# JS Advanced: Retake Exam 5 September 2017

Problems for exam preparation for the [“JavaScript Advanced” course @ SoftUni](https://softuni.bg/courses/javascript-advanced). Submit your solutions in the SoftUni judge system at <https://judge.softuni.bg/Contests/756/>.

# Problem 2. String Builder (Unit Testing)

You are given the following **JavaScript class**:

|  |
| --- |
| string-builder.js |
| **class** StringBuilder {  constructor(string) {  **if** (string !== ***undefined***) {  StringBuilder.*\_vrfyParam*(string);  **this**.**\_stringArray** = Array.from(string);  } **else** {  **this**.**\_stringArray** = [];  }  }   append(string) {  StringBuilder.*\_vrfyParam*(string);  **for**(**let** i = 0; i < string.**length**; i++) {  **this**.**\_stringArray**.push(string[i]);  }  }   prepend(string) {  StringBuilder.*\_vrfyParam*(string);  **for**(**let** i = string.**length** - 1; i >= 0; i--) {  **this**.**\_stringArray**.unshift(string[i]);  }  }   insertAt(string, startIndex) {  StringBuilder.*\_vrfyParam*(string);  **this**.**\_stringArray**.splice(startIndex, 0, ...string);  }   remove(startIndex, length) {  **this**.**\_stringArray**.splice(startIndex, length);  }   **static** *\_vrfyParam*(param) {  **if** (**typeof** param !== **'string'**) **throw new TypeError**(**'Argument must be string'**);  }   toString() {  **return this**.**\_stringArray**.join(**''**);  } } |

### Functionality

The above code defines a **class** that holds **characters** (strings with length 1) in an array. An **instance** of the class should support the following operations:

* Can be **instantiated** with a passed in **string** argument or **without** anything
* Functionappend(string) – **converts** the passed in **string** argument to an **array** and adds it to the **end** of the storage
* Function **prepend**(**string**) – **converts** the passed in **string** argument to an **array** and adds it to the **beginning** of the storage
* FunctioninsertAt(string, index) – **converts** the passed in **string** argument to an **array** and adds it at the **given** index (there is **no** need to check if the index is in range)
* Functionremove(startIndex, length) – **removes** elements from the storage, starting at the given index (**inclusive**), **length** number of characters (there is **no** need to check if the index is in range)
* FunctiontoString() – **returns** a string with **all** elements joined by an **empty** string
* All passed in **arguments** should be **strings.** If any of them are **not**, **throws** a type **error** with the following message: "**Argument must be a string**"

### Examples

This is an example how this code is **intended to be used**:

|  |  |  |
| --- | --- | --- |
| Sample code usage |  | Corresponding output |
| **let** str = **new** StringBuilder(**'hello'**); str.append(**', there'**); str.prepend(**'User, '**); str.insertAt(**'woop'**,5 ); **console**.log(str.toString()); str.remove(6, 3); **console**.log(str.toString()); | User,woop hello, there  User,w hello, there |

### Your Task

Using **Mocha** and **Chai** write **JS unit tests** to test the entire functionality of the StringBuilder class. Make sure it is **correctly defined as a class** and instances of it have all the required functionality. You may use the following code as a template:

|  |
| --- |
| describe(**"*TODO* …"**, **function**() {  ***it***(**"*TODO …*"**, **function**() {  *//* ***TODO:*** …  });  *//* ***TODO:*** …  }); |

### Submission

Submit your tests inside a describe() statement.