Javascript

**Objectif : Rendre dynamique son site web.**

## **Etape 1 : Découverte du langage Javascript**

Javascript est un langage qui a été créé principalement pour rendre des pages webs dynamiques (nous verrons en détails comment rendre dynamique des pages prochainement). Récupérer les consignes directement sur le github du Campus : <https://github.com/le-campus-numerique/begin-js>.

Pour utiliser du Javascript dans une page html, il suffit d’ajouter une balise :

**<script type="text/javascript">**

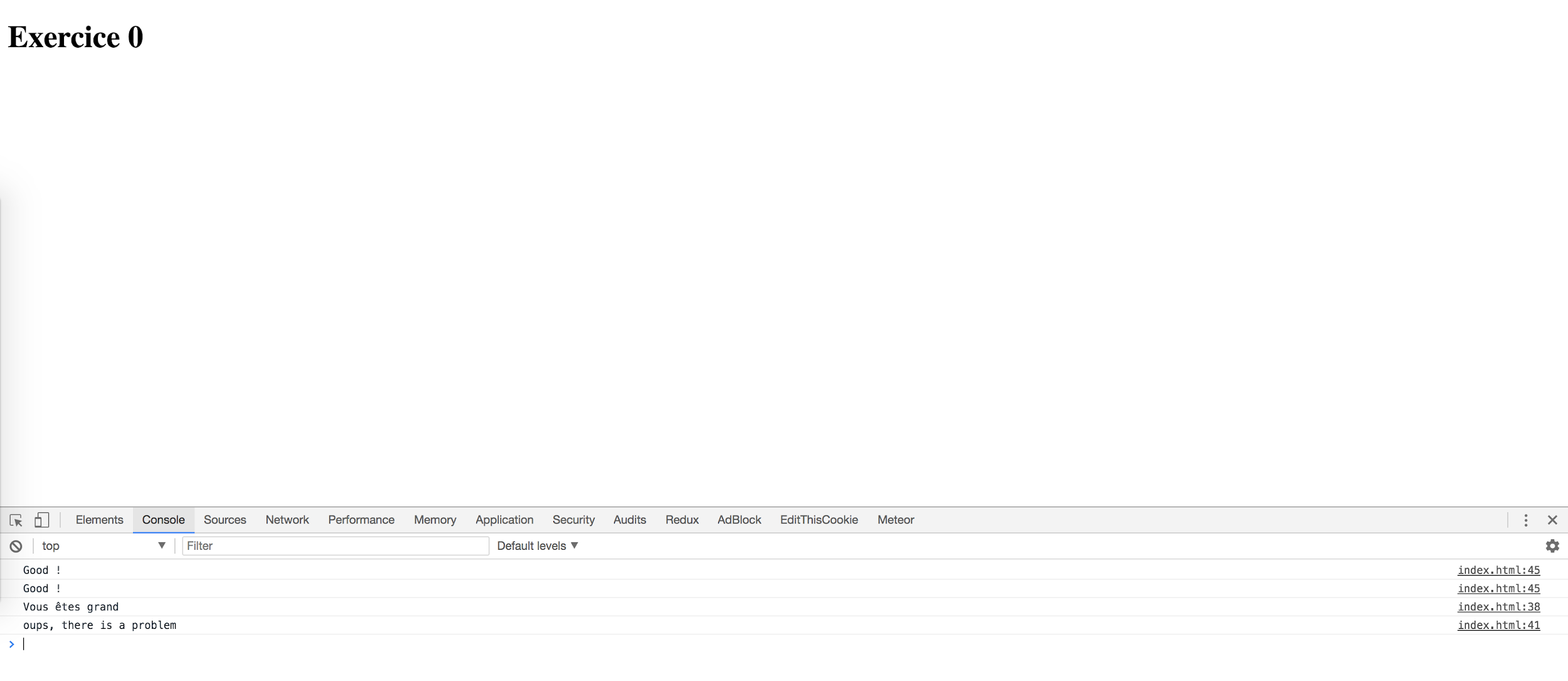
***// vous pouvez écrire du javascript ici***

**</script>**

À l’intérieur de votre balise *<body>* html.

Prenez par exemple, le fichier **index.html** du répertoire exo-0, et ouvrez la console javascript (sur chrome, faites un clic droit inspecter plus sélectionner l’onglet console).

Vous devriez avoir une page comme ceci :



La console est un outil fondamental qu’il faut toujours avoir ouvert lorsque vous faites du Javascript.

Vous vous souvenez surement du cours HTML/CSS que vous pouvez insérer du code CSS directement dans le fichier HTML. On vous a aussi probablement dit que ce n’était pas une bonne pratique car nous pouvons avoir plusieurs centaines de lignes de CSS et donc un fichier HTML énorme, c’est la même chose pour le Javascript, nous pouvons inclure du code Javascript dedans mais ce n’est pas le moyen recommandé.

À la place, nous pouvons écrire

**<script src="script.js"></script>**

Pour inclure le contenu d’un fichier Javascript directement.

C’est sur cette base que sont écrits les exercices 1 à 7.

Avant de commencer les exercices suivez ces quatres tutoriels Codecademy :

* <https://www.codecademy.com/courses/introduction-to-javascript/lessons/introduction-to-javascript/exercises/intro>
* <https://www.codecademy.com/courses/introduction-to-javascript/lessons/control-flow/exercises/control-flow-intro>
* <https://www.codecademy.com/courses/introduction-to-javascript/lessons/functions/exercises/intro-to-functions>
* <https://www.codecademy.com/courses/introduction-to-javascript/lessons/scope/exercises/scope>

Une fois fait, vous pouvez commencer les exercices.

**Tout votre travail doit être fait en utilisant Git.** Votre avancement devra être présenté dans des commits, sous la forme suivante : un commit minimum par exercice, avec un message reprenant l'intitulé de l'exercice.

**🚨 Commencez par forker le repository afin d’avoir une copie sur laquelle vous pourrez pusher.**

## *Modalité*

Travail en individuel

### *Objectif**de**l’activité*

* Programmer avec des variables
* Passer des paramètres à une fonction
* Programmer avec des objets
* Programmer avec des conditions
* Programmer une boucle
* Programmer avec des tableaux

### *Consignes*

* Réaliser les exercices : <https://github.com/le-campus-numerique/begin-js>

### *Productions**attendues*

* Un projet git avec vos réponses.

### *Ressources*

* Chrome dev tools *:* <https://www.youtube.com/watch?v=H0XScE08hy8>
* Introduction à Javascript avec Scrimba ( très très bien fait ) : <https://scrimba.com/g/gintrotojavascript>

## **Etape 2 : Manipulation du DOM avec JQuery**

Maintenant que nous avons appris les bases de Javascript, nous allons pouvoir rendre dynamiques nos pages web. Javascript a été créé dans ce but, si vous voulez en savoir un peu plus sur son histoire la [page wikipédia](https://en.wikipedia.org/wiki/JavaScript) vous apprendra notamment qu'il existe depuis 1995 ! Une belle longévité pour un langage du Web, et croyez-moi Javascript (incluant les différentes version et les évolutions) a encore de beaux jours devant lui.

Pour manipuler le DOM (Document Object Model, on en apprendra plus bientôt sur ce que c'est), on a plusieurs possibilités :

* soit utiliser des méthodes natives, écrites en Javascript ;
* soit utiliser une librairie nous facilitant la manipulation. La plus connue aujourd'hui est [JQuery](https://jquery.com/).

Nous allons prendre ce deuxième choix.

Pour mieux comprendre, voici un exemple de code en Javascript qui manipule le DOM

const login = document.getElementById('login');

const loginMenu = document.getElementById('loginMenu');

login.addEventListener('click', () => {

if(loginMenu.style.display === 'none'){

loginMenu.style.display = 'inline';

} else {

loginMenu.style.display = 'none';

}

});

Et voici son équivalent en Jquery :

$('#login').click(() => {

$('#loginMenu').toggle()

});

Pas besoin de comprendre ce code pour le moment, c'est le but du cours.

En une phrase, il fait : 'Quand je clique sur la div dont l'id est login alors j'affiche la div dont l'id est loginmenu' (et je la cache au clic suivant).

JQuery va nous permettre de faire des choses extraordinaires, pour en avoir un aperçu faites la première partie du cours (en Français !) [jQuery de Codecademy](https://www.codecademy.com/learn/learn-jquery).

Remarque 1: il existe deux cours Jquery sur codecademy, soyez vigilants à prendre celui indiqué.

Remarque 2: Les 4 premiers exercices sont assez guidés et permettent d'aborder les notions séparément. Les suivants, un peu moins et utilisent plusieurs notions à la fois. Les quatre premiers sont à réaliser en autonomie, prenez le temps de chercher, ce n'est pas une course, chacun son rythme.

## *Modalité*

Travail en individuel

### *Objectif**de**l’activité*

* Définir jQuery
* Sélectionner un élément du DOM
* Modifier un élément du DOM
* Réagir à des évènements utilisateurs
* Utiliser un plugin jQuery
* Réaliser une interface graphique dynamique
* Contrôler un formulaire

### *Consignes*

* Réaliser les exercices : <https://github.com/le-campus-numerique/begin-jquery>

### *Productions**attendues*

* Un projet git avec vos réponses.

### *Ressources*

* Cette série de vidéos jQuery sur Scrimba : <https://scrimba.com/playlist/pxZMUV>