Hash Set & Hash Map

Amreshwar
Anil
Bhavesh Pandey
Bhushan Ganesh Shelar
Chirag
Deepshikha Arora
Dev Raj Gaur
Dhanashree Sagane
Dhasthagiri Reddy
Dhruv
Farzana kauser
Hemant Verma
Hemanth T
Induja
Iti
jeevanantham
Keshav Dutt Kaushik
M S Haseeb Khan

Mahesh
Manisha
Manohar A N
Mohona Paul
Narayan
Naveen
Nutan
Pranjul Kesharwani
Praveen Kumar
Priyank
Pushpa
Rahul
Rishav Ghosh
Sai Kiran JNR
Shaik Lal Jan Basha
Shambhavi Belligatti
Shivam Shiv
Shivanand Patil
Tanya
Ved Verma
Yoga Nand Sattvik

HashSet —— Collection of unique elements

set there y no order

A[] = [1 3 -2 7 1 1 -2]

set1 = { 3, 1, 2 }

set2 = { 1, 3, 2 }

1	-2
3	7

Real world

- username.
- emails
- id in office.

Syntax

```
HashSet <Type> hs = new HashSet<>();
```

Basic Operations

How to add elements to hs ? hs.add(element);

How to check if hs contains a certain element ?

hs.contains(element); → return true or false

How to check size of hs ? hs.size()

How to remove an element from hs ? hs.remove(elem);

How to print the contents of hs ? sop(hs);

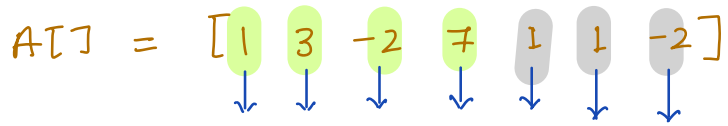
Demo

```java
public class Main {
    public static void main(String[] args) {
        HashSet<Integer> hs = new HashSet<>();
        // HashSet<int> hs = new HashSet<>(); Do not do this
        // Type argument cannot be of primitive type

        // hs doesn't contain duplicate values
        hs.add(100);
        hs.add(2);
        hs.add(4);
        hs.add(100);

        System.out.println("HashSet contains 100 or not: " + hs.contains(100));
        hs.remove( o: 100);
        System.out.println("HashSet contains 100 or not: " + hs.contains(100));

        // Print the hashset step by step
        // For Each loop
        // {2, 4}
        //     ^
        for (Integer val : hs) {
            System.out.println(val);
        }

    }
}
```

Difference b/w ArrayList and HashSet

Duplicate	Allowed	Ignored
Order	Order of insertion is maintained	Random order
looping strategy	foreach loop, for loop with index	foreach

Q> Given A[] as input. Add it elements to hashset and return the hashset.



1	-2
3	7

Try till 22:54
Type done once done

```java
import java.util.*;

public class Main {

  static HashSet<Integer> formHS(int[] A){
    HashSet<Integer> hs = new HashSet<>();

    for(int i = 0; i < A.length; i++){
      hs.add(A[i]);
    }
    return hs;
  }
  public static void main(String[] args) {
    int[] A = {1, 3, -2, 7, 1, 1, -2};

    System.out.println(formHS(A));
  }
}
```

Q> Given two hashsets as input. Print their common elements.

hs1 : { 0, -2, 4, 10 }
hs2 : { 1, -2, 3, 5, 4 }

Output ⟶ -2 4

Try till 22:10
Type done if correct output is achieved.

hs1 : { 1 2 3 }
hs2 : { 6 1 4 2 3 9 }

Output ⟶ 1 2 3

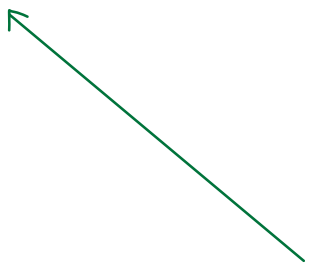
Break : 22:30

Idea Iterate over HS2 and find the elements of HS2 are present in HS1.

```
import java.util.*;

public class Main {
  static void intersect(HashSet<Integer> hs1, HashSet<Integer> hs2){
    for(Integer ele2 : hs2){
      if(hs1.contains(ele2)){
        System.out.print(ele2 + " ");
      }
    }
  }
  public static void main(String[] args) {
      HashSet<Integer> hs1 = new HashSet<>();
      hs1.add(0);hs1.add(-2);hs1.add(4);hs1.add(10);
      HashSet<Integer> hs2 = new HashSet<>();
      hs2.add(1);hs2.add(-2);hs2.add(3);hs2.add(5);hs2.add(4);
      intersect(hs1, hs2);
  }
}
```

HashMap — HashMap contains key-value pairs.

Key	Value
States	Population
Punjab	15
Haryana	18
UP	20
Delhi	18



Tell me the population of punjab ——

Syntax

```
                key , value
HashMap < KeyType , ValueType > hm = new HashMap<>();
          String , Long
```

How to add a key, value pair inside HM ?

```
hm.put ( key , value ) ;
hm.put ("punjab" , (5) ;
```

How to get the value against a key in HM ?

```
hm.get (key) ;
hm.get ("punjab") ;  ——→ return 15
```

Can there be duplicate values in HM ? yes

Can there be duplicate keys in HM ? NO

Features of HashMap

Duplicate values are allowed

Duplicate keys are not allowed

Order of key, value ⟶ random

```
public class Main {
    public static void main(String[] args) {
        HashMap<String, Integer> hm = new HashMap<>();
        hm.put("Delhi", 18);
        hm.put("Punjab", 20);
        hm.put("Haryana", 18);
        hm.put("Goa", 5);
        hm.put("Goa", 4); // Update the value against a key
        hm.put("MH", 40);
        // Print the HM
        System.out.println(hm + " size : " + hm.size());
        System.out.println("Population of Punjab " + hm.get("Punjab"));
        // hs contains
        // hm containsKey
        if (hm.containsKey("MH")) { // Always check first before you get.
            System.out.println(hm.get("MH"));
        }

        // Remove any key, value pair in HM
        hm.remove( key: "Goa");
        System.out.println("Get " + hm.get("Goa"));

        for (String key : hm.keySet()) {
            System.out.println("key-value " + key + " " + hm.get(key));
        }
    }
}
```

Q) Given A[] as input. Return the freq of array as key, value pair

A[] = { 1, 3, 2, -1, 1, 1, 3 }

Output = { 1 = 3, 3 = 2, 2 = 1, -1 = 1 }

freq map.

Try till 23 : 12

A[] = { 1, 3, 2, -1, 1, 1, 3 }

HM

1 : ~~1 2~~ 3 3 : ~~1~~ 2 2 : 1

-1 : 1

```java
import java.util.*;

public class Main {
  static HashMap<Integer, Integer> makeFreqMap(int[] A){
    HashMap<Integer, Integer> hm = new HashMap<>();

    for(int key : A){ // for each -> iterates over the values
      if(!hm.containsKey(key)){ // hm doesn't contain the key
        hm.put(key, 1);
      } else { // key already exists
        int value = hm.get(key);
        hm.put(key, value + 1);
      }
    }

    return hm;
  }
  public static void main(String[] args) {
      int[] A = {1, 4, 3, -2, 1, 1, 4, 5, 3};

      System.out.println(makeFreqMap(A));
  }
}
```