# JS Arrays and Strings – Lab

Problems with exercise and homework for the ["JS Front-End" Course @ SoftUni.](https://softuni.bg/trainings/4240/js-front-end-october-2023)

## Sum First and Last Array Elements

Write a function that receives an **array of numbers** and prints the sum of the **first** and **last** element in that array.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| [20, 30, 40] | 60 |
| [10, 17, 22, 33] | 43 |
| [11, 58, 69] | 80 |

### Hints

* You can access the **last element** in an array by subtracting 1 from **its length**:



## Reverse an Array of Numbers

Write a program, which receives a number n and an **array** of elements. Your task is to **create** a new array with n numbers from the original array, **reverse** it and **print** its elements on a single line, space-separated.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| 3, [10, 20, 30, 40, 50] | 30 20 10 |
| 4, [-1, 20, 99, 5] | 5 99 20 -1 |
| 2, [66, 43, 75, 89, 47] | 43 66 |

### Hints

* Use push() to add elements inside the new array



* Use **string interpolation** for the output



## Even and Odd Subtraction

Write a program that calculates the **difference** between the sum of the **even** and the sum of the **odd** numbers in an array.

### Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| [1,2,3,4,5,6] | 3 | 2 + 4 + 6 = 12, 1 + 3 + 5 = 9, 12 - 9 = 3 |
| [3,5,7,9] | -24 |  |
| [2,4,6,8,10] | 30 |  |

### Hints

* Parse each string to number



* Create two variables - for **even** and **odd** sum



* Iterate through all elements in the array with aloopand check if the number is odd or even
* Print the difference

## Substring

Write a function that **receives a string** and **two numbers**. The numbers will be a **starting index** and **count** of elements to substring. Print the result.

### Input / Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| 'ASentence', 1, 8 | Sentence |
| 'SkipWord', 4, 7 | Word |

### Hints

Create a new string that takes the needed amount of elements from the given string.



## Censored Words

Write a function that **receives a text as** a first parameter and a **single word** as a second. Find **all occurrences** of that word in the text and replace them with the corresponding count of **'\*'**.

### Input / Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| 'A small sentence with some words', 'small' | A \*\*\*\*\* sentence with some words |
| 'Find the hidden word', 'hidden' | Find the \*\*\*\*\*\* word |

### Hints

Save the new text in a new variable.



The repeat() function should take the length of the word and return that amount of stars '\*'.

## Count String Occurrences

Write a function that **receives a text** and a **single word** **that you need to search**. Print the number of all occurrences of this word in the text.

### Input / Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| 'This is a word and it also is a sentence',  'is' | 2 |
| 'softuni is great place for learning new programming languages',  'softuni' | 1 |

### Hints

Split the sentence into words and create a **counter** that stores how many times the searched word occurs.

