

2025년 상반기 K-디지털 트레이닝

예외 처리

[KB] IT's Your Life

✓ 다음 프로그램의 문제점을 설명하고, 그 결과를 확인하세요.

```
package ch11.sec02.exam01;

public class ExceptionHandlingExample1 {
    public static void printLength(String data) {
        int result = data.length();
        System.out.println("문자 수: " + result);
    }

    public static void main(String[] args) {
        System.out.println("[프로그램 시작]\n");
        printLength("ThisIsJava");
        printLength(null);
        System.out.println("[프로그램 종료]");
    }
}
```

✓ 앞의 프로그램에 예외 처리 코드를 추가하세요.

○ 예외 처리시 출력할 메시지

- 예외 객체의 메시지로 출력
- 스택 추적 내용을 모두 출력

✓ 다음 프로그램에 다중 예외 처리 코드를 추가하세요.

```
package ch11.sec03.exam01;

public class ExceptionHandlingExample {
    public static void main(String[] args) {
        String[] array = {"100", "100"};

        for(int i=0; i<=array.length; i++) {
            int value = Integer.parseInt(array[i]);
            System.out.println("array[" + i + "]: " + value);
        }
    }
}
```

✓ 다음 프로그램에 다중 예외 처리 코드를 추가하세요.

○ 처리할 예외

- `ArrayIndexOutOfBoundsException`
- `NullPointerException`
- `NumberFormatException`

○ `NullPointerException`와 `NumberFormatException`는 동일한 예외처리

```
package ch11.sec03.exam03;

public class ExceptionHandlingExample {
    public static void main(String[] args) {
        String[] array = {"100", "100", null, "200"};

        for(int i=0; i<=array.length; i++) {
            int value = Integer.parseInt(array[i]);
            System.out.println("array[" + i + "]: " + value);
        }
    }
}
```

- ✔ 다음과 같이 자원 클래스를 정의하고, AutoCloseable 인터페이스를 구현하세요.

```
package ch11.sec04;

public class MyResource {
    private String name;

    public MyResource(String name) {
        this.name = name;
        System.out.println("[MyResource(" + name + ") 열기]");
    }

    public String read1() {
        System.out.println("[MyResource(" + name + ") 읽기]");
        return "100";
    }

    public String read2() {
        System.out.println("[MyResource(" + name + ") 읽기]");
        return "abc";
    }
}
```

- ✔ 앞에서 정의한 **MyResource**에 대한 자동 닫기 예외처리 코드를 추가하세요.

```
package ch11.sec04;

public class TryWithResourceExample {
    public static void main(String[] args) {
        MyResource res = new MyResource("A");
        String data = res.read1();
        int value = Integer.parseInt(data);

        System.out.println();

        MyResource res = new MyResource("A");
        String data = res.read2();
        int value = Integer.parseInt(data);
        System.out.println();

        MyResource res1 = new MyResource("A");
        MyResource res2 = new MyResource("B");
        String data1 = res1.read1();
        String data2 = res2.read1();
    }
}
```

✓ 다음 코드가 예외 처리가 되도록 코드를 완성하세요.

```
package ch11.sec05;

public class ThrowsExample1 {
    public static void main(String[] args) {
        try {
            findClass();
        } catch(
            ) {
            System.out.println("예외 처리: " + e.toString());
        }
    }

    public static void findClass() {
        Class.forName("java.lang.String2");
    }
}
```


2025년 상반기 K-디지털 트레이닝

java.base 모듈

[KB] IT's Your Life

1 java.base 모듈

✓ 다음 조건을 만족하는 Student 클래스를 정의하세요.

- 멤버 필드를 초기화하는 생성자
- 멤버 필드에 대한 Getter/Setter 메서드
- equals(), hashCode(), toString() 메서드

```
package ch12.sec03.exam02;

public class Student {
    private int no;
    private String name;
}
```

- ✔ 다음과 같이 운영할 수 있도록 Member클래스를 record로 정의하세요.

```
package ch12.sec03.exam04;

public class RecordExample {
    public static void main(String[] args) {
        Member m = new Member("winter", "눈송이", 25);
        System.out.println(m.id());
        System.out.println(m.name());
        System.out.println(m.age());
        System.out.println(m.toString());
        System.out.println();

        Member m1 = new Member("winter", "눈송이", 25);
        Member m2 = new Member("winter", "눈송이", 25);
        System.out.println("m1.hashCode(): " + m1.hashCode());
        System.out.println("m2.hashCode(): " + m2.hashCode());
        System.out.println("m1.equals(m2): " + m1.equals(m2) );
    }
}
```

- ✓ Lombok 라이브러리를 다운로드 하세요.
- ✓ 프로젝트에 lib 폴더에 복사하세요.
- ✓ IntelliJ에 Lombok 플러그인을 설치하세요.
- ✓ Lombok 라이브러리를 프로젝트에 추가하세요.
- ✓ 프로젝트에 어노테이션 프로세싱(annotation processing) 기능을 활성화 하세요.

- ✓ 다음 조건을 만족하는 Member 클래스를 정의하세요.
 - Lombok을 이용하여 정의함
 - 기본 생성자, 전체 필드를 초기화하는 생성자,
 - Getter/Setter 메서드
 - equals(), hashCode(), toString() 메서드
 - 위의 메서드들이 제대로 생성되었는지 IntelliJ의 Structure 기능으로 확인하세요.

```
package ch12.sec03.exam05;

public class Member {
    private String id;
    private String name;
    private int age;
}
```

- ✓ 1부터 1000000까지의 합을 구하는 프로그램을 작성하고, 계산결과와 계산에 소요된 시간을 nano 초 단위로 출력하세요.

```
package ch12.sec04;

public class MeasureRunTimeExample {
    public static void main(String[] args) {

    }
}
```

✓ 다음 조건을 만족하는 프로그램을 작성하세요.

- 운영체제 명 출력
- 로그인 사용자 명과 홈 디렉토리 출력
- 시스템의 속성 정보 전체 출력

```
package ch12.sec04;

import java.util.Properties;
import java.util.Set;

public class GetPropertyExample {
    public static void main(String[] args) {
    }
}
```

✓ 다음 프로그램의 실행결과를 적어보고, 실제 결과와 비교하세요.

```
package ch12.sec05;

public class StringBuilderExample {
    public static void main(String[] args) {
        String data = new StringBuilder()
            .append("DEF")
            .insert(0, "ABC")
            .delete(3, 4)
            .toString();
        System.out.println(data);
    }
}
```


✓ 다음 조건을 만족하도록 프로그램을 수정하세요.

- split 메서드를 이용하여 "홍길동&이수홍,박연수"에서 이름 문자열만 추출하여 출력
- StringTokenizer를 이용하여 "홍길동/이수홍/박연수" 에서 이름 문자열만 추출하여 출력

```
package ch12.sec05;

import java.util.StringTokenizer;

public class StringTokenizerExample {
    public static void main(String[] args) {
        String data1 = "홍길동&이수홍,박연수";

        String data2 = "홍길동/이수홍/박연수";

    }
}
```

- ✓ 현재 날짜 데이터를 "2024.06.15 15:22:20"와 같은 포맷으로 출력하세요.

```
package ch12.sec08;

public class DateExample {
    public static void main(String[] args) {

    }
}
```

- ✓ **Calendar** 클래스를 이용하여 현재 날짜, 시간기준으로 다음과 같이 출력하세요.

```
package ch12.sec08;

import java.util.*;

public class CalendarExample {
    public static void main(String[] args) {

    }
}
```

2024년 6월 18일
화요일 오후
0시 25분 0초