

# 예외처리

# 예외와 런타임 예외

---

## ❖ 오류의 종류

### ○ 에러(Error)

- 하드웨어의 잘못된 동작 또는 고장으로 인한 오류
- 에러가 발생되면 프로그램 종료
- 정상 실행 상태로 돌아갈 수 없음

### ○ 예외(Exception)

- 사용자의 잘못된 조작 또는 개발자의 잘못된 코딩으로 인한 오류
- 예외가 발생되면 프로그램 종료
- 예외 처리 추가하면 정상 실행 상태로 돌아갈 수 있음

# 예외와 런타임 예외

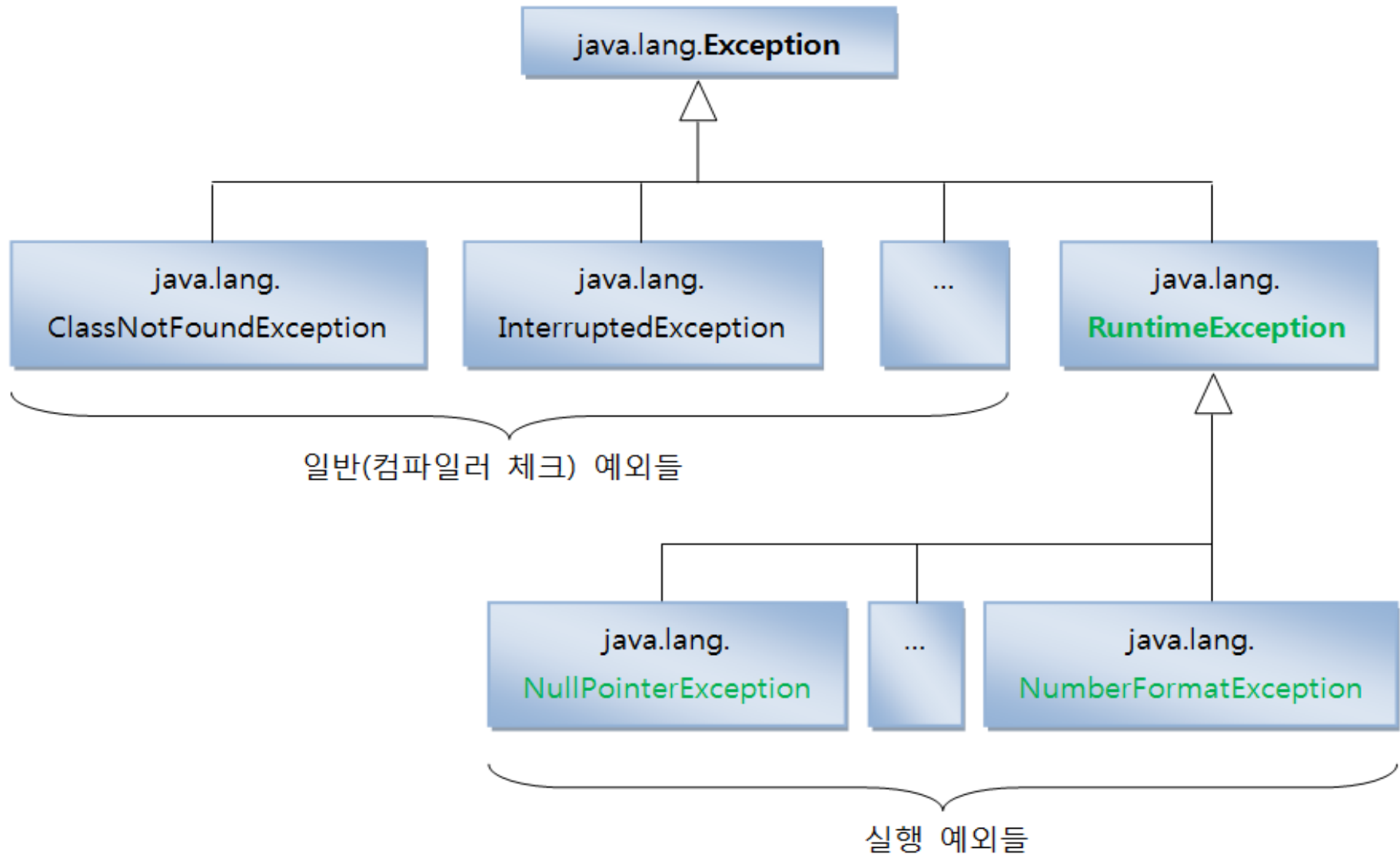
---

## ❖ 예외의 종류

- 일반(컴파일 체크) 예외(Exception)
  - 예외 처리 코드 없으면 컴파일 오류 발생
- 실행 예외(RuntimeException)
  - 예외 처리 코드를 생략하더라도 컴파일이 되는 예외
  - 경험 따라 예외 처리 코드 작성 필요

## 예외와 런타임 예외

### ❖ 예외 클래스



## 예외와 런타임 예외

---

### ❖ 실행 예외

- NullPointerException

- 객체 참조가 없는 상태
- null 값을 갖는 참조변수로 객체 접근 연산자인 도트(.) 사용했을 때 발생

```
String data = null;  
System.out.println(data.toString());
```

## 예외와 런타임 예외

---

### ❖ NullPointerException이 발생하는 경우: NullPointerExceptionExample.java

```
public class NullPointerExceptionExample {  
    public static void main(String[] args) {  
        String data = null;  
        System.out.println(data.toString());  
    }  
}
```

## 예외와 런타임 예외

---

### ❖ 실행 예외

- `ArrayIndexOutOfBoundsException`
  - 배열에서 인덱스 범위 초과하여 사용할 경우 발생

```
public static void main(String[] args) {  
    String data1 = args[0];  
    String data2 = args[1];  
  
    System.out.println("args[0]: " + data1);  
    System.out.println("args[1]: " + data2);  
}
```

## 예외와 런타임 예외

### ❖ **ArrayIndexOutOfBoundsException:** **ArrayIndexOutOfBoundsExceptionExample.java**

```
public class ArrayIndexOutOfBoundsExceptionExample {  
    public static void main(String[] args) {  
        if(args.length == 2) {  
            String data1 = args[0];  
            String data2 = args[1];  
            System.out.println("args[0]: " + data1);  
            System.out.println("args[1]: " + data2);  
        } else {  
            System.out.println("[실행 방법]");  
            System.out.print("java  ArrayIndexOutOfBoundsExceptionExample  ");  
            System.out.print("값1  값2");  
        }  
    }  
}
```



## 예외와 런타임 예외

### ❖ 실행 예외

- `NumberFormatException`

- 문자열을 숫자로 변환하는 방법

반환 타입	메소드명(매개 변수)	설명
int	<code>Integer.parseInt(String s)</code>	주어진 문자열을 정수로 변환해서 리턴
double	<code>Double.parseDouble(String s)</code>	주어진 문자열을 실수로 변환해서 리턴

- 타입별 `Wrapper` 클래스에서 `parseXXX()` 메서드 제공

- 문자열을 숫자로 변환 시도 시 실패하는 경우 `NumberFormatException` 예외 발생

## 예외와 런타임 예외

---

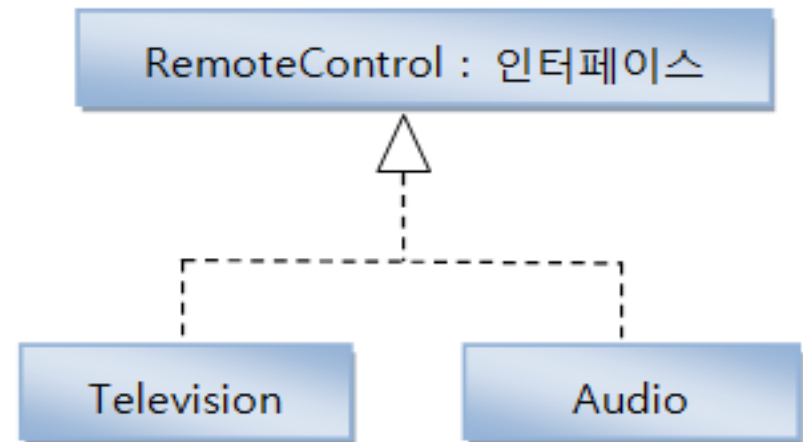
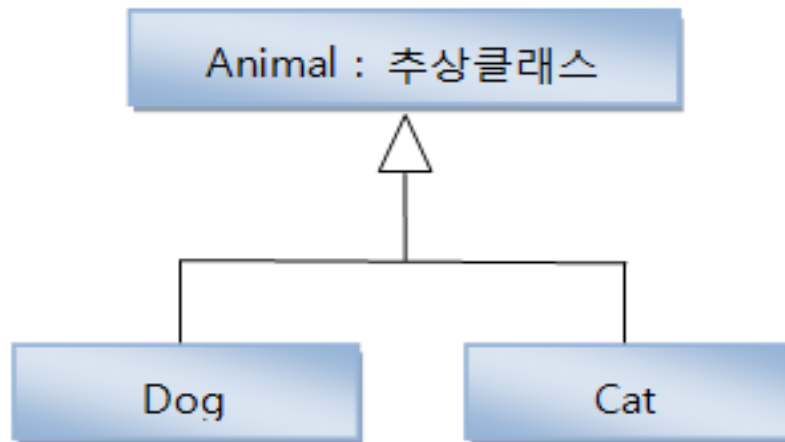
### ❖ NumberFormatException이 발생하는 경우: NumberFormatExceptionExample.java

```
public class NumberFormatExceptionExample {  
    public static void main(String[] args) {  
        String data1 = "100";  
        String data2 = "a100";  
  
        int value1 = Integer.parseInt(data1);  
        int value2 = Integer.parseInt(data2); // NumberFormatException 발생  
  
        int result = value1 + value2;  
        System.out.println(data1 + "+" + data2 + "=" + result);  
    }  
}
```

## 예외와 런타임 예외

### ❖ 실행 예외

- ClassCastException
  - 타입 변환이 되지 않을 경우 발생



## 예외와 런타임 예외

### ❖ 실행 예외

#### ○ ClassCastException

##### ▪ 정상 코드

```
Animal animal = new Dog();  
Dog dog = (Dog) animal;
```

```
RemoteControl rc = new Television();  
Television tv = (Television) rc;
```

##### ▪ 예외 발생 코드

```
Animal animal = new Dog();  
Cat cat = (Cat) animal;
```

```
RemoteControl rc = new Television();  
Audio audio = (Audio) rc;
```

##### ▪ instanceof 연산자로 캐스팅 가능여부 확인을 예외 발생 방지 가능

```
Animal animal = new Dog() ;  
if(animal instanceof Dog) {  
    Dog dog = (Dog) animal;  
} else if(animal instanceof Cat) {  
    Cat cat = (Cat) animal;  
}
```

```
Remocon rc = new Audio();  
if(rc instanceof Television) {  
    Television tv = (Television) rc;  
} else if(rc instanceof Audio) {  
    Audio audio = (Audio) rc;  
}
```

## 예외와 런타임 예외

### ❖ ClassCastException: ClassCastExceptionExample.java

```
public class ClassCastExceptionExample {
    public static void main(String[] args) {
        Dog dog = new Dog();
        changeDog(dog);

        Cat cat = new Cat();
        changeDog(cat);
    }

    public static void changeDog(Animal animal) {
        // if(animal instanceof Dog) {
        Dog dog = (Dog) animal; // ClassCastException 발생 가능
        // }
    }
}

class Animal { }
class Dog extends Animal { }
class Cat extends Animal { }
```