예외처리 : 에러에는 컴파일 에러와 실행 에러가 있는데 컴파일 에러는 문법적인 오류를 범한 경우에 해당하고 실행에러는 문법적으로 오류는 없는데 실행했을 때 문제가 발생하는 경우에 해당한다.

1. 컴파일에러 : 문법적 오류를 범한 경우이고 이는 사용자가 직접 고쳐야 한다.
2. 실행에러 : 배열의 공간을 2밖에 안만들었는데 3번째를 index한 이후 삽입한 경우처럼 문법은 맞는데 실행했을 때 오류가 생기는 경우로 try ~catch로 직접 처리하거나 throws를 이용해 넘겨줄 수 있다.
3. Try catch : try에 오류가 발생할 만한 것을 넣는다. catch에는 오류가 발생한 경우 행할 행동을 입력한다. e.printStackTrace(); 이걸 입력하면 e에 들어가 있는 어떤 오류가 발생했는지에 대한 메시지가 출력된다.
4. Throws

public void array() throws Exception{

int arr[] = {1,2};

arr[2] = 3;

}

* 이는 array를 부른 main으로 에러를 떠넘기게 된다. 따라서 원래는 main에 빨간 밑줄이 없었는데 생기게 되고 이를 main에서 해결해 줘야 한다. Exception이 모든 에러를 다 해결해주는 거에 해당한다.

<https://mvnrepository.com/artifact/org.jsoup/jsoup/1.12.1> 여기 들어가서 크롤링하는데 사용하는 jsoup 받을 수 있고 이후에 project에 추가하면 사용할 수 있다.

크롬으로 사이트 들어가서 f12를 누르면 각 항목을 어찌 표시해놨는지 확인이 가능한데 그 이름을 jsoup을 이용해 가져온 뒤 element에 저장된 것들 중 text에 해당하는 것만 뽑아오면 크롤링이 가능하다.

Vector list = new Vector(); 얘를 입력하면 컬렉션을 사용할 수 있는데 컬렉션은 배열과 비슷한데 여러 타입의 데이터를 크기에 상관 없이 유연하게 넣을 수 있다. 개수의 제한과 타입의 제한을 두지 않고 자유롭게 섞어서 넣을 수 있기에 편리하다. 넣을 때는 list.add( ) 를 이용해 넣는다.

ArrayList list = new ArrayList(); 얘를 입력하면 위에 컬렉션과 비슷한 기능을 가지지만 차이는 얘는 순서를 중요시 여기는 곳에 주로 사용한다.

* Collection(모음)

많은 양의 데이터를 다룰 때 사용하는 자료구조로 배열의 단점을 극복하였다. 대표적 단점인 같은 타입만 가능하다는 것과 크기 조절 불가 라는 단점을 극복하였다.

대표적 종류로 list타입, set타입, Queue 타입, map 타입이 있고 순서가 중요하면 list 타입을 사용하면 되는데 그 중 ArrayList 많이 사용한다.

set타입은 중복은 싫고 순서는 상관없다. 대표적인 것으로 hashset이 있다. 얘의 장점은 중복 걱정필요 없이 다 넣으면 알아서 중복은 넣지 않아 준다.

Queue 타입은 remove로 제거를 하게 되면 제일 먼저 들어오는 얘부터 remove 되는 특징을 가지고 있다. Queue 타입은 대표적인 것으로 LinkedList( )가 있다.

Map 타입은 애플은 사과 처럼 짝을 지어 데이터를 저장할 때 사용한다.

상속은 위에서 정의한 class를 이어받아 이어 받은 class가 가지고 있던 변수와 메소드를 가지는 것이다. 따라서 기존에 변수 한 개와 메소드 한 개인 class를 이어 받았다면 이후에 변수 한 개와 메소드 한 개를 새롭게 선언했을 때 이어 받은 class가 가진 변수는 2개 메소드는 2개가 되는 것이다. 이후 main에서 상속 받은 얘를 이용해 자유롭게 변수와 메소드 사용 가능하다.

Jframe 사용할 때도 얘를 우리가 편한 이름으로 상속 받은 뒤 그 이름으로 모든 기능에 접근해 사용 가능하다.