



# LES CHIFFRES ROMAINS – JEST : TDD

CODE KATA – FRAMEWORK DE TESTS UNITAIRES EN **JAVASCRIPT**

## AVANT DE COMMENCER

Faire **une tâche à la fois**. L'important est d'apprendre à travailler incrémentalement. Implémenter les tests avant d'implémenter la fonction testée. **TDD**

**Vous serez évalué sur la qualité de votre code et votre approche TDD.**

## CONVERTIR LES CHIFFRES ARABES EN CHIFFRES ROMAINS

1. L'institut national des normes et de la technologie des États-Unis n'a trouvé aucune autorité capable de décrire si l'année 1999 de vait être écrite sous la forme MDCCCCLXXXVIII, MCMXCIX ou MIM. En raison du manque de normalisation, un ensemble de règles supplémentaires a été fréquemment appliqué au cours des derniers siècles.

I, II, III	1, 2, 3
V	5
X	10
L	50
C	100
D	500
M	1000

2. Un seul symbole de petite valeur peut être soustrait de tout symbole de grande valeur.

3. Les symboles I, X, C et M peuvent être répétés trois fois de suite, mais pas plus. (Ils peuvent apparaître plus de trois fois s'ils apparaissent de manière non séquentielle, comme XXXIX.) V, L et D ne peuvent jamais être répétés.

- 4.
- I peut être soustrait de V et X uniquement.
  - X peut être soustrait de L et C uniquement.
  - C peut être soustrait de D et M uniquement.
  - V, L et D ne peuvent jamais être soustrait.

5. Un nombre écrit en chiffres arabes peut être décomposé de la manière suivante.

Exemple pour 1903, qui est composé de 1, 9, 0 et 3. Pour l'écrire en chiffres romains, chacun des chiffres non nuls doit être traité séparément.

Dans 1903, 1000 = M, 900 = CM et 3 = III. Donc 1903 = MCMIII.

6. TDD : Écrire une fonction qui convertit un nombre en chiffres arabes en chiffres romains.  
`arabicToRoman(int number) : string.`

7. TDD :

Gestion des cas particuliers :

- La fonction doit retourner une chaîne de caractère vide lorsqu'on passe 0 ou un nombre négatif en argument.

- La fonction doit lever une exception pour les nombres au-delà de 3999.

`"Number too high : [number]"`

### LIVRABLES ATTENDUS

1. Ensemble des fichiers des étapes de votre code de production représentant les cycles TDD ou alors le lien vers votre repository github privé où vous m'aurez ajouté en collaborateur afin que moi seul puisse voir l'historique de vos commits. 1 commit par cycle TDD.

2. Le fichier de vos tests unitaires.

Le tout dans un dossier comprimé :

Binôme : `binome-jest-tdd-nom1-nom2.zip`

Seul : `jest-tdd-nom-prenom`

Exemple : `jest-tdd-hassan-aunim.zip`