



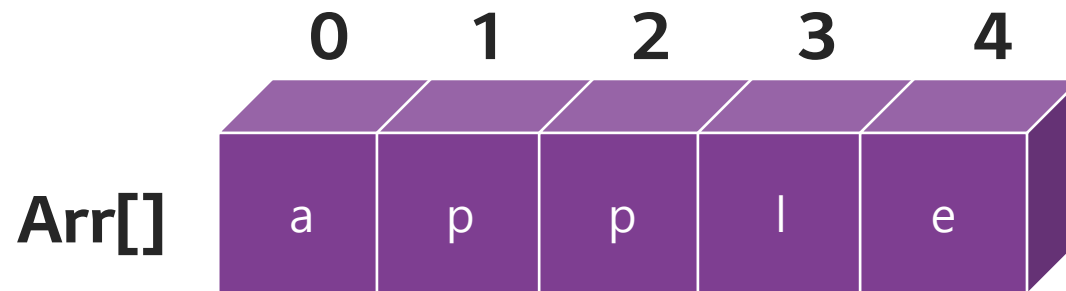
박수현 연구원

수업 진행방향



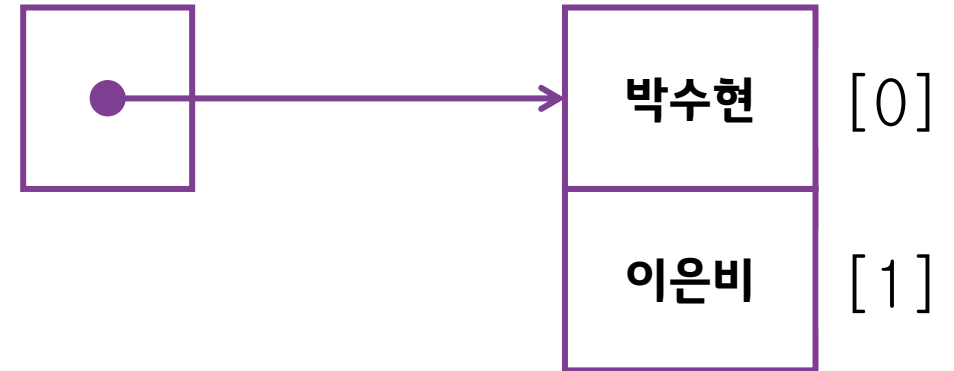
배열(Array)

같은 타입의 데이터들을 하나로 묶어 다루는 구조



1. 배열 선언 시, 고정된 크기로 선언

```
String[] nameArr = new String[2];  
nameArr[0] = "박수현";  
nameArr[1] = "이은비";
```



2. 배열의 크기를 바꿀 수 없다.

```
String[] nameArr = new String[2];  
  
nameArr[0] = "박수현";  
nameArr[1] = "이은비";  
nameArr[2] = "최성우";  
  
System.out.println(nameArr[2]);
```

Exception in thread "main" [java.lang.ArrayIndexOutOfBoundsException](#)
: [Index 2 out of bounds for length 2](#) at test2.main([test2.java:12](#))

ArrayList

- 가장 보편적으로 많이 사용하는 컬렉션 클래스로 객체를 저장
- 크기가 고정인 아닌 가변의 길이
- 원하는 위치의 추가나 삭제가 쉬움



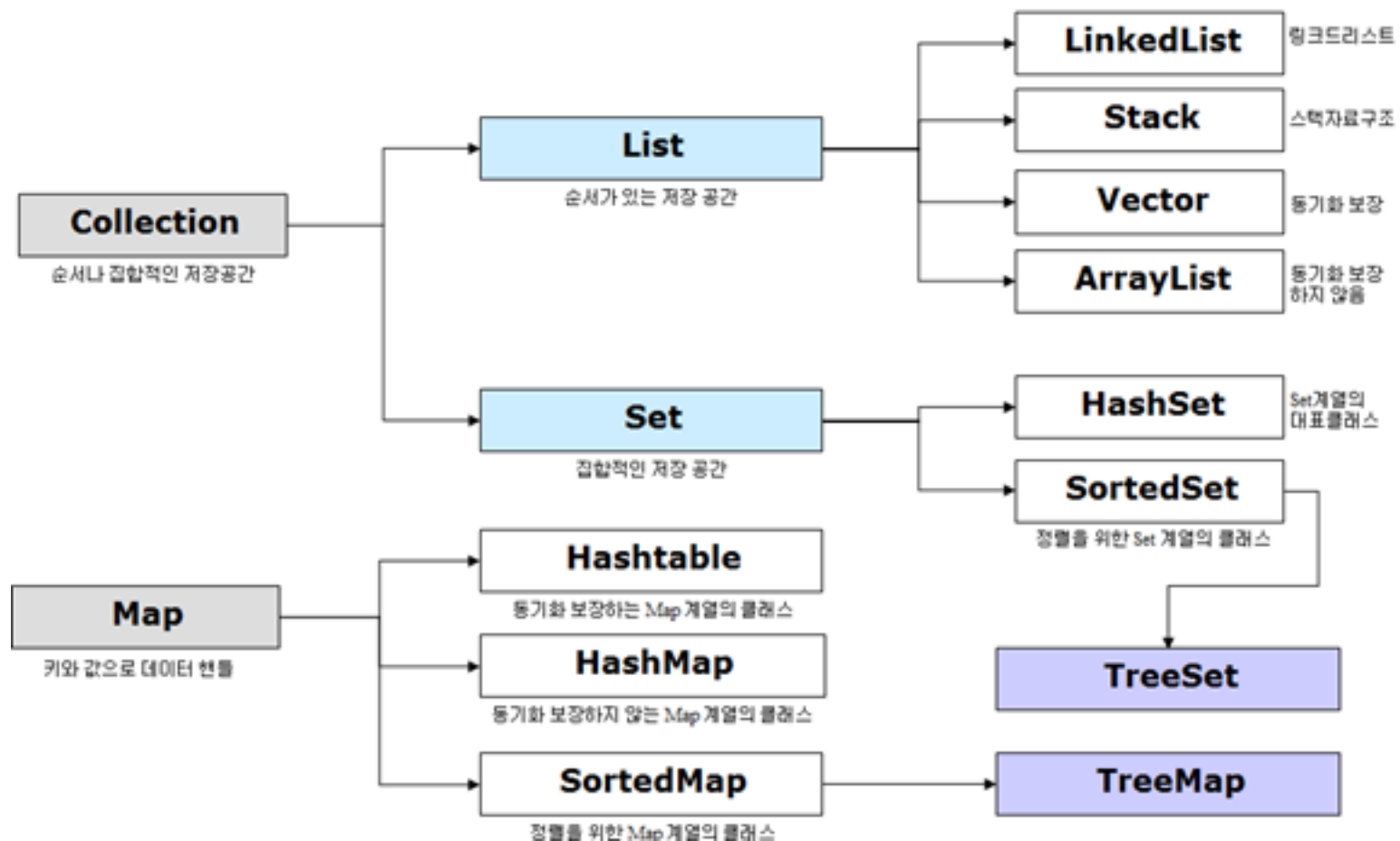
Collection Framework

: 다수의 데이터를 쉽고 효과적으로 처리할 수 있는 표준화된 방법을 제공하는 클래스의 집합

- 데이터

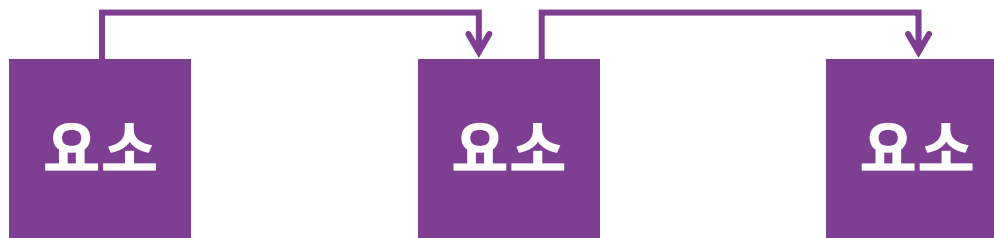
1. 객체들의 컨테이너라고도 불림
2. 요소의 개수에 따라 **자동 크기 조절**
3. 요소의 추가,삭제,에 따른 **요소의 이동 자동 관리**

- 여러 개의 객체를 보관할 수 있게 만들어진 클래스들의 집합
- 고정 크기의 배열을 다루는 불편함 해소

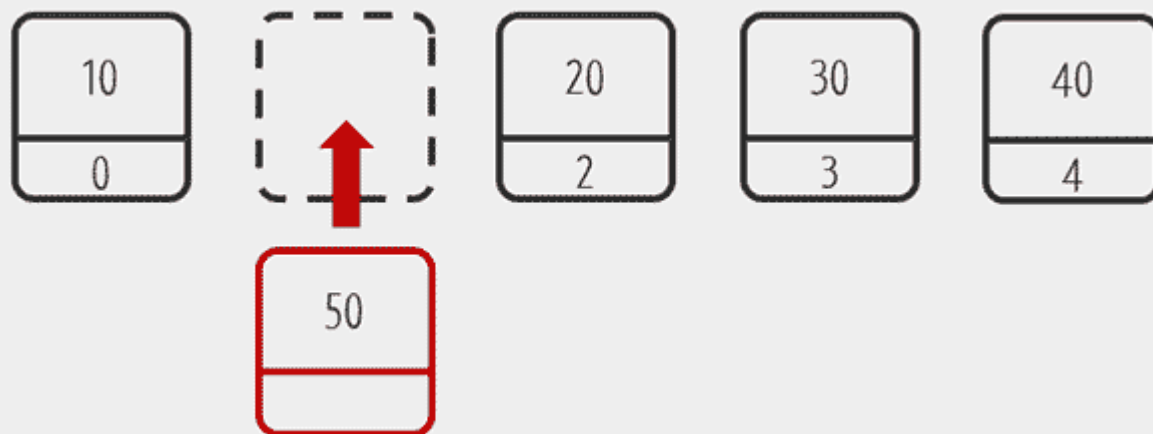


ArrayList

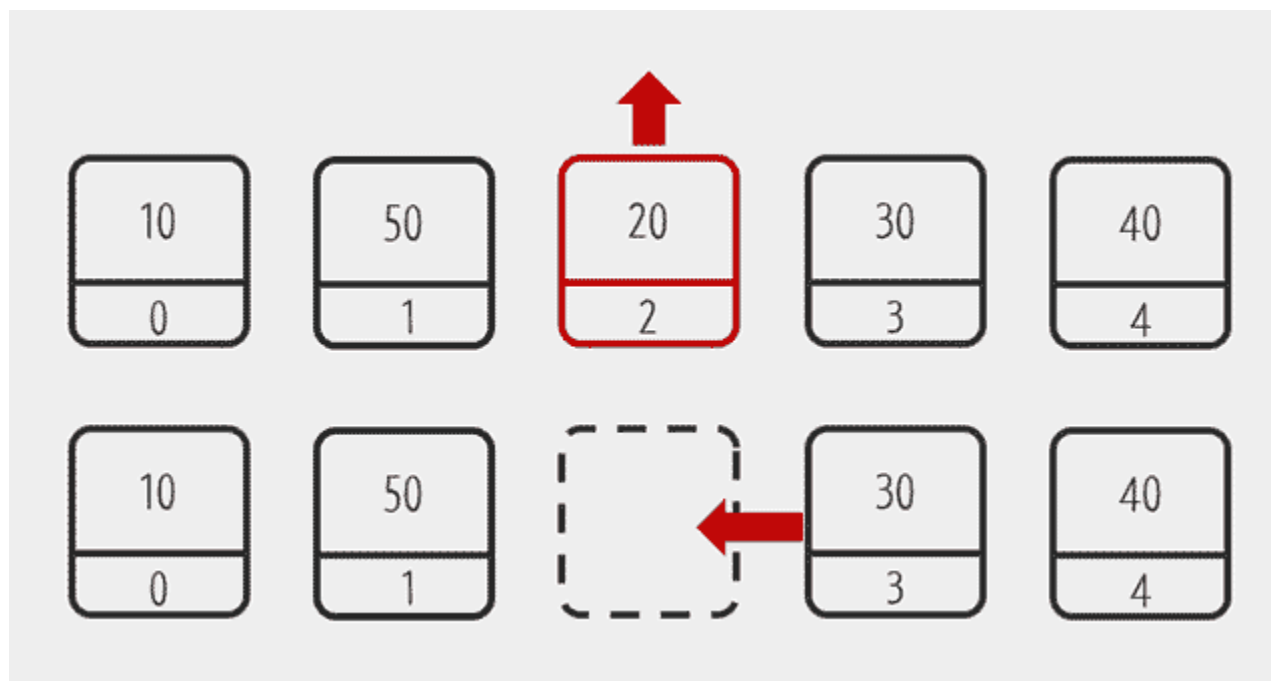
- 가장 보편적으로 많이 사용하는 컬렉션 클래스로 객체를 저장
- 크기가 고정인 아닌 가변의 길이
- 원하는 위치의 추가나 삭제가 쉬움



추가



삭제



각 팀원의 이름을 입력 받아서 ArrayList에 저장후 출력해보세요!

이름을 입력하세요 : 황해도

이름을 입력하세요 : 박수현

이름을 입력하세요 : 임승환

이름을 입력하세요 : 강예진

이름을 입력하세요 : 임명진

연구개발팀의 팀원은 황해도 박수현 임승환 강예진 임명진 입니다.

MusicPlayList클래스는 musiclist라는 이름으로 ArrayList객체를 생성하여 musiclist에 노래제목을 추가하고 삭제하는 프로그램 입니다.

MusicPlayList클래스는 아래와 같은 메뉴로 구성되어 있습니다.

```
====Music Play List====
```

```
[1] 노래 추가 [2]| 노래 조회 [3] 노래 삭제 [4] 종료 >>
```

[2] 노래 조회 선택 시,
재생 목록이 없다면 “재생 목록이 없습니다”를 출력하세요

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 2
===== 현재 재생 목록 =====
재생 목록이 없습니다!
```

[1] 노래추가 선택 시,
원하는 위치 혹은 마지막 위치에 노래를 추가할 것 인지 입력 받으세요.

```
====Music Play List====
```

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 1
```

```
=====
```

```
[1] 원하는 위치에 추가 [2] 마지막 위치에 추가 >>
```

[2] 마지막 위치에 추가 선택 시
추가 할 노래를 입력 받아서 musicList에 추가하세요.
추가가 완료 된 뒤 “추가가 완료되었습니다” 라는 문장을 출력하세요.

```
====Music Play List====
```

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 1
```

```
=====
```

```
[1] 원하는 위치에 추가 [2] 마지막 위치에 추가 >> 2
```

```
추가 할 노래 입력 >> 신호등
```

```
추가가 완료 되었습니다.
```


[1] 원하는 위치에 추가 선택 시
추가 할 위치와 노래를 입력 받아서 musicList에 추가하세요.
추가가 완료 된 뒤 “추가가 완료되었습니다” 라는 문장을 출력하세요.

===== 현재 재생 목록 =====

1. Butter
2. 신호등



===== 현재 재생 목록 =====

1. 롤린(Rollin')
2. Butter
3. 신호등

[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 1

=====

[1] 원하는 위치에 추가 [2] 마지막 위치에 추가 >> 1

추가 할 위치 입력 >> 0

추가 할 노래 입력 >> 롤린(Rollin')

추가가 완료 되었습니다.

[2] 노래조회 선택 시 현재 재생목록을 보여주세요

```
===== 현재 재생 목록 =====  
1. 롤린(Rollin')  
2. Butter  
3. 신호등
```

[3] 노래 삭제 선택 시 선택 삭제 할 지 노래 삭제 할 지 입력 받으세요

[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 3


=====

[1] 선택 삭제 [2] 전체 삭제 >> _____

[1] 선택 삭제 시, 삭제 할 노래를 선택하고 삭제 시 '노래가 삭제되었습니다' 라고 출력하세요

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 3
=====
[1] 선택 삭제 [2] 전체 삭제 >> 1
삭제 할 노래 선택 >> 2
노래가 삭제 되었습니다.
```

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 2
===== 현재 재생 목록 =====
1. 롤린(Rollin')
2. Butter
3. 신호등
```



```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 2
===== 현재 재생 목록 =====
1. 롤린(Rollin')
2. 신호등
```

[2] 전체 삭제 시 플레이 리스트 전체를 삭제하고, 삭제 시 '전체 리스트가 삭제되었습니다' 라고 출력하세요

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 3
```

```
=====
```

```
[1] 선택 삭제 [2] 전체 삭제 >> 2
```

```
전체 리스트가 삭제되었습니다.
```

```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 2
```

```
===== 현재 재생 목록 =====
```

```
1. 롤린(Rollin')
```

```
2. 신호등
```



```
[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 2
```

```
===== 현재 재생 목록 =====
```

```
재생 목록이 없습니다!
```

**[4] 종료 선택 시 프로그램을 종료(반복문 종료)하고
'뮤직 플레이리스트를 종료합니다' 라고 출력하세요**

[1] 노래 추가 [2] 노래 조회 [3] 노래 삭제 [4] 종료 >> 4
|뮤직 플레이리스트를 종료합니다.

다음시간에는?

클래스