
TP3 : Chaînes de caractère

Ce TP a pour but de vous familiariser avec la syntaxe du langage JavaScript en implémentant quelques calculs simples.

La documentation MDN de la classe Math peut vous être utile : https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Objets_globaux/Math

Compétences travaillées dans ce TP

- Manipulations de base de chaînes de caractères
- Organisation du code
- Utilisation d'une documentation

Préparation (valable pour tous les prochains TPS)

- 1- Créer un répertoire tp3
- 2- Dans TP3, créer 4 fichiers vides : tp3.html, string.js, stringTest.js et main.js
- 3- Dans tp3.html, charger les fichiers .j dans l'ordre mentionné ci-dessus
- 4- Dans main.js, ajouter une fonction main() vide et l'appel à cette fonction en fin de script

Exercice 1

Implémentation

Dans le fichier string.js, implémentez une fonction :

```
repeat(s, n)
```

qui renvoie une chaîne dont le contenu est celui de la chaîne s répété n fois (n'hésiter pas de se référer à la documentation MDN).

Exemple d'utilisation :

```
console.log(repeat("foo", 3)); // => foofoofoo
```

Test

- 1- Dans le fichier stringTest.js, ajoutez une fonction repeatTests() qui contient l'exemple d'utilisation ci-dessus.

- 2- Modifiez le main pour qu'il appelle repeatTests() et vérifiez le bon fonctionnement de votre fonction.
- 3- Complétez repeatTests() avec d'autres cas de test.
- 4- Changer l'ordre de l'appel des script .js dans le fichier HTML et interpréter le résultat

Exercice 2

En suivant la même méthodologie que précédemment, implémenter et tester une fonction :

```
truncate(s,n)
```

qui tronque la chaîne s à n caractères. La chaîne tronquée doit se terminer par "...". La taille de la chaîne renvoyée doit être inférieure ou égale à n, points de suspension compris.

Exemples d'utilisation :

```
console.log(truncate("unchanged text", 20)); // => unchanged text
console.log(truncate("unchanged text", 14)); // => unchanged text
console.log(truncate("truncated text", 10)); // => truncat...
```

Exercice 3

Un palindrome est un mot dont l'ordre des lettres reste le même qu'on le lise de gauche à droite ou de droite à gauche.

Implémenter et tester une fonction :

```
isPalindrome(s)
```

qui renvoie true si s est un palindrome (false sinon).

Exemples d'utilisation :

```
console.log(isPalindrome("bonjour")); // => false
console.log(isPalindrome("toto")); // => false
console.log(isPalindrome("kayak")); // => true
console.log(isPalindrome("ressasser")); // => true
```

Mêmes consignes que précédemment concernant la séparation en fichiers et les tests.

Exercice 4

Implémenter et testez une fonction :

```
swapCase(s)
```

qui renvoie une chaîne de caractères contenant tous les caractères de s avec une casse inversée (les minuscules sont transformées en majuscules et inversement).

Exemples d'utilisation :

```
console.log(swapCase("HELLO")); // => hello
console.log(swapCase("bye")); // => BYE
console.log(swapCase("GoOd NighT")); // => gOoD nIGHt
```

Indice : utiliser les méthodes toUpperCase() et toLowerCase() de la classe String.