

TP - Utiliser des fonctions

1 Un monde de fonctions

Nous avons vu ensemble comment écrire et utiliser des fonctions. Ce sont deux activités très différentes, et elles requièrent chacune de bien comprendre quels mécanismes sont en jeu lorsque l'on parle de fonctions. Comme nous l'avons évoqué, les fonctions ne sont pas indispensables à un programme informatique. Malgré tout, elles sont partout, c'est qu'elles sont tout de même bien pratiques !

1.1 Soyons utilisateurs

L'objectif du TP d'aujourd'hui est de se pencher plus précisément sur le rôle d'utilisateur de fonctions. Nous allons donc enfiler notre casquette "utilisateur", et simplement appeler des fonctions qui ont été développées par d'autres personnes. Vous avez déjà utilisé plusieurs fonctions sans forcément le savoir.

```
1 console.log("Salut !");  
2 tableau.push(0);  
3 Math.floor(3.14);  
4 document.getElementById("Grille");
```

Toutes ces lignes ne sont pas des instructions magiques, ce sont des appels de fonctions. Ces fonctions ont été développées par des équipes de développeurs pour nous doter d'outils qui nous facilitent la vie. Et voilà l'attrait principal des fonctions : Se faciliter la vie.

1.2 Des utilisateurs avertis

C'est bien tout ça, mais comment on les connaît ces fonctions ? Comment on sait qu'elles existent, ce qu'elles font, comment s'en servir ? La réponse à toutes ces questions - et bien d'autres encore - se trouve... Sur internet. Plus précisément, dans la documentation technique disponible en ligne.

1.2.1 Des fonctions à télécharger

Comme nous l'avons vu en cours, les créateurs de fonctions regroupent leur travail en "bundles". Ces "bundles" sont ce que l'on appelle des bibliothèques de fonctions, ou libraries en anglais. Nous pouvons télécharger ces bibliothèques dans nos projets, et les utiliser dans notre code en les important. Le téléchargement se fait avec npm, de la même manière que l'on installe typescript ou parcel.

1.2.2 La librairie bibliothèque standard

Mais alors, nous n'avons jamais rien téléchargé pour utiliser les fonctions que l'on a décrit plus haut. Et elles marchent directement, sans aucun problème. Comment se fait-ce ? Chaque langage de programmation (javascript compris) vient livré avec des fonctions clé en main. En effet, ces fonctions "de base" sont bien souvent indispensables pour énormément de programmes, et les avoir par défaut est un grand gain de temps. C'est ce que l'on appelle la bibliothèque standard du langage. Et elle est très fournie. Il y a donc déjà énormément de choses à explorer avant d'importer des fonctions qui viennent d'ailleurs. Chouette !

1.3 La documentation "officielle"

Javascript est un langage assez particulier. Il est standardisé et défini par une norme nommée ECMAScript. Concrètement, cela veut dire que des personnes décident de ce que doit pouvoir faire Javascript. Ce ne sont pas les mêmes personnes néanmoins qui décident de comment doit faire Javascript. Une analogie intéressante est simplement celle de la boîte de développement et de son client. Le client a un cahier des charges "Le langage doit pouvoir faire telle ou telle chose". Le développeur se charge de réaliser le langage, comme il le souhaite, mais en respectant son cahier des charges.

1.3.1 Qui est le développeur ?

La réponse courte est : Ça dépend.

La réponse un peu plus longue est : Ça dépend du moteur (l'interpréteur).

La réponse complète est : Ça dépend de chaque navigateur (Firefox, Chrome, Safari, Edge...), chaque environnement d'exécution de Javascript (node.js côté serveur, electron côté client lourd). La norme est fixée et doit être respectée par chaque développeur qui veut créer son moteur Javascript. En soit, cela ne change rien pour nous, utilisateurs. Comment Javascript fonctionne en interne ne nous regarde pas. Nous avons juste à connaître la syntaxe du langage, et les différentes fonctions que l'on veut utiliser.

1.4 Où se trouve la doc ?

Alors le plus simple serait de lire le cahier des charges ECMAScript non ? Non, ce cahier des charges est fait pour concevoir un langage de programmation. Essayez de le lire, juste pour rire. <https://ecma-international.org/ecma-262/10.0/index.html> Cela ne nous regarde aucunement. Nous devons donc trouver une documentation à destination des développeurs, utilisateurs du langage. Les personnes les mieux placées pour nous aider sont donc celles qui créent le langage.

1.5 Au hasard... Mozilla

Il existe plusieurs références en ligne, la plupart sont de très bonne qualité. Néanmoins, l'une d'elles est créée par un réseau de développeurs bien placé pour savoir de quoi ils parlent. Il s'agit des créateurs de Firefox : Mozilla. La fondation Mozilla est une association à but non lucratif, qui gère le développement de différents produits. La communauté Mozilla est une communauté de développeurs ouverte à tous et qui travaille à améliorer les produits de Mozilla. Ils maintiennent ensemble entre autre le navigateur Firefox et... Une documentation Javascript, c'est ce qui nous intéresse : https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/A_propos

1.6 Comment s'en servir ?

Bien qu'il soit très intéressant de se pencher sur les guides Javascript pour débutants, ainsi que sur les différentes ressources disponibles, ce n'est pas la manière dont vous vous servirez le plus souvent de cette documentation. Chercher une fonction en particulier se fait bien souvent directement sur Google. Il y a deux cas possibles.

- Soit vous connaissez le nom de la fonction (vous êtes tombés dessus dans un code, ou vous vous souvenez de son nom, mais pas de son fonctionnement), auquel cas vous tapez simplement le nom de la fonction sur Google. Vous tomberez directement sur la documentation Mozilla de la fonction dans les premières recherches.
- Soit vous cherchez à faire quelque chose avec un objet précis (un tableau, une chaîne de caractères), ou même pas, l'idée est floue. Dans ce cas de figure, il faut essayer de formuler au mieux ce que l'on cherche à faire, avec des termes génériques. Si votre recherche est pertinente, et avec un peu de réussite, vous aurez des questions/réponses venant de différents sites/forums (comme <https://stackoverflow.com/>). Et si vous trouvez une réponse à votre problème, il y a des chances qu'une fonction standard vous y aide. Auquel cas il vous restera à la rechercher afin de mieux la dompter, et la réutiliser à l'avenir.

2 Objectif du TP

Vous travaillerez à votre convenance seuls, ou en groupe. Pour chacun des énoncés qui suivent, trouver une fonction qui peut vous aider à résoudre le problème plus ou moins simplement. Enfin, résolvez le problème à l'aide de la fonction.

Chaque énoncé donnera lieu à un second exercice plus difficile qui mettra en oeuvre la fonction que vous venez de trouver. Ces exercices sont plus difficiles, ils ne sont pas obligatoires mais constituent un bon entraînement à la programmation. Il est recommandé d'essayer de les résoudre à plusieurs.

Certaines étapes donnent lieu à des questions. Vous répondrez à ces questions sur un document à part également à rendre.

2.1 Rendu du TP

Vous aurez deux séances pour réaliser le TP. Le rendu se fera sous forme d'un script `index.ts` et d'un document `reponses.pdf` par groupe. Précisez dans le document de réponse les membres éventuels de votre groupe. La date de rendu est fixée au vendredi 20 Novembre en fin de séance.

3 Commençons par un peu de mathématiques

3.1 Arrondir un nombre décimal

3.1.1 Premier exemple : arrondir 3 nombres

Soit le code suivant :

```
1 let nombreA : number = 15.3 ;
2 let nombreB : number = 12.2 ;
3 let nombreC : number = 7.8 ;
4
5 console.log(nombreA, nombreB, nombreC);
```

Je voudrais que mes nombres soient arrondis à l'entier le plus proche. Concrètement, mon log devrait m'écrire 15, 12, 8. Utiliser la fonction `Math.round()` afin d'arrondir mes trois nombres.

3.1.2 Pour aller plus loin : arrondir des moyennes de classe

Soit le code suivant :

```
1 let moyennesDeClasse : number[] = [13.4, 11.8, 17.2, 16.6, 9.7, 14.4,
2   10.6, 8.8, 11.6, 9.4, 13.6, 14.9, 18.1, 7.4];
3
4 let moyenneGenerale : number = 0;
5
6 for(let i : number = 0 ; i < moyennesDeClasse.length ; i++) {
7   moyenneGenerale += moyennesDeClasse[i];
8 }
9
10 moyenneGenerale /= moyennesDeClasse.length;
11 console.log(moyenneGenerale);
```

Mes moyennes de classe sont des nombres à virgule. J'aimerais arrondir ces moyennes à l'entier le plus proche. Tout cela pour calculer la moyenne générale de la classe.

Question : La moyenne de classe a un peu changé. Globalement, cet arrondi a-t-il été bénéfique pour la classe ?

3.2 Élever un nombre à une puissance

3.2.1 Premier exemple : quelques puissances de 2

Soit le code suivant :

```
1 let nombreDeBase : number = 2;
2 let quelquesPuissances : number[] = [2, 6, 12, 32];
```

J'aimerais écrire dans la console quelques puissances de 2. Je veux calculer le résultat de 2^2 , 2^6 , 2^{12} et 2^{32} . Utiliser la fonction `Math.pow()` afin d'élever mon nombre à ces puissances.

3.2.2 Pour aller plus loin : décomposition d'un nombre en base 10

Soit le code suivant :

```
1 let nombreDecompose : number[] = [2, 6, 4, 0, 2, 9, 7, 8, 5, 3, 9];
```

26 402 978 539, ce nombre représente le nombre total de vues du youtubeur PewDiePie au moment de la réalisation du TP. Écrire un code permettant d'écrire dans la console le nombre décomposé dans le tableau sous sa forme recomposée.

Question : Nous venons de recomposer le nombre 26 402 978 539 avec des produits de puissances de 10. Comment peut-on écrire mathématiquement cette opération ?

3.3 Générer un nombre aléatoire

3.3.1 Premier exemple : tirer une case d'un tableau au hasard

Soit le code suivant :

```
1 let nombresPremiers : number[] = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31];
```

Je souhaite écrire dans la console un nombre premier choisi de manière aléatoire dans mon tableau. Trouver une fonction dans Math capable de générer un nombre aléatoire. Utiliser cette fonction afin d'afficher un nombre aléatoirement pris dans le tableau.

Question : La fonction utilisée génère un nombre décimal entre 0 et 1. Quelle formule mathématique permet de générer un nombre entre a et b, avec x notre nombre aléatoire ?

3.3.2 Pour aller plus loin : mélange d'un tableau

Soit le code suivant :

```
1 let tableauRange : number[] = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
```

Mon tableau est bien rangé. Je souhaite le mélanger. Écrire un code qui mélange tableauRange et afficher le tableau une fois mélangé.

Question : Il existe un algorithme optimisé populaire permettant de réaliser ce type de mélange. Quel est le nom de cet algorithme ?

4 Manipuler des tableaux

4.1 Ajouter des éléments au début de mon tableau

4.1.1 Premier exemple : compléter une suite logique

Soit le code suivant :

```
1 const monTableauIncomplet : number[] = [3, 4, 5, 6, 7, 8, 9];
```

J'aimerais compléter mon tableau afin qu'il parte de 0 jusqu'à 9 au lieu de 3 à 9. Je sais ajouter un élément à la fin de mon tableau avec `Array.push()`. Cette fonction ne me permet pas d'ajouter des éléments au début. Trouver une fonction qui remplit ce rôle, et l'utiliser pour compléter mon tableau. Écrire le contenu de mon tableau dans la console.

4.1.2 Pour aller plus loin : miroir d'un tableau

Soit le code suivant :

```
1 const donneesALEndroit : number[] = [7, -3, -12, 4, 9, 8, -2, -1, 14, 13, -15];
```

Je voudrais créer un nouveau tableau à partir de ce tableau `donneesALEndroit`. Ce nouveau tableau serait un miroir du tableau existant. Créer ce nouveau tableau et l'afficher dans la console.

4.2 Trier un tableau

4.2.1 Premier exemple : tri par ordre alphabétique

Soit le code suivant :

```
1 const motsAuPif : string[] = ["Sommeil", "Viande", "Peintre", "Oreille", "Cintre"];
```

J'aimerais trier les mots de ce tableau par ordre alphabétique. Trouver une fonction permettant de trier le tableau. Afficher le tableau trié dans la console.

4.2.2 Pour aller plus loin : simplifier un problème grâce au tri

Soit le code suivant :

```
1 const leGrandEcart : number[] = [2, 7, 5, 9, 6, 12, 1, 11, 13, 8, 0];
```

Je dispose d'un tableau avec une série de nombres aléatoires entre 0 et 13. Ils ne sont pas tous représentés. J'aimerais savoir quel est le plus grand trou entre deux nombres qui se suivent. Afficher dans la console le plus grand écart entre deux nombre consécutifs contenus dans ce tableau.

Question : Vous vous heurterez probablement à un problème insoupçonné lors de la réutilisation de la fonction de tri. Quel est-il ? Comment parvenir à le surmonter ?

5 Les chaînes de caractères

5.1 Modifier la casse d'une chaîne de caractères

5.1.1 Premier exemple : trouver le bon ton

Soit le code suivant :

```
1 let petitMessageDiscret : string = "HEY, PSSST, VIENS VOIR J'AI  
   QUELQUE CHOSE À TE DIRE..."  
2 let criDAlerte : string = "attention ! un homoursporc vous attaque !"
```

L'effet n'est pas celui désiré... Trouvez des fonctions permettant de réécrire ces phrases respectivement en minuscule et en majuscule, afin d'obtenir le ton recherché. Écrire ces fonctions dans la console.

5.2 Remplacer des morceaux de chaîne

5.2.1 Premier exemple : modifier une phrase

Soit le code suivant :

```
1 let phraseAModifier : string = "Les cours de M.Poesmann c'est un peu  
   comme #, et je m'y connais !";
```

Cette phrase n'est pas complète. Il faudrait remplacer le hashtag par autre chose... Trouver comment faire à l'aide d'une fonction. Et afficher le résultat dans la console.

5.2.2 Pour aller plus loin

Soit le code suivant :

```
1 let phraseACensurer : string = "Fuck not lest ye be fucked, FUCked are  
   the rich fUCKers for these fuCKers shall fuck the earth, FUCked  
   are the meek fucKers, for they shall be truly fUcked, FuCk thy  
   enimies ere' they FuCk you, and if some fUcKer fucks you on your  
   right cheek--fuck his shit up.";
```

Cette phrase est bien vulgaire. Nous souhaitons la censurer en remplaçant toutes les occurrences de "fuck", par "*****", et ce, peu importe la casse. Utilisez différentes fonctions pour arriver à vos fins, et écrivez la version censurée de la phrase dans la console.

Question : La fonction `String.replaceAll()` semble bien pratique pour affronter ce problème. Tentez d'utiliser cette fonction sous différents navigateurs. Que se passe-t-il ? Enquêtez sur internet afin de comprendre ce qui se passe exactement, et expliquez avec vos mots le problème rencontré.