

23.12.26[java]

⌚ 작성 일시	@2023년 12월 26일 오전 9:55
≡ 태그	

예외/에러

예외와 오류의 차이는 무엇일까?
예외 : 예측 가능한 범위내의 에러
에러 : 의도한 결과가 나오지 않을경우
에러 **ex)**
구문 에러
;) {
예외 **ex)**
[존재하지 않는 파일을 열고자 한다.
{에러가 발생할수 밖에 없는 구조}]
파일명 **101.txt** 를 열려고 한다.
101.txt 경로상에 존재하지 않는다.

try/catch

try/catch

```
// 예외처리 오류 발생시 c의 값을 -1로 받아라(에러 조건문)
int c;
try{
    c = 4/0;
} catch (ArithmeticException f){ // 해당오류 발생시 아래 명령어 수행
    c = -1;
    System.out.println(c);    -1
}
```

try/catch/finally

try/catch/finally

```
package FlieIo.JavaExm12_26_01;

public class Exm_02 {
    2 usages
    public void beRun(){
        System.out.println("Ok");
    }

    public static void main(String[] args) {
        Exm_02 exm_02 =new Exm_02();
        int c;
        try{
            c=4/0;
            exm_02.beRun();
        }catch(ArithmeticException e){
            c=-1;
        }finally {
            exm_02.beRun();
        }
    }
}
```

Exception(처리 속도가 빠름)/RuntimeException(포괄적으로 사용)

Exception/RuntimeException

```
package FlieIo.JavaExm12_26_01;

import javax.annotation.processing.FilerException;

3 usages
class FoolExcetion extends RuntimeException{    //실행 시 예외 처리
                                                //예외 발생이 애매한 경우
                                                //처리 속도가 더결린다.
}

no usages
class ceb extends Exception{    //컴파일 발생 시 예외 처리
                                //예측 가능한경우
                                //RuntimeException 확인 후 사용
}

public class Exm_02 {
4 usages
public void sayNick(String nick)throws FoolExcetion{
    try {
        if ("바보".equals(nick)) {
            throw new FoolExcetion();
        }
        System.out.println(nick);
    }catch (FoolExcetion e){
        System.out.println("FoolExcetion 발생");
    }
}

public static void main (String[]args){
    Exm_02 exm_02 = new Exm_02();
    exm_02.sayNick("바드아");           바드아
    exm_02.sayNick("바보");             FoolExcetion 발생
    exm_02.sayNick("야호");             야호
    exm_02.sayNick("바보");             FoolExcetion 발생
}
```

throw

throw

```
public class Exm_02 {
```

4 usages

```
public void sayNick(String nick){  
    try {  
        if ("바보".equals(nick)) {  
            throw new FoolExcetion();  
        }  
        System.out.println(nick);  
    }catch (FoolExcetion e){  
        System.out.println("FoolExcetion 발생");  
    }  
}  
  
public static void main (String[]args){  
    Exm_02 exm_02 = new Exm_02();  
    exm_02.sayNick("바드아");    바드아  
    exm_02.sayNick("바보");    에러 발생  
    exm_02.sayNick("야호");    야호  
    exm_02.sayNick("바보");    에러 발생  
}
```

throws

throws

```
public class Exm_02 {
```

4 usages

```
public void sayNick(String nick) throws FoolExcetion{
```

```
    if ("바보".equals(nick)) {  
        throw new FoolExcetion();  
    }  
    System.out.println(nick);  
}
```

```
public static void main (String[]args) {  
    Exm_02 exm_02 = new Exm_02();  
    try {  
        exm_02.sayNick("바드아");           바드아  
        exm_02.sayNick("바보");             예외발생  
        exm_02.sayNick("야호");  
        exm_02.sayNick("바보");  
    } catch (FoolExcetion e) {  
        System.out.println("예외발생");  
    }  
}
```

try/catch/finally 활용

```

1 package FlieIo.JavaExm12_26_01;
2
3 import java.util.Scanner;
4
5 public class Exm_03 {
6     public static void main(String[] args) {
7
8         Scanner scanner = new Scanner(System.in);
9         System.out.println("두 정수 입력");
10        try{
11
12            System.out.println("나누는 수 : ");
13            int dividend = scanner.nextInt();
14            System.out.println("나누어 지는 수 : ");
15            int divisor = scanner.nextInt();
16            int result=dividend/divisor;
17            System.out.println("결과 : " + result);
18
19        }catch(ArithmeticException e){
20            System.out.println("0으로 나누기 못함");
21        }catch(Exception e){
22            System.out.println("정수를 입력하세요");
23        } finally {
24            scanner.close();
25        }
26        System.out.println("정상 종료");
27    }
28
29    no usages
30    public static int divie(int diviend, int divisor){
31        return diviend / divisor;
32    }
33 }

```

ex1

두 정수 입력
나누는 수 :
10
나누어 지는 수 :
2
결과 : 5
정상 종료

ex2

두 정수 입력
나누는 수 :
10
나누어 지는 수 :
0
0으로 나누기 못함
정상 종료

ex3

두 정수 입력
나누는 수 :
qa
정수를 입력하세요
정상 종료

성적관리 프로그램 [이중 배열 사용]

성적관리 프로그램

```

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    // 국, 영, 수, 과, 사
    final int STUDENT_COUNT = 5;
    final int SUBJECT_COUNT = 5;
    String[] st = new String[]{"주희", "희진", "혁일", "민선", "상빈"};
    int no;

    int[][] score = new int[STUDENT_COUNT][SUBJECT_COUNT];
    for (int i = 0; i < STUDENT_COUNT; i++) {

        System.out.println("학번을 입력하십시오");
        no = sc.nextInt();
        if (no > STUDENT_COUNT) {
            System.out.println("잘못 쳤으면 다시 해이지??");
            return;
        }

        System.out.println("\n" + "학생 " + (no) + st[no-1] + "의 정보");

        System.out.println("점수 기입");

        System.out.println("국어");
        score[no-1][0] = sc.nextInt();
        System.out.println("영어");
        score[no-1][1] = sc.nextInt();
        System.out.println("수학");
        score[no-1][2] = sc.nextInt();
        System.out.println("과학");
        score[no-1][3] = sc.nextInt();
        System.out.println("사회");
        score[no-1][4] = sc.nextInt();
    }

    sc.close();
    System.out.println("\n번호 이름 국어 영어 수학 과학 사회 총점 평균");
    for (int i = 0; i < STUDENT_COUNT; i++) {
        int total = 0;
        no = i + 1;
        for (int j = 0; j < SUBJECT_COUNT; j++) {
            total += score[i][j];
        }
        int avg = total / SUBJECT_COUNT;
        System.out.printf("%-3d %-4s %-4d %-4d %-4d %-4d %-4d %-4d %-4d \n",
            no, st[i], score[i][0], score[i][1], score[i][2], score[i][3], score[i][4], total, avg);
    }
}

```

번호	이름	국어	영어	수학	과학	사회	총점	평균
1	주희	99	99	99	99	99	495	99
2	희진	99	299	99	99	99	695	139
3	혁일	99	99	99	99	99	495	99
4	민선	99	99	99	99	99	495	99
5	상빈	99	99	99	99	99	495	99