

TP JavaScript n°1 - DWWM - 03.04.2025

Mise en pratique des connaissances et compétences

Mots clé : javascript, variables, structures de contrôle, boucles, objets littéraux, tableaux, fonctions

groupe DWWM

- M.TOULOUSE -

TP - JAVASCRIPT - DWWM - Jeudi 03.04.2025

Codage d'un système de gestion de commandes d'un restaurant en Vanilla JavaScript.

Ce TP n'est pas à faire en groupe et doit être rendu **individuellement**.

Pensez à bien **indenter** et **commenter** tous vos codes !

- Merci de me retourner le TP **avant 17h00** au courriel suivant : `frontend@netc.fr`
- Le dossier du TP devra être zippé.
- Le TP devra être compressé en respectant **STRICTEMENT** le format suivant qui utilise un underscore (__) entre chaque token écrit **sans accents** dans le nom de fichier :
 - `prenom_nom_tp1_javascript_030425.zip`

Exemple :

Je m'appelle Frédéric DUBOIS, mon fichier sera alors nommé : `frederic_dubois_tp1_javascript_030425.zip`

Si vous ne parvenez pas à envoyer votre dossier .zip (gmail pose souvent des problèmes), utilisez l'excellent service français **SMASH**, gratuit, sans inscription ni abonnement : <https://fr.fromsmash.com>

Prenez le temps d'analyser les tâches demandées avant de coder !

- Le codage **n'est pas uniquement** une affaire de technologie mais de **réflexion**.
- Il est suggéré de ne **pas utiliser l'ordinateur** durant la première heure du TP.
- Passer **directement** à la phase de codage sur écran est souvent très **sournois et contreproductif**.
- Travaillez avec du **papier et un stylo** afin d'envisager votre stratégie de codage face à la problématique.
- Faites des schémas, des **croquis**, envisagez des **scénarios** de codage, annotez vos idées
- Formulez **délicatement et consciencieusement** sous forme de phrase en français vos étapes de codage

1. Gestion des commandes d'un restaurant

1.1. Contexte

Vous êtes en charge de développer un programme pour gérer les commandes d'un restaurant.

Chaque commande doit inclure :

- **une entrée**
- **un plat principal**
- **un dessert**
- **une boisson**

L'utilisateur choisira ces quatre éléments pour constituer un repas complet, et cela sera enregistré comme une commande.

1.2. 🎯 Objectifs pédagogiques

À la fin de ce TP, vous serez capable de :

1. Manipuler des tableaux et des objets pour gérer des données dynamiques.
2. Structurer un programme en JavaScript avec des fonctions claires et réutilisables.
3. Utiliser des méthodes comme `map` , `filter` , et `reduce` pour manipuler des tableaux.

1.3. 📌 Prérequis

- Connaissance de base en JavaScript.
- Compréhension des **tableaux** et des **objets littéraux**.
- Connaissance des **fonctions** et des **méthodes de tableaux**.

1.4. 📄 Données initiales

Les données initiales sont déjà fournies dans le code ci-dessous. Vous ne devez **EN AUCUN CAS** les modifier !

Vous disposez de 4 tableaux d'objets représentant les entrées, les plats, les desserts et les boissons du restaurant.

Chaque objet a un `id`, un `nom` et un `prix`.

Vous disposez également d'un tableau `commandes` qui est initialement vide. Ce dernier **modélise les commandes passées** par les clients.

- Données du premier objet (Salade César) du tableau `entrees` :

Propriété	Signification	Valeur
id	Identifiant unique de l'élément	101
nom	Nom de l'élément	🥗 Salade César
prix	Prix de l'élément en euros	8

- *Menu du restaurant :*

javascript

```
const entrees = [
  { id: 101, nom: "🥗 Salade César", prix: 8 },
  { id: 102, nom: "🍲 Soupe du jour", prix: 7 },
  { id: 103, nom: "🦐 Crevettes grillées", prix: 10 },
  { id: 104, nom: "🧀 Assiette de fromages", prix: 9 }
];

const plats = [
  { id: 201, nom: "🍕 Pizza Margherita", prix: 12 },
  { id: 202, nom: "🍔 Burger", prix: 15 },
  { id: 203, nom: "🍝 Pâtes Carbonara", prix: 14 },
  { id: 204, nom: "🥩 Steak", prix: 20 }
];

const desserts = [
  { id: 301, nom: "🍰 Gâteau au chocolat", prix: 6 },
  { id: 302, nom: "🍦 Glace vanille", prix: 5 },
  { id: 303, nom: "🍓 Tarte aux fraises", prix: 7 },
  { id: 304, nom: "🍮 Crème caramel", prix: 6 }
];

const boissons = [
  { id: 401, nom: "🥤 Soda", prix: 3 },
  { id: 402, nom: "☕ Café", prix: 2 },
  { id: 403, nom: "🍵 Thé", prix: 2 },
  { id: 404, nom: "🍹 Cocktail", prix: 8 }
];
```

- Commandes initiales vides:

javascript

```
const commandes = [];
```

2. PARTIE N°1 : aucune boucle ni aucune fonction

Cette première partie du TP est la **plus simple**.

Elle vous permet de vous **familiariser** avec les **données initiales**.

Vous **n'avez pas le droit** d'utiliser de **boucle** ni de **fonction** pour afficher les informations du menu du restaurant.

Vous devez **afficher les informations** du menu du restaurant dans la **console de Visual Studio Code**.

Vous devez **respecter** avec **EXACTITUDE** la **mise en forme** de l'affichage **imposé**.

Indications :

- Utiliser `console.log` pour afficher les informations dans la console.
- Formatez les messages avec des `templates littéraires` ou `backticks`.
- Accédez aux informations des éléments directement à **partir de leurs indices**.

2.1. 🚩 Etapes préliminaires

QUESTION 1:

Afficher dans la **console de Visual Studio Code** les informations de la Salade César, de la pizza Margherita, du gâteau au chocolat et du café.

Bien entendu vous devez utiliser les notions de **tableaux**, d'**objets** et de **console.log** pour afficher les informations.

Il est **interdit** de faire un simple copier-coller des informations dans la console ! 😊

Affichage exact attendu dans la console de **Visual Studio Code** :

```
-----
Bienvenue dans notre restaurant !
-----

-----
PARTIE N°1
-----

-----
101 - 🥗 Salade César : 8 €
201 - 🍕 Pizza Margherita : 12 €
301 - 🍰 Gâteau au chocolat : 6 €
402 - ☕ Café : 2 €
-----
```

QUESTION 2:

Quel est le prix total d'un menu composé de la Salade César + Pizza Margherita + Gâteau au chocolat + Café ?

Indication : Additionnez les prix des éléments en accédant directement à partir de leurs indices.

Affichage exact attendu dans la console de **Visual Studio Code** :

```
-----
Menu:

🥗 Salade César
🍕 Pizza Margherita
🍰 Gâteau au chocolat
☕ Café

28 €
-----
```

QUESTION 3:

Ajoutez "🍔 Burger" à la liste de commandes (tableau `commandes`) initialement vide.

Indication : accédez directement à l'élément correspondant dans le tableau `plats` et ajoutez-le au tableau commandes.

Affichage exact attendu dans la console de Visual Studio Code :

```
-----
Commande en cours :
🍔 Burger 15 €
-----
```

QUESTION 4:

Ajoutez "🦐 Crevettes grillées" à la liste de commandes (tableau `commandes`) initialement vide.

Indication : accédez directement à l'élément correspondant dans le tableau `entrees` et ajoutez-le au tableau commandes.

Affichage exact attendu dans la console de Visual Studio Code :

```
-----
Liste de commandes :
🍔 Burger 15 €
🦐 Crevettes grillées 10 €
-----
```

QUESTION 5:

Calculer le prix total de la commande précédente et l'afficher dans la console.

Indication : Additionnez les prix des éléments en accédant directement à leurs indices.

Affichage exact attendu dans la console de Visual Studio Code :

```
-----
Commande en cours :
🍔 Burger 15 €
🦐 Crevettes grillées 10 €
Prix total : 25 €
-----
```

2.2. 🖋️ Récapitulatif de l'affichage intégral de la partie n°1 dans la console de Visual Studio Code :

Si tout votre code est **correct**, vous devez obtenir l'affichage suivant dans la console de Visual Studio Code à la fin de la partie n°1 du TP 👍

Bienvenue dans notre restaurant !

PARTIE N°1

101 - 🥗 Salade César : 8 €
201 - 🍕 Pizza Margherita : 12 €
301 - 🍰 Gâteau au chocolat : 6 €
402 - ☕ Café : 2 €

Menu:

🥗 Salade César
🍕 Pizza Margherita
🍰 Gâteau au chocolat
☕ Café

28 €

Commande en cours :

🍔 Burger 15 €

Commande en cours :

🍔 Burger 15 €
🦐 Crevettes grillées 10 €

Commande en cours :

🍔 Burger 15 €
🦐 Crevettes grillées 10 €

Prix total : 25 €

3. PARTIE N°2 : Manipuler les données avec des boucles

Cette partie du TP est **encore plus optimisée** et vous demande de manipuler les données avec des **boucles**.

Vous devez donc **OBLIGATOIREMENT** utiliser des **boucles** pour afficher les informations demandées.

Vous devez **respecter** avec **EXACTITUDE** la **mise en forme** de l'affichage **imposé**.

3.1. 🚩 Etapes préliminaires

QUESTION 1:

Récupérer et afficher le prix de la Salade César dans la console de **Visual Studio Code**.

Affichage exact attendu dans la console de **Visual Studio Code** :

```
-----
PARTIE N°2
-----
🥗 Salade César : 8 €
-----
```

QUESTION 2:

Calculer le prix total d'un menu composé de la Salade César + Pizza Margherita + Gâteau au chocolat + Café.

Indication : Utilisez des boucles pour trouver les prix des éléments

Affichage exact attendu dans la console de **Visual Studio Code** :

```
-----
Menu :
```

```
🥗 Salade César
🍕 Pizza Margherita
🍰 Gâteau au chocolat
☕ Café
```

```
28 €.
-----
```

QUESTION 3:

Quels sont les plats (toutes catégories confondues) qui coûtent moins de 10€ ?

Indication : Parcourez séparément chacun des **tableaux** avec une boucle **for** et affichez les plats dont le prix est inférieur à 10€.

Affichage exact attendu dans la console de **Visual Studio Code** :

Plats à moins de 10€ :

 Salade César : 8 €
 Soupe du jour : 7 €
 Assiette de fromages : 9 €
 Gâteau au chocolat : 6 €
 Glace vanille : 5 €
 Tarte aux fraises : 7 €
 Crème caramel : 6 €
 Soda : 3 €
 Café : 2 €
 Thé : 2 €
 Cocktail : 8 €

3.2. Récapitulatif de l'affichage intégral de la partie n°2 dans la console de Visual Studio Code :

Si tout votre code est **correct**, vous devez obtenir l'affichage suivant dans la console de Visual Studio Code à la fin de la partie n°2 du TP 👍

PARTIE N°2

🥗 Salade César : 8 €

Menu :

🥗 Salade César
 🍕 Pizza Margherita
 🍰 Gâteau au chocolat
 ☕ Café

28 €.

Plats à moins de 10€ :

🥗 Salade César : 8 €
 🍲 Soupe du jour : 7 €
 🧀 Assiette de fromages : 9 €
 🍰 Gâteau au chocolat : 6 €
 🍦 Glace vanille : 5 €
 🍰 Tarte aux fraises : 7 €
 🍰 Crème caramel : 6 €
 🍷 Soda : 3 €
 ☕ Café : 2 €
 🍵 Thé : 2 €
 🍹 Cocktail : 8 €

4. Partie N°3 : Refactoriser avec des fonctions

Cette partie du TP est plus évoluée et vous demande de refactoriser le code en utilisant des **fonctions**.

Vous devez donc **OBLIGATOIREMENT** utiliser des **fonctions** pour afficher les informations demandées.

QUESTION 1 :

Récupérer le **prix d'un plat** par son nom à l'aide d'une fonction.

Vous devez créer une fonction `obtenirPrixPlat` qui prend en paramètre le nom du plat et retourne son prix.

Affichage exact attendu dans la console de `Visual Studio Code` :

 PARTIE N°3

 Salade César : 8 €

QUESTION 2 :

Calculer le **prix total** d'une commande avec une fonction qui utilise la méthode `reduce`.

Indication : Créez une fonction `calculerTotalCommande` qui prend en paramètre un tableau de commandes et retourne le prix total.

Affichage exact attendu dans la console de Visual Studio Code :

 Calcul du prix total de la commande :

Commande :

```
[
  { id: 202, nom: '🍔 Burger', prix: 15 },
  { id: 103, nom: '🦐 Crevettes grillées', prix: 10 },
  { id: 101, nom: '🥗 Salade César', prix: 8 },
  { id: 202, nom: '🍔 Burger', prix: 15 },
  { id: 301, nom: '🍰 Gâteau au chocolat', prix: 6 },
  { id: 402, nom: '☕ Café', prix: 2 }
]
```

Prix total : 56 €.

4.1. Récapitulatif de l'affichage intégral de la partie n°3 dans la console de Visual Studio Code :

Si tout votre code est **correct**, vous devez obtenir l'affichage suivant dans la console de Visual Studio Code à la fin de la partie n°3 du TP 👍

PARTIE N°3

🥗 Salade César : 8 €

Calcul du prix total de la commande :

Commande :

```
[
  { id: 202, nom: '🍔 Burger', prix: 15 },
  { id: 103, nom: '🦐 Crevettes grillées', prix: 10 },
  { id: 101, nom: '🥗 Salade César', prix: 8 },
  { id: 202, nom: '🍔 Burger', prix: 15 },
  { id: 301, nom: '🍰 Gâteau au chocolat', prix: 6 },
  { id: 402, nom: '☕ Café', prix: 2 }
]
```

Prix total : 56 €.

5. Prolongement & enrichissement du TP

Dans le cas où vous auriez terminé ce TP **avant le temps imparti**, je vous propose de coder :

- 🗑 une fonction `supprimerCommande` : supprime une commande par son `id`.
- 🏷 une fonction `trierCommandes` : trie les commandes par total (du plus élevé au plus bas).
- 🔍 une fonction `rechercherCommande` : recherche une commande par son `id` ou par un critère (plat spécifique)
- 📊 une fonction `statistiquesCommandes` : affiche des statistiques (nombre de commandes, plat le plus demandé ...)
- ✖ une fonction `supprimerDuMenu` : supprime une entrée, un plat ou un dessert du menu.
- 💎 une fonction `ajouterReduction` : applique une réduction à une commande spécifique ou à toutes les commandes.
- 🎁 Ajouter un **système de points de fidélité** où chaque commande génère des points en fonction du total dépensé.

6. Critères d'évaluation

Votre code sera évalué sur les critères suivants :

- Code **fonctionnel** et **sans erreurs**
- **Clarté** et **lisibilité** du code
- Utilisation appropriée des **commentaires**
- **Respect des consignes** et de la **mise en forme**
- **Utilisation correcte** des boucles et des fonctions
- **Utilisation appropriée** des méthodes de tableaux
- **Efficacité** du code

7. Conseils

- **Bien lire les fiches** que je vous aies fournies
- Les **réponses aux problématiques** posées dans ce TP s'y trouvent à l'intérieur.

8. Le mot de la fin

Soignez très **précis** et **soigneux** dans votre travail.

Les **détails** visuels sont très souvent **chronophages**, ils mettent du temps à se mettre en place et prendre forme, c'est tout à fait normal, **pas d'inquiétude** !

Bonne journée de codage et bon courage !

M.TOULOUSE