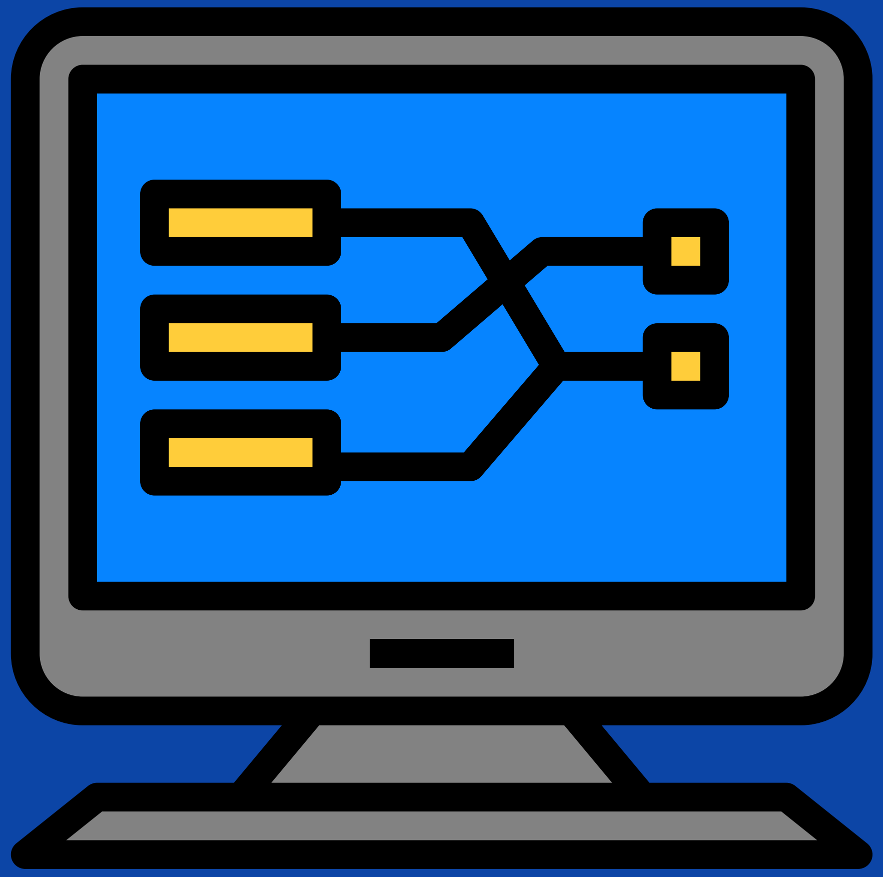
Java Collection

Set

Techtorial

SET

- SET IS UNORDERED COLLECTION
- SET CONTAINS UNIQUE ELEMENTS ONLY



HASHSET

1. HASHSET DOESN'T MAINTAIN THE INSERTION ORDER
2. HASHSET STORES THE ELEMENTS USING HASHING
3. HASHSET ALLOWS NULL VALUE
4. HASHSET CLASS IS NOT SYNCHRONIZED

Techtorial

HASHSET METHODS

1. ADD();
2. CONTAINS();
3. SIZE();
4. REMOVE();
5. CLEAR();
6. ISEMPTY();

Techtorial

CONVERTING FROM ARRAYLIST TO HASHSET

While we are creating the set we can assign the ArrayList to set.

```
List<String> computers=new ArrayList();
HashSet<String> computerNames=new HashSet(computers);
```

CONVERTING FROM HASHSET TO ARRAYLIST

Java allows us to convert Set to ArrayList. We need to assign HashSet object to ArrayList.

```
HashSet<String> students=new HashSet();
List<String> allStudents=new ArrayList(students);
```

LINKEDHASHSET

LinkedHashSet contains unique elements only.

It provides all optional set operations(methods) and permits null elements

LinkedHashSet is also non synchronized.

Main difference is LinkedHashSet maintains insertion order

HOW TO CREATE LINKEDHASHSET?

LinkedHashSet is class in java. We can use the Set interface to declare HashSet or we can declare using LinkedHashSet.

1- Set <String> set=new LinkedHashSet();
2- LinkedHashSet<String> linkedHashSet=new LinkedHashSet();

Techtorial

TREESET

TreeSet contains unique elements only

Accessing to the treeset elements is very fast

TreeSet doesn't not allow null element

TreeSet maintains ascending order.

TreeSet can use all the methods which set maintains.

Techtorial

TREESET METHODS

first() : This method take **first** element from TreeSet.

last() : This method take **last** element from TreeSet.

pollFirst(): This method take **first** element from TreeSet and then **remove** the element.

pollLast(): This method take **Last** element from TreeSet and then **remove** the element.

descendingSet(): This method reverse the order in TreeSet but it will not affect original TreeSet.

Techtorial

MAP

01

MAP

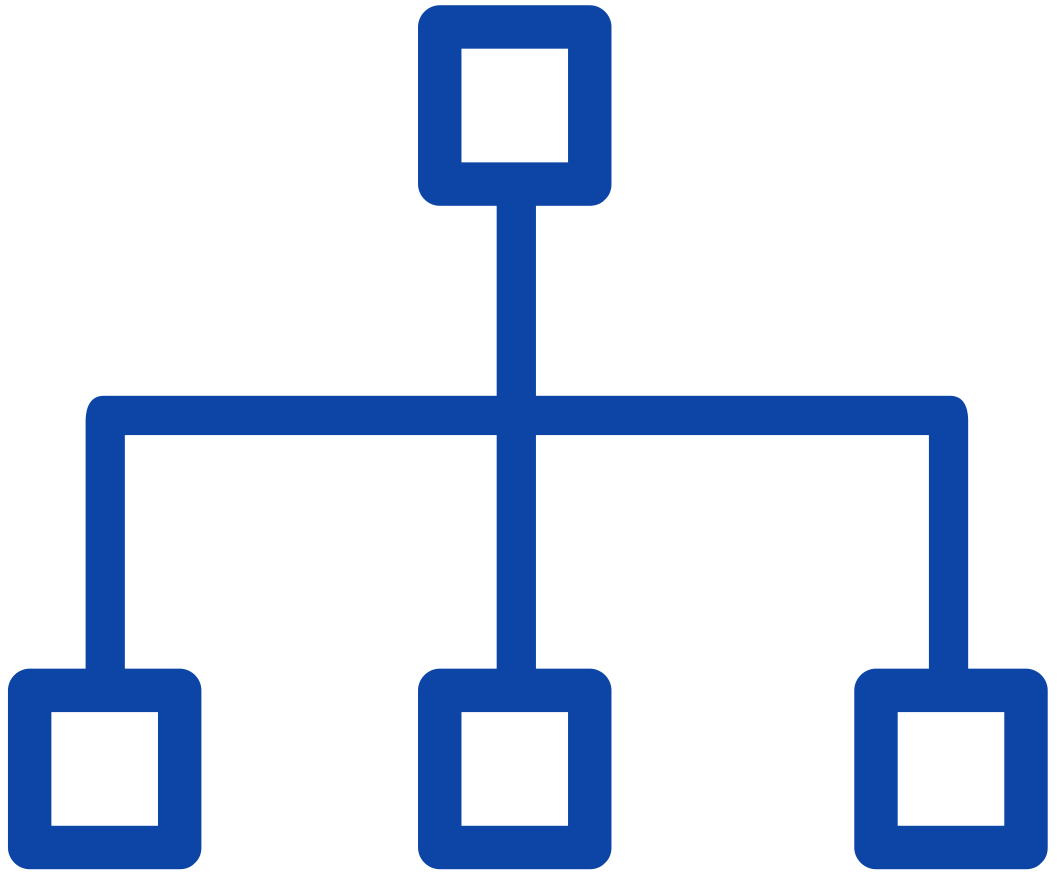
Map Collection

MAP IS WORKING WITH KEY AND VALUE SYSTEM

MAP DOESN'T ALLOW DUPLICATE KEYS BUT CAN HAVE DUPLICATE VALUES

MAP CAN'T BE ITERATED, that's why we need to get all the keys as set collection from the map.

HASHMAP



1-) HASHMAP CONTAINS VALUES BASED ON THE KEY.
2-) HASHMAP CONTAINS ONLY UNIQUE KEYS.
3-) HASHMAP CLASS HAVE ONE NULL KEY AND MULTIPLE NULL VALUES.
4-) HASHMAP IS NOT SYNCHRONIZED.
5-) HASHMAP MAINTAINS NO ORDER.

MAP METHODS

Put(Object key, Object Value)

THIS METHOD TAKES TWO PARAMETER AND INSERT ENTRY IN TO THE MAP.

PutAll(Map)

This method takes one parameter as value of Map and it will insert the specified map in to the map.

Get(Object key) Method

This method will take one parameter as a key and return the value matching with the key.

Remove(Object key) method

It will remove the specified value with related keys from the map. It will take one parameter as a key.

Keyset() method

It return the Set object which contains all the keys in the map.

Keyset() method

TO USE THE KEYSET FIRST WE NEED TO CONVERT SET. AS A RESULT, IT WILL RETURN THE SET OF KEYS. AFTER CONVERTING TO THE SET, WE NEED TO USE FOREACH LOOP TO PRINT THE KEY ON THE MAP. THEN USING LOOP WE CAN PRINT OUT ALL THE VALUES OF THE MAP.

Replace(key , value)

It will replace the specified value with the new value for a specified key.

ContainsValue(Object value) Method

IT WILL TAKE ONE PARAMETER AS VALUE AND RETURN TRUE IF THE VALUE IS EXIST

ContainsKey(Object key) Method

IT WILL TAKE ONE PARAMETER AS KEY AND RETURN TRUE IF THE KEY IS EXIST

VALUES() METHOD

IT RETUNS A COLLECTION VIEW OF THE VALUES CONTAINED IN THE MAP.

ENTRYSET() METHOD

IT WILL RETURN THE SET CONTAINING ALL THE KEYS AND VALUES

Clear() - Size() - isEmpty()

Techtorial

LinkedHashMap

Techtorial

LINKEDHASHMAP

- LinkedHashMap is working with key and values.
- LinkedHashMap contains unique elements.
- LinkedHashMap may have one null key and multiple null values.
- LinkedHashMap is non synchronized.
- LinkedHashMap maintains insertion order.

Techtorial

TREEMAP

Techtorial

TREEMAP

01 Contain values based on key

02 TreeMap contains unique elements.

03 TreeMap can not have null key but can have multiple null values.

04 TreeMap is non synchronized.

05 TreeMap maintains ascending order.

HASHTABLE

Techtorial

HASHTABLE

01 HashTable contains values based on the key.

02 HashTable contains unique elements.

03 HashTable can not have null key or value

04 HashTable is synchronized.

Techtorial

HASHMAP	HASHTABLE
01 HashMap is not synchronized. It is not thread-safe	01 HashTable is synchronized. It is thread-safe.
02 HashMap allows one null key and multiple null values.	02 HashTable doesn’t allow any null key or value
03 HashMap is fast.	03 HashTable is slow

Techtorial