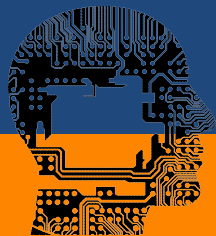




한국IT진흥부설

정보보호교육학원

아이섹



# 자바 프로그래밍 기초

## 컬렉션(Collection)

By SoonGu Hong(Kokono)

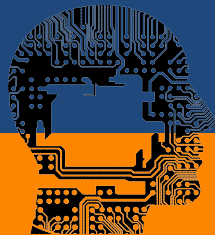


한국IT진흥부설

정보보호교육학원 아이섹

# 파이썬 기초

## 1. 컬렉션이란?

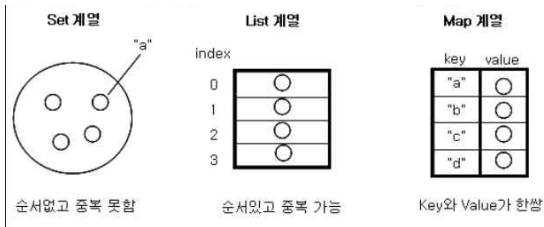


## 1-1. 컬렉션이란?

- 프로그램을 개발하다보면 다수의 객체를 저장해두고 필요할 때마다 꺼내서 사용하는 경우가 많습니다.
- 이 때 가장 먼저 생각나는 자료구조가 배열이겠죠??
- 하지만 배열은 크기가 고정되어 있어 불특정 수의 객체를 저장하는데 있어 알고리즘을 직접 만들어야 하는 불편함이 있습니다.
- 우리는 이 문제를 해결하기 위해 잘 만들어진 자료구조 API 컬렉션을 사용합니다.



## 1-2. 대표적인 컬렉션 인터페이스



- 컬렉션 API는 `java.util` 패키지로 제공됩니다.
- 여러가지 인터페이스를 지원하는데 대표적으로 List, Set, Map 인터페이스가 있습니다.
- 우리는 해당 인터페이스의 하위 구현체를 통해 다수의 객체를 저장하고 마음껏 활용할 수 있습니다.

\* 이미지 출처: <https://hanabie.tistory.com/36>

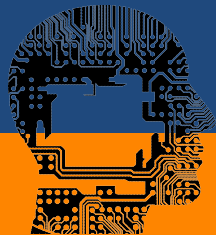


한국IT진흥부설

정보보호교육학원 아이섹

# 파이썬 기초

## 2. List 인터페이스



## 2-1. List 컬렉션

- List 컬렉션은 객체를 일렬로 늘어놓은 구조를 가지고 있습니다. 그리고 객체를 **인덱스로 관리**하기 때문에 배열과 유사한 점이 많습니다.
- 실제로 List 인터페이스를 구현한 클래스 중 ArrayList는 배열자료구조 형태를 가지고 있습니다.
- 대표적인 구현 클래스는 ArrayList, LinkedList, Vector가 있고 객체를 모음 저장할 수 있으며, 검색, 삭제, 수정기능을 사용할 수 있습니다.
- 또한 리스트는 동일한 주소를 가진 객체를 **중복저장**할 수 있습니다.

## 2-2. List 인터페이스 주요메서드

기능	메서드	설명
객체 추가	boolean add(E e)	객체를 맨 끝에 추가
	void add(int index, E e)	인덱스에 객체를 추가
	E set(int index, E e)	인덱스의 객체를 수정
객체 검색	boolean contains(Object o)	객체가 저장된 여부
	E get(int index)	저장된 객체를 반환
	boolean isEmpty()	빈 컬렉션인지 확인
	int size()	저장된 객체 수 반환
객체 삭제	void clear()	저장된 모든 객체 삭제
	E remove(int index)	인덱스의 객체 삭제
	boolean remove(Object o)	해당 객체를 삭제

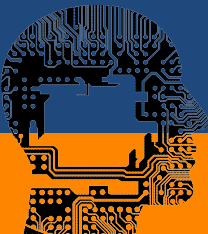


한국IT진흥부설

정보보호교육학원 아이섹

# 파이썬 기초

## 3. Set 인터페이스





### 3-1. Set 컬렉션

- Set 컬렉션은 List와 달리 **저장 순서가 유지되지 않고 중복 저장을 허용하지 않습니다.**
- Set 컬렉션은 수학의 **집합**과 유사하다고 생각할 수 있습니다. 합집합을 하면 중복 숫자가 배제되는 것 처럼말이죠.
- 또한 큰 주머니에 비유될 수 있습니다. 순서가 유지되지 않기 때문에 처음 들어간 물건이 꺼낼때 처음에 나오지 않을 수도 있겠죠?
- Set 인터페이스의 대표적인 구현 클래스는 HashSet, TreeSet, Hashtable이 있습니다.
- Set은 **인덱스가 없습니다.**

## 3-2. Set 인터페이스 주요메서드

기능	메서드	설명
객체 추가	<code>boolean add(E e)</code>	객체를 추가
객체 검색	<code>boolean contains(Object o)</code>	객체가 저장된 여부
	<code>boolean isEmpty()</code>	빈 컬렉션인지 확인
	<code>int size()</code>	저장된 객체 수 반환
객체 삭제	<code>void clear()</code>	저장된 모든 객체 삭제
	<code>boolean remove(Object o)</code>	해당 객체를 삭제

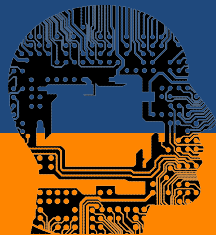


한국IT진흥부설

정보보호교육학원 아이섹

# 파이썬 기초

## 4. Map 인터페이스



## 4-1. Map 컬렉션

- Map 컬렉션은 **키(key)**와 **값(value)**으로 구성된 Entry객체를 저장하는 구조를 가지고 있습니다.
- 여기서 키와 값은 모두 객체입니다.
- **키는 중복 저장할 수 없지만 값은 중복 저장이 가능합니다.**
- 만약 동일 키로 다른 값을 저장하면 해당 키의 값이 수정됩니다.
- Map 인터페이스의 대표적인 구현 클래스로는 HashMap, TreeMap, Hashtable, Properties가 있습니다.

## 4-2. Map 인터페이스 주요메서드

기능	메서드	설명
객체 추가	V put(K key, V value)	주어진 키와 값 추가
객체 검색	boolean containsKey(Object key)	키가 저장된 여부
	V get(Object key)	키에 해당되는 값 반환
	boolean isEmpty()	빈 컬렉션인지 확인
	int size()	저장된 키의 수 반환
	Set<K> keySet	모든 키를 Set으로 반환
	Collection<V> values()	모든 값을 반환
객체 삭제	void clear()	저장된 모든 객체 삭제
	V remove(Object key)	키와 일치하는 Entry삭제

**감사합니다**  
**THANK YOU**