

UN PREMIER PAS DANS LE LANGAGES JAVASRIPT

A graphic of a computer window with a black background. Inside, there is a snippet of JavaScript code. The code defines a function with a parameter and some internal logic.

```
function nombre_funcion(parametros){  
    instrucciones;  
}
```

OBJECTIFS

- Comprendre les principes de base de la programmation JavaScript
- Rendre une page HTML dynamique
- Savoir mettre en application des fonctions simples
- Savoir créer des fonctions permettant de rendre une page HTML dynamique

SITES DE RÉFÉRENCE

→ MathSoup – Johan Segura – Pr. de Maths

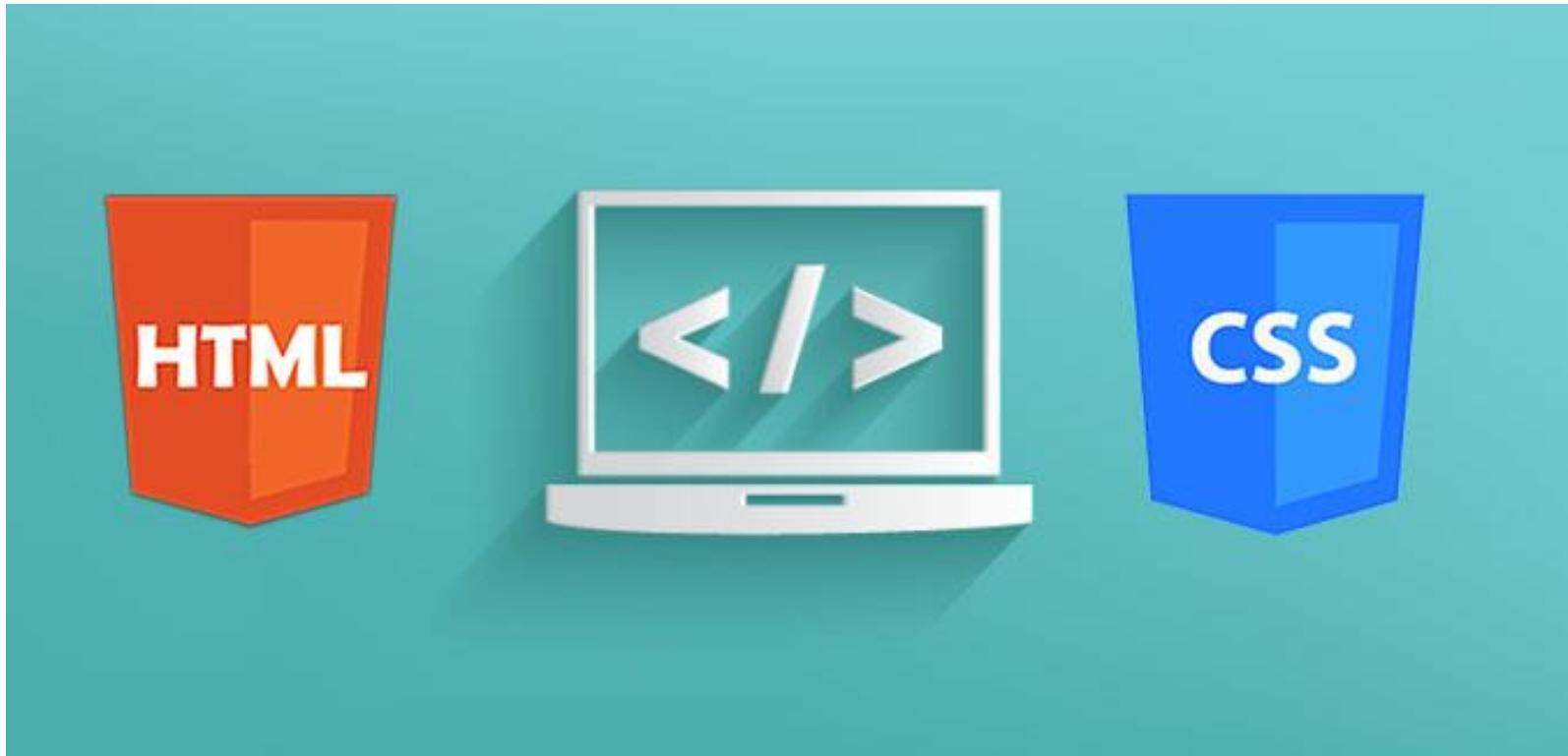
<http://mathsoup.xyz/lycee/index.php?d=ICN-ISN&c=JavaScript>

→ OpenClassRoom

<https://openclassrooms.com/fr/courses/6175841-apprenez-a-programmer-avec-javascript/6278709-demarrez-votre-programme-avec-la-fonction-main>

PRÉREQUIS

Savoir programmer en HTML et CSS



Chapitre 1

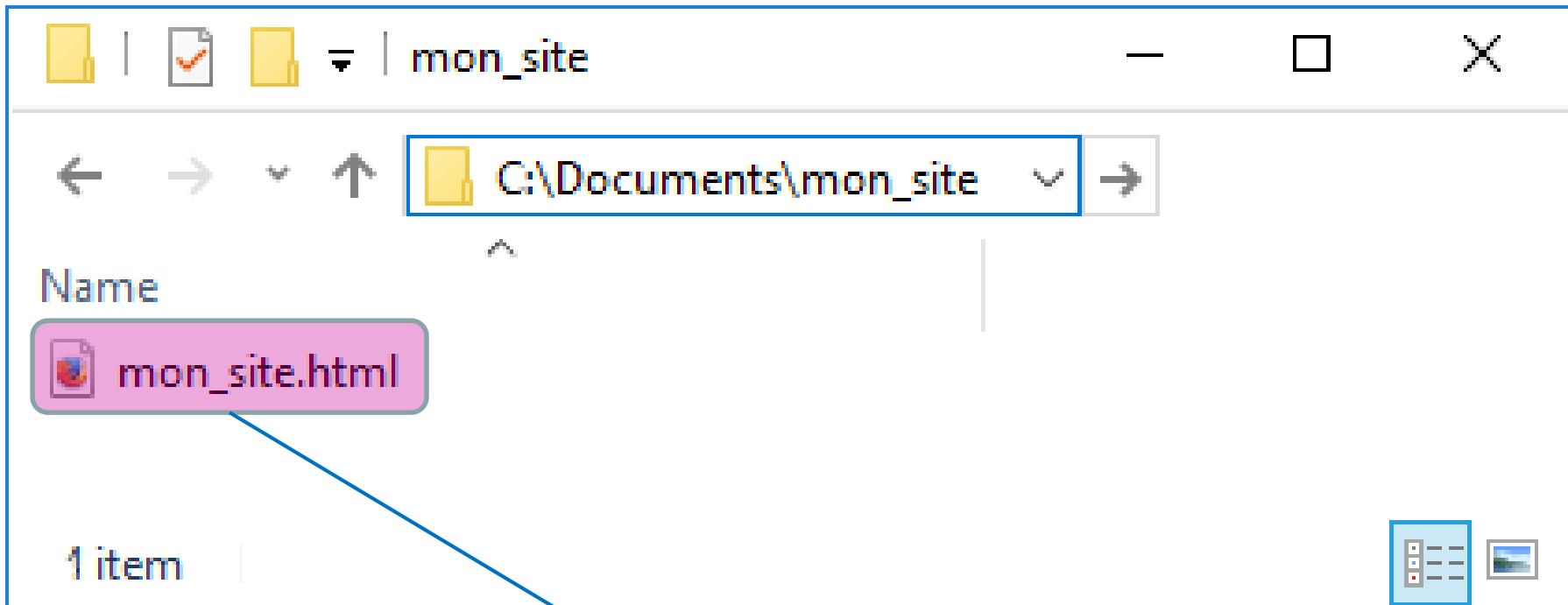
Un environnement de développement

INTRODUCTION - OBJECTIFS

Présenter des outils et solutions pour programmer efficacement en JavaScript :

- ➡ **Notepad++** : Un **éditeur** facilitant la programmation avec *coloration syntaxique* et de nombreuses fonctionnalités
- ➡ Un **inspecteur** permettant de sélectionner et d'éditer une partie de la page web
- ➡ Une **console** permettant de tester du code et d'interagir avec la page web sur laquelle on travaille
- ➡ Un **débogueur** permettant d'exécuter un programme ligne par ligne

I. UN PEU D'ORGANISATION



Une page web peut se résumer au minimum en un unique fichier .html.

mon_site.html :

```
1 <html>
2   <head>
3     <style type="text/css">
4       ... le style CSS se met ici
5     </style>
6
7     <script type="text/javascript">
8       ... le code JavaScript se met ici
9     </script>
10
11   </head>
12
13   <body>
14     ... la structure HTML ici
15   </body>
16
17 </html>
```

length : 252 lines : 17 Ln : 1 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 INS

La structure principale du site entre les balises<body> permettant de détailler le squelette du site en HTML

The screenshot shows an open Notepad++ window displaying an HTML file named "mon_site.html". The code is as follows:

```
1 <html>
2   <head>
3     <style type="text/css">
4       ... le style CSS se met ici
5     </style>
6
7     <script type="text/javascript">
8       ... le code JavaScript se met ici
9     </script>
10
11   </head>
12
13   <body>
14     ... la structure HTML ici
15   </body>
16
17 </html>
```

A callout bubble highlights the CSS section between lines 3 and 5, with the text "... le style CSS se met ici". Another callout bubble highlights the JavaScript section between lines 7 and 9, with the text "... le code JavaScript se met ici".

Une zone de style entre les balises `<styles>` permettant de soigner la forme de la page dans le langage CSS.

C:\Documents\mon_site\mon_site.html - Notepad++

Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution TextFX Compléments

```
1 <html>
2   <head>
3     <style type="text/css">
4       ... le style CSS se met ici
5     </style>
6
7     <script type="text/javascript">
8       ... le code JavaScript se met ici
9     </script>
10
11   </head>
12
13   <body>
14     ... la structure HTML ici
15   </body>
16
17 </html>
```

length : 252 lines : 17 Ln : 1 Col : 1 Sel : 0 | 0 Dos\Windows UTF-8 INS

Une zone de script entre les balises `<script>` permettant de dynamiser et de gérer les interactions dans le langage JavaScript.

Question:

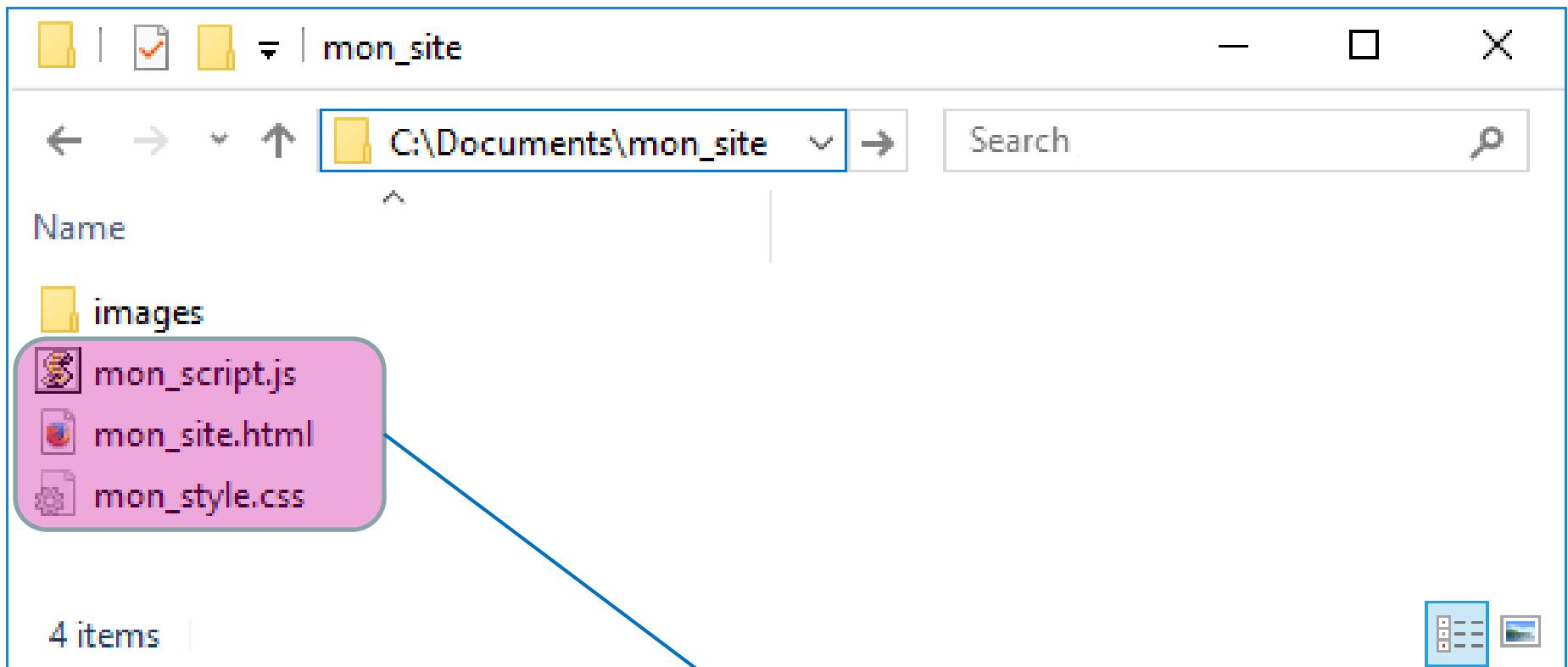
Au vu des cours précédents
sur le HTML/CSS,
pensez-vous que ce soit une
bonne manière d'écrire et de
gérer le site dans un seul et
même fichier
« mon_site.html »?

Réponse :

NON !!!

- ➡ Le fichier peut vite devenir très long et le programmeur finira par se sentir perdu dans son propre code
- ➡ Les éditeurs de développement sont généralement adaptés à un langage unique et ne rempliront pas leur rôle éclairant sur trois langages à la fois (HTML, CSS, JavaScript).
- ➡ Lorsque l'on travaille sur la CSS, par exemple, on n'a pas besoin de voir le code JavaScript et inversement.
- ➡ Les messages d'erreurs permettant de déboguer un programme ne renverront pas à la bonne ligne.

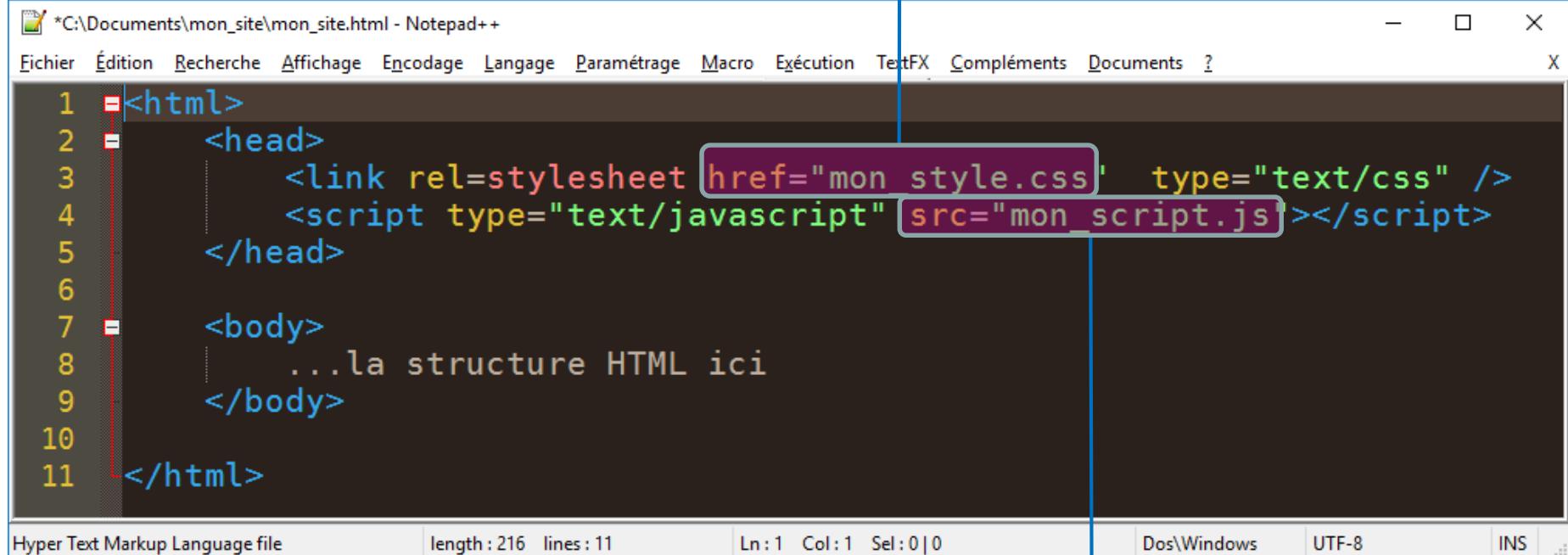
Solution:



Il est important de séparer chacun des langages dans des fichiers séparés.

Remarque :

Appel aux fichier CSS



The screenshot shows an open Notepad++ window displaying an HTML file named "mon_site.html". The code includes a head section with a CSS link and a script tag, and a body section containing placeholder text. The "mon_style.css" and "mon_script.js" files are highlighted in purple, indicating they are external resources.

```
<html>
    <head>
        <link rel="stylesheet" href="mon_style.css" type="text/css" />
        <script type="text/javascript" src="mon_script.js"></script>
    </head>
    <body>
        ...la structure HTML ici
    </body>
</html>
```

Appel aux fichier JavaScript

Les chemins vers la feuille de style et le script (*attributs href et src*) peut aussi être une adresse internet...

Exercice 1:

- Crée un dossier *Mon site* à l'intérieur duquel vous allez placer les trois fichiers à télécharger.
 - ↳ Cliquer sur le lien suivant ...
http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/ressources/_exemple_site/exemple_site.zip
- Visualisez ensuite le site *mon_site.html* avec le navigateur Firefox.
 - ↳ Clic-droit sur le fichier "mon_site.html", puis "Ouvrir avec" et choisir "Firefox"

II. L'éditeur : Notepad++

2.1 Première utilisation



Editeur de texte libre générique adapté à de nombreux langages de programmations. Il propose notamment :

- ➡ **La coloration syntaxique** de code pour le HTML, le CSS et le JavaScript (entre autres)

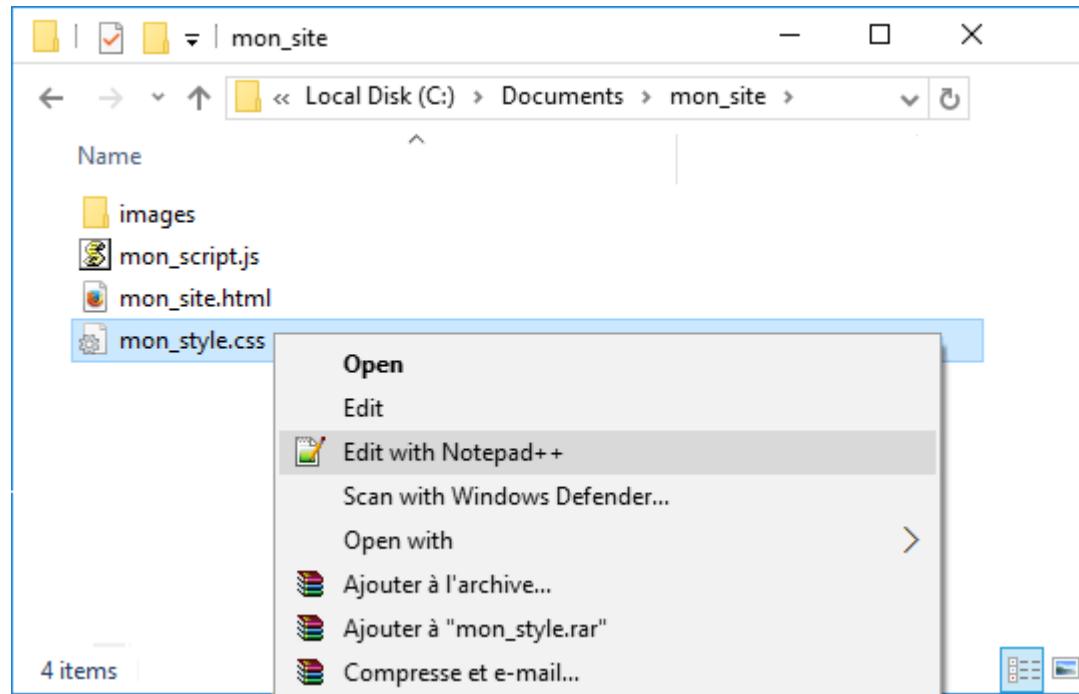
- ➡ Les différentes parties du code (variables, fonctions, balises,etc.) sont reconnues et colorées par l'éditeur pour rendre la lecture plus intuitive...



- La numérotation des lignes et des colonnes pour repérer rapidement les erreurs indiquées par le débogueur
- Un système d'**autocomplétion** permettant notamment de fermer les balises HTML (très pratique).
 - ↳ L'éditeur essaye de deviner la fin des mots que vous tapez pour accélérer la saisie
- L'exécution rapide de programmes et des raccourcis

Exercice 2:

→ Editez les trois fichiers *mon_site.html*, *mon_style.css* et *mon_script.js*.



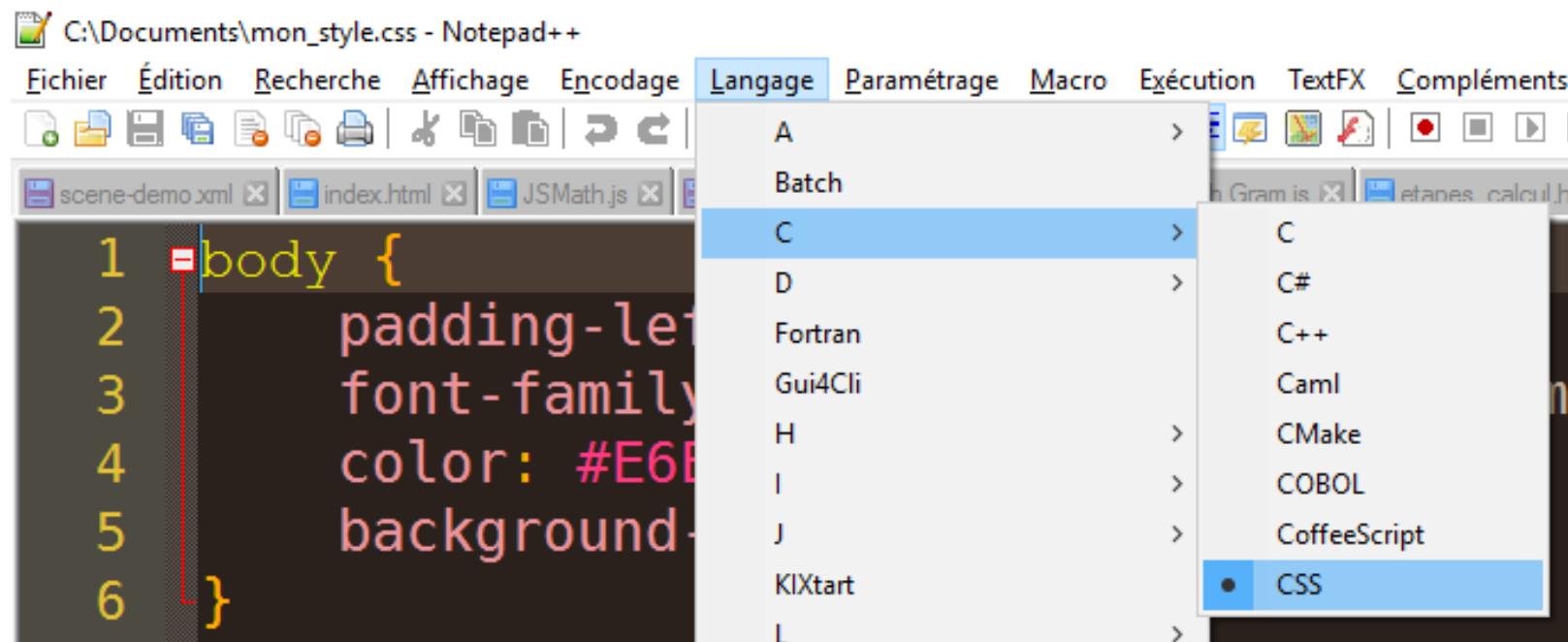
Pour éditer un fichier avec notepad++ : faire clic-droit sur le fichier à ouvrir, puis "Ouvrir avec Notepad++"

Exercice 2 (suite):

- Modifiez le titre dans le fichier mon_site.html et **sauvegardez.**
 - ↳ Pour sauvegarder : faire "Fichier" puis "Enregistrer" ou utiliser le raccourci clavier "Ctrl + S")
- Visualiser le résultat dans le navigateur en **Rafraîchissant** la page à l'aide du bouton REFRESH ou avec F5
- Il est aussi possible de lancer directement Firefox depuis Notepad++ à l'aide du menu *Exécution* puis *Launch with Firefox*

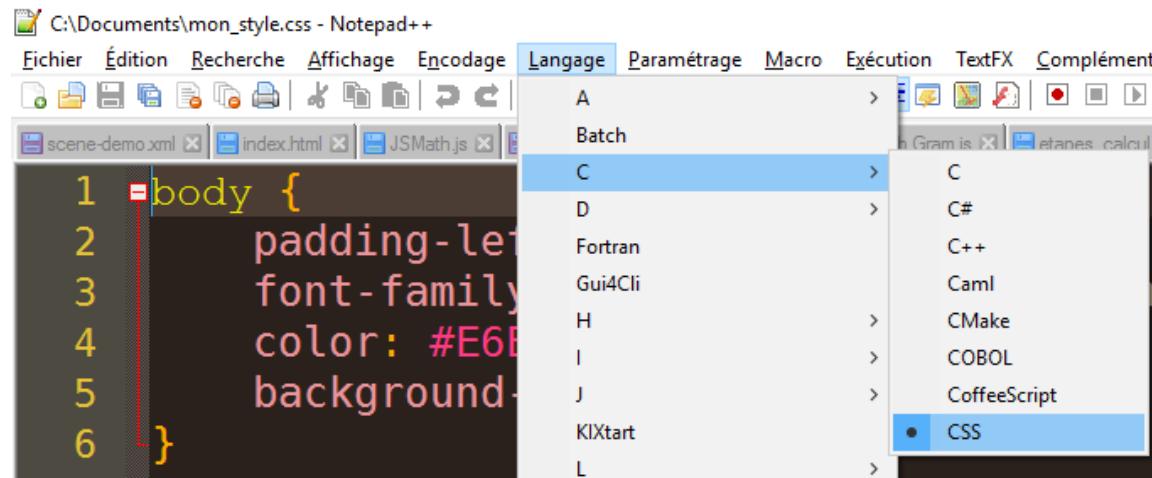
2.2 Choix du langage

Normalement, le langage est détecté automatiquement. Si ce n'est pas le cas (par exemple si le fichier contient plusieurs langages), il est possible de le sélectionner dans le menu *Langage*.



Exercice 3:

→ Vérifiez que pour chacun des trois fichiers le langage est correctement détecté dans le menu langage (pour les trois fichiers HTML, CSS et JavaScript).



→ Vérifiez que pour le HTML l'autocomplétion ferme automatiquement les balises (sinon activez l'option Autoclose dans le menu *TextFX* puis *TextFX Settings*)

2.3 Quelques paramètres et raccourcis

Les raccourcis les plus utiles :

- **Ctrl+C et Ctrl+V** pour **Copier et Coller.**
- **Ctrl+X** pour **Couper**
- **Alt-Tabulation** permet de passer rapidement d'une fenêtre à une autre (de l'éditeur au navigateur quand on programme).

Quelques raccourcis supplémentaires:

- **Duppliquer une ligne** : Ctrl + d
- **Sélectionner du texte sans la souris** : Shift + flèches directionnelles
- **Décaler une ou plusieurs lignes** : Shift + Ctrl + flèches directionnelles
- **Faire défiler verticalement** : Ctrl + flèches haut et bas
- **Déplacer le curseur mot à mot** : Ctrl + flèches droite et gauche
- **Indenter une ligne ou plusieurs** : Quand une ligne est sélectionnée, Tabulation
- **Dé-indenter une ligne ou plusieurs** : Quand une ligne est sélectionnée, Shift+Tabulation

Affichage et économie d'énergie

Notepad++ propose différents styles d'affichage:

- *Paramétrage > Configurateur de couleurs syntaxiques*
- *Settings > Style configurator* (si interface en anglais)

Le *Thème Bespin* est agréable.

III. Les outils de développement de Firefox

3.1 Introduction

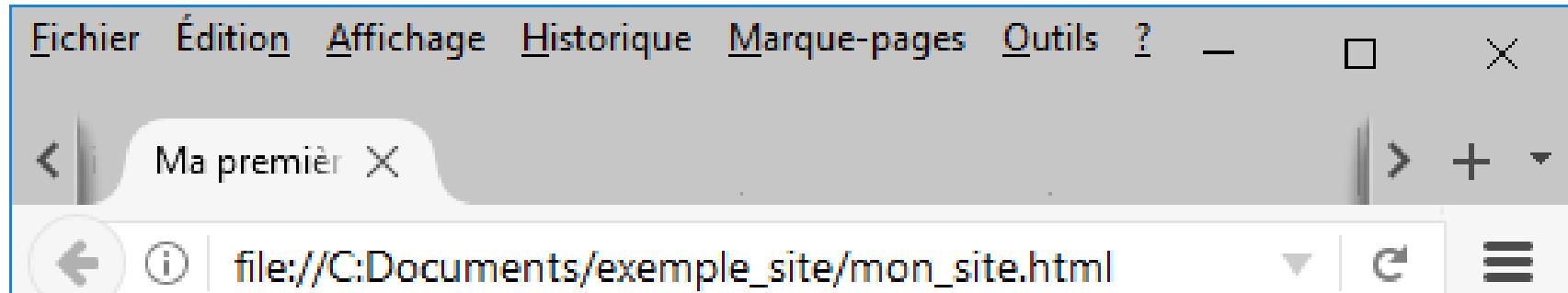
Les **navigateurs web**, dans leur utilisation habituelle, se placent du point de vue de l'utilisateur (internaute).



Ils proposent également des outils de développement très utiles pour les *développeurs web* (programmeurs).

Barre de menus

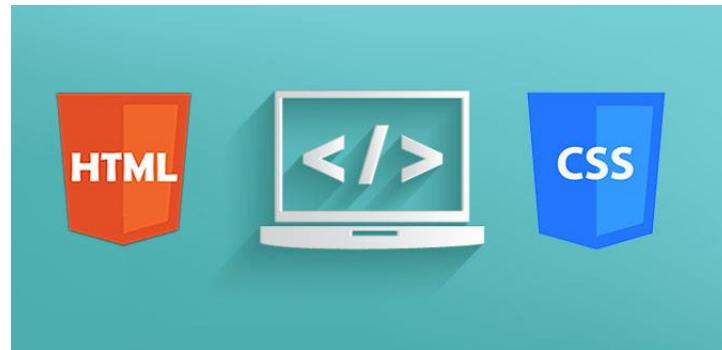
Commencez par vérifier que vous accédez à la **barre de menus** de Firefox (pour accéder aux menus Ficher, Edition, etc.).



Si ce n'est pas le cas, en haut de la fenêtre, faire **clic-droit** puis **Barre de menus**.

3.2 Code source d'une page

On appelle **code source** le programme qui est derrière le rendu graphique d'une application.



- ➡ Le HTML, le CSS et le JavaScript sont des langages interprétés (ils sont exécutés au fur et à mesure par l'interpréteur).
- ➡ Leur code source est public et accessible par n'importe qui.

Comment accéder au code source ?

Environnement de développement +

mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/TD - IDE.html

MAcadémic lycée Stex Atrium ARENA BqPost BqPop SMC W Wpéd M gmail Y y_mail mailFree fbk météo_marseille Processing sTokens google ecosia TKebab

1. CODE SOURCE

DÉFINITION

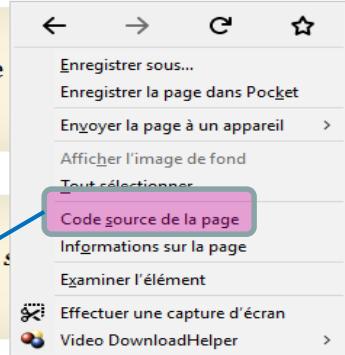
On appelle **code source** le programme qui est derrière le rendu graphique d'une application.

REMARQUE

Dans certains langages compilés ⁽⁷⁾, le code source est caché. Le HTML, le CSS et les langages interprétés ⁽⁸⁾. Leur code source est public et accessible par n'importe qui.

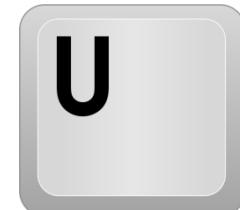
MÉTHODE

Pour afficher le code source d'une page web, faire *Clic-droit* sur la page, puis *Code source de la page* (cliquer sur le background) ou directement par le raccourci clavier **Ctrl+U**.



Clic-droit sur la page, puis Code source de la page

Sinon :



Exercice 4:

Afficher le code source des page suivantes:

- http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/TD%20-%20IDE.html#footnote_6
- https://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal
- <https://www.google.fr/>

Remarque:

Les développeurs web peuvent ainsi facilement s'inspirer du code des autres.... lorsqu'il est lisible.

3.3 Inspecteur de FireFox



L'**inspecteur** est un outil qui permet de visualiser le code d'un élément précis de la page.

Pour ouvrir l'**inspecteur**, il y a plusieurs manières :

- menu **Outils** puis **Développement web**, puis **Inspecteur**
- En cliquant sur  , puis  , puis **Inspecteur**
- Avec le raccourci clavier **Ctrl+Shift+C**
- En faisant **Clic-droit** sur un élément de la page puis **Examiner l'élément**

Exemple:

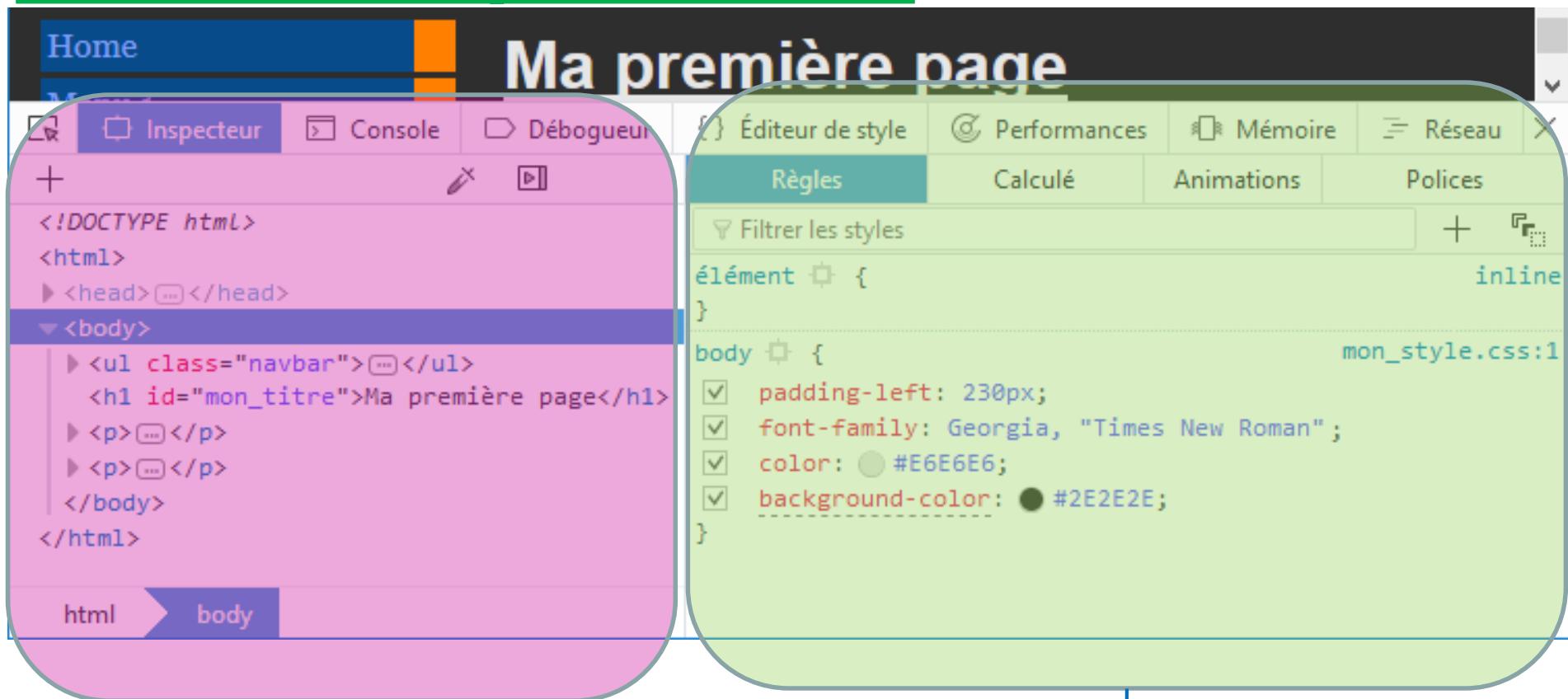
Aller sur l'exercice 7 du site « Mathsoup »:

→ <http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/TD%20-%20IDE.html>

Inspectez l'image suivante :



Présentation de l'inspecteur de FireFox



HTML : il est possible de modifier n'importe quelle partie. Notamment le texte de la page.

CSS : il est possible de modifier les attributs CSS et d'en ajouter

Cela permet aux développeurs de visualiser directement le rendu de certaines modifications avant de se lancer dans la programmation.

Exercice 5: Aller sur l'exercice 8 du site « Mathsoup » :

<http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/TD%20-%20IDE.html>

→ Utilisez l'inspecteur, du côté HTML, pour :

- ↳ Modifier un texte de cet énoncé
- ↳ Mettre ce **mot** en italique plutôt qu'en gras.
- ↳ Ajouter un élément à cette liste de questions
- ↳ Modifier la largeur de l'image de l'inspecteur

→ Utilisez l'inspecteur, du côté CSS, pour :

- ↳ Modifier la taille du texte dans les *div* de classe *consignes*
- ↳ Modifier la largeur et la hauteur du logo *Google* sur la page d'accueil de [Google](#)

3.4 Console de FireFox

Utilité :

- Affiche les évènements de la page dont les erreurs JavaScript et CSS.
 - Permet d'exécuter des commandes JavaScript directement.
-
- 
- Très utile au développeur qui veut tester des petits morceaux de code.

Pour ouvrir **la console**, il y a plusieurs manières :

→ menu **Outils** puis **Développement web**, puis **Console**



→ En cliquant sur , puis , puis **Console**



→ Avec le raccourci clavier ***Ctrl+Shift+K***

→ Lorsque l'inspecteur est déjà ouvert, en cliquant sur l'onglet **console**



Exercice 6: Aller sur l'exercice 9 du site « Mathsoup » :

<http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/TD%20-%20IDE.html>

→ Dans le navigateur, revenez sur la page "mon_site.html" et ouvrez la console.

↳ Cliquer sur le lien suivant ...

http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/ressources/_exemple_site/exemple_site.zip

→ Activez l'affichage des erreurs CSS en cochant le bouton CSS dans la console :



→ Corrigez les deux erreurs qui empêchent le titre de s'afficher correctement dans le fichier "mon_style.css".

Zone de saisie de la console

- La console comprend une **zone de saisie** tout en bas pour tester et exécuter des commandes en JavaScript

➤ Taper une commande en JS ici

- Il est possible d'exécuter des **commandes simples** ou d'appeler des **fonctions** contenus dans les scripts JS de la page. L'autocomplétion assiste la saisie. On valide avec *Entrée* et le résultat s'affiche au dessus.

Exercice 7:

→ Dans le navigateur, revenez sur la page "mon_site.html" et ouvrez la console.

↳ Cliquer sur le lien suivant ...

http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/ressources/_exemple_site/exemple_site.zip

1 Tapez l'une après l'autre les commandes suivantes :

x=6

y=x+1

x

y

y=x++

X

y

2

Lancez une alerte JavaScript en tapant :

```
alert("Attention derrière toi c'est  
horrible ! !");
```

3

Le fichier "mon_script.js" contient une fonction *BONJOUR()*. Appelez cette fonction en tapant dans la console :

BONJOUR()

4

Il est aussi possible d'agir sur les éléments de la page avec JS (nous reverrons ça plus tard). Tapez par exemple :

```
document.querySelector("body").style.backgrou
```

```
nColor="red"
```

```
document.  
getElementById("mon_titre").innerHTML="Un  
titre bien plus accrocheur"
```

Remarques

- Certaines de ces commandes marchent sur n'importe quelle page web (affectations, alertes)
- La fonction *BONJOUR()* vient du fichier "mon_script.js" qui est dans votre dossier personnel.
 - ↳ Cette fonction ne marchera pas sur un autre site
- La fonction *BONJOUR()* ne peut marcher sur un autre site que si il contient un élément d'id "mon_titre"
- Certains développeurs facétieux cachent des easter egg en JS qui se révèlent par la console...

Messages d'erreur

- La console affiche les **messages d'erreur** lorsqu'un programme JavaScript est bogué
 - ↳ Le **type de l'erreur** (faute de frappe, parenthèse manquante, etc.) est indiqué
 - ↳ Le **numéro de la ligne et de la colonne** de l'erreur est aussi indiqué

Exercice 8:

→ Ouvrir le fichier "mon_script.js" »

↳ Cliquer sur le lien suivant ...

http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/ressources/_exemple_site/exemple_site.zip

→ Le fichier "mon_script.js" contient une partie mise en commentaire entre les caractères /* et */. Cette partie contient des erreurs, c'est pourquoi elle a été commentée.

- 1 Décommentez cette partie du programme.
- 2 En rafraîchissant la page "mon_site.html", allez voir les messages d'erreur dans la console.
- 3 Corrigez les erreurs faites, et relancez pour voir le résultat (indice : "missing" signifie "manquant").

3.5 Le Débogueur de FireFox

Utilité :

- Le débogueur permet **d'avancer pas à pas** dans un programme JavaScript pour examiner plus en détail, comprendre ce qui fonctionne mal et le corriger.

Pour ouvrir le **débogueur**, il y a plusieurs manières :

- menu **Outils** puis **Développement web**, puis **Débogueur**
- En cliquant sur  , puis  , puis **Débogueur**
- Avec le raccourci clavier **Ctrl+Shift+K**
- Lorsque l'inspecteur est déjà ouvert, en cliquant sur l'onglet **Débogueur**



Activer le débogueur

- Pour exécuter un programme **pas à pas**, il faut placer des **points d'arrêts** sur chaque ligne où souhaite que le programme marque une pause :

The screenshot shows the 'Débogueur' tab selected in the browser's developer tools. On the left, the 'Sources' panel lists a file named 'mon_script.js'. The code editor on the right displays the following JavaScript function:

```
19 function boucle_test(){
20     var N=prompt("Nombre d'itérations souhaité");
21     N=parseInt(N); //parseInt() convertit N en nombre entier (Int) car prompt renvoie une chaîne de caractères
22     var p=document.querySelector("#fin"); //selectionne le dernier paragraphe d'un document
23     var text="Voici une énumération : <br/>";
24     p.innerHTML=text;
25     for(var i=0;i<N;i++){
26         text=p.innerHTML;
27         text=text + "Numéro " + i.toString() + "<br/>"; //toString() convertit i en une chaîne de caractères
28         p.innerHTML=text;
29     }
30     return;
31 }
32 }
```

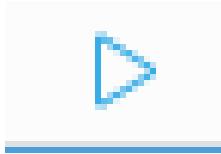
Breakpoints are visible as small red arrows pointing to the left margin next to lines 23, 24, 26, and 28. A pink callout box at the bottom left identifies the script as 'nom du script analysé mon_script.js'. A green callout box at the bottom right details the breakpoints: 'Des points d'arrêts sont placés sur les lignes 23, 24, puis 26 à 28.'

nom du script analysé
mon_script.js

Détail du script
Des points d'arrêts sont placés sur les
lignes 23, 24, puis 26 à 28.

Activer le débogueur

- Pour **ajouter ou enlever un point d'arrêt**, il suffit de cliquer sur le numéro de la ligne.
- Pour **avancer jusqu'au prochain point d'arrêt**, cliquer sur



- Pour **avancer d'une seule ligne de code**, cliquer sur



Exercice 9:

→ Ouvrir le fichier "mon_script.js »

↳ Cliquer sur le lien suivant ...

http://mathsoup.xyz/mathsoup.xyz/content/NSI/JavaScript/ressources/_exemple_site/exemple_site.zip

→ Le script *mon_script.js* contient une fonction **boucle_test()** que nous allons étudier en détail à l'aide du débogueur :

- 1 Dans la **console**, exécutez la fonction *boucle_test()* pour observer son résultat.
- 2 Rechargez la page, puis dans le **débogueur** placez des points d'arrêts sur les lignes de l'image précédente.
- 3 Observez le fonctionnement de la boucle pas à pas, et les changements des variables dans la partie droite du débogueur.
- 4 En vous inspirant fortement de la fonction *boucle_test*, créez une fonction *boucle_menu()* qui modifie le menu de navigation à gauche de la page web pour y ajouter autant de choix que l'on décide : *Menu 1*, *Menu 2*, *Menu 3* etc.

Indications

- Copier-coller la fonction *boucle_test()* et l'appeler *boucle_menu*
- Il faut sélectionner le menu de navigation qui est un élément ** d'*id = navbar*
- Il faut ajouter les titre des menus dans dans la liste, entre balises ** et **.
- Vous pouvez choisir avant la boucle d'affecter la chaîne vide "" à la variable *text* si vous ne voulez pas laisser le texte "*Voici une énumération*"
- Sauvegardez et rechargez la page web pour que Firefox tienne compte des changements apportés à *mon_script.js*

Rendu

- Menu 0
- Menu 1
- Menu 2
- Menu 3
- Menu 4
- Menu 5
- Menu 6
- Menu 7
- Menu 8
- Menu 9
- Menu 10
- Menu 11
- Menu 12
- Menu 13
- Menu 14
- Menu 15

Ma première pa

Bienvenue sur ma page. Pour l'ins

Lorem ipsum dolor sit amet, cons
 varius turpis tincidunt. Quisque c
 fringilla scelerisque enim, vel con
 sequat consequat. Nullam tinc