

Polytech Lyon 3A	TP Web N°3	Javascript	2025-2026
------------------------	---------------	-------------------	-----------

Objectifs :

Première initiation au langage Javascript.

Comme dans les deux premiers TP Web les différentes réalisations seront regroupées et publiées sur un serveur Web.

Pré-requis :

HTML et CSS (TP Web n°1 et 2).

Conditions de réalisation :

Durée : 3 heures

Matériel :

- Une machine Windows/Linux reliée au réseau Intranet;

Logiciels :

- le navigateur **Firefox** (obligatoire) ;
- le logiciel de développement **Visual Studio Code** ;
- le système d'exploitation sur lequel le serveur Web (**apache** ou **nginx**) est installé et lancé.

Documentation :

Les sites : www.w3schools.com et <https://developer.mozilla.org/fr/docs/Web/JavaScript>.

Sommaire

1	JavaScript.....	1
2	Utilisation de la « Console » de Firefox	2
3	Utilisation du « Débogueur » de Firefox	3
4	Développement de la fonction <i>Range(stop, [start], [step])</i>	4
5	Insertion de la page <i>exercices.html</i> dans notre site Web.....	5
6	Ampoule.....	5
7	Planètes (facultatif)	6
7.1	Evénements javascript	6
7.2	Les événements <i>onmouseover</i> et <i>onmouseout</i>	6

1 JavaScript

JavaScript est **LE** langage de script du web.

JavaScript est utilisé dans des milliards de pages Web pour ajouter des fonctionnalités aux pages HTML, pour valider les données saisies dans les formulaires, pour communiquer avec le serveur (avec AJAX par exemple), et bien plus encore.

Tout en javascript est objet : une chaîne, un nombre, un tableau, une date,...

Avec HTML DOM (Document Object Model), javascript peut accéder à tous les éléments d'un document HTML.

Quand une page HTML est chargée, le navigateur crée un **Document Object Model** de la page.

Avec HTML DOM, javascript a toute la puissance dont il a besoin pour créer des pages HTML dynamique :

- il peut changer tous les éléments HTML de la page,
- il peut changer tous les attributs HTML de la page,

Polytech Lyon 3A	TP Web N°3	Javascript	2025-2026
------------------------	---------------	-------------------	-----------

- il peut changer tous les styles CSS de la page,
- il peut réagir à tous les événements de la page.

Nous vous proposons ici une première initiation au langage **javascript**. Nous allons toujours beaucoup nous appuyer sur le site www.w3schools.com, mais aussi sur le site de référence du langage javascript : <https://developer.mozilla.org/fr/docs/Web/JavaScript>.

2 Utilisation de la « Console » de Firefox

On vous fournit les fichiers *exercices.html* et *exercices.js*. Copier ces deux fichiers dans un répertoire et ouvrir le fichier *exercices.html* avec le navigateur **Firefox** (obligatoire !).

Appuyer sur la touche F12. Des **outils de développement intégrés à Firefox** s'affichent dans le bas de la page. Aller dans l'onglet **Console** comme indiqué ci-dessous et taper les instructions suivantes :

```
a = 5 ;
b = 3*a + 2 ;
print("b = " + b) ;
a
```

The screenshot shows the Firefox Developer Tools interface with the 'Console' tab selected. The title of the page is 'Exercices javascript'. Below the title, there is a section for 'Fibonacci' with a description 'Calcul de la suite de Fibonacci de rang "n":'. There is an input field for 'Rentrer le rang à calculer :', a 'Calculer Fibonacci' button, and a status message 'Résultat du calcul de la suite de Fibonacci en attente...'. The console shows the following commands and their outputs:

```
>> a = 5;
<< 5
>> b = 3*a + 2;
<< 17
>> print("b = " + b);
<< "b = 17"
>> a
<< 5
>>
```

Ainsi on peut dans la Console de Firefox tester des instructions javascript !

Exécuter les commandes suivantes dans la console pour déterminer les principaux types de JavaScript :

```
typeof "John";
```

Polytech Lyon 3A	TP Web N°3	JavaScript	2025-2026
------------------------	---------------	------------	-----------

```

typeof 3.14;
typeof NaN;
typeof false;

typeof {};
typeof {name:'John', age:34};
({}) instanceof Object;

typeof [1,2];
[1,2] instanceof Array;
[1,2] instanceof Object;

typeof new Date();
new Date() instanceof Date;
new Date() instanceof Object;

typeof function () {};
typeof (() => {});
(() => {}) instanceof Function;
(() => {}) instanceof Object;

typeof un_truc;
typeof null;
typeof undefined;

```

3 Utilisation du « Débogueur » de Firefox

Tester la suite de Fibonacci depuis la page *exercices.html* avec différentes valeurs.

Puis retourner dans la Console de Firefox et tester directement la fonction ***Fibonacci*** depuis la Console :

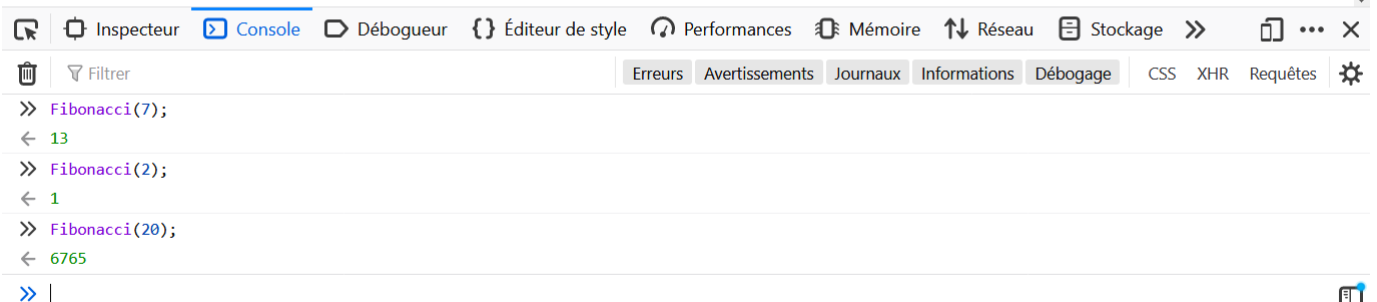
Exercices javascript

Fibonacci

Calcul de la suite de Fibonacci de rang "n" :

Rentrer le rang à calculer : Calculer Fibonacci

Résultat : Fibonacci(7) = 13



Polytech Lyon 3A	TP Web N°3	Javascript	2025-2026
------------------------	---------------	------------	-----------

Examiner le code html de la page *exercices.html* et le code javascript de *exercices.js*. En particulier étudier la fonction *Fibonacci*.

Aller dans l'onglet « *Débogueur* » des outils de développement de Firefox. Nous allons exécuter en mode débogage, c'est-à-dire pas-à-pas ou instruction-par-instruction, une partie de la fonction *Fibonacci*.

Sélectionner le fichier *exercices.js* dans la partie à gauche.

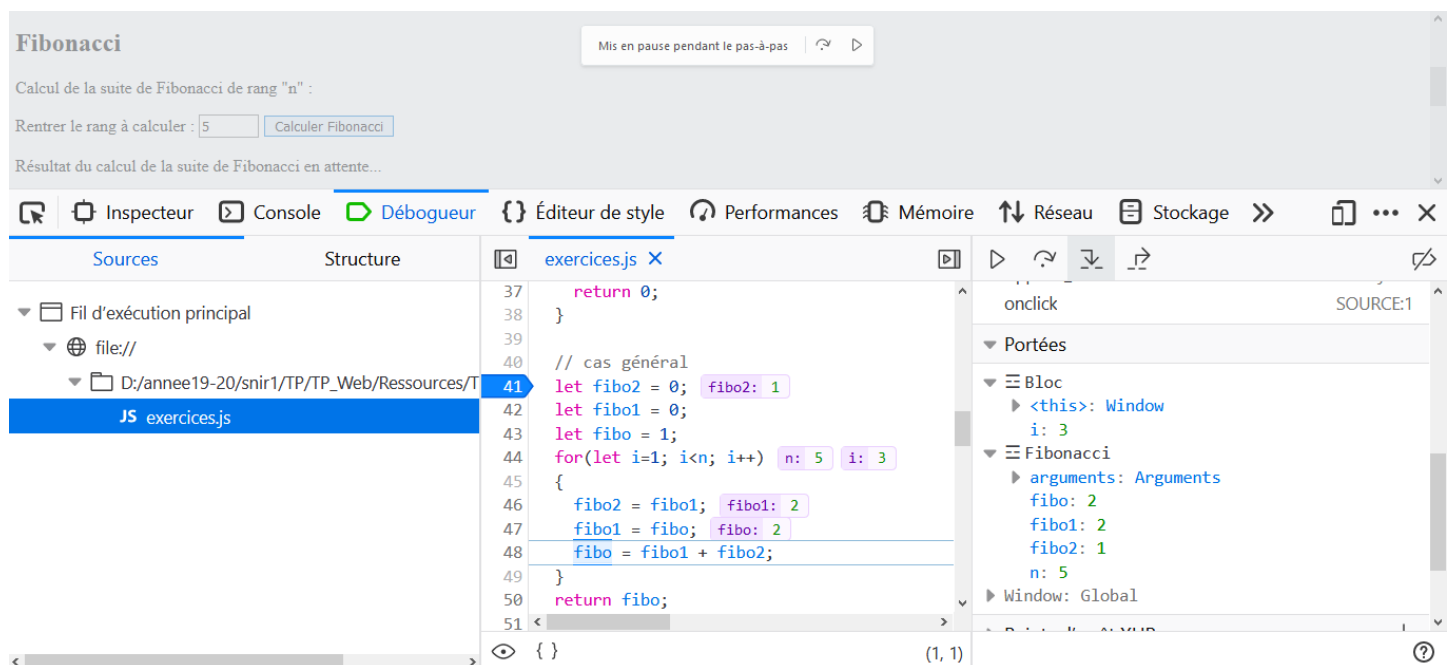
Placer un point d'arrêt par exemple à la ligne 41 `"let fibo2 = 0 ;"`.

Rentrer une valeur pour le rang à calculer dans la page html, et cliquer sur le bouton « Calculer Fibonacci ».

L'exécution du code javascript s'est arrêtée à la ligne 41 au point d'arrêt. On peut visualiser les variables et leur évolution dans la partie à droite.

On retrouve les icônes classiques d'un débogueur toujours à droite.

Exécuter avec le débogueur, pas-à-pas, la fonction *Fibonacci*.



4 Développement de la fonction *Range(stop, [start], [step])*

De façon similaire à la suite de Fibonacci, développer dans *exercices.js* le code permettant d'exécuter l'appel à la fonction *Range(stop, [start], [step])* que l'on trouve dans le fichier *exercices.html*. Vous avez deux fonctions à écrire :

- la fonction *Appeler_Range()* qui est appelée lors du clic sur le bouton « Range(stop, [start], [step]) » ;
- La fonction *Range(stop, [start], [step])* elle-même.

Description de la fonction *Range* :

Avec l'[API des tableaux](#) et une [boucle for](#) écrire la fonction `Range(stop, [start], [step])` qui prend trois arguments en paramètre et renvoie le tableau des entiers compris entre `start` (inclus) et `stop` (exclu) avec un pas de `step` :

- si `start` n'est pas défini, sa valeur par défaut est 0 ;
- si `step` n'est pas défini, sa valeur par défaut est 1 ;

Polytech Lyon 3A	TP Web N°3	Javascript	2025-2026
------------------------	---------------	------------	-----------

- par simplification, on ne traitera que le cas où `step` est strictement positif ;
- si `stop` est inférieur ou égal à `start`, alors le tableau vide est retourné ;
- exemple `Range(10, 0) == range(10) == [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]`
- exemple `Range(10, 4, 2) == [4, 6, 8]`
- exemple `Range(-2, 0) == []`

Range(stop, [start], [step])

Fonction `range(stop, [start], [step])` qui prend trois arguments en paramètre et renvoie le tableau des entiers compris entre `start` (inclus) et `stop` (exclu) avec un pas de `step`.

Stop : [Start] : [Step] : `Range(stop, [start], [step])`

Range = [0,1,2,3,4,5,6,7,8,9]

Range(stop, [start], [step])

Fonction `range(stop, [start], [step])` qui prend trois arguments en paramètre et renvoie le tableau des entiers compris entre `start` (inclus) et `stop` (exclu) avec un pas de `step`.

Stop : [Start] : [Step] : `Range(stop, [start], [step])`

Range = [4,6,8]

Range(stop, [start], [step])

Fonction `range(stop, [start], [step])` qui prend trois arguments en paramètre et renvoie le tableau des entiers compris entre `start` (inclus) et `stop` (exclu) avec un pas de `step`.

Stop : [Start] : [Step] : `Range(stop, [start], [step])`

Range = []

5 Insertion de la page *exercices.html* dans notre site Web

Vous devez construire un fichier **js_exercices.html** qui contiendra le code du fichier *exercices.html* dans notre site Web. Le fichier *exercices.js* sera placé dans un dossier nommé **javascript**.

Reportez-vous au fichier *js_exercices.pdf* pour visualiser le résultat à obtenir.

Demandez au professeur de vous montrer le résultat à obtenir.

6 Ampoule

Le tutorial javascript de w3schools propose, pour commencer, 5 exemples dans le chapitre **"JS Introduction"** :

- 1) JavaScript Can Change HTML Content
- 2) JavaScript Can Change HTML Attributes
- 3) JavaScript Can Change HTML Styles (CSS)
- 4) JavaScript Can Hide HTML Elements
- 5) JavaScript Can Show HTML Elements

Tester avec attention ces 5 exemples sur le site de w3schools.

Polytech Lyon 3A	TP Web N°3	Javascript	2025-2026
------------------------	---------------	-------------------	-----------

L'objectif est d'insérer le 2ième exemple (celui avec l'ampoule) dans notre site web (voir fichier *js_ampoule.pdf*) mais avec la modification suivante :

- un clic sur l'ampoule l'allumera ou l'éteindra !

Indications :

- dans la balise , l'attribut « onclick » exécutera une fonction que l'on pourra appeler par exemple « OnOffAmpoule() »
- l'attribut « src » de l'image renvoie une chaîne javascript de type « string » qui contient le nom de l'image. Javascript propose pour les objets « string » une méthode « search » qui sera très utile dans notre cas présent https://www.w3schools.com/jsref/jsref_search.asp !

Pour rajouter l'exemple de l'ampoule à notre site web, créer un fichier **js_ampoule.html** et créer un fichier **ampoule.js** qui sera placé dans le dossier *javascript* créé précédemment.

7 Planètes (facultatif)

7.1 Événements javascript

Lire le paragraphe sur les événements du tutorial Javascript de w3schools (*JS Tutorial* puis *JS Events*).

Tester les exemples.

Faire les 3 petits exercices à la fin du paragraphe sur les événements.

7.2 Les événements *onmouseover* et *onmouseout*

Soit l'exemple « [onmouseover an image map](#) » que l'on trouvait sur w3schools dans le cours sur JAVASCRIPT (*JS HTML DOM* puis *DOM Events* puis *Mouse Events*). Le lien suivant est encore actif :

http://www.w3schools.com/js/tryit.asp?filename=tryjs_imagemap.

Lorsque l'on teste cet exemple, on constate deux défauts :

- 1) l'indication initiale "*Mouse over the sun and the planets and see the different descriptions*" ne réapparaît plus après avoir passé la souris sur une des planètes.
- 2) Lorsque l'on clique sur une des 3 planètes, l'affichage de la planète se fait par lien hypertexte sur un fichier html (les fichiers *sun.htm*, *mercure.htm* et *venus.htm*).

On demande de corriger ces deux défauts en exploitant l'évènement *onmouseout* :

- sur l'occurrence de l'évènement *onmouseout*, réafficher l'indication initiale "*Mouse over the sun and the planets and see the different descriptions*" (en appelant une nouvelle fonction *writeDefault()*) ;
- rajouter une image (vide initialement) à côté de l'image des planètes et dans cette image :
 - sur l'occurrence de *onmouseover* alors afficher l'image gif de la planète correspondante ;
 - sur l'occurrence de *onmouseout* alors ne plus afficher l'image de la planète.

Voir fichier *js_planets.pdf* correspondant. Demander au professeur une démo au vidéo projecteur.

On vous fournit les fichiers *planets.gif*, *sun.gif*, *merglobe.gif* et *venglobe.gif*.