



# Piscine Discovery

## Cellule 3-3

Staff 42 [piscine@42.fr](mailto:piscine@42.fr)

*Résumé: Dans cette cellule, nous allons voir comment utiliser du javascript.*

*Version: 2*

# Table des matières

<b>I</b>	<b>Un mot sur cette Piscine Discovery</b>	<b>2</b>
<b>II</b>	<b>Introduction</b>	<b>3</b>
<b>III</b>	<b>Consignes générales</b>	<b>4</b>
<b>IV</b>	<b>Exercice 03 : To do or not to do</b>	<b>5</b>
<b>V</b>	<b>Rendu et peer-évaluation</b>	<b>6</b>

# Chapitre I

## Un mot sur cette Piscine Discovery

Bienvenue !

Vous allez commencer la première *cellule* de cette piscine de découverte de la programmation informatique. Nous voulons à la fois vous faire voir ce qu'est le code qui compose les logiciels que vous utilisez tous les jours, et à la fois expérimenter le peer-learning, modèle pédagogique de 42.

La programmation fait appel à de la logique (et non des maths). Elle vous fournit des briques élémentaires, que vous assemblez comme vous le souhaitez. Il n'y a jamais LA solution à un problème. Il y aura votre solution, il y aura celles de chacun de vos voisins. Lent ou rapide, moche ou beau, si ça fait le boulot c'est tout ce qu'il faut ! Cet assemblage de briques va constituer une suite d'ordres (calcul, affichage, ...) que l'ordinateur va effectuer, dans l'ordre que vous aurez choisi.

Plutôt que de vous donner un cours avec une seule solution pour chacun des problèmes, et qui sera probablement dépassée dans quelques années, nous avons choisi de vous mettre en situation de peer-learning. Vous allez chercher les éléments qui pourraient vous servir pour votre challenge, trier ceux qui sont effectivement intéressants en les testant et manipulant, et créer votre propre programme. Pour cela, discutez avec les autres, échangez vos points de vue, trouvez ensemble de nouvelles idées, et enfin testez par vous-même pour vous convaincre que ça marche.

La peer-évaluation est un moment clef pour découvrir d'autres façons de faire, de même que des cas particuliers auxquels vous n'avez pas pensé et qui pourraient mettre à mal votre programme (pensez à votre degré d'énervement face à un logiciel qui plante). Tels des clients différents qui ne font pas attention aux mêmes choses, chaque évaluateur sera différent du précédent. Et qui sait, vous aurez peut-être fait une nouvelle connaissance pour des collaborations ultérieures.

A la fin de cette piscine, vous n'aurez pas fait les mêmes choses que les autres participant(e)s, vous n'aurez pas validé les mêmes projets, vous aurez choisi de faire tel challenge plutôt qu'un autre.... et c'est normal ! C'est une expérience à la fois collective, et à la fois personnelle. Tout le monde sera gagnant de ce qu'il ou elle aura vécu durant cette période.

Bon courage à tous et à toutes, on espère que cette découverte vous plaira.

# Chapitre II

## Introduction

Ce que cette cellule va vous faire voir :

- Découverte du fonctionnement d'une page web.
- Découverte du javascript.

# Chapitre III


## Consignes générales

Sauf contradiction explicite, les consignes suivantes seront valables pour tous les jours de cette Piscine.

- Seul ce sujet sert de référence : ne vous fiez pas aux bruits de couloir.
- Ce sujet peut être mis à jour jusqu'à une heure avant le délai de rendu.
- Les exercices sont très précisément ordonnés du plus simple au plus complexe. En aucun cas nous ne porterons attention ni ne prendrons en compte un exercice complexe si un exercice plus simple n'est pas parfaitement réussi.
- Attention aux droits de vos fichiers et de vos répertoires.
- Vos exercices seront évalués par vos camarades de Piscine.
- Le shell `/bin/bash` sera le shell de référence pour les exercices.
- Vous ne devez laisser aucun autre fichier que ceux explicitement spécifiés par les énoncés des exercices dans votre répertoire lors de la peer-évaluation.
- Vous avez une question ? Demandez à votre voisin de droite. Sinon, essayez avec votre voisin de gauche.
- Toutes les réponses à vos questions techniques se trouvent dans les **man** ou sur Internet.
- Pensez à discuter sur le forum Piscine de votre Intra et sur Slack !
- Lisez attentivement les exemples car ils peuvent vous permettre d'identifier un travail à réaliser qui n'est pas précisé dans le sujet à première vue.
- Réfléchissez. Par pitié, par Thor, par Odin !

# Chapitre IV

## Exercice 03 : To do or not to do

	Exercice : 03
To do or not to do	
Dossier de rendu : <i>ex03/</i>	
Fichiers à rendre : <code>index.html</code> , <code>todo.js</code>	
Fonctions Autorisées : HTML, CSS, JS	

Pour cet exercice nous allons faire un mini gestionnaire de tâches en local. Aucune lib n'est autorisée. Le design importe peu du moment que la structure présentée ci-dessous est respectée. Soyez créatif mais concentrez vous avant tout sur le côté fonctionnel.

La to do list sera représentée par un `div` qui aura pour attribut `id` la valeur `'ft_list'`. Ce bloc contiendra la liste des to do. Chaque to do sera représenté par un `div` contenu dans le bloc `'ft_list'`. Quand un to do est créé, il est placé en haut de la liste. À vous de créer l'élément et de le placer au bon endroit (manipulation de DOM).

Il faut également un bouton de création avec pour valeur d'affichage `'New'`. Ce dernier déclenche lors d'un clic, une fenêtre de texte (voir fonction `prompt`) qui permet de renseigner le nouveau to do. Une fois le texte validé, s'il n'est pas vide, il devra apparaître en haut de la liste.

Pour supprimer un to do de la liste, il suffit de cliquer dessus. Une fenêtre de confirmation doit s'ouvrir et demander si oui ou non vous souhaitez supprimer ce to do. Si vous confirmez, le to do doit définitivement disparaître du DOM, il ne faut donc pas juste le "cacher".

Petite implémentation supplémentaire, votre to do list devra être sauvegardée sous forme de cookies. Si la liste contient des to do et que vous fermez le navigateur, cette même liste devra être chargée et affichée dans `'ft_list'`. Si le(s) cookie(s) n'existe(nt) pas alors la liste sera vierge.

# Chapitre V

## Rendu et peer-évaluation

- Dans votre dossier `discovery_piscine` à la racine de votre home créez un nouveau dossier `cell103` et déplacez vous dedans.
- A partir de maintenant, tous les exercices devront être dans le bon dossier de rendu. L'exercice 00 dans le dossier `ex00`, l'exercice 01 dans le dossier `ex01`, etc... vous avez compris la logique.



Attention, durant votre soutenance tout ce qui n'est pas présent dans le dossier du jour ne sera pas vérifié.