08. Hash Map And Hash Set

14:08

Introduction
Basic Problems

1. HashSet

- HashSet is a collection of unique elements.
- HashSet doesn't have any sequence of elements.

Syntax :
HashSet<Type> hs = new HashSet<Type>();

Here Type can be of any class.

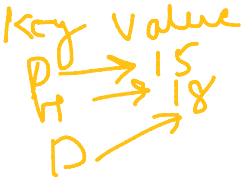
Basic Operations :

- **Add** : Used to add elements in hash set.
- **Contains** : Used to check whether hash set contains a
- **Size** : Used to get the size of hash set
- **Remove** : Used to remove the element from the hash set.

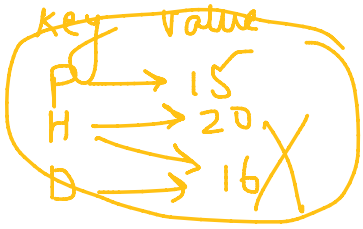
Problem 1. Given two hash sets, return the common elements in them.

2. HashMap

- Data structure that contains key value pair.
- Duplicate values are allowed.



- Duplicate keys are not allowed.



- No order of data, key-value pair are in random order.

Syntax :
HashMap<keyType, valueType> hm = new HashMap<keyType, valueType>();

Basic Operations :

- **Add** -> hm.put("Vishal", 27);
- **Contains** -> hm.containsKey("Vishal");
- **Get** -> hm.get("Vishal")
- **Update** -> hm.put("Vishal", 28);
- **Size** -> hm.size();
- **Remove** -> hm.remove("Vishal");
- **hm.containsKey(arr.get(i)**
- **Print** ->

```
// print
    // 1. get all keys
    // hm.keySet()-> returns a set of keys of HashMap
    // 2. Use keys to iterate over the map
    for(String state : hm.keySet()){
       System.out.println(state + " -> " + hm.get(state));
    }
```

Problem 1. Given an integer array as input, return the corresponding frequency map.