다.

[자바는 자주 사용되는 예외 클래스를 표준 라이브러리에서 제공]



ArithmeticException : 정수를 0으로 나눌경우 발생

ArrayIndexOutOfBoundsException : 배열의 범위를 벗어난 index를 접근할 시 발생

FileNotFoundException : 파일을 못 찾을 때 발생

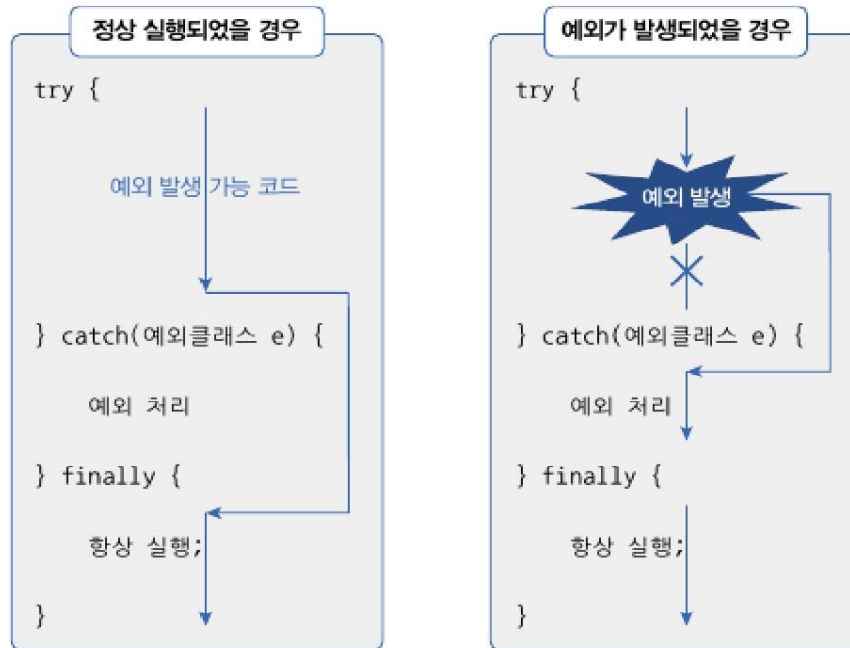
ClassCastException : 변환할 수 없는 타입으로 객체를 반환 시 발생

NullPointerException : 존재하지 않는 레퍼런스를 참조할 때 발생

IllegalArgumentException : 잘못된 인자를 전달할 때 발생

■ 예외 처리 코드

- 예외가 발생했을 때 프로그램의 갑작스러운 종료를 막고, 정상 실행을 유지할 수 있도록 처리하는 코드를 예외 처리 코드라고 한다.
- try-catch-finally 블록은 "생성자"와 "메소드" 내부에서 작성한다.



※ try ~ catch 블록에서 return(메소드 종료)을 사용하더라도 finally 블록은 항상 실행된다.
finally 블록은 옵션으로 생략 가능하다.