

JavaScript – Session 1

Le code doit être écrit dans un fichier index.html (attention, un seul index par dossier).
Vous pouvez utiliser le template base disponible sur la page exos @ campus.ow-lab.com.
Le point de départ de vos programmes est la fonction init() activée avec un attribut onload sur body.
Le code JS sera rédigé dans une balise script, située dans le fichier HTML.
Facultatif: chaque programme pourra être exécuté indépendamment au click sur un élément HTML.

Attention => on n'utilisera jamais ici de variables globales. Utiliser le **scope des fonctions**

Exercice 1: Paramètre à afficher

Écrire une fonction displayTextBox() prenant une string en paramètre et l'affichant dans une alert box.
Exécuter la méthode au rechargement de la page et depuis la console du navigateur.

Exercice 2: Sur une ligne

Écrire une fonction prenant deux paramètres a et b et les affichant dans la console, à la suite sur une même ligne, séparés par un espace. Commencer avec les paramètres de son choix. Puis tenter avec les deux strings suivantes:

- C'est une belle journée...
- Pour s'exercer à JS!&

Exercice 3: Multiplication

Écrire une fonction prenant deux paramètres a et b et les multipliant. Si le résultat est de type number, le retourner et afficher le résultat dans la console. Sinon, une alerte indiquera une erreur.

Exercice 4: Affichage d'objets

Créer une div HTML portant l'id res_user1
Créer deux objets littéraux user1 et user2 (attention à la portée des variables).
Chaque objet a 2 membres:
1. name: string
2. age: number

Affecter des valeurs cohérentes aux objets selon leur type.

Écrire une fonction getName(user), retournant le nom de l'utilisateur passé en paramètre

Écrire une fonction getAge(user), retournant l'âge de l'utilisateur passé en paramètre.

Écrire une fonction displayUserInfo(user).
Cette méthode affiche dans la div les informations de l'utilisateur passé en paramètre.
Par exemple: 'Cet utilisateur se nommant Joe est âgé(e) de 23 ans'.

La div doit être vidée à chaque nouvel appel de fonction.
Faire marcher le code depuis la console du navigateur.

Trouver des idées pour améliorer ce programme.

Exercice 5: Easy DOM

Déclarer 5 div et un p en HTML.

Styliser sommairement ces éléments,

Identifier chaque div et le p avec un attribut id.

Utiliser une méthode de manipulation du DOM fournie par JS pour stocker les div dans un array.

Parcourir ce tableau avec une boucle.

À l'exécution, le programme doit afficher l'id de chaque div dans le p en sautant des lignes.

Le résultat doit ressembler à ceci :



Listing des ids =>

```
id n° 1 : div_1  
id n° 2 : div_2  
id n° 3 : div_3  
id n° 4 : div_4  
id n° 5 : div_5
```

Trouver des idées pour améliorer ce programme.

Exercice 6: Click en couleur

Utiliser le code de l'exercice 6.

Ajouter un attribut onclick sur chaque div.

Le click exécute une méthode colorDiv().

Cette fonction ajoute un attribut style appliquant un background-color à la div cliquée.