

# Javascript - Tableaux

# Pratique: Calendrier

```
let jours = [ 'lun', '007', 'mer', 'jeu', 'ven', 'sam', 'BUG' ];
```

Quelles instructions JavaScript faut-il exécuter pour effectuer les opérations suivantes ?

1. Retirer la dernière valeur du tableau jours
2. Afficher les valeurs du tableau dans la console
3. Ajouter la valeur 'dim' à la fin du tableau
4. Remplacer la valeur '007' par 'mar'
5. Afficher le nombre de valeurs du tableau dans la console
6. Afficher la troisième valeur du tableau dans la console
7. Quelle est la valeur finale du tableau ? Faites un console.log

# Pratique: Épicerie

```
let fruits = [ 'Mangue', 'Raisin', 'Figue', 'Kiwi' ];
```

Écrire un programme qui:

1. Affiche la liste de fruits disponibles;
2. Demande au client quel fruit il désire acheter:
  - s'il est présent dans le tableau (indexOf) fruits: le retirer du tableau (splice), et afficher 'ok!',
  - sinon, afficher 'indisponible...'.
3. Affiche à nouveau la liste de fruits disponibles.

# Exercice: Jeu du Pendu

Principe du jeu: Le joueur doit deviner un mot, lettre par lettre. Il perd après 10 mauvaises tentatives.

Nous allons utiliser les variables suivantes:

```
let mauvaisesLettres = [ ]; // sera complété pendant le jeu
```

```
let lettresAttendues = [ 's', 'u', 'p', 'e', 'r' ];
```

À chaque tentative, le joueur va proposer une lettre, puis:

- si la lettre fait partie du mot à deviner (dont les lettres sont stockées dans le tableau lettresAttendues), cette lettre sera supprimée du tableau;
- sinon, elle sera ajoutée à la fin du tableau mauvaisesLettres.

Le joueur gagne la partie quand il a deviné toutes les lettres attendues du mot. (c.a.d. le tableau lettresAttendues ne contient plus aucun élément)

Le joueur perd la partie dès que le tableau mauvaisesLettres contient 10 éléments. (c.a.d. le nombre de tentatives autorisées)

# Exercice: Jeu du Pendu

Mots clés à utiliser: alert, prompt, if, else, =, ===, indexOf, length, push, splice, et for ou while

Conseil: avant de créer une boucle for, implémenter l'algorithme qui sera exécuté à chaque itération de cette boucle. (une itération = une tentative du joueur)

Étapes proposées:

1. Afficher le nombre de lettres à trouver, et de tentatives restantes.
2. Si mauvaisesLettres a une longueur de 10, afficher perdu!.
3. Demander une lettre au joueur, en utilisant prompt().
4. Si la lettre fait partie du tableau lettresAttendues, la supprimer du tableau. Sinon l'ajouter au tableau mauvaisesLettres. Afficher un message au joueur dans chaque cas.
5. Si le nombre de lettresAttendues restantes à trouver est nul, afficher bravo!.
6. Créer une boucle for permettant de jouer jusqu'à la fin de partie (perdu ou bravo).

BONUS: Dans le cas où le joueur devine une lettre qui apparaît plus d'une fois dans le mot, retirer toutes les occurrences de cette lettre en une seule fois.