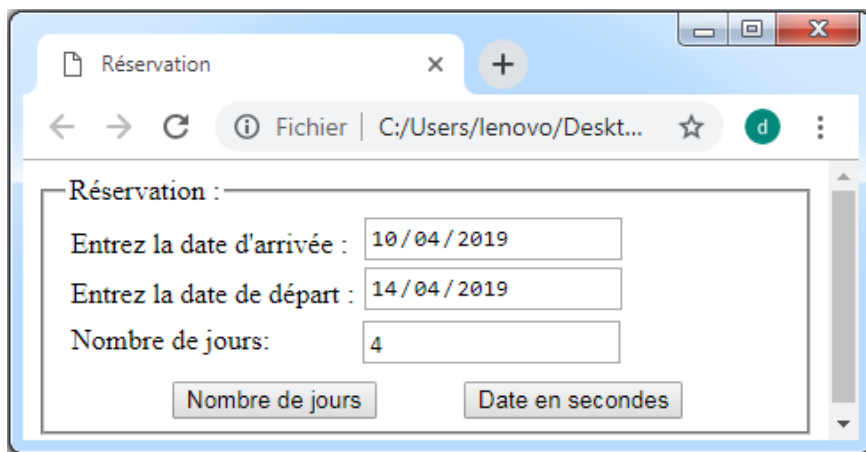

Atelier N°2 : Les objets prédéfinis du langage JS (STRING, DATE et MATH)

Objectifs spécifiques

Manipuler les méthodes de certains objets natifs (Date, String, Math) du langage JS ainsi que leurs propriétés.

Exercice n°1 : Objet date (DS 2019)

On se propose dans cet exercice de créer un formulaire qui rassemble à la figure 1 et qui est formé avec trois champs de saisie. Deux champs de type input type="date", un champ de type="number".



Travail demandé :

1. Calculer le nombre de jours entre les deux dates en cliquant sur le bouton « Nombre de jours ». Le nombre de jours calculés sera affiché dans le troisième champ du formulaire.

2. Afficher la différence entre les deux dates en millisecondes avec la fonction `alert()` en cliquant sur le bouton « Date en secondes ».

Notez Bien: pour la question 2, et afin de simplifier la fonction à développer, on supposera que la date de départ est supérieur à la date date d'arrivée et que les dates sont proposées durant le même mois et la même année.

Exercice n°2 : Objet Math : *calculatrice (DS 2020)*

On se propose dans cet exercice de créer une "**calculatrice de durée ou de temps**" qui ressemble à **la figure 1**. Cette calculatrice est formée de deux zones :

- **Zone A :** Contenant les durées à additionner ou à soustraire. Elle est formée de :
 - 6 "**input**" de type "**number** ": 2 pour les heures, 2 pour les minutes et 2 pour les secondes.
 - 3 "**input**" de type "**button**":
 - 1 bouton "**Addition**" pour2 faire l'addition des durées.
 - 1 bouton "**Reset**" pour Annuler le contenu des champs du formulaire.
 - 1 bouton "**Soustraction**" pour soustraire une durée d'une autre.
- **Zone B :** Contenant le résultat d'addition ou de la soustraction. Elle est formée de :
 - 3 "**input**" de type "**number** ": 1 pour les heures, 1 pour les minutes et 1 pour les secondes.

Notez Bien :

- Les champs minutes et secondes ne peuvent pas contenir des valeurs supérieur à 59.
- Dans le cas de la soustraction on supposera qu'il n'y a pas de contrainte lors du calcul de la différence et on retranche toujours la valeur la plus petite de la valeur la plus grande.
- Il est possible d'utiliser la méthode `Math.round()` qui retourne la valeur d'un nombre arrondi à l'entier le plus proche (*Exemple : `Math.round(5.05)=5`*)

The diagram shows a web application titled "Ma Calculatrice toute simple: Calculatrice de durée ou de temps". It features two input sections labeled "Zone A" and "Zone B".

Zone A: Contains two rows of input fields for hours, minutes, and seconds. The first row has values 1, 25, and 20. The second row has values 0, 40, and 20. Below these are three buttons: "ADDITION", "RESET", and "SOUSTRACTION".

Zone B: Contains a single row of input fields with values 2, 5, and 40.

Result: A box on the right labeled "Le résultat de l'addition des 2 durées de la zone A" displays the result of adding the durations from Zone A.

Figure 1 – Calculatrice de durée

Exercice n°3 : Chaînes de caractères (DS 2019)

On se propose dans cet exercice de créer un formulaire qui rassemble à la figure 2 et dont le rôle consiste à déterminer le nombre d'occurrences d'un caractère dans une, deux ou trois phrases proposées par l'utilisateur.

The screenshot shows a web browser window with a single tab titled "Nouvel onglet". The address bar is empty. The main content area contains a form titled "Les phrases à utiliser :".

Form Fields:

- Phrase 1 :
- Phrase 2 :
- Phrase 3 :

Search Section:

- Rechercher dans :** Three checkboxes are present: "Phrase 1", "Phrase 2", and "Phrase 3".
- Résultat :** A section containing:
 - Caractère recherché :** A dropdown menu with options a, b, c, and d.
 - Lancer la recherche** : A button.
 - Résultat :** An empty text input field.

Figure 2 – Formulaire « Recherche nombre d’occurrences »

Le principe de cette application est assez simple, l'utilisateur va saisir trois phrases dans les trois champs de type texte du formulaire (phrase 1, phrase 2 et phrase 3). Puis, il va sélectionner au niveau de la liste « Caractère à rechercher » un caractère quelconque. Ensuite et au niveau de la zone « Rechercher Dans », il va cocher aussi la ou les phrases concernées par l'opération de recherche. Enfin et une fois toutes les conditions déjà citées sont remplies, l'utilisateur clique sur le bouton « Lancer la recherche », le nombre d'occurrence calculé sera affiché dans le champ de type texte « Résultat ».

Notez bien : On suppose toujours que les trois phrases sont saisies et au moins une case à cocher et un caractère de la liste sont sélectionnés.

Travail demandé :

Ecrire les fonctions JavaScript nécessaires afin d'obtenir les fonctionnalités décrites précédemment.

Exercice 4 : DS ISET SFAX 2015

Le but de cet Exercice est de calculer le total d'une commande effectuée par un client. Pour cela, on doit créer deux tableaux TAB_QTE et TAB_PRIX ayant la même taille et qui contiennent respectivement les Quantités et leurs Prix.

Voici la liste des fonctions à développer en Javascript :

Noms des fonctions	Indications
Taille()	Lit et retourne un entier qui représente la taille d'un tableau qui doit être numérique et positive
Remplir(Tab,n)	Qui permet de remplir un tableau Tab par n valeurs positives et numériques
Affichage(Tab)	Qui permet d'afficher les éléments d'un tableau dans la page
Produit (q,p)	Qui retourne le produit des 2 entiers p et q
Calcul(Tab1,Tab2)	Qui retourne la somme totale des quantités multipliés par les prix élément par élément des deux tableaux Tab1 et Tab2 en utilisant la fonction Produit
Affichage_Fen(s)	Qui permet d'afficher le total retourné par la fonction Calcul dans une nouvelle fenêtre ayant les caractéristiques suivantes : largeur=300, hauteur=300, avec affichage des barres d'adresse, d'état et de menu.
Affichage_TabHtml(Tab,type)	Qui permet d'afficher le contenu d'un tableau ainsi que son type (Prix ou Quantité) dans un tableau HTML

	Ex :			
	Prix	4	5	6

Travail demandé

On vous demande d'écrire les codes HTML et Javascript correspondants tout en tenant compte des éléments suivants :

■ A l'ouverture de la page et à l'aide de la fonction Taille() cette fenêtre s'affiche :

■ Après avoir saisi la taille du tableau, on vous demande de remplir les deux tableaux TAB_QTE et TAB_PRIX (fonction Remplir) et de les afficher (fonction Affichage) (ligne 49->53).

On appelle ensuite la fonction Calcul() ; la valeur retournée par cette fonction est affichée dans une nouvelle fenêtre en utilisant la fonction Affichage_Fen() (ligne 54->55).

Et finalement, on affiche les deux tableaux TAB_QTE et TAB_PRIX en utilisant la fonction Affichage_TabHtml (Tab,type) (ligne 56->57).

Exercice 5 DS ISET SFAX 2015

Soit le formulaire suivant qui permet à un étudiant de vérifier s'il a le droit au logement universitaire ou bien non :

Demande de logement universitaire

Nom: Prenom: CIN

Entrer votre CIN de nouveau:

Sexe: ☒ Mâle ☐ Femelle

☐ Redoublant

Combien de fois vous avez bénéficié du logement universitaire auparavant: 0 fois ▼

0 fois
1 fois
2 fois

Travail demandé

On vous demande de compléter les 2 fichiers « Formulaire.html » & « Fonctions.js »

1) Ajouter le script nécessaire dans la ligne 2 du fichier « Formulaire.html » pour appeler le fichier « Fonctions.js »

2) a) Donner le script de la fonction ValideCIN() permettant de tester si le numéro CIN inséré est numérique ; Si non, elle doit afficher une alerte.

b) Appeler la fonction ValideCIN() dans le code HTML (la ligne 7)

3) a) Donner le script de la fonction VerifCIN() permettant de vérifier si la valeur du numéro CIN qui est tapée 2 fois est la même ou non; Si non elle doit afficher une alerte.

b) Appeler la fonction VerifCIN() dans le code HTML (la ligne 8)

4) a) Donner le script de la fonction Resultat() permettant d'afficher une alerte indiquant si l'étudiant a le droit au logement universitaire ou bien non suivant ce modèle :

■ Si l'étudiant est redoublant => Pas de logement dans tous les cas ;

■ Si l'étudiant n'est pas redoublant alors :

o Si l'étudiant est un garçon alors il a le droit au logement s'il n'en a pas bénéficié auparavant (0 fois) ;

o Si l'étudiant est une fille alors elle a le droit au logement si elle en a bénéficié au plus une fois (0 fois ou 1 fois) ;

b) Appeler la fonction Resultat() dans le code HTML lorsqu'on clique sur le bouton Valider (la ligne 18)