14.Exception

1)try

Try -> 에러처리나면 " ~~ " 이런방식으로 해결하세요!

Thorow -> 에러처리나면 문제가 되는 부분을 자신을 호출한 부분으로 던지세요

예외가 발생할수도 있고 아닐수도 있음 예외가 만약 발생하게 된다면 예외 객체가 생성됨

(내가 new 해서 만들지 않아도 자동 생성)

그 객체는 메시지와 private한 데이터 + 메소드들을 품고 있으며 에러가 나면 그 메시지들을 뿌려줌

*에러 발생부분을 try{} 안에 넣어줌 / catch()안에 객체 이름과 변수 설정

Try{

예외 발생 내용

}catch(예외 객체 이름 변수){

예외발생시 우회적으로 갈 로직

Sysout.----------

}

```
try {
    System.out.print("사칙연산할 첫번째 수는 ? ");
    i = scanner.nextInt();
    break;
}catch(InputMismatchException e) {
    System.out.println("예외 메세지 " + e.getMessage());
    System.out.println("정수를 반드시 입력하세요");
    scanner.nextLine();
}
```

에러나면 catch절 수행 안나면 try절 수행

Try절 안에 try를 넣지 않음 / catch절을 여러 개 생성하는 방식

exception하나로 처리해도 되지만, 각 에러마다 다른 메시지를 부리고 싶을땐 각각 만들어줘야됨

2)trows

throwsEx class

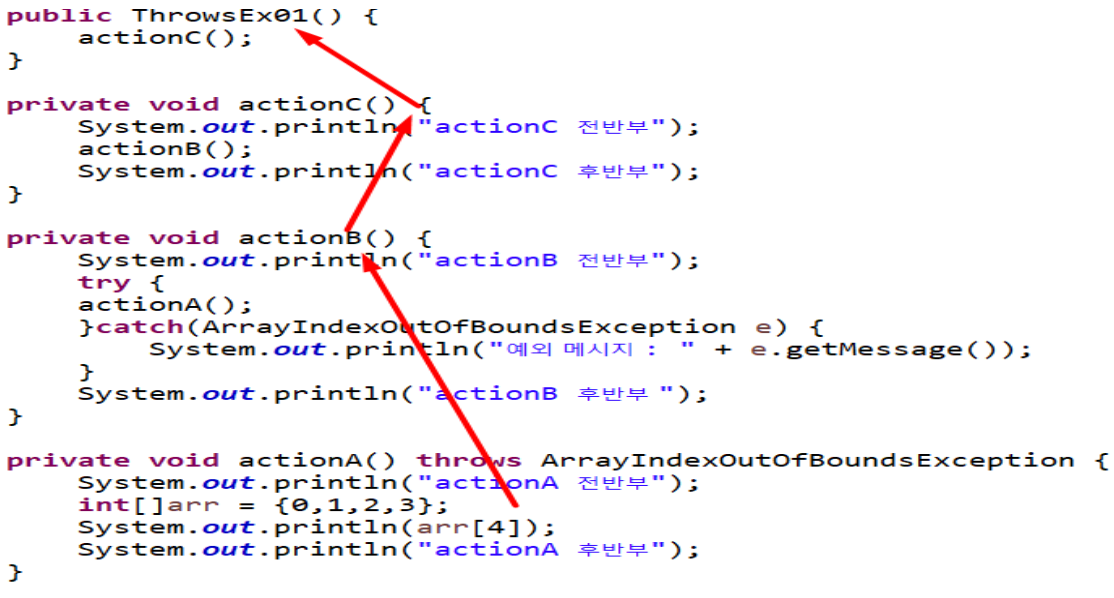
생성자 함수 -> action();

Action C

Action B

Action A

Action A에서 에러가 나면 다시 자기를 호출한 B로 돌아감 최종적으로는 객체의 에러메세지가 뿌려짐



```
public ThrowsEx01() {
    actionC();
}

private void actionC() {
    System.out.println("actionC 전반부");
    actionB();
    System.out.println("actionC 후반부");
}

private void actionB() {
    System.out.println("actionB 전반부");
    try {
    actionA();
    }catch(ArrayIndexOutOfBoundsException e) {
        System.out.println("예외 메시지 : " + e.getMessage());
    }
    System.out.println("actionB 후반부 ");
}

private void actionA() throws ArrayIndexOutOfBoundsException {
    System.out.println("actionA 전반부");
    int[]arr = {0,1,2,3};
    System.out.println(arr[4]);
    System.out.println("actionA 후반부");
}
```