## JS Advanced: Exam 18.11.2018

# **Problem 2. SoftUniFy (Unit Testing)**

You are given the following JavaScript class:

```
app.js
class SoftUniFy {
   constructor() {
        this.allSongs = {};
   downloadSong(artist, song, lyrics) {
        if (!this.allSongs[artist]) {
            this.allSongs[artist] = {rate: 0, votes: 0, songs: []}
        this.allSongs[artist]['songs'].push(`${song} - ${lyrics}`);
        return this;
   playSong(song) {
        let songArtists = Object.keys(this.allSongs).reduce((acc, cur) => {
            let songs = this.allSongs[cur]['songs']
                .filter((songInfo) => songInfo
                    .split(/ - /)[0] === song);
            if(songs.length > 0){
                acc[cur] = songs;
            return acc;
        }, {});
        let arr = Object.keys(songArtists);
        let output = "";
        if(arr.length > 0){
            arr.forEach((artist) => {
                output += `${artist}:\n`;
                output += `${songArtists[artist].join('\n')}\n`;
            });
        } else {
            output = `You have not downloaded a ${song} song yet. Use SoftUniFy's function
downloadSong() to change that!`
```



© <u>Software University Foundation</u>. This work is licensed under the <u>CC-BY-NC-SA</u> license.

















```
return output;
}
get songsList() {
    let songs = Object.values(this.allSongs)
        .map((v) => v['songs'])
        .reduce((acc, cur) => {
            return acc.concat(cur);
        }, []);
   let output;
    if (songs.length > 0) {
        output = songs.join('\n');
    } else {
        output = 'Your song list is empty';
    return output;
}
rateArtist() {
    let artistExist = this.allSongs[arguments[0]];
   let output;
    if (artistExist) {
        if (arguments.length === 2) {
            artistExist['rate'] += +arguments[1];
            artistExist['votes'] += 1;
        }
        let currentRate = (+(artistExist['rate'] / artistExist['votes']).toFixed(2));
        isNaN(currentRate) ? output = 0 : output = currentRate;
    } else {
        output = `The ${arguments[0]} is not on your artist list.`
   return output;
```

## **Functionality**

An **instance** of the **SoftUniFy** class should support the following operations:

















- Should contain allSongs property that is initialized as an empty object.
- downloadSong(artist, song, lyrics) adds the given information to the allSongs in format: artist: {rate: 0, votes: 0, songs: []}.

**songs** property, should contain **all songs** from the current artist in format:

```
[song1 - song1Lyrics, song2 - song2Lyrics....]
```

should return the entire Class.

• playSong(song) searches all already downloaded songs and returns them in format:

```
artist:
song - lyrics
```

...

If we do not have at least one downloaded song or the one we have given is not found should return the following string:

- `You have not downloaded a {song} song yet. Use SoftUniFy's function downloadSong() to change that!`
- songsList() gets all already downloaded songs in format:

```
song1 - song1Lyrics
song2 - song2Lyrics
song3 - song3Lyrics
```

If we do not have at least one downloaded song return the following string:

```
`Your song list is empty`
```

rateArtist() sums the values of all votes on the current artist and return the average rate.

Keep in mind that, this function has two overloads!

#### TODO

Using **Mocha** and **Chai** write **JS unit tests** to test the entire functionality of the **SoftUniFy** class. You may use the following code as a template:

Wrap all the tests in one describe statement and submit it..



© <u>Software University Foundation</u>. This work is licensed under the <u>CC-BY-NC-SA</u> license.















```
it("TODO ...", function() {
         // TODO: ...
      });
    // TODO: ...
});
```

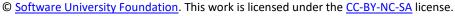
Don't forget to require the chai library!

## **Examples**

```
Input 1 (downloadSong)
let sofunify = new SoftUniFy();
sofunify.downloadSong('Eminem', 'Venom', 'Knock, Knock let the devil in...');
sofunify.downloadSong('Eminem', 'Phenomenal', 'IM PHENOMENAL...');
sofunify.downloadSong('Dub Fx', 'Light Me On Fire', 'You can call me a liar.. ');
console.log(sofunify.allSongs);
```

```
Output 1 (downloadSong)
{ Eminem:
    { rate: 0,
      votes: 0,
      songs:
        [ 'Venom - Knock, Knock let the devil in...',
           'Phenomenal - IM PHENOMENAL...'] },
  'Dub Fx':
   { rate: 0,
      votes: 0,
      songs: [ 'Light Me On Fire - You can call me a liar.. ' ] } }
```



















```
Input 1 (rateArtist)
let sofunify = new SoftUniFy();
console.log(sofunify.rateArtist('Eminem'));
console.log(sofunify.rateArtist('Eminem', 50));
```

#### Output 1 (rateArtist)

The Eminem is not on your artist list.

The Eminem is not on your artist list.

### Hints

If at first look you do not understand exactly how a certain function works, just do your own tests by calling it to find out exactly what result returns...

# GOOD LUCK... C(:



