

## Les variables

### Déclaration des variables

Dans l'exercice ci-dessous vous devez créer les variables suivantes :

- **num** qui contient le nombre 255
- **txt** qui contient la chaîne de caractère 255
- **bin** qui contient la valeur booléenne *vraie*
- **flt** qui contient la valeur 7,23

### Opération sur les variables

Reprenez la déclaration des variables précédentes et :

- Appliquez à **num** l'opérateur **d'incrément**
- Concaténez à la chaîne **txt** la chaîne " est ici une chaîne de caractères"
- Inversez la valeur logique de **bin**
- Ajoutez à **flt** la valeur contenue dans **num**

### Types des variables

Dans l'exercice ci-dessous vous devez créer les variables suivantes :

- **num** qui contient le nombre 23
- **txt** qui contient la chaîne de caractère 33
- **tmp** qui contient le nombre 0
- Affichez le résultat de **txt + num** avec la commande `console.log( txt + num );`
- Affichez le résultat de **num + txt**
- Convertir la chaîne contenue dans txt en nombre avec **parseInt()** et mettre le résultat dans **tmp**
- Ajouter à **num** le contenu de **tmp**

## Les fonctions

### Fonction sans argument

Dans cet exercice vous devez créer une fonction nommée **isoNum** qui renvoie le nombre 42

### Fonction avec arguments

Créez maintenant une fonction nommée **myFct** qui :

- prend 2 arguments (arg1 et arg2)
- renvoie comme résultat : La somme de arg1 et arg2 =  $\text{arg1} + \text{arg2}$
- Appeler la fonction **myFct** en passant les 2 arguments 23,27, le résultat affiché -> La somme de 23 et 27 = 50

## Les tests conditionnels

### Test sur la longueur d'une chaîne de caractères




Ecrivez une fonction nommée `isTooLong` qui :

- prend une chaîne de caractères comme argument
- renvoie vrai si la longueur de la chaîne dépasse 10 caractères
- renvoie faux dans les autres cas



Utilisez `str.length` pour connaître la taille chaîne de caractères `str`

### Test sur le type des paramètres

- Reprenez la fonction `myFct` et complétez là pour qu'elle renvoie `-999999999` si un argument n'est pas un nombre
-  Utilisez `typeof( varia )` pour connaître le type de la variable `varia`. Si `varia` est un nombre `typeof` retourne la chaîne de caractère `"number"`.
  -  Vous pouvez tester si deux conditions sont vraies simultanément avec un AND logique (`condition1 ET condition2`) comme ceci : `( test1 && test2 )`.
  -  Vous pouvez tester si l'une OU l'autre des conditions sont vraies avec un OR logique (`condition1 OU condition2`) comme ceci : `( test1 || test2 )`.

### Tests imbriqués

1. Créez une fonction nommée `aprec` qui :
  - prend une note sur 20 en argument
  - renvoie `Hideux` si la note est entre 0 et 7 exclus
  - renvoie `Moche` si la note entre 7 et 10 exclus
  - renvoie `Nice` si la note entre 10 et 15 exclus
  - renvoie `GG` si la note entre 15 et 20 inclus
  - renvoie `BUG` si la note n'est pas correcte
2. Grâce à la fonction `prompt()`, demander l'âge de l'utilisateur, puis à l'aide d'une condition, déterminer et afficher via une boîte de dialogue si l'utilisateur est mineur ou majeur. On considère qu'un individu est majeur à partir de 18 ans.

## Les tableaux

### Création d'un tableau

- Créez un tableau qui se nomme `tablo` et qui contient 1, 2 et 3

### Accédez aux données d'un tableau et les modifier

- Reprenez `tablo` qui contient 1, 2 et 3 de l'exercice précédent
- Accédez au 2ème élément du tableau `tablo` et rangez la valeur dans une variable nommée `item`
- Remplacez le 2ème élément du tableau par le nombre 5

## Ajoutez des données au tableau

- Reprenez tablo qui contient 1, 2 et 3 de l'exercice précédent
- Ajoutez une ligne de code qui ajoute la valeur 13 à la fin du tableau
- Ajoutez une autre ligne de code qui ajoute la valeur 14 à la fin du tableau

## Opérations

## Filter

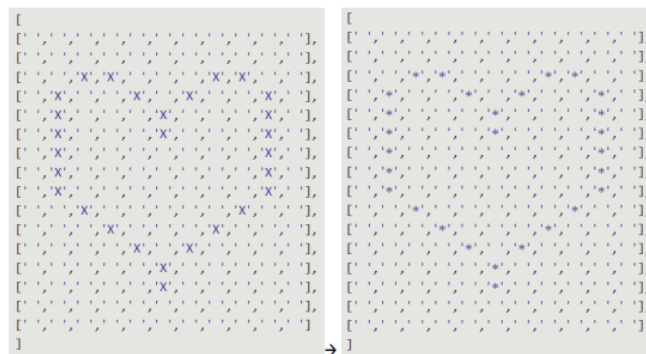
1. Vous avez un tableau avec des mots aléatoires, mais votre programme n'accepte pas les mots commençant par la lettre majuscule « Z ». Supprimez les mots non acceptés et renvoyez le nouveau tableau.
2. Écrivez une fonction qui prend un tableau d'éléments et ne renvoie que les entiers.

## Map

1. Ecrivez le code qui permet à partir d'un tableau d'entier, de renvoyer un tableau contenant les valeurs au carré

```
const input = [1, 2, 3, 4, 5];==>[1, 4, 9, 16, 25];
```

2. Créez un robot qui peut enlever les tatouages. Votre fonction accepte en argument un tableau appelé skinScan
3. Votre tâche est donc de créer une fonction qui va effacer les X et les remplacer par \*. Toutes les valeurs qui ne sont pas X doivent être laissées identiques.



## Reduce

Calculer la moyenne de chaque étudiant

```
const students = [
  { name: "Alice", scores: [90, 85, 92] },
  { name: "Bob", scores: [75, 80, 85] },
  { name: "Charlie", scores: [90, 95, 85] },
  { name: "Jack", scores: [100, 100, 100] }
];
```

## Sans filet

Afficher le prix moyen des produits par catégorie

```
const products = [
  { name: "Product 1", price: 20, category: "Electronics" },
  { name: "Product 2", price: 30, category: "Clothes" },
  { name: "Product 3", price: 40, category: "Electronics" },
  { name: "Product 4", price: 50, category: "Clothes" },
  { name: "Product 5", price: 60, category: "Clothes" },
  { name: "Product 6", price: 70, category: "Electronics" },
  { name: "Product 7", price: 80, category: "Clothes" },
  { name: "Product 8", price: 90, category: "Electronics" },
];
➔ [
  { category: 'Clothes', average: 55 },
  { category: 'Electronics', average: 55 }
]
```

## Les boucles

### Boucle et fonction

1. Créez une fonction multiplication qui
  - Prend un argument un chiffre
  - En utilisant la boucle for, affiche la table de multiplication du chiffre en paramètre
2. Créez une fonction puissance qui :
  - prend deux arguments
  - réalise le calcul du premier argument élevé à la puissance du deuxième sans utiliser les fonctions mathématiques de JS

💡 Utilisez une boucle par exemple  $x^n = x * x * x * x * \dots n \text{ fois}$

### Boucle et tableau

1. On souhaite stocker les positions d'une série de 20 objets mobiles (mobs) dans un jeu vidéo. Pour cela on va créer 2 tableaux :
  - posX qui contient la série des abscisses des objets
  - posY qui contient la série des ordonnées des objets

Créez une fonction initXY :

- qui prend en 1er argument l'abscisse du 1er objet
- qui prend en 2ème argument l'ordonnée du 1er objet
- qui calcule les abscisses et ordonnées des objets de la série sachant qu'il doivent être espacés de 40 pixels (+40) les uns des autres en abscisse et de 30 pixels (+30) en ordonnée

2. Créez une fonction myTable qui :
  - Prend en argument 2 variables x et y
  - Construit, en utilisant la fonction [writeln\(\)](#) , un tableau HTML, sans en-tête, avec X lignes et Y colonnes dans un élément <table>. Ajouter dans les cellules un contenu aléatoire de votre choix.
  - Demandez à l'utilisateur de saisir les valeur x et y, via la fonction prompt()

### Modifiez un tableau avec push et pop

On peut aussi manipuler les tableaux comme des flux en utilisant les fonctions pop, push, shift et unshift.

Créez un tableau contenant des nombres aléatoires et créez une fonction sortNumbers(tInf, tInf, tSup) qui va ranger :

- tous les éléments de tInf inférieurs à 10 dans le tableau tInf
- et tous les éléments de tInf supérieurs ou égaux à 10 dans le tableau tSup
- 💡 la fonction doit fonctionner quel que soit le tableau tInf

## Les objets

### Littéral

Créez une fonction qui prend un objet comme argument et renvoie une chaîne qui représente une personne. Les attributs de la personne devront être extraits des trois propriétés de l'objet:

- nom
- age
- adresse

La chaîne doit avoir le format suivant: X a Y ans. Il habite à Z. (où X est le nom de la personne, Y est l'âge et Z est l'adresse).

### Objet et tableau

- Définir un objet personnalisé 'member' (membres) avec les attributs 'id' (identifiant), 'name' (nom) et 'grade' et une méthode 'toString' personnalisée.
- Créer un objet personnalisé 'team' qui contient des membres.
- Créer une instance de 'team' et y ajouter des membres.
- Afficher les membres de 'team' en utilisant la fonction 'toString' de 'member'
- Créer une méthode 'toHtmlTable' dans 'team'
- Afficher les membres de 'team' sous la forme d'un tableau HTML , en utilisant la fonction 'toHtmlTable'

## Paramètre d'URL

Le but est de récupérer les paramètres d'une url de type

<https://www.google.com/search?q=meilleur+ide+js&client=firefox>

Rien qu'en utilisant les méthodes de l'objet string, écrivez la fonction **getParam** qui:

- renvoie la valeur du paramètre en argument
- Renvoie tous les paramètres et leurs valeurs dans le cas contraire

Exemples:

- `getParam(q) ==> meilleur ide js`
- `getParam() ==> q : meilleur ide js | client : firefox`

## Calcul d'âge

Ecrivez une fonction **getAge**, qui permet de calculer l'âge d'une personne à partir de sa date de naissance.

## Anagramme

- abc et cba sont des anagrammes.
- créatif et réactif sont également des anagrammes

Ecrivez la fonction **isAnagram** qui vérifie si 2 chaînes sont des anagrammes ou non

## Mélanger un tableau

Créez une fonction **randomize** qui à partir d'un tableau renvoie un tableau avec les valeurs mélangées

## Suite de Fibonacci

La suite de Fibonacci est une suite de nombres entiers de 0, 1, 1, 2, 3, 5, 8 ....

Les deux premiers termes sont 0 et 1. Tous les autres termes sont obtenus en ajoutant les deux termes précédents. Cela signifie que le n<sup>ième</sup> terme est la somme des (n-1)<sup>ème</sup> et (n-2)<sup>ème</sup> terme.

Ecrivez la fonction **fibonacci**, qui aura un paramètre

Exemple : `fibonacci(8) ==> 21`

## Chiffrement de César

Le chiffrement de César, également connu sous le nom de chiffrement par décalage, est l'une des formes les plus anciennes et les plus simples de chiffrement d'un message. Il s'agit d'un type de chiffrement par substitution dans lequel chaque lettre du message d'origine est remplacée par une lettre correspondant à un certain nombre de lettres décalées vers le haut ou vers le bas dans l'alphabet.

Exemple: texte = ABCD, Clé = 10 ==> KLMN

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Décalage de 10 vers la droite, A est remplacé par K ,B par L ..

Décalage de 10 vers la droite, C est remplacé par M

Décalage de 10 vers la droite, D est remplacé par

Créez la fonction `cesar` qui aura comme paramètres, la chaîne à coder et le décalage.