# Hash Tables

Data Structures
Week #4

# Practice

Go to GeeksforGeeks HashTable

https://www.geeksforgeeks.org/hashtable-in-java/

Go to GeeksforGeeks HashMap

https://www.geeksforgeeks.org/java-util-hashmap-in-java-with-examples/

Go to GeeksforGeeks HashSet

https://www.geeksforgeeks.org/hashset-in-java/

Practice HashMap design with LeetCode

https://leetcode.com/problems/design-hashmap//

# Hash Table Usage

# Hash Table Usage

Finding duplicates

# Hash Table Usage

Finding duplicates

Finding Sums

# Hash Table Usage

Finding duplicates

Finding Sums

Finding sequences

# Hash Table Usage

Finding duplicates

Finding Sums

Finding sequences

LRU (Least Recently Used) Cache

# HashTable

# HashTable

It is similar to HashMap, but is synchronized

# HashTable

It is similar to HashMap, but is synchronized

Stores key/value pair in hash table

# HashTable

It is similar to HashMap, but is synchronized

Stores key/value pair in hash table

In Hashtable we specify an object that is used as a key, and the value we want to associate to that key. The key is then hashed, and the resulting hash code is used as the index at which the value is stored within the table.

# HashTable


```java
Hashtable<String, Integer> topTestScores =
    new Hashtable<String, Integer>();
topTestScores.put("John", 99);
topTestScores.put("Sandra", 100);
topTestScores.put("Sammy", 85);
// Prints 100
System.out.println(topTestScores.get("Sandra"));
```


```python
topTestScores =
    {'John': 99, 'Sandra': 100, 'Sammy': 85}
# Prints 85
print topTestScores['Sammy']
```

# HashMap

# HashMap

**It is similar to HashTable, but is not synchronized**

# HashMap

It is similar to HashTable, but is not synchronized

HashMap doesn’t allow duplicate keys but allows duplicate values. That means A single key can’t contain more than 1 value but more than 1 key can contain a single value.

# HashMap

It is similar to HashTable, but is not synchronized

HashMap doesn’t allow duplicate keys but allows duplicate values. That means A single key can’t contain more than 1 value but more than 1 key can contain a single value.

HashMap allows null key also but only once and multiple null values within the table

# HashMap

It is similar to HashTable, but is not synchronized

HashMap doesn’t allow duplicate keys but allows duplicate values. That means A single key can’t contain more than 1 value but more than 1 key can contain a single value.

HashMap allows null key also but only once and multiple null values within the table

This class makes no guarantees as to the order of the map; in particular, it does not guarantee that the order will remain constant over time

# HashSet

# HashSet

Underlying data structure for HashSet is hashtable

# HashSet

Underlying data structure for HashSet is hashtable

As it implements the Set Interface, duplicate values are not allowed

# HashSet

Underlying data structure for HashSet is hashtable

As it implements the Set Interface, duplicate values are not allowed

Objects that you insert in HashSet are not guaranteed to be inserted in same order. Objects are inserted based on their hash code

# HashSet

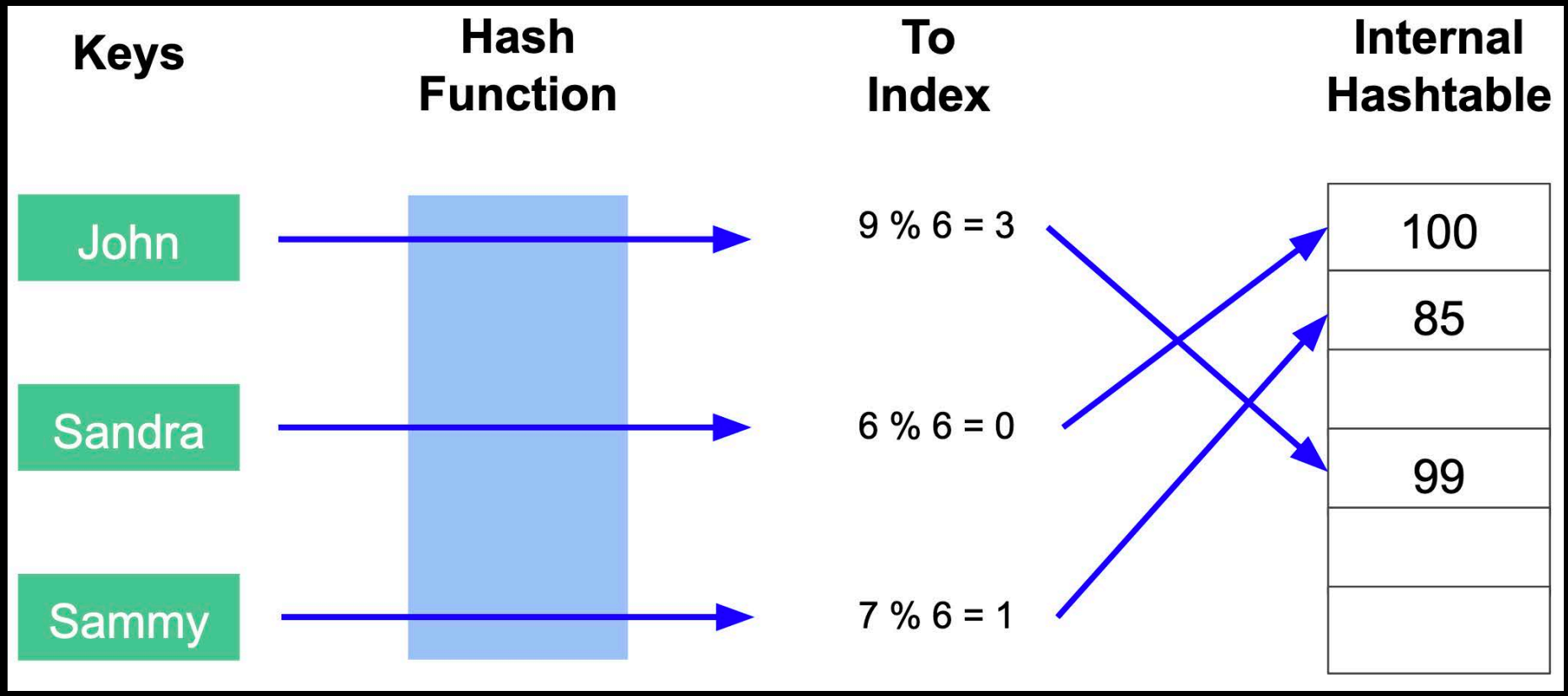
Underlying data structure for HashSet is hashtable

As it implements the Set Interface, duplicate values are not allowed

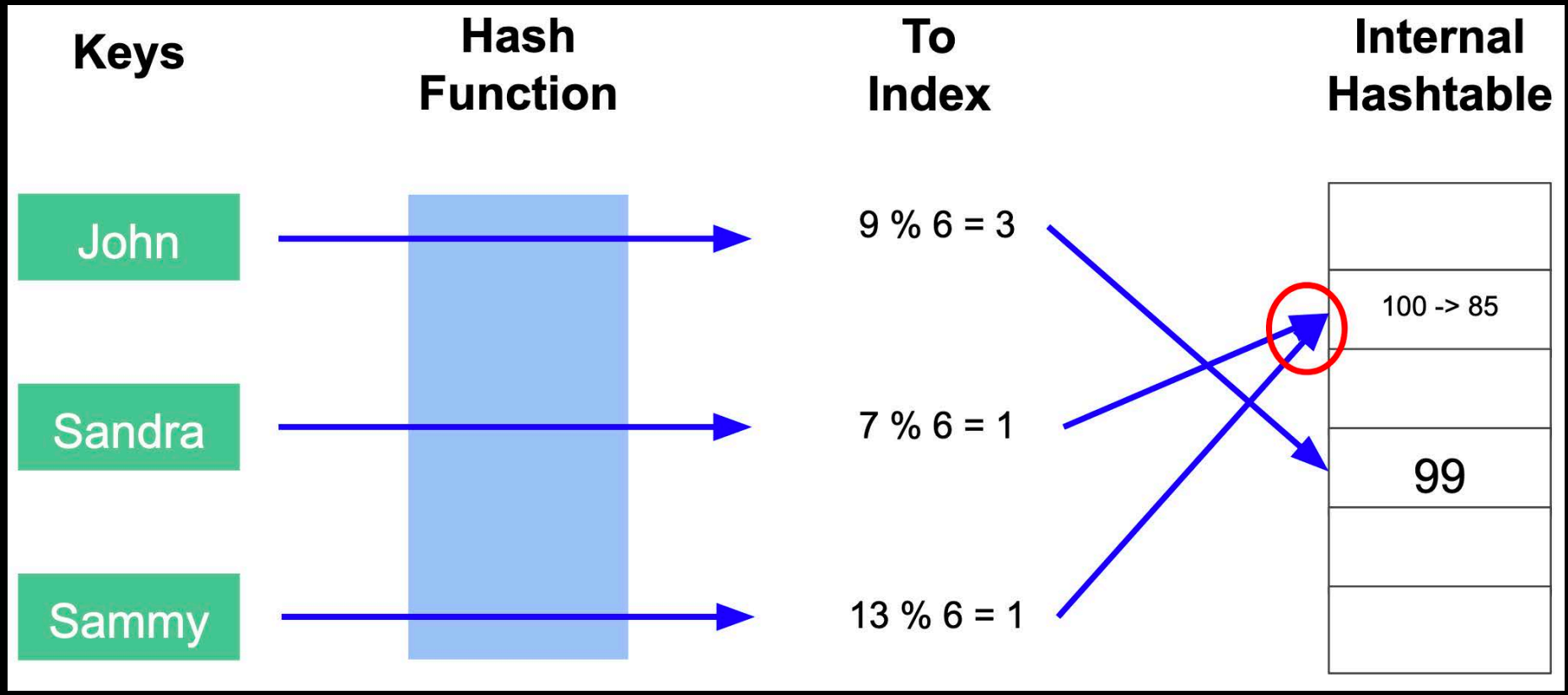
Objects that you insert in HashSet are not guaranteed to be inserted in same order. Objects are inserted based on their hash code

NULL elements are allowed in HashSet

# Hash Function



# Hash Function - Collisions



Understand

Matchtch

Psuedocode / Plan

Implement

Revieweview

Evaluate

# Practice Exercise #1

Given an array of integers, return indices of the two numbers such that they add up to a specific target.

You may assume that each input would have exactly one solution, and you may not use the same element twice.

**Example 1:**
Given nums = [2, 7, 11, 15], target = 9,

Because nums[0] + nums[1] = 2 + 7 = 9,
return [0, 1].

# Practice Exercise #2

You' are given strings representing the types of stones that are jewels, and s representing the stones you have.  Each character in J is a type of stone you have.  You want to know how many of the stones you have are also jewels.
The letters in J are guaranteed distinct, and all characters in J and S are letters. Letters are case sensitive, so 'a' is considered a different type of stone from 'A'.

**Example 1:**
**Input:** J =""aA", S =""aAAbbbb""
**Output:** 33

**Example 2:**
**Input:** J = "z", S = "ZZ"; "ZZ"
**Output:**  0

**Note:** S and J will consist of letters and have length at most 50.
The characters in J are distinct.

# Questions