Hasmap

Hapmap<Key , value>;

Book

Java Books	:	6
C++	;	7
Python	:	8
Ruby	:	5
Dart	:	8
Javascript	:	10

Company

Employee	Salary
emp 1	10,000
emp 2	20,000
emp 3	30,000
emp 4	40,000
emp 5	50,000

key

Value

<Integer, Integer>

<Integer, string>

<string, Integer>

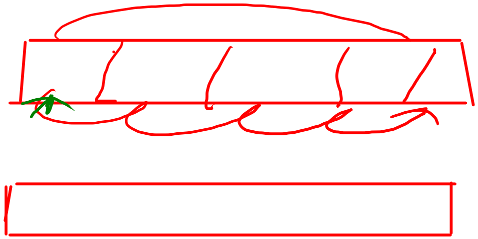
→ [1, 100] -2^{31} to 2^{31}

[2, −3, −5, 1, 5, 7, 2]

−ve to +ve

−5

Array as Hashmap



← O(n)

2 :	2
3 :	1
5 :	2
7 :	1

→ O(1)

type	price
Diesel	90
Petrol	90
CNG	100

key

value

+ add

+ delete

+ search

+ update

* key must be unique

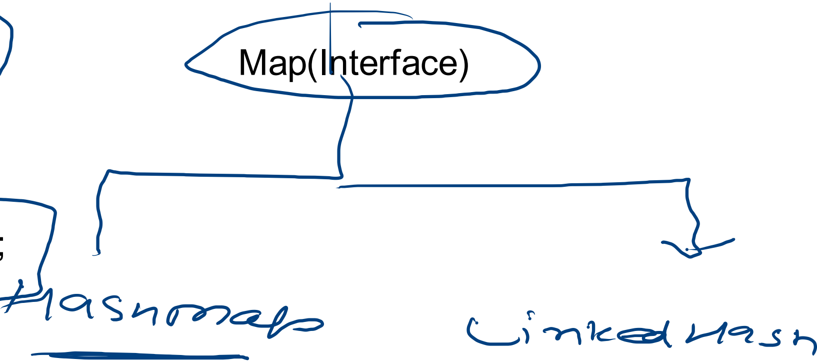
Ⓢ {2, 2, 3, 6, 5, 5}

Syntax

Industry

```
Map<key,value> objName = new HashMap<>();
```

```
HashMap<key,value> objName = new HashMap<>();
```



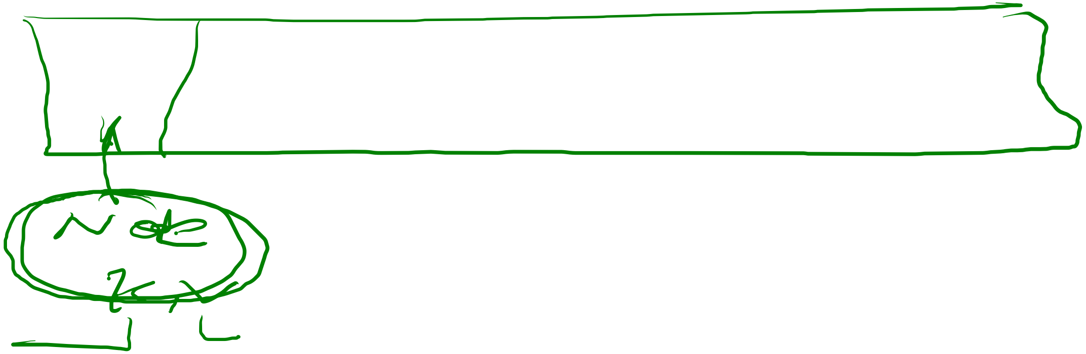
put (key, value)

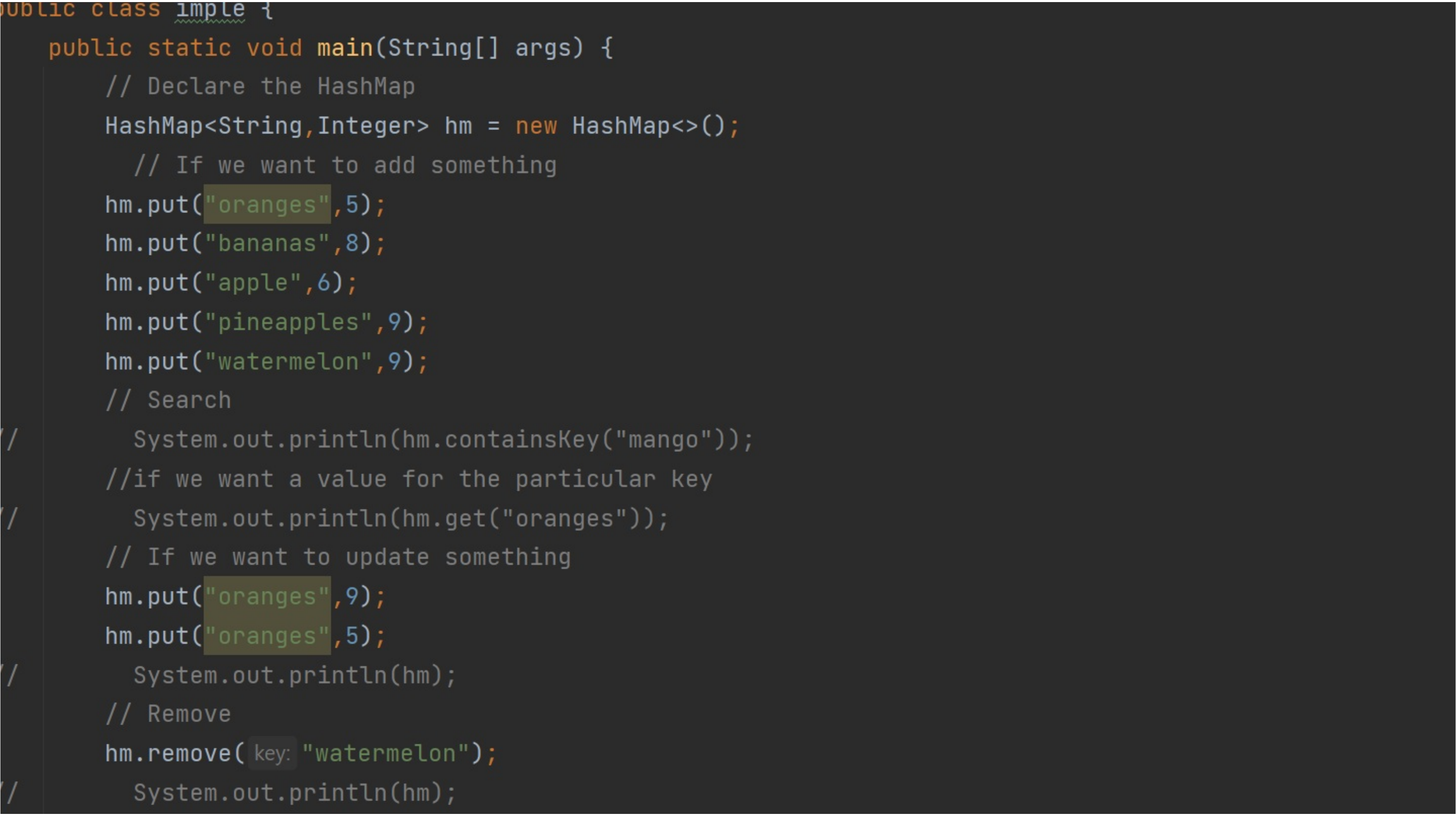
// search

Contains key
- If the key will exist then it will give true
- Not exist then it will give false

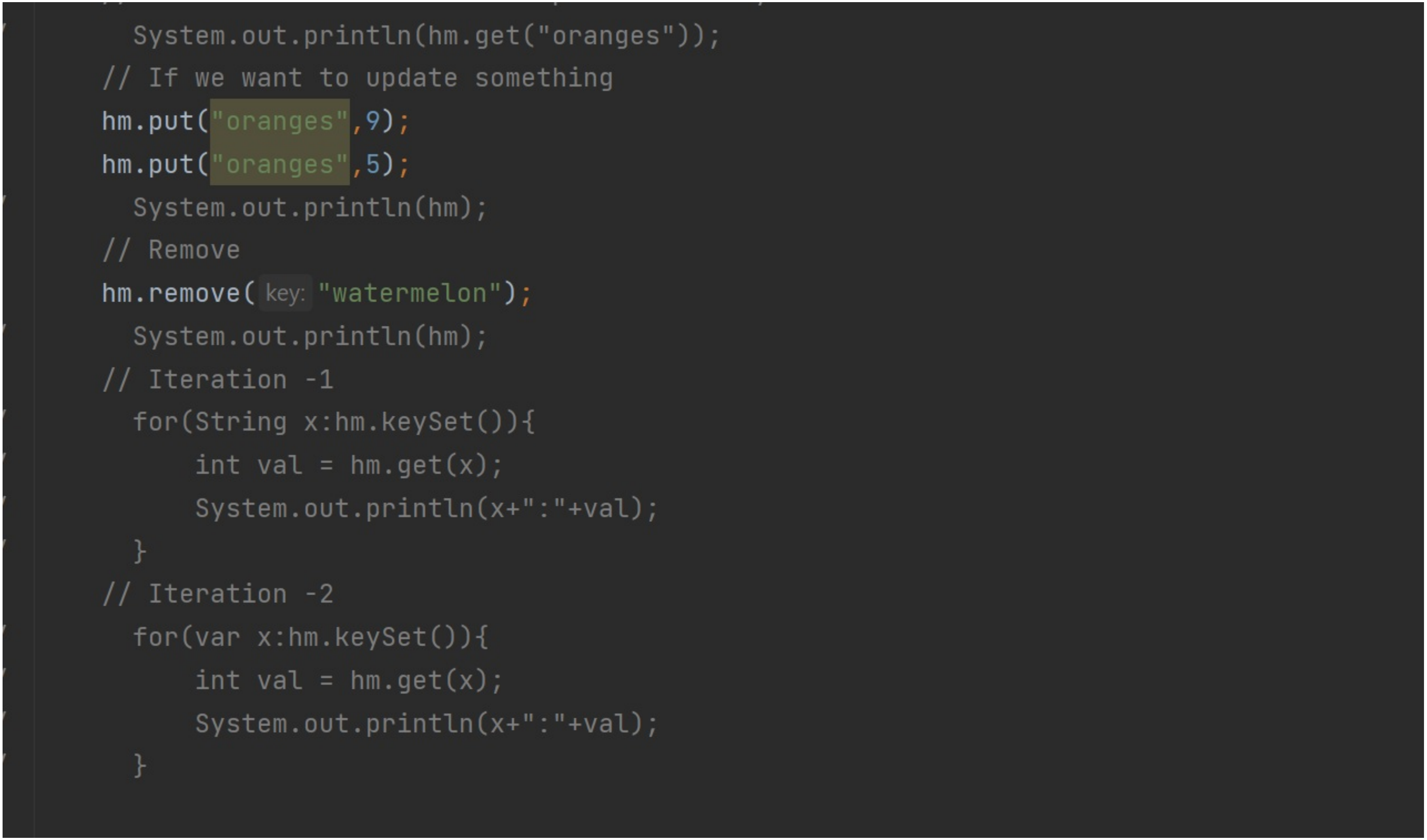
//
Get
- key exist, then it will give the value
- key not exist then it print null

Array of Linked





```java
public class imple {
    public static void main(String[] args) {
        // Declare the HashMap
        HashMap<String,Integer> hm = new HashMap<>();
          // If we want to add something
        hm.put("oranges",5);
        hm.put("bananas",8);
        hm.put("apple",6);
        hm.put("pineapples",9);
        hm.put("watermelon",9);
        // Search
//        System.out.println(hm.containsKey("mango"));
        //if we want a value for the particular key
//        System.out.println(hm.get("oranges"));
        // If we want to update something
        hm.put("oranges",9);
        hm.put("oranges",5);
//        System.out.println(hm);
        // Remove
        hm.remove( key: "watermelon");
//        System.out.println(hm);
```



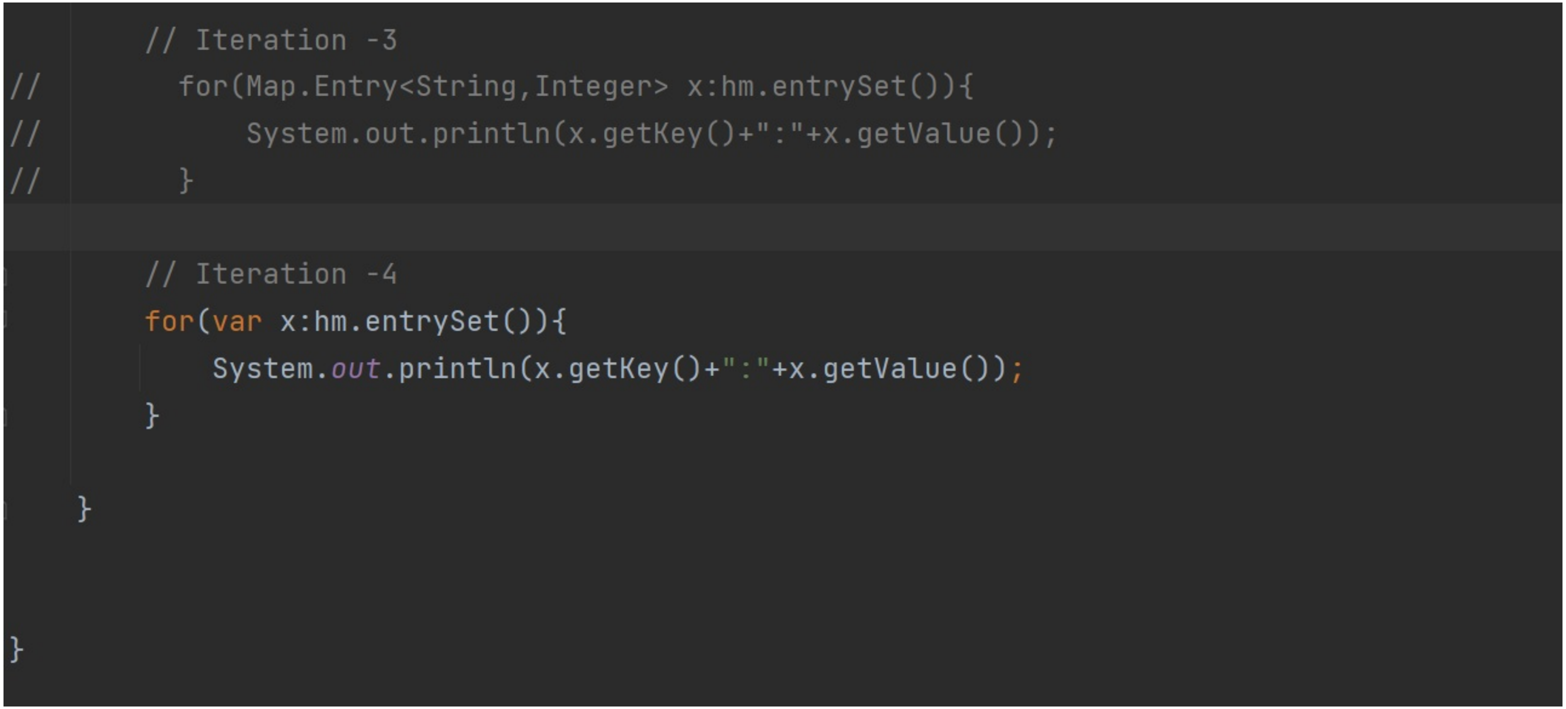
```
  System.out.println(hm.get("oranges"));
// If we want to update something
hm.put("oranges",9);
hm.put("oranges",5);
  System.out.println(hm);
// Remove
hm.remove( key: "watermelon");
  System.out.println(hm);
// Iteration -1
  for(String x:hm.keySet()){
      int val = hm.get(x);
      System.out.println(x+":"+val);
  }
// Iteration -2
  for(var x:hm.keySet()){
      int val = hm.get(x);
      System.out.println(x+":"+val);
  }
```

"Java" : 5

"Java" : 7

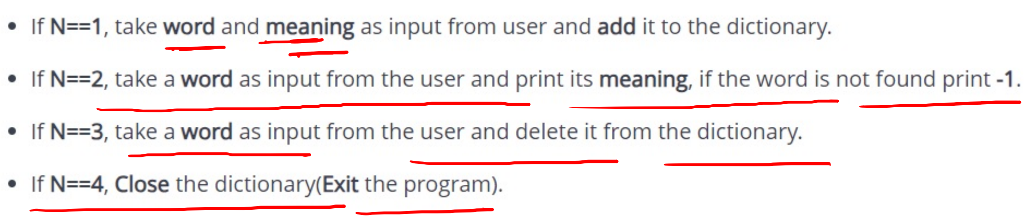
→ 3 (++";

output: null

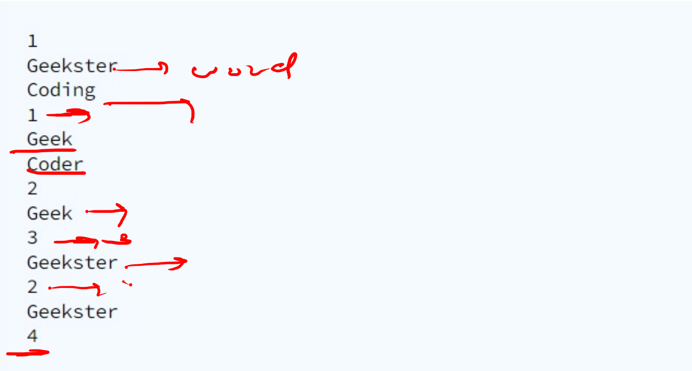


```
        // Iteration -3
//        for(Map.Entry<String,Integer> x:hm.entrySet()){
//            System.out.println(x.getKey()+":"+x.getValue());
//        }

        // Iteration -4
        for(var x:hm.entrySet()){
            System.out.println(x.getKey()+":"+x.getValue());
        }

    }


}
```

[2, 2, 3, 5, 8]



- If N==1, take **word** and **meaning** as input from user and **add** it to the dictionary.
- If N==2, take a **word** as input from the user and print its **meaning**, if the word is not found print **-1**.
- If N==3, take a **word** as input from the user and delete it from the dictionary.
- If N==4, **Close** the dictionary(**Exit** the program).



```
1
Geekster
Coding
1
Geek
Coder
2
Geek
3
Geekster
2
Geekster
4
```

word

word	meaning
~~Geekster~~	~~coding~~
Geek	Coder

Output: Coder
-1

{5, 6, 8, 9, 5, 3, 2, 1}

{

5 : 2

6 : 1

8 : 1

9 : 1

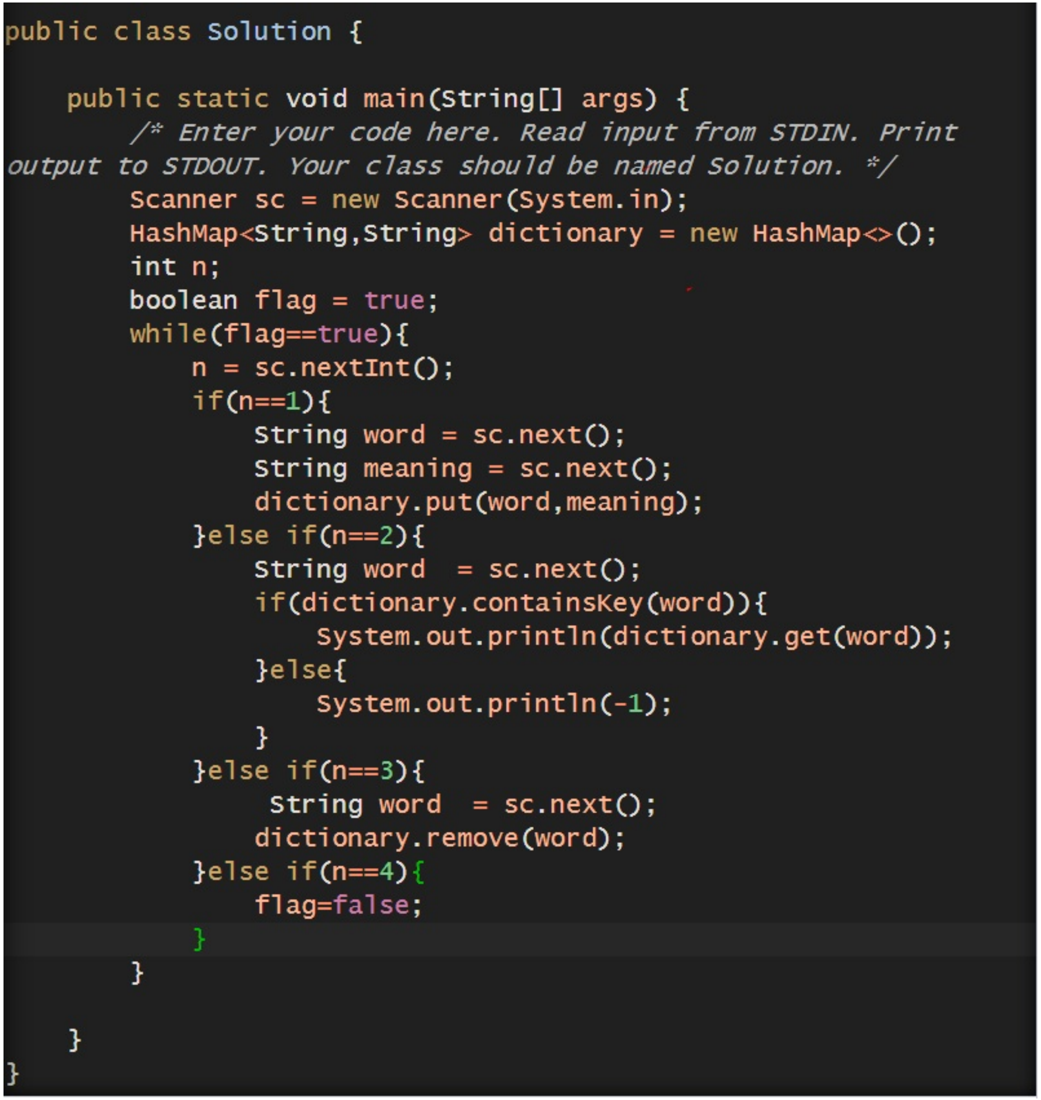
5 : 1

3 : 1

2 : 1

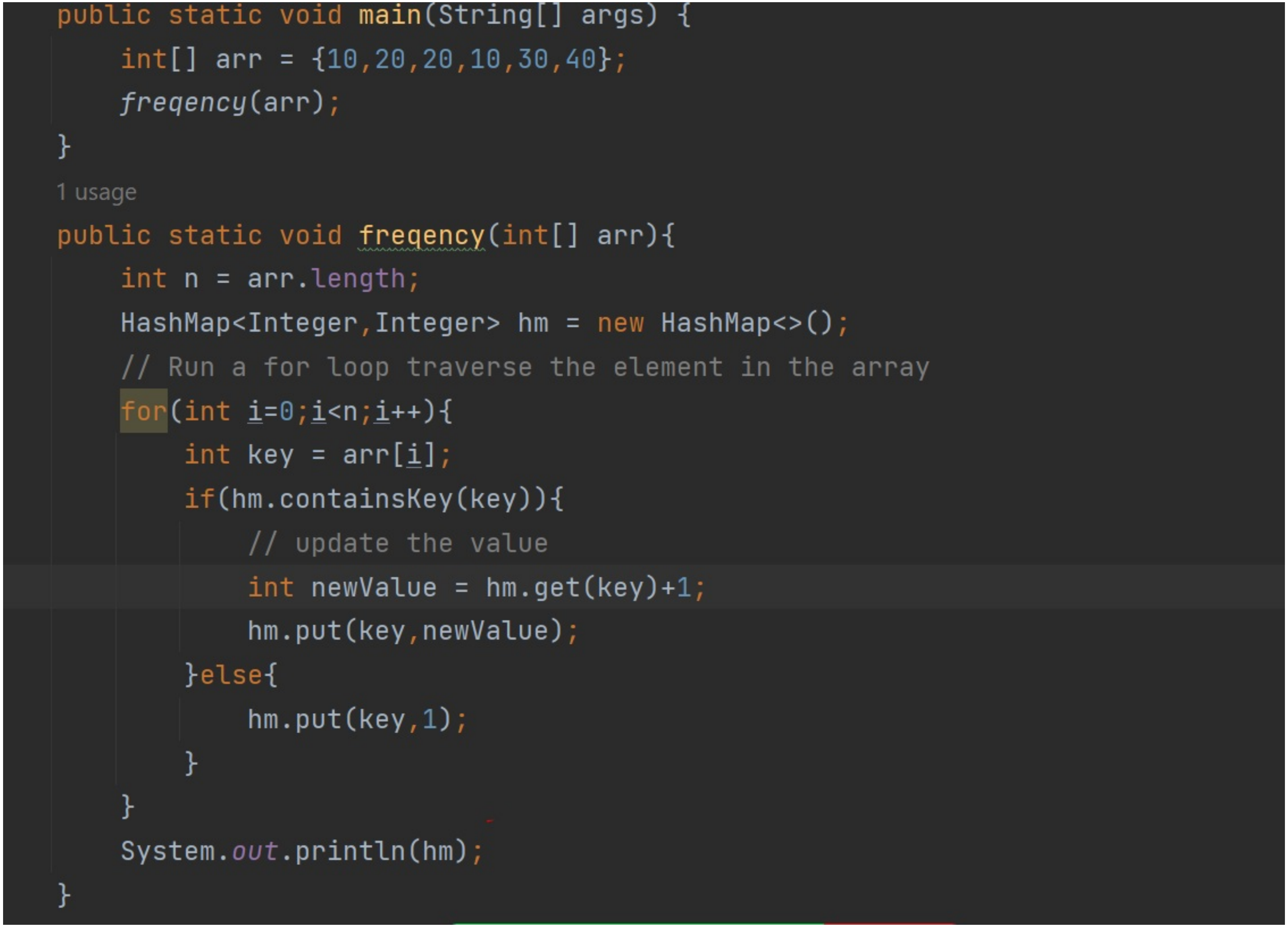
1 : 1

Word Meaning



```java
public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print
output to STDOUT. Your class should be named Solution. */
        Scanner sc = new Scanner(System.in);
        HashMap<String,String> dictionary = new HashMap<>();
        int n;
        boolean flag = true;
        while(flag==true){
            n = sc.nextInt();
            if(n==1){
                String word = sc.next();
                String meaning = sc.next();
                dictionary.put(word,meaning);
            }else if(n==2){
                String word  = sc.next();
                if(dictionary.containsKey(word)){
                    System.out.println(dictionary.get(word));
                }else{
                    System.out.println(-1);
                }
            }else if(n==3){
                 String word  = sc.next();
                dictionary.remove(word);
            }else if(n==4){
                flag=false;
            }
        }

    }
}
```

Frequency storage



```
public static void main(String[] args) {
    int[] arr = {10,20,20,10,30,40};
    freqency(arr);
}
1 usage
public static void freqency(int[] arr){
    int n = arr.length;
    HashMap<Integer,Integer> hm = new HashMap<>();
    // Run a for loop traverse the element in the array
    for(int i=0;i<n;i++){
        int key = arr[i];
        if(hm.containsKey(key)){
            // update the value
            int newValue = hm.get(key)+1;
            hm.put(key,newValue);
        }else{
            hm.put(key,1);
        }
    }
    System.out.println(hm);
}
```