# Coding Challenge 2020Q1 Problem Set

Handling data transformations using map, filter, reduce
and functional programming techniques

Team Name (Optional):

Members:

February 7, 2020

**For each problem, follow the directions stated on the description of the problem.** The programming language for this challenge will be **javascript/typescript**. Please note that the correct solution may come in the form of either providing the code snippet that answers the problem description or providing the console output for the given code block.

# Reference Page: Map, Filter, Reduce

If you are already familiar with the map, filter, reduce functions please feel free to skip this page.

## map()

The Array.map method creates a new array populated with the results of calling a provided function on every element in the calling array.[1]

**Syntax**:

```
array.map(function(currentValue, index, arr), thisValue)
```

**Example**:

```
let arr = [1, 5, 6]
arr.map(x => x*2) // returns [2, 10, 12]
```

## filter()

The Array.filter method creates a new array with all elements that pass the test implemented by the provided function.[1]

**Syntax**:

```
let newArray = arr.filter(callback(element[, index[, array]])[, thisArg])
```

**Example**:

```
let arr = [1, 2, 3, 4, 5, 6]
arr.filter(x => x % 2 == 0) // returns [2, 4, 6]
```

## reduce()

The Array.reduce method executes a reducer function (that you provide) on each element of the array, resulting in a single output value.[1]

**Syntax**:

```
arr.reduce(callback(accumulator, currentValue[, index[, array]] )[, initialValue])
```

**Example**:

```
let arr = [1, 2, 3, 6]
arr.reduce((accum, curr) => (accum + curr), 0) // returns 12 (the sum)
```

## ES6 Spread Operator ...

Spread syntax `(...)` allows an iterable such as an array expression, string, or object to be expanded in places where zero or more arguments (for function calls) or elements (for array literals) are expected, or an object expression to be expanded in places where zero or more key-value pairs (for object literals) are expected.[1]

**Example:**

```
var user = { name:  'person', age:  23 }
var result = {...user, likable:  true}
console.log(result) // { name:  ’person’, age:  23, likable:  true }
```

[1] https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array

# Part 1: Map, filter, reduce on iterable data structures

1. **(1 pt.)** Using `map`, finish the return statement in the given function which should take in a list of numbers and **return a list of numbers where each element in the list is multiplied by 10**.
   **Expected Result:**
   `nums => [1,2,3,4,5,6,7,8,9,10]`
   `q1(nums) => [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]`

```
function q1(nums) : number[] {
    // Your Code Here
    return

}
```

2. **(1 pt.)** Using `reduce`, write a function **returns the average rating** of the movies in the catalog.
**Expected Result:**
`movies => (moviesCatalog :  Movies[])`
`q2(movies) => 4.2`

```
const moviesCatalog : Movie[] = [{
    "id": 70111470,
    "title": "Die Hard",
    "boxart": "http://cdn-0.nflximg.com/images/2891/DieHard.jpg",
    "rating": 4.0
}, {
    "id": 654356453,
    "title": "Bad Boys",
    "boxart": "http://cdn-0.nflximg.com/images/2891/BadBoys.jpg",
    "rating": 5.0
}, {
    "id": 65432445,
    "title": "The Chamber",
    "boxart": "http://cdn-0.nflximg.com/images/2891/TheChamber.jpg",
    "rating": 4.0
}, {
    "id": 64248315,
    "title": "The Wolf of Wall Street",
    "boxart": "http://cdn-0.nflximg.com/images/2891/WolfOnWallStr.jpg",
    "rating": 3.0
}, {
    "id": 64245743,
    "title": "The GodFather",
    "boxart": "http://cdn-0.nflximg.com/images/2891/TheGodFather.jpg",
    "rating": 5.0
}];
class Movie {
    public id : number;
    public title: String;
    public boxart: String;
    public rating: number;
}

function q2(movies: Movie[]) : number {
    // Your Code Here
    return

}
```

3. **(1 pt.)** Using a chain of maps, filters, or reduces. Write a function that returns a list of movies that:
1. **(1/3 pt.)** starts with ’The’ (account for case insensitivity)
2. **(1/3 pt.)** has a rating > 3
3. **(1/3 pt.)** append property imageFileName which is the base file name coming from the boxart property, i.e. {..., boxart: http://cdn-0.nflximg.com/images/2891/DieHard.jpg} => {..., imageFileName: DieHard.jpg}.
*Note that the imageFileName comes from the string after the last slash (/) in the boxart property

Please refer to the movie data used in problem 2.

```
function q3(movies: Movie[]) : number {
    // Your Code Here
    return




}
```

## Part 2: Function Composition and Currying

4. **(1 pt.)** What is the output to the console when the following code is executed?
***Note**: pipe, map, and filter have been re-written to be used for function composition.

```
const pipe = (...fns) => args => fns.reduce((arg, f) => f(arg), args)
const map = fn => arr => arr.map(fn)
const filter = fn => arr => arr.filter(fn)
const classmates = [{ name: "Charlie",
    age: 20,
    sex: "male",
    likabilityScore: 95,
    isKind: true
}, {
    name: "Riley",
    age: 23,
    sex: "female",
    likabilityScore: 78,
    isKind: false
}, {
    name: "Alex",
    age: 25,
    sex: "female",
    likabilityScore: 77,
    isKind: true
}, {
    name: "Skyler",
    age: 21,
    sex: "male",
    likabilityScore: 86,
    isKind: false}]
const mysteryFunction1 = arr => map(x => x.isKind ?
{...x, likabilityScore: x.likabilityScore + 4} : x)(arr)
const mysteryFunction2 = arr => filter(x => x.likabilityScore >= 80)(arr)

var closeFriends = pipe(
    filter(x => x.age >= 21),
    mysteryFunction1,
    mysteryFunction2,
    map(x => (x.name))
)(classmates)

console.log(closeFriends)
```

5. **(1 pt.)** What is the output to the console when the following code is executed? [2]

```
export const pipe = (...fns) => x => fns.reduce((y, f) => f(y), x);
export const flip = fn => a => b => fn(b)(a);
export const trace = value => label => {
    console.log(`${ label }: ${ value }`);
    return value;
};

const flippedTrace = flip(trace);
const g = n => n + 1;
const f = n => n * 2;

const h = pipe(
    g,
    flippedTrace('after g'),
    f,
    flippedTrace('after f')
);

h(20);
```

[2] https://medium.com/javascript-scene/curry-and-function-composition-2c208d774983

## Tiebreaker

**Please ignore this problem unless further instruction to do so is given.**

6. **(1 pt.)** What is the output to the console when the following code is executed?
***Note**: `Array.pop()` removes the last element from an array and returns that value to the caller.[1]

```
var sequence = [1, 7, 12, 8, 2, 0, 3, 9, 6, 10, 5, 11, 4]
var wordScramble = [
    ['A', 'Z', 'Y', 'M', 'C',' X', 'N', 'D', 'O', 'P', 'T', 'G', 'Q'],
    ['M', 'D', 'T', 'R', 'V',' S', 'L', 'P', 'I', 'C', 'X', 'O', 'N'],
    ['N', 'E', 'R', 'X', 'N', 'D', 'P', 'R', 'J', 'T', 'C', 'V', 'U'],
    ['H', 'X', 'M', 'B', 'E', 'I', 'N', 'P', 'J', 'X', 'E', 'V', 'O'],
    ['N', 'X', 'J', 'T', 'O', 'P', 'C', 'G', 'H', 'L', 'S', 'A', 'K'],
    ['O', 'F', 'L', 'D', 'V', 'B', 'R', 'W', 'R', 'O', 'H', 'F', 'B'],
    ['K', 'J', 'U', 'M', 'E', 'T', 'Y', 'E', 'B', 'M', 'C', 'V', 'I'],
    ['M', 'X', 'R', 'A', 'Z', 'I', 'T', 'F', 'K', 'V', 'U', 'M', 'X'],
    ['U', 'S', 'U', 'J', 'V', 'P', 'U', 'B', 'Q', 'X', 'I', 'N', 'I'],
    ['R', 'L', 'I', 'H', 'C', 'Y', 'D', 'J', 'N', 'G', 'T', 'R', 'E'],
    ['F', 'K', 'C', 'R', 'M', 'U', 'I', 'X', 'Z', 'E', 'R', 'L', 'I'],
    ['W', 'M', 'O', 'S', 'D', 'F', 'Y', 'T', 'P', 'U', 'L', 'C', 'B'],
    ['N', 'Y', 'J', 'K', 'I', 'R', 'P', 'L', 'F', 'O', 'X', 'B', 'U']
]

var result: any = wordScramble.reduce((accum, curr) =>
                                accum + curr[sequence.pop()],
                                '')
console.log(result)
```