EX1)

Client-side sorting

Suppose you have a table of data (at least 3 columns, one of String type, one of Number type and one of Date objects presented as string on the html table)
One example will be showing the employees table with Name, Salary and Birthday as the columns. From the example I have shown below, you must add a column that represent a date (birthday or first day at work, any date column you wish)

Company	Contact	Country	Salary
Alfreds Futterkiste	Maria Anders	Germany	1000
Centro comercial Moctezuma	Francisco Chang	Mexico	400
Ernst Handel	Roland Mendel	Austria	2000
Island Trading	Helen Bennett	UK	500
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada	650
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy	200

When I click on one of the table's columns, the whole table should be sorted according to the column I clicked.
From the example above if I click the **Company** column the whole table will be sorted ascending according to the Company column. If I click **Company** column again you are going to sort the table descending (only if I click it for the second time in a row).
If I click on Salary column, you are going to sort the table ascending according to the salary.

It is important to right only ONE SORTING FUNCTION which will take as arguments the array that you need to sort, the column that is going to sort on and a Boolean value that signifies whether it will sort ascending or descending.
Basically, the function signature must be something like:

```
const sortData = (array, key, asc=true) => {
 // sort the array
}
```

Do not write different sorting functions for each column, if we add a new column later, I do not want to change the sorting function or add new lines to handle the new column.
Your sorting function should be able to handle new columns.

**Optional
Show a simple icon next to the column name that signifies whether you have sorted ascending or descending. If you have sorted ascending show an arrow going up, if you have sorted descending show an arrow going down.

If the exercise is not clear, have a look at a live demo of what you have to implement
http://webapplayers.com/inspinia_admin-v2.9/table_data_tables.html

Do not worry much about CSS styling, it is enough for you to just define borders of the html table.

EX2) **Optional

Theory

Explain how Stack, Heap, Web APIs, Event Loop and Call-back Queue are linked together in JavaScript.

Do not explain each of them separately, explain shortly what they are and how they are combined.

This exercise is optional, you still get full points if you solve the first exercise, but you get extra points for the second one.

It is extremely important for you to know how to answer this question even if you do not include it in your assignment.