Functions

DEADLINE: 31/03/2021

FOLDER STRUCTURE

```
FL1_HW10/*
    └─ homework/*
        └─ index.html*
        └─ index.js*
        └─ .eslintrc.js
```

* - required

TASK

Task #1
Write a function - *isEquals*
It should accept two arguments and returns **true** if first one value equals second one or **false** otherwise.
Tip: no need for if/else clause nor ternary operator
For example:

```
isEquals(3, 3) // => false
```

Task #2
Write a function - *numberToString*
It should accept one argument as a number and return it as a string
Tip: Don't worry about incoming number – it's always valid
For example:

```
numberToString(1258) // => '1258'
```

Task #3
Write a function - *storeNames*
It should accept an **arbitrary** number of strings and return an array of that strings
For example:

```
storeNames('Tommy Shelby', 'Ragnar Lodbrok', 'Tom Hardy')
// => ['Tommy Shelby', 'Ragnar Lodbrok', 'Tom Hardy']
```

Task #4
Write a function - *getDivision*
It should accept two arguments as numbers and return their division. But the function *never returns a value smaller than 1*. If second parameter is greater than first one, function will change their order.

Tip: consider reusing *isBigger* function
For example:

```
getDivision(4, 1) // => 4
getDivision(2, 8) // => 4
```

Task #5
Write a function - *negativeCount*
It should accept an array of numbers and return the count of negative values from the array.
For example:

```
negativeCount([4, 3, 2, 9])   // => 0
negativeCount([0, -3, 5, 7])  // => 1
```

Task #6
Write a function – *letterCount*
It accepts two string arguments and returns an integer of the count of occurrences the 2nd argument is found in the first one.
If no occurrences can be found, a count of 0 should be returned.
For example:

```
letterCount("Marry", "r") // => 2
letterCount("Barny", "y") // => 1
letterCount("", "z")      // => 0
```

Task #7
Our basketball team (**x** – our team) completed the championship. The result of each match look like **"x:y"**.
Results of all matches are recorded in the collection like this: **["95:74", "107:107", "99:110", ...]**

Write a function – *countPoints*
It should accept a collection of football games scores and count the points of our team in the championship.
Rules for counting points for each match:

- if x > y - 3 points
- if x < y - 0 point
- if x = y - 1 point

Tip: there are 8 matches in the championship

- 0 <= x <= 4
- 0 <= y <= 4
- Consider reusing of *isBigger* function

For example:

```
countPoints(['100:90', '110:98', '100:100', '95:46', '54:90', '99:44', '90:90', '111:100']) // => 17
```

RESTRICTIONS

- Usage of Math object is forbidden;

BEFORE SUBMIT

- Remove all unnecessary files that you might have included by mistake
- Verify that all functionality is implemented according to requirements
- Make sure you code is well-formatted, and validated via validator (w3org Markup Validation Service)
- Add comments if the code is difficult to understand
- Fix warnings/errors in the browser console
- Verify that the name of the folders and files meet the requirements
- Make sure there are no errors/warnings in the browser console
- Run the linter and fix all warnings and errors.

HOW TO

Use linter :

- In order to use npm package manager you should install nodejs (https://nodejs.org/)
- Install eslint to check your code (npm install -g eslint)
  - open a terminal(or cmd)
  - run eslint (i.e. eslint ./js/task1.js)

  Code should be without ‘errors’

SUBMIT

The folder should be uploaded to gitlab repository '**FL-HU1**' into **master** branch