

# JavaScript Memo

Alexandre Bonhomme <hello@alexandrebonhomme.com>

## Les variables

Une variable sert à stocker une information. Il faut créer une variable et lui donner un nom avant de pouvoir l'utiliser. C'est comme une boîte avec une étiquette collée dessus.

```
// Le mot clé "var" permet de créer une nouvelle variable
var nomDeMaVariable;
var nouvelleVariable = "Je suis un texte";
```

Pour utiliser une variable il faut l'appeler par son nom.

```
// affiche "Je suis un texte" dans la console.
console.log(nouvelleVariable);
```

Pour modifier la valeur d'une variable on utilise le signe =.

```
// ajoute 1 à la variable
variableNombre = variableNombre + 1;

// remplace le contenu de la variable par un texte
variableNombre = "nouveau contenu"
```

## Les conditions

Une condition définit l'exécution d'une partie de code. Une condition s'écrit sous la forme d'une expression qui peut être *vraie* ou *fausse*.

```
var nombre = 20;

if (nombre > 10) {
  // instructions à exécuter si la condition est vraie
}
```

Il est possible de définir une seconde portion de code à exécuter si la condition n'est pas vraie grâce au mot clé **else**.

```
if ((nombre > 10) && (nombre < 100)) {
  // instructions à exécuter si la condition est vraie
} else {
  // instructions à exécuter si la condition n'est pas vraie
}
```

On peut décomposer une condition complexe en plusieurs conditions simples grâce aux opérateurs logique **&&** (et), **||** (ou) et **!** (non).

## Les boucles

Une boucle permet de répéter un ensemble d'instructions tant qu'une condition est *vraie*. Une boucle est composée de 3 éléments : une variable compteur (**var i = 0**), une condition d'arrêt (**i < 10**) et un pas d'incrémentation (**i = i + 1**).

```
// affiche 10 fois la valeur de i (eg. 0, 1, 2, etc.)
for (var i = 0; i < 10; i = i + 1) {
  console.log(i);
}
```

Il existe une seconde syntaxe pour écrire une boucle :

```
var i = 0;
while (i < 10) {
  // affiche la valeur de i
  console.log(i);

  // incrémente la valeur de i
  i = i + 1;
}
```

## Les tableaux

Un tableau est une structure de données qui permet de stocker plusieurs informations dans une variable. On obtient la taille d'un tableau grâce à **.length**.

```
// on déclare un tableau avec les "[ ]"
var nomDuTableau = [1, 3, 0, 5];

// affiche chaque élément du tableau sur une nouvelle ligne
for (var i = 0; i < nomDuTableau.length; i = i + 1) {
  console.log(nomDuTableau[i]);
}
```

## Les fonctions

Une fonction permet de rassembler un ensemble d'instructions afin de pouvoir les exécuter plusieurs fois. Elle peut avoir des *paramètres*, qui seront alors utilisés de la même manière que des variables au sein de la fonction, afin d'obtenir du code générique. Une fonction renvoie une valeur qui peut-être stipulée grâce au mot clé **return**. Celle-ci peut-être enregistrée dans une variable afin d'être utilisée ultérieurement. Par défaut la valeur renvoyée est **undefined**.

```
// on déclare en premier lieu la fonction avec ses paramètres (s'il y en a)
function nomDeLaFonction(param1, param2) {
  return param1 + param2;
}

// on peut ensuite utiliser la fonction en lui donnant des valeurs qui seront copiées dans les paramètres
var resultat = nomDeLaFonction(40, 2);
```