|  |
| --- |
| **Javascript – TP n°3 « Todo List »** |

Objectifs du TP :

* Comprendre le fonctionnement du protocole HTTP
* Tester des requêtes à l’aide de Postman
* Mieux comprendre l’API Promise de Javascript et son utilisation
* Appréhender la syntaxe async/await en langage Javascript
* Utiliser la méthode [fetch()](https://developer.mozilla.org/fr/docs/Web/API/Fetch_API/Using_Fetch) pour récupérer des informations depuis un serveur

Déroulé du TP :

1. Installez Postman sur votre poste.

Lien de téléchargement : <https://www.postman.com/downloads>

Testez les requêtes suivantes dans Postman :

* GET <https://jsonplaceholder.typicode.com/todos>
* GET <https://jsonplaceholder.typicode.com/todos/1>
* POST <https://jsonplaceholder.typicode.com/todos>
* PUT <https://jsonplaceholder.typicode.com/todos/1>
* DELETE <https://jsonplaceholder.typicode.com/todos/1>

Est-ce une API RESTful ? Pourquoi ?

A quoi servent chacune de ces méthodes ?

1. Passons en mode développement.  
   Initialisez un nouveau projet appelé « js-todolist »
2. Dans votre page HTML, ajoutez les éléments suivants :

* Un titre de page Liste de tâches
* Un bouton [Charger la liste]
* Un tableau comprenant les colonnes suivantes :

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Titre | Terminé | Actions |

1. Lorsque je clique sur le bouton [Charger la liste],
   * appelez la requête correspondante afin de récupérer la liste des tâches
   * alimentez le tableau avec l’ensemble des tâches reçues

***/!\ N’écrivez pas de code HTML dans le code Javascript. Passez par le DOM pour générer les lignes et les cellules***

***(!) Essayez de créer une méthode dédiée à la génération d’une ligne de tableau***

* + Pour chaque tâche, Si le champ completed est à true, affichez le libellé « Oui » en vert
  + Pour chaque tâche, Si le champ completed est à false, affichez le libellé « Non » en rouge

***(!) Essayez de créer une méthode dédiée à la génération de cette cellule***

1. Juste en dessous du tableau, affichez le nombre de tâches récupérées depuis l’API

***(!) Essayez de créer une méthode dédiée à ce calcul***

1. En dessous du tableau, créez un formulaire d’ajout avec :
   * Un champ Titre
   * Un bouton Valider (type button)
2. Appelez la méthode POST afin de créer une tâche.  
   Lorsque vous recevez la réponse, ajoutez les informations reçues dans le tableau

***/!\ N’oubliez pas de mettre à jour le nombre d’éléments en bas du tableau***

|  |
| --- |
| ***Note :***  *L’API POST vous renvoi un ID correspondant à l’objet que vous venez de créer. Dans le cas de cet exercice, elle vous renverra toujours le même ID pour un nouvel objet crée.*  *Il s’agit d’une API de test. Dans la réalité, l’API vous renverra un nouvel ID pour chaque objet crée.* |

1. Dans la colonne Action, pour chaque tâche, ajoutez le bouton [Terminer]
   * Au clic du bouton, envoyez la requête correspondante permettant de passer le statut Terminé de la tâche à « Oui »
   * Lors de la réception de la réponse, mettez à jour la tâche correspondante
2. Dans la colonne Action, pour chaque todo, ajoutez le bouton [Supprimer]
   * Au clic du bouton, envoyez la requête correspondante permettant de supprimer une tâche
   * Lors de la réception de la réponse, supprimez la tâche correspondante

***/!\ N’oubliez pas de mettre à jour le nombre d’éléments en bas du tableau***