Overview: 10 Days of JavaScript

Day 0 Hello World!

This series focuses on learning and practicing JavaScript. Each challenge comes with a tutorial article, and you can view these articles by clicking either the *Topics* tab along the top or the *article* icon in the right-hand menu.

**Objective**

In this challenge, we review some basic concepts that will get you started with this series. Check out the tutorial to learn more about JavaScript's lexical structure.

**Task**

A *greeting* function is provided for you in the editor below. It has one parameter, parameterVariable. Perform the following tasks to complete this challenge:

1. Use console.log() to print Hello, World! on a new line in the console, which is also known as *stdout* or *standard output*. The code for this portion of the task is already provided in the editor.
2. Use console.log() to print the contents of  (i.e., the argument passed to *main*).

You've got this!

**Input Format**

| **Data Type** | **Parameter** | **Description** |
| --- | --- | --- |
| string | parameterVariable | A single line of text containing one or more space-separated words. |

**Output Format**

Print the following two lines of output:

1. On the first line, print Hello, World! (this is provided for you in the editor).
2. On the second line, print the contents of parameterVariable.

**Sample Input 0**

Welcome to 10 Days of JavaScript!

**Sample Output 0**

Hello, World!

Welcome to 10 Days of JavaScript!

**Explanation 0**

We printed two lines of output:

1. We printed the literal string Hello, World! using the code provided in the editor.
2. The value of  parameterVariable passed to our *main* function in this *Sample Case* was Welcome to 10 Days of JavaScript!. We then passed our variable to *console.log*, which printed the contents of .

'use strict';

process.stdin.resume();

process.stdin.setEncoding('utf-8');

let inputString = '';

let currentLine = 0;

process.stdin.on('data', inputStdin => {

inputString += inputStdin;

});

process.stdin.on('end', \_ => {

inputString = inputString.trim().split('\n').map(string => {

return string.trim();

});

main();

});

function readLine() {

return inputString[currentLine++];

}

/\*\*

\* A line of code that prints "Hello, World!" on a new line is provided in the editor.

\* Write a second line of code that prints the contents of 'parameterVariable' on a new line.

\*

\* Parameter:

\* parameterVariable - A string of text.

\*\*/

function greeting(parameterVariable) {

// This line prints 'Hello, World!' to the console:

console.log('Hello, World!');

// Write a line of code that prints parameterVariable to stdout using console.log:

}

function main() {

const parameterVariable = readLine();

greeting(parameterVariable);

}

Solution: console.log(parameterVariable);

# Day 0:Data Types

**Objective**

Today, we're discussing data types. Check out the attached tutorial for more details.

**Task**

Variables named firstInteger, firstDecimal, and firstString are declared for you in the editor below. You must use the  operator to perform the following sequence of operations:

1. Convert  secondInteger to an integer (Number type), then sum it with fristInteger  and print the result on a new line using console.log.
2. Convert  secondDecimal to a floating-point number (Number type), then sum it with firstDecimal  and print the result on a new line using console.log.
3. Print the concatenation of firstString and  secondString on a new line using console.log. Note that firstString must be printed first.

**Input Format**

| **Data Type** | **Parameter** | **Description** |
| --- | --- | --- |
| string | secondInteger | The string representation of an integer you must sum with firstInteger. |
| string | secondDecimal | The string representation of a floating-point number you must sum with firstDecimal. |
| string | secondString | A string of one or more space-separated words you must append to secondString . |

**Output Format**

Print the following three lines of output:

1. On the first line, print the sum of firstInteger and the integer representation of secondInteger.
2. On the second line, print the sum of firstDecimal and the floating-point representation of secondDecimal.
3. On the third line, print firstString concatenated with secondString. You must print firstString  before secondString .

**Sample Input 0**

12

4.32

is the best place to learn and practice coding!

**Sample Output 0**

16

8.32

HackerRank is the best place to learn and practice coding!

**Explanation 0**

When we sum the integers 4 and 12, we get the integer 16 .  
When we sum the floating-point numbers 4.0  and 4.32 , we get 8.32  . When we concatenate HackerRank with is the best place to learn and practice coding!, we get HackerRank is the best place to learn and practice coding!.

**You will not pass this challenge if you attempt to assign the Sample Case values to your variables instead of following the instructions above.**

'use strict';

process.stdin.resume();

process.stdin.setEncoding('utf-8');

let inputString = '';

let currentLine = 0;

process.stdin.on('data', inputStdin => {

inputString += inputStdin;

});

process.stdin.on('end', \_ => {

inputString = inputString.trim().split('\n').map(string => {

return string.trim();

});

main();

});

function readLine() {

return inputString[currentLine++];

}

/\*\*

\* The variables 'firstInteger', 'firstDecimal', and 'firstString' are declared for you -- do not modify them.

\* Print three lines:

\* 1. The sum of 'firstInteger' and the Number representation of 'secondInteger'.

\* 2. The sum of 'firstDecimal' and the Number representation of 'secondDecimal'.

\* 3. The concatenation of 'firstString' and 'secondString' ('firstString' must be first).

\*

\* Parameter(s):

\* secondInteger - The string representation of an integer.

\* secondDecimal - The string representation of a floating-point number.

\* secondString - A string consisting of one or more space-separated words.

\*\*/

function performOperation(secondInteger, secondDecimal, secondString) {

// Declare a variable named 'firstInteger' and initialize with integer value 4.

const firstInteger = 4;

// Declare a variable named 'firstDecimal' and initialize with floating-point value 4.0.

const firstDecimal = 4.0;

// Declare a variable named 'firstString' and initialize with the string "HackerRank".

const firstString = 'HackerRank ';

// Write code that uses console.log to print the sum of the 'firstInteger' and 'secondInteger' (converted to a Number type) on a new line.

// Write code that uses console.log to print the sum of 'firstDecimal' and 'secondDecimal' (converted to a Number type) on a new line.

// Write code that uses console.log to print the concatenation of 'firstString' and 'secondString' on a new line. The variable 'firstString' must be printed first.

}

function main() {

const secondInteger = readLine();

const secondDecimal = readLine();

const secondString = readLine();

performOperation(secondInteger, secondDecimal, secondString);

}

**Solution:**

const sumIntegers = firstInteger + Number(secondInteger);

console.log(sumIntegers);

const sumDecimals = firstDecimal + Number(secondDecimal);

console.log(sumDecimals);

const concatenatedString = firstString + secondString;

console.log(concatenatedString);

**Day 1: Arithmetic Operators**

**Day 8:**

**WE ARE DEPRECATING THIS JAVASCRIPT CHALLENGE AND WILL REMOVE IT SOON.**

**Objective**

In this challenge, we practice creating buttons in JavaScript. Check out the attached tutorial for learning materials.

**Task**

Complete the code in the editor so that it creates a clickable button satisfying the following properties:

* The button's id is btn.
* The button's initial text label is . After each click, the button must increment by . Recall that the button's text label is the JS object's innerHTML property.
* The button has the following style properties:
  + A width of 96px.
  + A height of 48px.
  + The font-size attribute is 24px.

The .js and .css files are in different directories, so use the *link* tag to provide the CSS file path and the *script* tag to provide the JS file path:

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="css/button.css" type="text/css">

</head>

<body>

<script src="js/button.js" type="text/javascript"></script>

</body>

</html>

**Submissions**

This is a new style of challenge involving Front-End rendering. It may take up to  seconds to see the result of your code, so *please be patient after clicking Submit*. The *Submissions* page contains screenshots to help you gauge how well you did.

Ask questions in the [Discussions forum](https://www.hackerrank.com/forum) and submit any bug reports to support@hackerrank.com. Enjoy!

**Explanation**

Initially, the button looks like this:

initial

After the first  4 clicks, it looks like this:

four clicks

After 5 more clicks, it looks like this:

nine clicks