

# Activités Javascript

Afin de faire ces exercices, il est préférable d'avoir réalisé les tutoriels de 1 à 6 du site [codecademy](https://www.codecademy.com). Une fois fait, ces exercices vous permettront de valider les acquis.

## Exo-0 - Lecture de code Javascript.

Avant d'écrire du code javascript, il est important de savoir en lire, c'est le but de cette partie.

Rendez vous dans le répertoire exo-0

### Section 1 : Commentaires

L'objectif est de commenter chaque ligne de code avec une phrase en Français.

Par exemple :

```
var a = 38; // je déclare une variable nommée a qui prend pour valeur l'entier 38.
```

Faire ou compléter une fiche mémo qui reprends les différentes expressions.

### Section 2 : Utilisation de la console

En utilisant les variables déclarées plus haut, afficher dans la console les différentes consignes.

### Section 3 : Undefined

Décommenter la ligne puis recharger la page.

```
// console.log(myVariable);
```

Que se passe t'il et pourquoi ?

## Exo-1 - Les variables

La température aujourd'hui est de 294 Kelvin.

1. Pour commencer, créer une variable `kelvin` qui prends pour valeur 294;

Nous pouvons aussi exprimer une température en Celsius. La température en Celsius est de 273 degrés inférieure à celle en Kelvin.

1. Convertir la température en Kelvin en stockant le résultat dans une variable nommée `celsius`.

Ecrire un commentaire pour expliquer cette ligne de code.

Nous pouvons aussi exprimer une température en Fahrenheit avec la formule : `Fahrenheit = Celsius * (9/5) + 32`

1. En utilisant cette équation, calculer la température en Fahrenheit et stocker le résultat dans une variable nommée `fahrenheit`. Ecrire un commentaire pour expliquer cette ligne de code.

Lorsque nous faisons cette conversion, il est très probable d'avoir un nombre avec des décimales, nous allons donc arrondir ce nombre.

1. En utilisant la méthode `.floor()` de la bibliothèque standard, modifier le résultat de la variable `fahrenheit` pour obtenir un entier. Ecrire un commentaire pour expliquer cette ligne de code.
2. Nous voulons modifier le programme afin de demander la température en Kelvin à l'utilisateur et lui afficher ensuite les conversions dans le console du navigateur. Remplacer la première ligne de code par :

```
var kelvin = prompt('Quelle est la température en Kelvin aujourd'hui ?');
```

et relancer le programme.

## Exo-2 - Les conditions

Vous avez été recruté pour écrire un programme pour gérer les inscriptions d'une course à pied. Tous les adultes courent le matin et tous les personnes qui ont moins de 18 ans l'après midi. Suivez les instructions afin de développer la logique du programme.

1. Créer une variable pour stocker si un coureur est déjà inscrit ou pas. Quel est le type de cette variable ?
2. Créer une variable pour stocker l'âge du coureur.
3. Une personne qui s'inscrit un jour à un numéro de coureur supérieur à quelqu'un inscrit plus tôt. Ecrire une condition qui vérifie si la personne n'est pas déjà inscrite. Ajouter 1000 à son numéro si ce n'est pas le cas. Vous pouvez utiliser cette expression pour avoir un numéro de coureur aléatoire. `var raceNumber = Math.floor(Math.random()*1000);`
4. Ecrire une nouvelle condition en dessous de la première permettant d'afficher pour :
  1. Les personnes qui ont plus de 18 ans et déjà inscrites : 'You will race at 9:30 am'
  2. Les personnes déjà inscrites ou qui ont plus de 18 ans ( mais pas les deux ) 'You will race at 11:00 am ' plus leur numéro de dossard.
  3. Les personnes de moins de 18 ans ou qui ne sont pas encore inscrites. 'You will race at 12:30 pm ' plus leur numéro de dossard.
5. Entrer différentes combinaisons de valeurs pour les deux variables et vérifier le bon fonctionnement de votre programme.
6. Ajouter un block `else` dans votre code disant simplement : 'Go to see the registration desk' pour les personnes ne rentrant pas dans les cas précédents.

## Exo-3 - Les fonctions

Vous allez programmer la logique du jeu Pierre, Feuille, Ciseau. Vous jouerez contre l'ordinateur.

Pour se souvenir des règles, cliquer [ici](#).

Notre code va se décomposer en 4 parties : \* Récupérer le choix de l'utilisateur \* Récupérer le choix du programme. \* Comparer les choix \* Annoncer le gagnant.

1. Créer une fonction `getUserChoice()` qui prends en paramètre un seul argument `userInput`
2. Puisque nous ne pouvons pas être certain que l'utilisateur va taper 'Rock' ou 'rock', nous allons dans un premier temps utiliser la fonction `toLowerCase()` pour rendre minuscule la chaîne de caractère `userInput`.
3. Nous devons aussi nous assurer que l'utilisateur rentre bien un des trois choix : 'rock', 'paper', ou 'scissors'. A l'intérieur de la fonction `getUserChoice()`, écrire une condition qui vérifie que la valeur rentrée par l'utilisateur est bien l'une des trois attendues i.e. 'rock', 'paper', ou 'scissors'. Si c'est le cas, la fonction retourne la chaîne de caractères `userInput`, sinon afficher un message d'erreur dans la console.
4. Maintenant c'est à l'ordinateur de faire un choix. Créer une nouvelle fonction `getComputerChoice()` qui ne prends pas de paramètre. A l'intérieur, en utilisant la fonction `Math.random()`, tirer un nombre aléatoire entre 0 et 2. Ensuite, étant donné le résultat, la fonction doit retourner 'rock', 'paper', ou 'scissors'.

Maintenant, nous devons déterminer le gagnant.

1. Créer une fonction `determineWinner()` qui prends deux paramètres `userChoice` et `computerChoice`

Cette fonction va comparer les choix et va retourner soit : \* Tied : égalité \* Won : l'utilisateur a gagné. \* Lost : l'utilisateur a perdu.

1. Commençons par le cas d'égalité, à l'intérieur de la fonction `determineWinner()`, écrire une condition qui vérifie si le choix de l'utilisateur est égal à celui de l'ordinateur. Si c'est le cas, retourner la chaîne de caractères 'Tied'.
2. Si ce n'est pas une égalité alors quelqu'un a gagné. Commencer par vérifier si `userChoice` est `rock`. A l'intérieur de cette condition, vérifier si le choix de l'ordinateur est 'scissors', si oui alors retourner 'Won' sinon retourner 'Lost'.
3. Ensuite, écrire une nouvelle condition pour déterminer si le choix de l'utilisateur est 'paper'. A l'intérieur de cette condition, en fonction du choix de l'ordinateur, retourner 'Lost' ou 'Won'
4. Traiter maintenant le cas où l'utilisateur a choisi 'scissors'.
5. Nos fonctions sont prêtes, nous pouvons maintenant commencer le jeu. Créer une fonction `playGame()`, à l'intérieur de celle ci, créer une variable `uChoice` qui prends pour valeur le résultat de la fonction `getUserChoice()` en ajoutant l'argument de votre choix : 'rock', 'paper', or 'scissors'. Ensuite créer une autre variable `computerChoice` qui prends pour valeur le résultat de l'appel à la fonction `getComputerChoice()`. Afficher chacune de ses variables dans la console.
6. Affichons maintenant le vainqueur. A l'intérieur de la fonction `playGame()`, appeler la fonction `determineWinner()` avec en paramètre les valeurs des variables `uChoice` et `computerChoice`. Pour connaître le vainqueur vous devez mettre l'appel de la fonction à l'intérieur d'un `console.log()`. Maintenant appeler la fonction `playGame()`
7. On va ajouter un petit code secret afin d'être certain de gagner dans tous les cas : si l'utilisateur rentre 'bomb' alors il a gagné. Modifier votre code en fonction.

## Exo-4 - Les tableaux.

```
var secretMessage = ["Learning", "isn't", "about", "what", "you", "get", "easily",  
"the", "first", "time,", "it's", "about", "what", "you", "can", "figure", "out.",  
"-2015,", "Chris", "Pine,", "Learn", "JavaScript"];
```

Vous aurez besoin de la [documentation sur les tableaux](#), notamment pour trouver les bonnes méthodes.

1. Utiliser une méthode pour enlever le dernier éléments du tableau `secretMessage`.
2. Utiliser une méthode pour ajouter les mots 'to' and 'program' comme des éléments distincts à la fin du tableau.
3. Changer le mot 'easily' par 'right' en accédant au bon index du tableau.
4. Utiliser une méthode pour supprimer le premier éléments du tableau.
5. Utiliser une méthode pour ajouter la chaine "Programming" en début de tableau.
6. Utiliser une méthode pour remplacer les mots 'get', 'right', 'the', 'first', 'time' par 'know'.
7. Utiliser la méthode `.join` pour afficher le message secret à la console.

## Exo-5 - Les boucles

Les baleines (!) traduisent les phrases comme : 'Turpentine and turtles' par: 'UEIEAUE'.

Il y a une règle simple : Pas de consonnes, uniquement les voyelles, pas d'espace et toutes les lettres en majuscules.

Pour faire cette traduction, nous allons utiliser des boucles.

1. Créer une variable nommée `input` qui prends pour valeur n'importe quelle phrase ou mot à traduire.
2. Nous allons aussi créer un tableau de lettres. Puisque la langue des baleines contient que des voyelles, créer une variable `vowels` de type tableau contenant toutes les voyelles.
3. Nous allons stocker la chaine traduite dans un tableau. Créer donc une variable `resultArray` dont la valeur est un tableau vide.
4. Maintenant notre but est de trouver toutes les voyelles dans notre chaine initiale, on peut le faire en vérifiant si la première lettre est une voyelle, puis la deuxième etc... Créer donc une boucle qui parcours toutes les lettres de notre chaine de caractères. A l'intérieur de la boucle afficher simplement la lettre courante.
5. Maintenant nous devons vérifier si la lettre courante est une voyelle autrement dit si la lettre courante est une lettre du tableau `vowels`. Utiliser la méthode `indexOf` des tableaux pour faire ça.
6. Bonus: A la place de `indexOf`, écrire une seconde boucle.
7. Si la lettre est une voyelle, utiliser la méthode `.push()` sur le tableau `resultArray` afin d'y ajouter la lettre courante `input[i]`. Vérifier que le tableau `resultArray` contient que des voyelles.
8. Si vous affichez `resultArray`, vous observerez qu'il y a des virgules entre chaque élément. Pourquoi ? Utiliser les méthodes `join('')` et `toUpperCase()` pour un bon formatage.
9. Lancer votre programme et chanter la traduction !

# Exo-6 : Les objets

Vous devez avoir fait le cours 8 ( exercices 1 à 7 ) de CodeAcademy pour faire cet exercice.

Nous voulons créer et extraire des informations à propos de votre équipe de sports préférés. ( Basketball, soccer, tennis, or water polo... vous choisissez ) C'est votre boulot de créer ces informations en utilisant les structures de données en Javascript appropriées : arrays, objects, etc.

1. Nous voulons une structure de données permettant de stocker des informations sur notre équipe, créer une variable 'team' dont la valeur est un object vide.
2. Notre équipe a des joueurs qui jouent des matches. Nous voulons représenter les deux. Ajouter deux propriétés à votre objet : `_players` et `_games` toutes les deux étant des tableaux vides.
3. Remplir le tableau `_players` avec des données dans le format spécifique :

```
{
  firstName: "Pablo",
  lastName: "Sanchez",
  age: 11
}
```

1. Remplir le tableau `_players` avec des données dans le format spécifique :

```
{
  opponent: "Broncos",
  teamPoints: 42,
  opponentPoints: 27
}
```

1. Nous voulons ajouter un nouveau joueur à votre équipe, créer une fonction `addPlayer` qui prends trois arguments : `first`, `last`, et `age` et ajoute à l'objet `team` un joueur.
2. Appeler cette méthode et vérifier son bon fonctionnement.
3. Faire la même chose pour un match.
4. Appeler vos fonctions plusieurs fois pour avoir un object avec suffisamment de données.
5. Calculer la somme des points de votre équipe sur tous les matches joués.
6. Calculer la moyenne des points de l'équipe adverse sur tous les matches.
7. Ecrire une fonction qui permet de trouver le joueur le plus âgé.
8. Bonus: Trier les joueurs par ordre alphabétique.