

## 예외처리 관련 문제

이름 :

1. 예외에 대한 설명 중 틀린 것은 무엇입니까?

- ① 예외는 사용자의 잘못된 조작, 개발자의 잘못된 코딩으로 인한 프로그램 오류를 말한다.
- ② RuntimeException의 하위 예외는 컴파일러가 예외 처리 코드를 체크한다.
- ③ 예외는 try~catch 블록을 사용해서 처리된다.
- ④ 모든 예외 클래스의 최상위는 Exception이다.

2. Exception이 발생하는 코드이다. 실행 결과와 같이 출력될 수 있도록 메인 메서드에 코드를 작성하십시오.

```
public class Example {  
    public static void prn() throws Exception {  
        throw new Exception("예외가 발생!!");  
    }  
  
    public static void main(String[] args) {  
        prn();  
    }  
}
```

[실행 결과]

예외가 발생되어 처리합니다 java.lang.Exception: 예외가 발생!!

toString()

3. Exception이 발생할 때 메시지를 리턴하는 메서드는 무엇인가?

- ① getError()
- ② getMessage()
- ③ printMessage()
- ④ traceMessage()

4. parseInt() 메서드를 사용할 때 잘못된 데이터를 입력하게 되면 발생하는 Exception은 무엇인가?

- ① ArrayIndexOutOfBoundsException
- ② ClassCastException
- ③ NumberFormatException
- ④ NumberError

5. ( )안에 들어갈 예약어는?

- ① 예외처리를 위해 try~catch문장을 사용할 수도 있지만, 예외를 직접 처리하지 않고 미룰 때 사용하는 예약어는 ( )이다.
- ② 사용자가 예외를 직접 발생시키기 위해 사용하는 예약어는 ( )이다.

6. double d = 10/0;을 실행하게 되면 발생하는 Exception은 무엇인가?

- ① ArithmeticException
- ② NullPointerException
- ③ NegativeArraySizeException
- ④ 에러나지 않는다.

7. 아래에서의 코드를 참고한 후 4번째 줄의 코드에서 발생할 수 있는 예외클래스를 생각한 후 해당 예외 발생시 "부적절한 나눗셈을 시도하였습니다" 라는 구문이 출력될 수 있도록 try-catch를 이용하여 예외처리하는 구문을 기술하시오.

```
1: Scanner sc = new Scanner(System.in);
2: int num1 = sc.nextInt();
3: int num2 = sc.nextInt();
4: System.out.println("두 수의 나눗셈 결과 : " + num1 / num2 );
```

8. try~catch~finally 블록에 대한 설명 중 틀린것은 무엇입니까?

- ① try{} 블록에는 예외가 발생할 수 있는 코드를 작성한다.
- ② catch{} 블록은 try{} 블록에서 발생한 예외를 처리하는 블록이다.
- ③ try{} 블록에서 실행 후 finally{} 블록은 실행되지 않고 catch{} 블록 실행 후만 실행된다.
- ④ catch{} 블록은 예외의 종류별로 여러 개를 작성할 수 있다.

9. 다음과 같은 메소드가 있을 때 예외를 잘못 처리한 것은 무엇입니까?

```
public void method1() throws NumberFormatException, ClassNotFoundException { ... }
```

- ① try{ method1(); }catch(Exception e){ }
- ② void method2() throws Exception { method1();}
- ③ try{ method1(); }catch(Exception e){ }catch(ClassNotFoundException cnfe){ }
- ④ try{ method1(); }catch(ClassNotFoundException cnfe){ }catch(NumberFormatException nfe){ }

10. 0으로 나누기 연산시 발생할 수 있는 예외클래스에 대해 기술하시오.

11. 오류와 발생하는 예외를 올바르게 짝지으시오.

- |                            |   |                         |
|----------------------------|---|-------------------------|
| int[] list; list[0] = 0;   | • | •에러(error)              |
| 자바 가상 기계가 클래스를 찾을 수 없는 경우  | • | •체크예외(cheked exception) |
| 파일을 읽던 프로그램이 파일의 끝에 도달한 경우 | • | •컴파일 오류                 |
| 파일의 끝에 도달했는데도 파일을 읽으려고 시도  | • | •예외가 발생하지 않는다           |

12. 아래에서의 1번 코드를 2번 코드로도 기술 가능하다. 그 이유에 대해 서술하시오.

```
1.
try {
    int result = 10 / 0;
} catch(ArithmeticException e) {
    e.printStackTrace();
}

2.
try {
    int result = 10 / 0;
} catch(RuntimeException e) {
    e.printStackTrace();
}
```

13. 배열에서 부적절한 인덱스에 접근하려고 할 때 발생할 수 있는 예외클래스에 대해 기술하시오.

14. 배열의 크기로 음수를 제시했을 때 발생할 수 있는 예외클래스에 대해 기술하시오.

15. 다음 프로그램의 출력을 쓰시오.

```
try {
    int[] array=new int[-10];
    System.out.println("try");
} catch (NumberFormatException e){
    System.out.println("숫자 형식 오류");
} catch (NegativeArraySizeException e){
    System.out.println("배열 크기 음수 오류");
} catch (Exception e){
    System.out.println("오류");
} finally {
    System.out.println("처리 끝");
}
```