# JS Syntax Fundamentals - Exercises

Problems with exercise and homework for the "JS Front-End" Course @ SoftUni.

## 1. Ages

Write a function that determines whether based on the given age a person is: baby, child, teenager, adult, elder.

The input comes as a **single number parameter**. The bounders are:

- 0-2 (age) is a baby;
- 3-13 (age) is a child;
- 14-19 (age) is a teenager;
- 20-65 (age) is an adult;
- >=66 (age) is an elder;
- In all other cases print "out of bounds";

The **output** should be printed to the console.

### **Examples**

Input	Output
20	adult
1	baby
100	elder
-1	out of bounds

### 2. Vacation

You are given a group of people, the type of the group, and the day of the week they are going to stay. Based on that information calculate how much they have to pay and print that price on the console. Use the table below. In each cell is the price for a single person.

The output should look like that: `Total price: {price}`. The price should be formatted to the second decimal point.

	Friday	Saturday	Sunday
Students	8.45	9.80	10.46
Business	10.90	15.60	16
Regular	15	20	22.50















There are also **discounts** based on some conditions:

- Students if the group is bigger than or equal to 30 people you should reduce the total price by 15%
- Business if the group is bigger than or equal to 100 people 10 of them can stay for free
- Regular if the group is bigger than or equal to 10 and less than or equal to 20 reduce the total price by 5%

Note: You should reduce the prices in that EXACT order.

### **Examples**

Input	Output
30, "Students", "Sunday"	Total price: 266.73
40, "Regular", "Saturday"	Total price: 800.00

## 3. Leap Year

Write a JS function to check whether a year is a leap. Leap years are either divisible by 4 but not by 100 or are divisible by **400**. The **output** should be following:

• If the year is a leap, print: "yes"

• Otherwise, print: "no"

## **Examples**

Input	Output
1984	yes
2003	no
4	yes

## 4. Print and Sum

Write a function that displays numbers from given start to given end and their sum. The input comes as two number parameters. Print the result like the examples below:

Input	Output
5, 10	5 6 7 8 9 10 Sum: 45
0, 26	0 1 2 26 Sum: 351









50, 60	50 51 52 53 54 55 56 57 58 59 60
	Sum: 605

# 5. Multiplication Table

You will receive a number as a parameter. Print the 10 times table for this number. See the examples below for more information.

### **Output**

**Print** every row of the table in the following format:

{number} X {times} = {product}

### **Constraints**

The number will be an **integer** will be in the interval [1...100]

### **Examples**

Input	Output	Input	Output
5	5 X 1 = 5 5 X 2 = 10	2	2 X 1 = 2 2 X 2 = 4
	5 X 3 = 15		2 X 3 = 6
	5 X 4 = 20 5 X 5 = 25		2 X 4 = 8 2 X 5 = 10
	5 X 6 = 30		2 X 3 = 10 2 X 6 = 12
	5 X 7 = 35		2 X 7 = 14
	5 X 8 = 40 5 X 9 = 45		2 X 8 = 16 2 X 9 = 18
	5 X 9 = 45 5 X 10 = 50		2 X 9 = 18 2 X 10
			= 20

# 6. Sum Digits

Write a function, which will be given a single number. Your task is to find the sum of its digits.

# **Examples**

Input	Output
245678	32
97561	28
543	12

## 7. Chars to String

Write a function, which receives 3 parameters. Each parameter is a single character. Combine all the characters into one string and print it on the console.















### **Examples**

Input	Output
'a', 'b', 'c'	abc
'%', '2', 'o'	%20
'1', '5', 'p'	15p

### 8. Reversed Chars

Write a program that takes 3 parameters (characters) and prints them in reversed order with a space between them.

### **Examples**

Input	Output
'A', 'B', 'C'	СВА
'1', 'L', '&'	& L 1

### 9. Fruit

Write a function that calculates how much money you need to buy fruit. You will receive a string for the type of fruit you want to buy, a number for weight in grams, and another number for the price per kilogram.

Print the following text on the console:

`I need \${money} to buy {weight} kilograms {fruit}.`

Print the weight and the money **rounded** to two decimal places.

The **input** comes as **three arguments** passed to your function.

The **output** should be printed on the console.

Input	Output
'orange', 2500, 1.80	I need \$4.50 to buy 2.50 kilograms orange.

Input	Output
'apple', 1563, 2.35	I need \$3.67 to buy 1.56 kilograms apple.













### 10. Same Numbers

Write a function that takes an integer number as an input and check if all the digits in a given number are the same or not.

Print on the console true if all numbers are the same and false if not. On the next line print the sum of all digits.

The **input** comes as an integer number.

The **output** should be printed on the console.

### **Examples**

Input	Output
2222222	true 14

Input	Output
1234	false
	10

#### **Road Radar** 11.

Write a function that determines whether a driver is within the speed limit. You will receive the speed and the area. Each area has a different limit:

- On the motorway, the limit is 130 km/h
- On the interstate, the limit is 90 km/h
- In the city, the limit is 50 km/h
- Within a residential area, the limit is 20 km/h

If the driver is within the limits, there should be a printed speed and the speed limit.

`Driving {speed} km/h in a {speed limit} zone`

If the driver is over the limit, however, your function should print the severity of the infraction and the difference in speeds.

`The speed is {difference} km/h faster than the allowed speed of {speed limit} -{status}`

For speeding up to 20 km/h over the limit, the status should be speeding.

For speeding up to 40 km/h over the limit, the status should be excessive speeding.

For anything else, **status** should be **reckless driving**.

The input comes as 2 string parameters. The first element is the current speed (number), the second element is the area.

The **output** should be printed on the console.

Input	Output
40, 'city'	Driving 40 km/h in a 50 zone
21, 'residential'	The speed is 1 km/h faster than the allowed speed of 20 - speeding
120, 'interstate'	The speed is 30 km/h faster than the allowed speed of 90 - excessive speeding













200, 'motorway'	The speed is 70 km/h faster than the allowed speed of 130 -
	reckless driving

#### **Cooking by Numbers 12.**

Write a program that receives 6 parameters which are a number and a list of five operations. Perform the operations sequentially by starting with the input number and using the result of every operation as a starting point for the next one. Print the result of every operation in order. The operations can be one of the following:

- chop divide the number by two
- dice square root of a number
- spice add 1 to the number
- bake multiply number by 3
- fillet subtract 20% from the number

The input comes as 6 string elements. The first element is the starting point and must be parsed to a number. The remaining 5 elements are the names of the operations to be performed.

The **output** should be printed on the console.

### **Examples**

Input	Output
'32', 'chop', 'chop', 'chop', 'chop'	16 8 4 2 1

Input	Output
'9', 'dice', 'spice', 'chop', 'bake', 'fillet'	3 4 2 6 4.8

# 13. Array Rotation

Write a function that receives an array and the number of rotations you have to perform.

Note: Depending on the number of rotations, the first element goes to the end.

## Output

Print the resulting array elements separated by a single space.

Input	Output
[51, 47, 32, 61, 21], 2	32 61 21 51 47
[32, 21, 61, 1], 4	32 21 61 1













[2, 4, 15, 31], 5	4 15 31 2
-------------------	-----------

## **Print Every N-th Element from an Array**

The input comes as two parameters – an array of strings and a number. The second parameter is N – the step.

The output is every element on the N-th step starting from the first one. If the step is 3, you need to return the 1-st, the **4-th**, the **7-th** ... and so on, until you reach the end of the array.

The **output** is the **return** value of your function and must be an **array**.

### **Example**

Input	Output
['5', '20', '31', '4', '20'],	['5', '31', '20']

Input	Output
['dsa', 'asd', 'test', 'tset'], 2	['dsa', 'test']

Input	Output
['1', '2', '3', '4', '5'],	['1']

### **Hints**

**Return all the elements** with **for** loop, **incrementing** the **loop variable** with the value of the **step** variable.

#### **List of Names 15.**

You will receive an array of names. Sort them alphabetically in ascending order and print a numbered list of all the names, each on a new line.

## **Example**

Input	Output
["John",	1.Bob
"Bob",	2.Christina
"Christina",	3.Ema
"Ema"]	4.John

### **Hints**

The **sort** function rearranges the array in ascending order

#### **Sorting Numbers 16.**

Write a function that sorts an array of numbers so that the first element is the smallest one, the second is the biggest one, the third is the second smallest one, the fourth is the second biggest one, and so on.

**Return** the resulting array.

Toput	Output
Input	Output













### 17. Reveal Words

Write a function, which receives two parameters.

The first parameter will be a string with some words **separated by** ', '.

The second parameter will be a string that contains templates containing '\*'.

Find the word with the same length as the template and replace it.

### **Example**

Input	Output
'great', 'softuni is ***** place for learning new programming languages'	softuni is great place for learning new programming languages
'great, learning', 'softuni is ***** place for ****** new programming languages'	softuni is great place for learning new programming languages

#### **Modern Times of #(HashTag)** 18.

The input will be a single string.

Find all special words starting with #. If the found special word does not consist only of letters, then it is invalid and should not be printed.

Finally, print out all the special words you found without the label (#) on a new line.

## **Example**

Input	Output
'Nowadays everyone uses # to tag a #special word in #socialMedia'	special socialMedia
'The symbol # is known #variously in English-speaking #regions as the #number sign'	variously regions number

#### **String Substring 19.**

The input will be given as **two** separated strings (a **word** as a first parameter and a **text** as a second).

Write a function that checks given text for containing a given word. The comparison should be case insensitive. Once you find a match, **print** the word and **stop** the program.

If you don't find the word print: "{word} not found!"













# **Example**

Input	Output
'javascript', 'JavaScript is the best programming language'	javascript
'python', 'JavaScript is the best programming language'	python not found!

#### **Pascal-Case Splitter** 20.

You will receive a single string.

This string is written in PascalCase format. Your task here is to split this string by every word in it.

Print them joined by comma and space.

Input	Output
'SplitMeIfYouCanHaHaYouCantOrYouCan'	Split, Me, If, You, Can, Ha, Ha, You, Cant, Or, You, Can
'HoldTheDoor'	Hold, The, Door
'ThisIsSoAnnoyingToDo'	This, Is, So, Annoying, To, Do















