

DECOUVERTE DU LANGAGE

EXERCICE 01

- Écrire, avec des comparaisons, un algorithme qui affiche l'état de l'eau « glace, liquide, vapeur » en fonction de sa température.
- Donnez son implémentation en JavaScript.

EXERCICE 02

- Écrire un algorithme qui détermine la catégorie sportive d'un utilisateur en fonction de son âge :
 - 6 à 7 ans : poussin
 - 8 à 9 ans : pupille
 - 10 à 11 ans : minime
 - 12 ans et plus : cadet
- Écrire le programme JavaScript associé.

EXERCICE 03

- Ecrivez un programme JavaScript qui permet d'échanger la valeur contenue dans deux variables. Par exemple, si A vaut 5 et B 13 au début de l'algorithme, faites-en sorte que A soit égal à 13 et B à 5 à la fin de l'algorithme.

EXERCICE 04

Proposez un programme JavaScript qui demande le poids en **kg** et la taille en **cm** à son utilisateur puis calcule son **IMC** et lui affiche un message en fonction du résultat obtenu.

Le calcul de l'IMC se fait en divisant le poids « exprimé en kg » par la taille au carré « exprimée en m ».

IMC	Message
<18,5	Insuffisance pondérale
18,5 <= IMC < 25	Poids normal
25 <= IMC < 30	Surpoids
IMC >= 30	Obésité

EXERCICE 05

- Une année bissextile comporte 366 jours au lieu de 365. Le jour supplémentaire « le 29 février », est placé après le dernier jour de ce mois qui compte habituellement 28 jours.
- Une année est bissextile :
 - *Si elle est divisible par 4 mais pas par 100* ou
 - *Si elle est divisible par 400*

Proposer un programme JavaScript qui demande une année à l'utilisateur et qui l'informe si cette année est bissextile ou non.

EXERCICE 06

1. Ecrire un programme qui affiche les nombres de 1 à 10.
2. Ecrire un programme qui affiche la somme des nombres de 1 à 10.

EXERCICE 07

Ecrire un programme JavaScript qui demande à l'utilisateur de saisir au clavier les nombres m et n et les afficher de n à m :

- a. Dans le cas où n est supérieur à m,
 - i. Afficher un message d'erreur et arrêter le programme.
 - ii. Permuter n et m, puis continuer.
 - iii. De manière décroissante
 - iv. Afficher la somme des nombres pairs de n à m.

EXERCICE 08

1. Lire 10 nombres et trouver le plus grand.
2. Lire 10 nombres et trouver le plus petit et son rang.
3. Lire n nombres et en calculer la moyenne.

EXERCICE 09

1. Dans un tableau de 10 entiers initialisé avec des valeurs aléatoires, trouver les rangs et les valeurs du plus petit et du plus grand élément, et les afficher dans la console.
2. Inverser les éléments d'un tableau de 10 éléments (sans passer par un tableau intermédiaire).
3. Trier un tableau de 10 nombres saisis au clavier, par ordre croissant.
4. Fusionner deux tableaux déjà triés par ordre croissant.
5. Dire si une valeur existe dans un tableau.

EXERCICE 10

Le jeu consiste à découvrir par essais successifs le prix d'un lot. Pour chaque essai, le joueur reçoit un message : « **Trop grand** », « **Trop petit** » ou « **BRAVO ! Vous avez trouvé en K essais** ». Le jeu est fini quand le joueur a trouvé le prix du lot. On propose d'écrire un programme qui joue le rôle de meneur de jeu ; l'exécution de ce programme vous fera tenir le rôle du joueur. Le programme doit définir le prix du lot en tirant un entier aléatoire entre 1 et 10 et dialoguer avec le joueur pendant le jeu.

EXERCICE 11

1. Créer une chaîne de caractères de 50 '- '.
2. Inverser une chaîne de caractères (sans supprimer l'originale).
3. Remplacer tous les caractères d'une chaîne par une '*'
4. Remplacer toutes les occurrences d'une lettre par une autre dans une chaîne de caractères.
5. Supprimer toutes les occurrences d'une lettre dans une chaîne de caractères.
6. Afficher la présence d'une lettre dans une chaîne (si oui, en afficher le nombre (quantité, si non, afficher "absent").
7. Compter le nombre d'occurrence de chaque lettre dans une chaîne.
8. Remplacer les double-espace (ou +) dans une chaîne de caractères par un espace.
9. Découper une chaîne de caractères en mots avec l'espace comme séparateur et les compter.
10. Inverser les mots d'une phrase.

EXERCICE 12

1. Ecrire la fonction *isMultiple*, qui prend deux entiers *n* et *m* et retourne *vrai* si et seulement si *n est un multiple de m*.
2. Ecrire la fonction *isEven*, qui prend un entier *i* et retourne *vrai* si et seulement si *i est un nombre pair*. « Notre méthode ne peut pas utiliser la multiplication, le module, ou la division comme opérateur en son sein »
3. Un nombre est *n* premier s'il a seulement deux diviseurs : *1* et *n*. Ecrire une fonction *isPremier* qui permet de tester si un nombre saisi par l'utilisateur est premier ou non.
4. Ecrire une fonction qui prend un entier *n* et retourne la somme des entiers inférieurs ou égaux à la valeur absolue de *n*.
5. Ecrire une méthode qui compte le nombre de voyelles dans une chaîne de caractères donnée.