



RE-BOOTCAMP

Strings and Loops

STRINGS

- Strings are object representation of character sequences
- Strings are immutable object - The original object value cannot be changed
- There is two ways to make a String object: using new keyword or by giving the literal value
 - Literal Strings will be stored in the String pool
 - Strings made with new keyword will be stored in the heap memory like other objects

STRINGS

- String values use quotation marks ("")
- Strings are combined by concatenation. The operator to do this is +
 - Anything concatenated to a String will be turned to a String
- Every character in a String has an index, which is just the position of the character. Indexes always start from 0

STRING METHODS

<code>equals()</code>	<code>length()</code>	<code>charAt()</code>	<code>contains()</code>
<code>indexOf()</code>	<code>toLowerCase()</code>	<code>toUpperCase()</code>	<code>replace()</code>
<code>substring()</code>	<code>trim()</code>	<code>startsWith()</code>	<code>endsWith()</code>

LOOPS

- Loops are a concept that allows us to execute certain statements in a code block for a repeated number of executions
- We have traditional for loop, while loop, do while loop, for each loop
- Iteration: Every instance of execution in a loop. Every time the loop cycles fully through the statements it goes to the next iteration of the loop.

FOR LOOP

```
for(int i = 0; i < 5; i++){  
    System.out.println(i);  
}
```

Each part of a for loop:

-> initialization: happens only once in the beginning

```
int i = 0;
```

-> termination condition: when the result is false the loop ends

```
i < 5;
```

-> update: increases or decreases a value

```
i++
```

WHILE LOOP

The amount of iterations in a while loop are more dynamic. The iterations are determined by the boolean statement in the while.

```
int i = 0;
while(i < 5){
    System.out.println(i);
    i++;
}
```

As long as the boolean statement is true the loop will continue to run. Before every iteration execution the statement will be evaluated and if it is false the while loop will terminate.

DO WHILE

Do while loop works like while loop, but the first iteration is always executed then the statement is checked.

```
int i = 0;  
do {  
    System.out.println(i);  
} while (false);
```

With a do while loop one iteration will always be guaranteed to run

STRING WITH LOOPS

Loops and Strings are used together to do dynamic actions

Key things to keep in mind:

- > How many times should we loop through a String
 - length times
- > How can we read each character of a String with a loop
 - `charAt(i)`

TASK - 1 - FINRA

Write a program that prints out the numbers from 1 to 30 but for numbers which are a multiple of 3, print "FIN" instead of the number and for numbers which are a multiple of 5, print "RA" instead of the number. For numbers which are a multiple of both 3 and 5, print "FINRA" instead of the number.

TASK - 2

Write a program that will count how many times "java" is found in any given String

TASK - 3

Given any String determine if it is Palindrome. Print "Palindrome" if it is and "Not Palindrome" if it is not

NESTED LOOPS

Having a loop inside of a loop means the inner loop will execute from beginning to end for every iteration of the outer loop

In this example the outer loop (i) will run for 5 times. Each time that outer loop runs the inner loop (j) will run for 3 times. For a total of 15 executions

```
for(int i=0; i < 5; i++) {  
    for(int j=0; j < 3; j++) {  
        System.out.println("i: " + i);  
        System.out.println("j: " + j);  
    }  
}
```

TASK - 4

Given any String print out how many times each character is found in the String

Ex: java

j - 1

a - 2

v - 1

TASK - 5

Create a program that will take any String and print the total sum of all the number in the String.

Note: numbers can be more than digits from 1-9 so if you have "14" next to each other it should be considered 14 and not 1 and 4 separate

Ex:

Input: "jav45ai15sgre1at82"

Output: $(45 + 15 + 1 + 82) = 143$

TASK - 6

Given two Strings determine if they are Anagrams -> Are built of the same characters

TASK - 7 - PASSWORD VALIDATION

1. Password MUST be at least 8 characters
2. Password should at least contain one upper case letter
3. Password should at least contain one lowercase letter
4. Password should at least contain one special characters
5. Password should at least contain a digit

if all requirements above are met, the password is valid, if not all are met it is invalid