

22 June 2023

HashMap:

>HashMap stores items in "key/value" pairs.

>We can access them by index of another type (eg: String)

```
1 // HashMap
2 // import java.util.*;
3 import java.util.HashMap;
4 import java.util.Iterator;
5 class Main
6 {
7     public static void main(String[] args)
8     {
9         HashMap<String, String> lovers = new HashMap<String, String>();
10
11         lovers.put("Romeo", "Juliet");
12         lovers.put("Aamir Khan", "Juhi Chawla");
13         lovers.put("Professor", "Neha");
14         lovers.put("Naruto", "Hinata");
15         System.out.println(love);
16         System.out.println(love.get("Aamir Khan"));
17     }
18 }
19
```

Handwritten notes and table:

key	value
Romeo	Juliet
AK	J C
Prof	Neha
Nar.	Hin

Using key, we can access value.

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL

```
{Professor=Neha, Aamir Khan=Juhi Chawla, Naruto=Hinata, Romeo=Juliet}
Juhi Chawla
(base) ingledarshan@192 NS 22 June 2023 %
```

Polls/Quizzes How it works?

Poll #5 LIVE

End Test

Q1. What will be the output of the following program? ...
Attempted: 16

```
public static void main(String args[]) {
    Map map = new HashMap();
    map.put("key1", "family");
    map.put("key2", "relatives");
    map.put("key3", "friends");
    System.out.println(map);
    String str = (String) map.get("key1");
    System.out.print(str);
}
```

Handwritten notes and table:

k1	k2	k3
family	rel.	friends

Results:

Option	Percentage
{key3=relatives, key2=relatives, key1=family} family	25%
{key1=family, key2=relatives, key3=relatives} family	63%
{key3=relatives, key2=relatives, key1=family} [EMPTY]	6.3%
{key1=family, key2=relatives, key3=relatives} [EMPTY]	6.3%

Output

Print 26 space separated numbers that denote the number of occurrences of each character from 'a' to 'z'.

Example

Sample Input:-

newton school

Sample Output:-

0 0 1 0 1 0 0 1 0 0 0 1 0 2 3 0 0 0 1 1 0 0 1 0 0 0
a b c d e f g h i j k l m n o p q r s t u v w x y z

n-2

e-1

h-1

w-1

l-1

t-1

o-3

s-1

c-1

3
4
5
6
7
8
9
10
11
12
13
14
15

Generate Expected Output

← →

NS 22 June 2023

J Main.java 1 x

J Main.java > Main > main(String[])

```
74 {
75     Run | Debug
76     public static void main(String[] args)
77     {
78         Scanner sc = new Scanner(System.in);
79         String input = sc.nextLine().toLowerCase();
80         int [] occurrences = new int[26];
81         for(char c : input.toCharArray())
82         {
83             if(c >= 'a' && c <= 'z')
84             {
85                 occurrences[c - 'a']++;
86             }
87         }
88         for(int count : occurrences)
89         {
90             System.out.print(count + " ");
91         }
92     }
93 }
```

input = newton school

0	1	2	3	4	...	24	25
0	1	1	0	0	...	0	0

a=97

b=98

c=99

d=100

⋮

Newton School
0 0 1 0 1 0 0 1 0 0 0 1 0 2 3 0 0 0 1 1 0 0 1 0 0 0
o (base) ingledarshan@192 NS 22 June 2023 %

Code

Ln 89, Col 10 Spaces: 4 UTF-8 LF {} Java