

# Conventions de codage

## Nommage des variables & fonctions : camelCase

```
1. var myFabulousVariable;  
2. function myWonderfulFunction()
```

## Position des accolades : en dessous de la ligne

```
1. function myWonderfulFunction()  
2. {  
3. }  
4.  
5. if (condition)  
6. {  
7. }
```

→ Éventuellement : possible de ne pas mettre d'accolade si on est sûrs qu'il n'y aura qu'une seule ligne :

```
1. for (var i in values)  
2.     total += values[i];
```

## Toponymie : anglais + mots-clés spécifiques + verbes pour les fonctions

```
1. function getProperty(...)  
2. function setProperty(...)  
3. function hasProperty(...)  
4. function isCompleted(...)  
5. function randomizeArray(...)  
6. function computeCalculation(...)
```

→ On utilise des mots complets plutôt que des abréviations : lisibilité et compréhension

```
1. function calculateDistanceBetween(...) // plutôt que « calcDistBetw »  
2. function randomizeVerticalPositions(...) // plutôt que « randVertPos »
```

→ Chaque variable doit être nommée avec un sens précis, et pas « truc » ou « machin » ni avec des abréviations (sauf dans les formules mathématiques)

---

*On doit lire le code de manière fluide et compréhensive, un code bien écrit n'a presque pas besoin de commentaires (mais on en met quand même !)*

---

Commentaires des fonctions :

```
1. /**
2.  * Calcule la moyenne des valeurs du tableau "values"
3.  */
4. function getAverage(values)
```

Commentaires du code : au dessus de chaque bloc de code, sauf si l'action est absolument évidente

```
1. // Créé et initialise le tableau qui servira à ...
2. var values = [];
3. values.push(32);
4. values.push(54);
5.
6. // Calcule la moyenne des valeurs
7. var average = getAverage(values);
8.
9. console.log(average); // (pas besoin de commentaire ici...)
```

Sauts de ligne : 1 seul à la fois, entre chaque bloc, aucun après ou avant une accolade

```
1. /**
2.  * Calcule la moyenne des valeurs du tableau "values"
3.  */
4. function getAverage(values)
5. {
6.     var total = 0;
7.
8.     for (var i in values)
9.         total += values[i];
10.
11.     return total / values.length;
12. }
```

Espaces : entre chaque opérateur, mais pas après ou avant une parenthèse

```
1. for (var i in values)
2. if (condition1 && condition2)
3. var result = x + y / z;
```