1. Write a function to check whether the input is a string or not.

```
"My random string" -> true
12 -> false
```

```
function isString(input) {
    if (typeof input === 'string') {
        return true;
    } else {
        return false;
    }
}

console.log(isString(2)); // false
console.log(isString(' ')); //true
console.log(isString('bit')); //true
console.log(isString([1, 2, 4, 0])); //false
```

2. Write a function to check whether a string is blank or not.

```
"My random string" -> false
" " -> true
12 -> false
false -> false
```

```
function isBlank(input) {
    if (typeof input === 'string' && input.length === 0) {
        return true;
    } else {
        return false;
    }
}

console.log(isBlank(''));
console.log(isBlank('abc'));
```

3. Write a function that concatenates a given string n times (default is 1).

```
"Ha" -> "Ha"
"Ha", 3 -> "HaHaHa"
```

```
function repeat(str, count) {
    if (typeof count == "undefined") {
        count = 1;
    }

    if(count < 1) {
        return "";
    }
    var res = "";
    for (var i = 0; i < count; i++) {
```

```
            res += str;
        }

        return res;
    }

    console.log(repeat('Ha'));
    console.log(repeat('Ha', 2));
    console.log(repeat('Ha', 4));
    console.log(repeat('Ha', -5));
```

4. Write a function to count the number of letter occurrences in a string.

   `"My random string", "n" -> 2`

```
function countOcurances(str, letter) {
    var count = 0;
    for (var i = 0; i < str.length; i++) {
        var element = str[i];
        if (element === letter) {
            count++;
        }
    }
    return count;
}

var occ = countOcurances("My random string", "n");
console.log();
```

5. Write a function to find the position of the first occurrence of a character in a string. The result should be in human numeration form. If there are no occurrences of the character, the function should return -1.

```
function positionOfFirst(inputString, character) {

    for (var index = 0; index < inputString.length; index++) {
        var currentCharacter = inputString[index];
        if (currentCharacter === character) {
            return index + 1;
        }
    }
    return -1;
}

var characterFirstPosition = positionOfFirst("This is input string", "k");
console.log(characterFirstPosition);
```

6. Write a function to find the position of the last occurrence of a character in a string. The result should be in human numeration form. If there are no occurrences of the character, function should return -1.

```
function positionOfLast(inputString, character) {
```

```
    var lastIndex = inputString.length - 1;

    for (var index = lastIndex; index >= 0; index--) {
        var currentCharacter = inputString[index];
        if (currentCharacter === character) {
            return index + 1;
        }
    }
    return -1;
}

var characterFirstPosition = positionOfLast("This is input string", "i");
console.log(characterFirstPosition);
```

7. Write a function to convert string into an array. Space in a string should be represented as “null” in new array.

```
"My random string" -> ["M", "y", null, "r", "a"...]
"Random" -> ["R", "a", "n", "d", "o", "m"]
```

```
function stringIntoArray(inputString) {
    var space = " "
    var outputArray = [];

    for (var index = 0; index < inputString.length; index++) {
        var element = inputString[index];
        if (element === space) {
            outputArray[index] = null;
        } else {
            outputArray[index] = element;
        }
    }

    return outputArray;
}

var output = stringIntoArray("This is simple string");
console.log(output);
```

8. Write a function that accepts a number as a parameter and check if the number is prime or not.
Note: A prime number (or a prime) is a natural number greater than 1 that has no positive divisors other than 1 and itself.

```
7 -> true
```

```
function isPrimeNumber(num) {

    if (num === 1) {
        return false;
    } else if (num === 2) {
        return true;
```

```
    }

    for (var x = 2; x < num; x++) {
        if (num % x === 0) {
            return false;
        }
    }

    return true;
}

var isPrime = isPrimeNumber(37);
console.log(isPrime);
```

9. Write a function that replaces spaces in a string with provided separator. If separator is not provided, use “-” (dash) as the default separator.

```
"My random string", "_" -> "My_random_string"
"My random string", "+" -> "My+random+string"
"My random string" -> "My-random-string"
```

```
function replaceSpace(string, sep) {
    var output = "";
    sep = sep || "-";
    for (var index = 0; index < string.length; index++) {
        var element = string[index];
        if (element === " ") {
            output += sep;
        } else {
            output += element;
        }
    }

    return output;
}

var result = replaceSpace("Random string with space", "%");
console.log(result);
```

10. Write a function to get the first n characters and add “...” at the end of newly created string.

```
"This is long string", 7 -> "This is..."
```

```
function stringChop(input, size) {
    var output = "";

    for (var index = 0; index < input.length; index++) {
        var element = input[index];
        output += element;
        if (index === (size - 1)) {
            output += "...";
            break;
        }
```

```
    }

    return output;
}

var result = stringChop("This is long string", 7);
console.log(result);
```

11. Write a function to convert an array of strings into an array of numbers. Filter out all non-numeric values.

```
["1", "21", undefined, "42", "1e+3", Infinity] -> [1, 21, 42, 1000]
```

```
function filterNonNumbers(inputArray) {
    var numArray = [];
    var positionFixer = 0;

    for (var index = 0; index < inputArray.length; index++) {
        var element = inputArray[index];
        var number = parseFloat(element);

        if (!isNaN(number) && isFinite(number)) {
            numArray[index - positionFixer] = number;
        } else {
            positionFixer++;
        }
    }

    return numArray;
}

var output = filterNonNumbers(["1", "21", undefined, "42", "1e+3", Infinity]);
console.log(output);
```

12. Write a function to calculate how many years there are left until retirement based on the year of birth. Retirement for men is at age of 65 and for women at age of 60. If someone is already retired, a proper message should be displayed.

```
function calculateAge(yearOfBirth) {
    var age = parseInt(2017 - yearOfBirth);
    return age;
}

function isInReitrement(age, gender) {
    gender = gender || "m"

    if (gender === "m") {
        return age >= 65;
    } else if(gender === "f") {
        return age >= 60;
    }
}

function untilRetirement(yearOfBirth, gender) {
```

```
    var age = calculateAge(yearOfBirth);
    var gender = gender || "m";

    if (isInReitrement(age, gender)) {
        return "Person is already in retirement"
    }

    if (gender === "m") {
        return 65 - age;
    } else {
        return 60 - age;
    }
}

var output = untilRetirement(1989, "f")
console.log(output);
```

13. Write a function to humanize a number (formats a number to a human-readable string) with the correct suffix such as 1st, 2nd, 3rd or 4th.

```
1 -> 1st
11 -> 11th
```

```
    function humanizeNumber(num) {
        if (typeof num == "undefined") {
            return;
        }

        if (num % 100 >= 11 && num % 100 <= 13) {
            return num + "th";
        }

        switch (num % 10) {
            case 1:
                return num + "st";
            case 2:
                return num + "nd";
            case 3:
                return num + "rd";
        }
        return num + "th";
    }

    // Output
    console.log(humanizeNumber());
    console.log(humanizeNumber(1));
    console.log(humanizeNumber(8));
    console.log(humanizeNumber(301));
    console.log(humanizeNumber(402));
```