**Vector 클래스**

Vector() 10개의 데이터를 저장할 수 있는 길이의 객체 생성 저장 공간이 부족할 경우 10개씩 증가

Vector(int size) Vector(int size,int incr) : size개의 데이터 저장할 수있는 길이의 객체 생성, 저장

공간 부족한 경우 incr개씩 증가

**벡터의 용량과 개수**

* 벡터의 크기: 벡터에 들어 있는 요소의 개수
  + ex) int n=v.size();
* 백터의 용량 :요소를 수용할 수있는 크기(벡터는 자동으로 조절 가능)
  + ex)int c=v.capacity();

Vector 클래스 Vector는 서로 다른 타입을 가지는 참조형 데이터를 저장하는 가변 길이의 배열 기본형 데이터 저장할 수 없음

기본형 데이터 저장하기 위해서는 기본형 데이터를 Wrapper 클래스 타입의 객체로 생성해야 함

Vector 클래스

ArrayList클래스와 같이컬렉션 프레임웍에서 가장 많이 사용되는 컬렉션 클래스

List인터페이스 구현하기 때문에 List인터페이스 특징 가짐

1. 저장순서 유지

2. 중복 허용

ex) v.add(1,100); 인덱스 1위치에 100 요소 삽입

새로운 요소 삽입 후에 나머지 요소들 한칸씩 뒤로 이동함

String nextLine(); // '\n'을 포함하는 한 라인을 읽고 '\n'을 버린 나머지만 리턴

void close(); Scanner의 사용 종료

boolean hasNExt(); 현재 입력된 토큰이 있으면 true, 아니면 새로운 입력이 들어올 때까지 무한정 기다려서

새로운 입력이 들어오면 그 때 true 리턴. ctrl+z가 입력되면 끝이므로 false 리턴

**Vector 내의 Element 알아내기**

1. get()메서드

2. elementAt()메서드

Vector에서 요소 Remove

v.remove(1);// 벡터v가 가리키고 있는 1번째 인덱스의 값을 삭제

v.removeAllElements(); 벡터의 모든 요소 삭제

**Iterator**

public boolean hasNext(); // 다음 요소가 있는 지 판별

public Object next(); //다음으로 이동하는 next 메서드

public void remove(); // 읽어 온 요소를 삭제하는 remove 메서드

iterator 개체는 컬렉션 개체의 iterator 메서드를 호출하여 얻어올 수 있음 그리고 hasNext메서드로 이동이 가능한지 확인한

후에 next메서드로 해당 위치에 보관한 개체를 참조하여 원하는 작업 수행

<http://ehpub.co.kr/tag/hasnext-%EB%A9%94%EC%84%9C%EB%93%9C/>

<http://tcpschool.com/java/java_collectionFramework_iterator>

<https://wikidocs.net/214>

**instanceof**

참조변수가 참조하고 있는 인스턴스의 실제 타입 알아보기 위해 사용

연산 결과로 boolean값 반환

ex) true->참조변수가 검사한 타입으로 형변환이 가능함

**배열에 이용되는 for문**

for(변수:배열)

ex) int arr[]={1,2,3,4,5}; for(int num: arr) System.out.println(num);

arr배열에 들어있는 값들을 하나씩 num변수에 대입시킴

배열의 자료형과 for문 변수의 자료형은 같아야 함

* <https://tbbrother.tistory.com/53>
  + 문제 2839