

Collection Framework

🕒 작성일시	@2022년 7월 26일 오전 10:57
📄 강의 번호	JAVA
📄 유형	
📎 자료	
☑ 복습	<input type="checkbox"/>
≡ 학습 소스 출처 1	
≡ 학습 소스 출처 2	
📅 날짜	

Collection

add 가 가장 중요.

Collection s<T> : add(T t);

set<interface> : 중복 x, 순서 x

List : 순서 O(인덱스 관리 존재) , 중복 O

Data : 나중에 쓸 데이터

저장방식 : Data를 찾기 편하게 저장해야함.

선형데이터 : 하나의 데이터를 기준으로 앞뒤로 하나이하씩 있는것.

ex) Array , stack , queue ,list(array, pointer)

순회방식 : loop

비선형데이터 :

ex) tree (1:N) , graph (N:M)

리스트

pointer 리스트

추가 수정 삭제 → 빠름

조회 → 느림

배열 리스트

추가 수정 삭제 → 느림

조회 → 빠름

—remove() 할 시, 만약 뒤에 값이 더있다면 끌어옴

```
//ex
List<Integer> list=new ArrayList<>();
// 1 2 3 형태의 리스트
list.add(1);
list.add(2);
list.add(3);
list.remove(0); //0번째 원소 삭제
//결과는 null 2 3 이 아니라 2 3 의 형태가 된다
인덱스      0   1 2           0 1
```

Set

업데이트가 없다

→ 업데이트는 수정할 위치의 값을 삭제하고 그 위치에 수정할값을 추가하는 과정

셋은 순서가 없기 때문에 굳이 업데이트의 필요성이 없다.

Map

key 와 value 형태로 저장

가져올땐 .get() 을 이용하면 key,value 형태로 반환