**6 types primitifs :**

*- undefined*

Lorsqu’on ne donne pas un argument (ce qu'il y a dans la parenthèse d’une fonction genre console log)

Lorsque la fonction n'est pas faite pour nous retourner une valeur

Lorsqu’elle n'est pas définie

- *booléen*

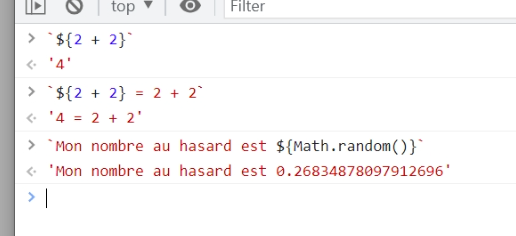
-*number*

-*string*

Chaîne de caractères (toujours entre "")

Il y a 3 sortes de guillemets : les français, les anglais et les backtiks `` sur la même touche que le £ et le µ via (grâce à alt. gr)

Les backtiks ont été rajoutés vers 2015 et permettent d'interpréter les var et du code en les mettant ensemble dans une structure spéciale `${blablabla}` Ce qui est dans ces accolades sera interpréter mais pas le reste, le reste sera toujours un/des string (voir capture d'écran)



-*BigInt*

Pour garantir la validité des grands nombres. Car JS n'est pas fait pour stocker des nombres immenses (qui sont déjà énormes). Le BigInt garanti l'intégrité de ce chiffre mais on parle de centaines de millions ce genre de choses, donc peu de chance de beaucoup en croiser

-*les symboles*

Qui représente un identifiant unique

**3 types non-primitifs**

Ce sont à la base des objets

-null

À +/-le même but qu'undifined avec la différence qu'on veut montrer qu'on a VOLONTAIREMENT mis aucune valeur. Ce n’est rien (même pas zéro, il n'y a pas de valeur)

Donc undefined c'est "on ne sait pas la valeur et on sait pas pourquoi" et null "on sait pas la valeur et c'est fait sciemment"

-object

-function

**Les variables**

Il y a des mots clefs disponibles lors de la création de variables en JS. Ces scopes limiteront la portée des variables.

Du coup, lorsqu'on essaie d'interagir avec une var qui n'est pas accessible depuis là où on est dans notre code, on aura un message d'erreur.

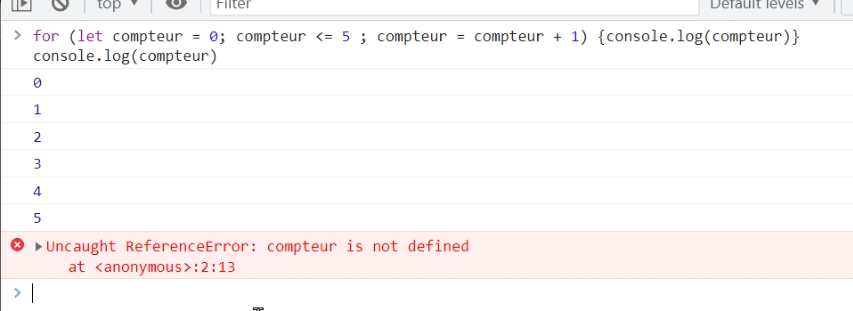
Cela va permettre de réutiliser les noms des variables à différents endroits sans pour autant

4 types de variables différentes

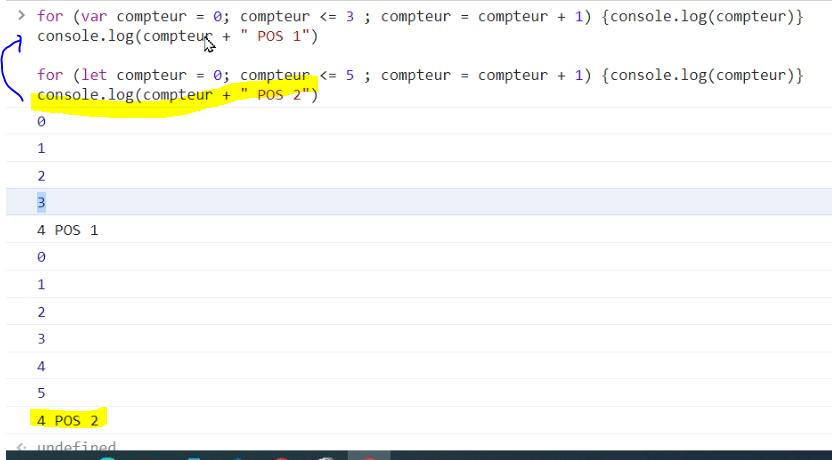
-*var*

-*let*

Limite la portée au bloc déclaré entre brackets.

.

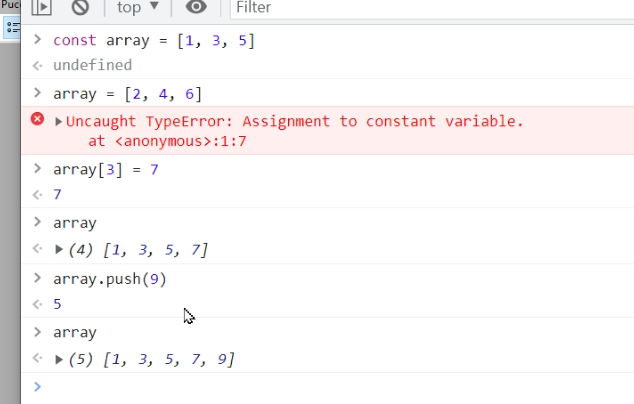
Exemple 2 – 2 variables, quel résultat dans la console ?



-*conso*

Comme un let (limité au bloc) mais on ne peut pas REASSIGNER une valeur à const. On doit donc lui donner ne valeur dès le début sinon elle ne sert à rien.

Attention que la valeur ne peut pas être réassignée mais peut être modifiée



-*maVariable*

Variable

Une expression est quelque chose qui renvoyer une valeur (on peut afficher, mettre dans une variable, dans un calcul)