

---

## TP2 : HTML & JavaScript

### Objectifs :

- Réalisation des pages web dynamiques à l'aide du langage JavaScript .

### Outils :

- Visual Studio Code ou tout autre IDE
- 

### Exercice 1

Dans cet exercice on se propose de créer une page web avec un formulaire dans l'objectif de saisir des données et d'effectuer des calculs.

- Etape 1 : Code HTML

Ecrire le code HTML suivant dans un fichier «index.html»

---

```
<html>
  <head>
    <title> TP JavaScript </title>
    <style>

    </style>
    <script language="javascript">

    </script>
  </head>

  <body>
    <fieldset>
      <legend> Calcul du Prix TTC </legend>
      Prix unitaire HTC : <br>
      <input type="text" size="20" id="PU"/>
      <br>
      Taux TVA : <br>
      <input type="text" size="20" id="TVA"/>
      <br>
      Quantité : <br>
      <input type="text" size="20" id="QT"/>
      <br>
      <input type="button" value="Calculer" Onclick="Calculer_PrixTTC();" />
      <br>
      Prix TTC : <br>
      <input type="text" size="20" id="PT"/>
    </fieldset>
  </body>
</html>
```

---

- **Etape 2 : Mise en forme de la page web et code CSS**

Au niveau de la balise **<style>** ajouter le code CSS nécessaire pour la mise en forme de la page web :

---

```
<style>
  body{
    // Ajouter un code ici
  }
  fieldset{
    // Ajouter un code ici
  }
  #PU{
    // Ajouter un code ici
  }
  #TVA{
    // Ajouter un code ici
  }
  #QT{
    // Ajouter un code ici
  }
  #PT{
    // Ajouter un code ici
  }
</style>
```

---

- **Etape 3 : Ajout du code JavaScript**

Au niveau de la balise **<script>** ajouter le code JavaScript suivant :

```
<script language="javascript">

function Calculer_PrixTTC()
{
  var Champ_Texte_PU ;
  var Champ_Texte_TVA ;
  var Champ_Texte_QT ;
  var Champ_Texte_PTTC ;

  var Prix_Unitaire ;
  var Taux_TVA ;
  var Quantite ;
  var Prix_TTC ;

  Champ_Texte_PU = document.getElementById('PU') ;
  Champ_Texte_TVA = document.getElementById('TVA') ;
  Champ_Texte_QT = document.getElementById('QT') ;
  Champ_Texte_PTTC = document.getElementById('PT') ;

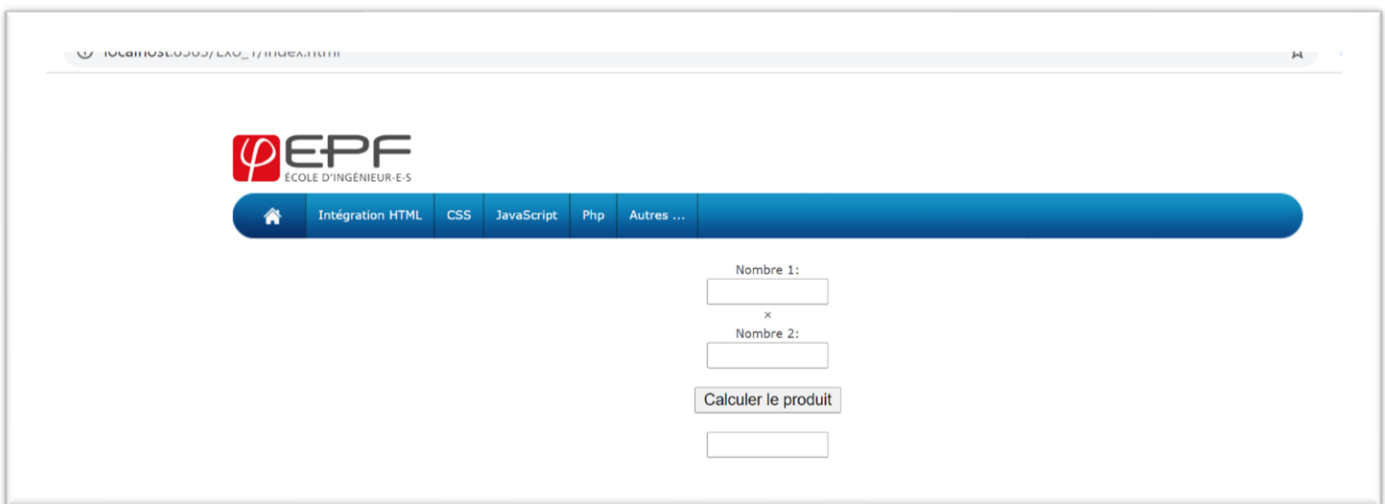
  Prix_Unitaire = parseFloat(Champ_Texte_PU.value);
  Taux_TVA = parseFloat(Champ_Texte_TVA.value);
  Quantite = parseFloat(Champ_Texte_QT.value);

  Prix_TTC =
```

```
(Prix_Unitaire*Quantite)+(Prix_Unitaire*Quantite*Taux_TVA/100);  
  
Champ_Texte_PTTC.value = Prix_TTC ;  
  
}  
</script>
```

## **Exercice 2**

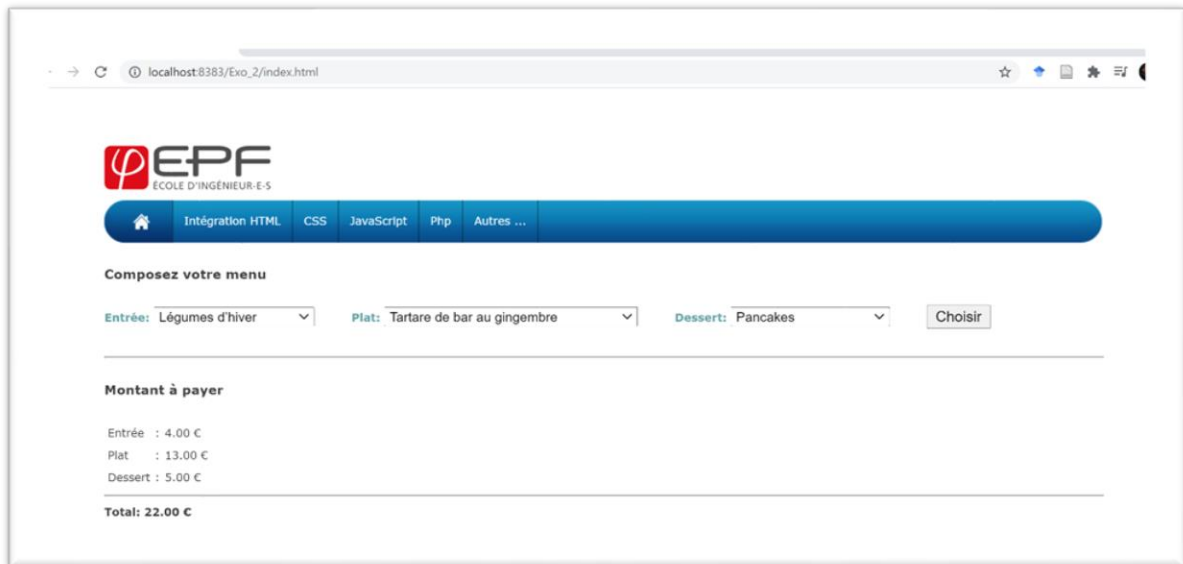
Dans cet exercice on vous demande d'écrire un code JavaScript permettant de calculer un produit de deux nombres comme le montre la figure ci-après :



The screenshot shows a web browser window displaying a page from the EPF (Ecole d'Ingénieur-e-s) website. The page features a navigation bar with the EPF logo and a menu with links: 'Intégration HTML', 'CSS', 'JavaScript', 'Php', and 'Autres ...'. Below the navigation bar, there is a form for calculating the product of two numbers. The form consists of two input fields labeled 'Nombre 1:' and 'Nombre 2:', with a multiplication symbol 'x' between them. Below the input fields is a button labeled 'Calculer le produit'. At the bottom of the form, there is an empty input field for the result.

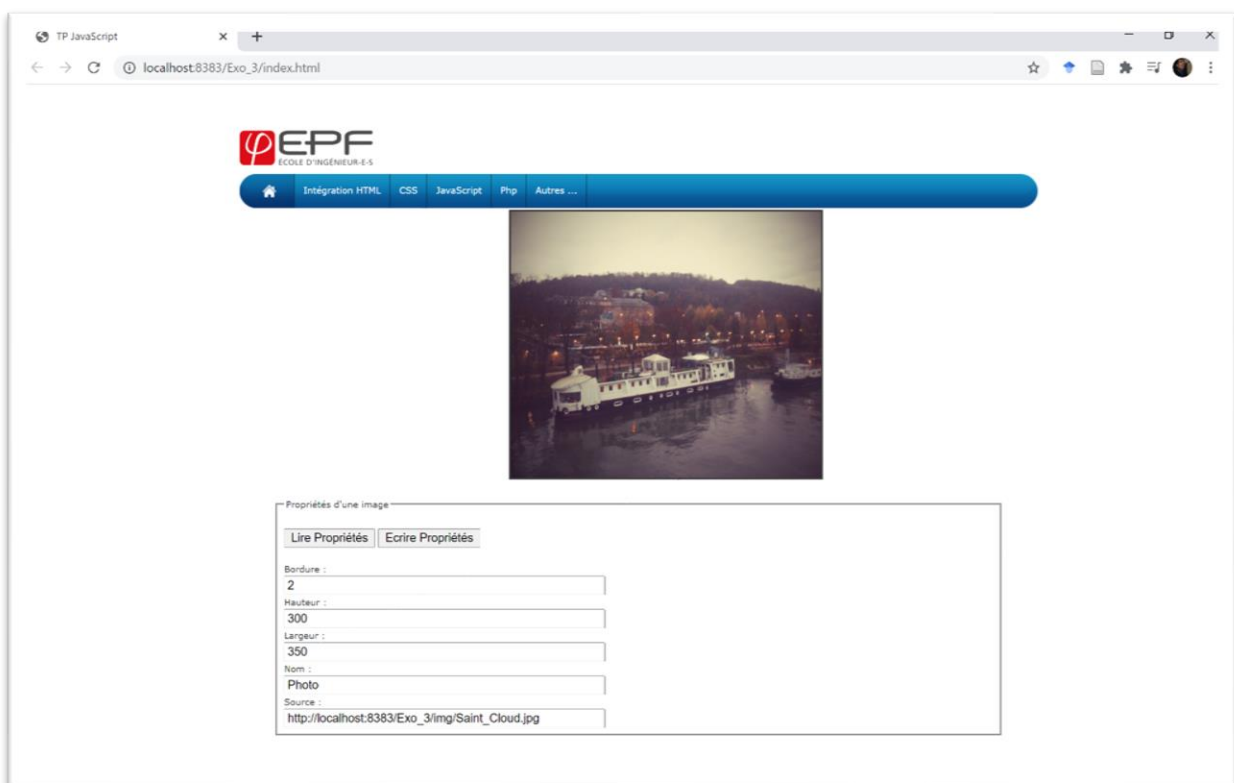
## **Exercice 3**

Dans cet exercice vous allez créer une page web qui permet de choisir son menu au restaurant : une entrée, un plat principal et un dessert. L'appui sur le bouton «Choisir» permet de calculer l'addition comme il est montré sur la figure ci-après :



## Exercice 4

Dans cet exercice on vous demande d'écrire deux fonctions JavaScript qui permettent d'agir sur une image d'une page web. La première fonction permet de lire et d'afficher les paramètres de l'image (i.e. largeur de l'image, hauteur de l'image, etc.). La deuxième fonction permet de modifier les paramètres de l'image (i.e. modifier la largeur de l'image, la hauteur de l'image, etc.).



Sur l'exemple de la figure ci-dessus, le bouton « Lire Propriétés » est associé à une fonction JavaScript qui permet d'agir sur l'image et de récupérer ses paramètres : bordure, hauteur,

largeur, nom et la source. Le bouton « Ecrire Propriétés » déclenche une fonction JavaScript qui modifie les paramètres de l'image en appliquant les valeurs données dans les champs de textes.

### **Exercice 5 (facultatif)**

On se propose d'écrire sous JavaScript des fonctions permettant de trier, dans l'ordre croissant, les éléments d'un tableau d'entiers.

Il existe différents algorithmes de tri : tri par sélection, tri par insertion, tri à bulles, etc. Selon, votre choix écrivez un code JavaScript qui permet de trier un tableau aléatoire et de l'afficher dans la page web.

## Annexe 1 : Quelques algorithmes de tri

### I.1. Tri par Sélection.

#### I.1.1.Principe.

On parcourt le tableau de gauche à droite (avec un indice  $i$ ), à chaque étape, on échange l'élément situé en  $i$  avec le minimum du tableau à partir de  $i$  (i.e. on sélectionne le minimum).

#### I.1.2.Exemple.

Les éléments permutés sont indiqués en rouge et les éléments déjà classés en vert.

4	3	6	1	2	8
1	3	6	4	2	8
1	2	6	4	3	8
1	2	3	4	6	8
1	2	3	4	6	8
1	2	3	4	6	8

#### I.1.3.Algorithme.

```

Procédure Selection();
Var
    Indice_Min, Min, i, j;
Début
    (* Tri des éléments du Tableau *)
    Pour i ← 1 à (N-1) faire
        Min ← T[i];
        Indice_Min ← i;
        Pour j ← i+1 à (N) faire
            Si Min > T[j] alors
                Min ← T[j];
                Indice_Min ← j;
            FinSi
        FinPour j
        Permuter(i, Indice_Min);
    FinPour i
Fin

```

Algorithme Tri par Sélection.

### I.2. Tri par Insertion.

#### I.2.1. Principe.

Le tri par insertion consiste à parcourir le tableau de gauche à droite, en maintenant le sous-tableau, situé à gauche de  $i$  trié. On cherche ensuite à *insérer* le  $i^{\text{ème}}$  élément dans sous-tableau, en faisant « descendre » vers la gauche dans le sous-tableau trié.

#### I.2.2. Exemple.

L'élément qu'on va insérer (i) est en vert, et les comparaisons en rouge.

4	3	6	1	2	8
---	---	---	---	---	---

[illegible]

### I.2.3. Algorithme

```

Procédure Insertion() ;
Var
    i,j ;
Début
    (* Tri des éléments du Tableau *)
    Pour i ← 1 à (N) faire
        Pour j ← i à (1) faire
            Si T[j-1] > T[j] alors
                Permuter(T[j],T[j-1]);
            FinSi
        FinPour j
    FinPour i
Fin

```

### Algorithme Tri par Insertion