# Lab: Java Advanced Sets and Maps

Problems for exercises and homework for the [“Java Advanced” course @ SoftUni](https://softuni.bg/trainings/1975/java-advanced-may-2018).

You can check your solutions here: <https://judge.softuni.bg/Contests/1029/Sets-And-Maps-Lab>

## Sets

### Parking Lot

Write program that:

* Record **car number** for every car that enter in **parking lot**
* Remove **car number** when the car go out
* Input will be string in format **[direction, carNumber]**
* input end with string **"END"**

Print the output with all car numbers which are in parking lot

#### Examples

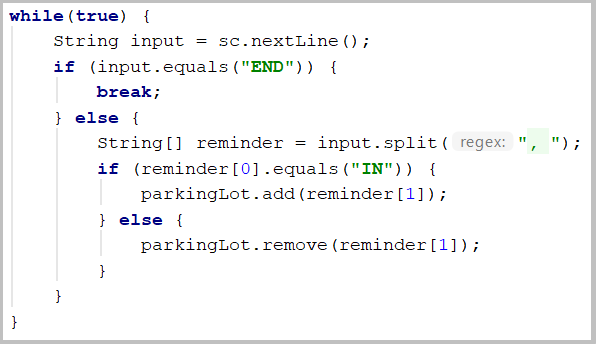
|  |  |
| --- | --- |
| **Input** | **Output** |
| IN, CA2844AA  IN, CA1234TA  OUT, CA2844AA  IN, CA9999TT  IN, CA2866HI  OUT, CA1234TA  IN, CA2844AA  OUT, CA2866HI  IN, CA9876HH  IN, CA2822UU  END | CA2822UU  CA2844AA  CA9999TT  CA9876HH |
| IN, CA2844AA  IN, CA1234TA  OUT, CA2844AA  OUT, CA1234TA  END | Parking Lot is Empty |

#### Hints

* Car numbers are **unique**
* Use the methods **isEmpty()**

#### Solution

You might help yourself with the code below:



### SoftUni Party

There is a party in SoftUni. Many guests are invited and they are two type: VIP and regular. When guest come check if he/she exist in any of two reservation lists

All reservation numbers will be with 8 chars

All VIP numbers start with digit

There will be 2 command lines. First is "PARTY" - party is on and guests start coming. Second is "END" – then party is over and no more guest will come

Output have to all guest, who didn't come to the party (VIP must be first)

#### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Input** | **Output** | **Input** | **Output** |
| 7IK9Yo0h  9NoBUajQ  Ce8vwPmE  SVQXQCbc  tSzE5t0p  PARTY  9NoBUajQ  Ce8vwPmE  SVQXQCbc  END | 2  7IK9Yo0h  tSzE5t0p | m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  xys2FYzn  MDzcM9ZK  PARTY  2FQZT3uC  dziNz78I  mdSGyQCJ  LjcVpmDL  fPXNHpm1  HTTbwRmM  B5yTkMQi  8N0FThqG  m8rfQBvl  fc1oZCE0  UgffRkOn  7ugX7bm0  9CQBGUeJ  END | 2  MDzcM9ZK  xys2FYzn |

### "Voina" - Number game

Write program that:

* Read 20 numbers for both players
* Numbers will be **Integer**, separated with **" "** **(single space).**
* Every player can hold only **unique** numbers
* Each Round both players get **top number** from their own. Player with bigger number get both numbers and add it on the **bottom** of his own numbers
* Game end after **50 rounds** or if any player **lose all** of his numbers
* Output must be **"First Player Win!"**, **"Second Player Win!"** or **"Draw!"**

#### Examples

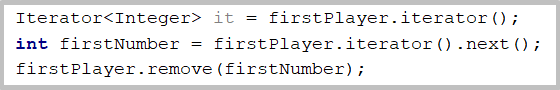
|  |  |
| --- | --- |
| **Input** | **Output** |
| 26 58 16 92 44 65 65 77 57 23 71 57 7 52 85 44 32 70 38 23  43 95 33 51 62 93 57 55 0 31 32 95 68 34 30 51 37 32 11 97 | Second player win! |
| 74 78 82 42 19 39 29 69 20 42 31 77 57 36 76 26 4 9 83 42  15 43 80 71 22 88 78 35 28 30 46 41 76 51 76 18 14 52 47 38 | First player win! |

#### Hints

* Use Iterator<E> and .next() for finding top number in decks
* Think where to check if any player is without cards
* When you find top number, be sure to remove it immediately

#### Solution

You might help yourself with the code below:



## Sets

### Count Same Values in Array

Write a program that counts in a given array of double values the number of occurrences of each value.

#### Examples

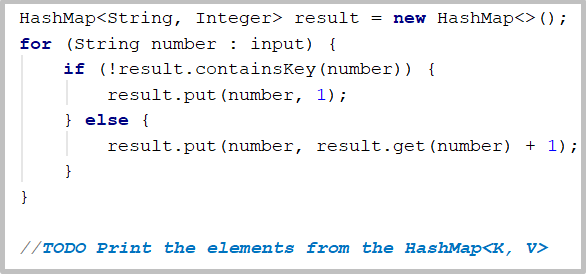
|  |  |
| --- | --- |
| **Input** | **Output** |
| -2.5 4 3 -2.5 -5.5 4 3 3 -2.5 3 | 3 - 4 times  4 - 2 times  -2.5 - 3 times  -5.5 - 1 times |
| 2 4 4 5 5 2 3 3 4 4 3 3 4 3 5 3 2 5 4 3 | 2 - 3 times  3 - 7 times  4 - 6 times  5 - 4 times |

#### Hints

* Use HashMap<K, V>

#### Solution

You might help yourself with the code below:



### Academy Graduation

Write a program that:

* Read from console **number** of student for a track
* Read on **pair of rows**:
  + First line is **name** of student
  + Second line is his **score** for different number of courses
* Print on console “**{name}** is graduated with **{average scores)**”

#### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3  Gosho  3.75 5  Mara  4.25 6  Pesho  6 4.5 | Gosho is graduated with 4.375  Mara is graduated with 5.125  Pesho is graduated with 5.25 |
| 5  Gruio  4.36 5.50 3.30 5.63 2.57 5.75 2.81 4.89  Trendafilka  3.10 5.35 3.30 3.35 5.64 4.99 2.75 4.68  Mite  3.45 3.23 3.03 5.42 5.46 4.15 2.26 5.95  Roza  2.08 3.48 3.36 2.73 2.96 4.54 3.70 3.85  Ganio  4.75 4.92 3.78 4.79 4.82 4.75 2.81 2.13 | Ganio is graduated with 4.09375  Gruio is graduated with 4.351249999999999  Mite is graduated with 4.11875  Roza is graduated with 3.3375  Trendafilka is graduated with 4.145 |

#### Hints

* Think about **proper type** of map
* **Value** can be **array**
* **Nested loop** and one more **variable** will be need for average score

#### Solution

You might help yourself with the code below:

