-자료형 클래스 : 여러 개의 데이터를 담아놓고 사용(처리)하기 위해 사용

-제네릭을 사용하지 않을 경우 object가 됨 -> 기능을 제대로 쓰려면 클래스의 형변환이 필요.

-Map : 키와 값으로 이루어진 자료 보관 형태. 여러 가지 Map이 있지만, 보통 HashMap, TreemMap을 사용함. 우리나라에서는 HashMap을 많이 사용한다. 배열과 다른점은 순차적이지 않으며, 주소를 통해 값을 관리함(해시코드를 기반으로 관리함)

-HashMap 생성  
 HashMap<클래스명, 클래스명> 변수명 = new HashMap<클래스명, 클래스명>();  
 ex) HashMap<String, String> map = new HashMap<String, String>();

|  |  |  |
| --- | --- | --- |
| map | key | value |
|  | “a” | “abc” |
|  | “가” | “가나다” |

<사람이 인식하고 있는 형태>

|  |  |  |
| --- | --- | --- |
| map | key | value |
|  | “a” | “abc” 객체의 주소 |
|  |  | ” |

<실제 형태>  
  
들어갈 데이터 형태가 정해져있으며, 키를 통해 값을 확인함.   
  
1) put(키, 값) : map에 해당 데이터를 추가한다. 킷값이 이미 있다면 덮어씌워짐(굳이 replace 안쓰는 이유)  
 map.put("1","가나다");  
 map.put("2","라마바");  
 map.put("3","사아자");  
2) get(키) : 키에 해당하는 값을 돌려준다.  
 System.out.println(map.get("3")); -> “사아자” 출력  
3) remove(키) : 키에 해당하는 내용을 제거한다.  
 map.remove("1")  
 System.out.println(map.get("1")); -> “null” 출력. 값만 없을 뿐 공간 존재  
4) cantainsKey(값), containsValue(값) : 값에 해당하는 키나 값이 있으면 boolean으로 제공  
 System.out.println(map.containsKey("1")); -> “false” 출력  
 System.out.println(map.containsValue("사아자")); “true” 출력

-Set : 단순히 값을 담아놓는 역할을 함(추가적으로 값 취득 못함). 배열이나 Iterator로 변환 가능.  
 hasNext() : 다음에 값이 있는지 확인  
 next() : 다음으로 이동하며, 다음 값을 돌려준다.

-Iterator : 값을 순차적으로 보관. 값 추가 불가능. 단지 가져오는 기능밖에 없음. 취득도 순차적으로 되며 뒤로는 못감. 레코드판 생각할 것.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 크기 | 값 추가 | 값 취득 | 순차적 |
| Array | 고정 | 크기  안에서는 가능 | 가능 | O |
| List | 유동적 | 가능 | 가능 | O |
| Map | 유동적 | 가능 | 가능 | X |
| Set | 유동적 | 가능 | 불가능 | O |
| Iterator | 고정 | 불가능 | 가능 | O |

Set과 Iterator는 Array, List, Map 일종의 하위호환(?)이므로 잘 쓰이지 않음.  
값이 순차적을 때는 Array, List를 사용하는데, 크기가 유동적이면 List를 씀.

- List<HashMap<클래스, 클래스>> list = new ArrayList<HashMap<클래스, 클래스>>(); 형태로 사용하거나 HashMap만 단독으로 사용하는 경우가 많음.  
 <List 구조>

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | map 1 주소 | |  |  | | --- | --- | | key | value | |  |  | |  |  |   map1 객체 |
| 1 | map 2 주소 | |  |  | | --- | --- | | key | value | |  |  | |  |  |   map2 객체 |
| 2 | map 3 주소 | |  |  | | --- | --- | | key | value | |  |  | |  |  |   Map3 객체 |
| 3 |  |  |
| 4 |  |  |

\*편의상 map에 숫자 붙임

-객체의 주소 연결/분리  
