

## Rapport de atelier 2

**Objectif** : L'objectif principal de ce Travail pratique et de se familiariser avec les concepts POO de langage javascript.

### Exercice 1 :

Les fonctions implémentées :

- **Fonction constructeur Voiture** : C'est une fonction constructeur qui initialise les propriétés d'une voiture comme le modèle, la marque, l'année, le type et le carburant.
- **Méthode alarmer** : Une méthode spécifique à Hyundai qui affiche un message d'alarme pour une voiture Hyundai particulière .
- **La méthode sort()** : sort() est une méthode intégrée de JavaScript qui permet de trier les éléments d'un tableau . on l'utilise pour trier les voitures

Les classes implémentées :

**Classe Hyundai** héritant de **Voiture** : C'est une sous-classe de Voiture qui ajoute les propriétés spécifiques à une Hyundai, telles que la série et si elle est hybride ou non. Elle hérite des propriétés de Voiture .

**Classe Ford** héritant de **Voiture** : C'est une sous-classe de Voiture qui ajoute les propriétés spécifiques à une Ford, telles que les options. Elle hérite des propriétés de Voiture.

```
C:\> Invite de commandes
Microsoft Windows [version 10.0.19045.5555]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\youss>node "D:\atelier 1\exo 2\exercice1.js"
Ford Mustang - 2018
Ford Focus - 2019
Hyundai i30 - 2020
Hyundai Tucson - 2022
```

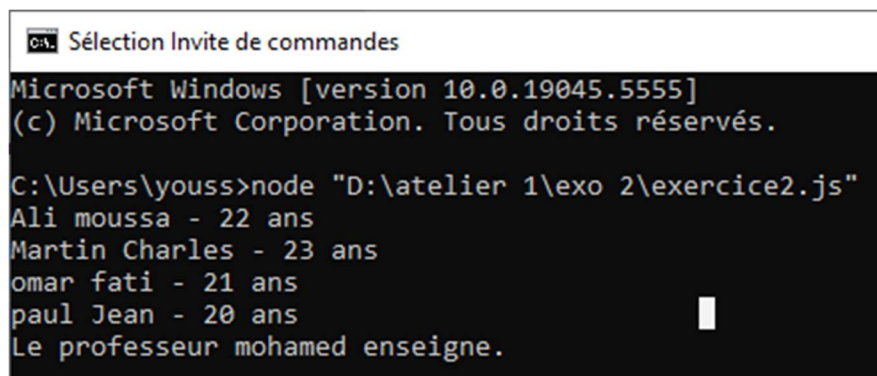
## Exercice 2 :

Les fonctions implémentées :

- **Fonction constructeur Etudiant** : C'est une fonction constructeur qui initialise les propriétés d'un étudiant, comme le nom, le prénom, l'âge, et le CNE (Code National de l'Étudiant).
- **Méthode etudier du prototype Etudiant** : Cette méthode affiche un message indiquant que l'étudiant est en train d'étudier
- **Méthode enseigner du prototype Professeur** : Cette méthode affiche un message indiquant que le professeur est en train d'enseigner.
- **Fonction constructeur Professeur** : C'est une fonction constructeur qui initialise les propriétés d'un professeur, comme le nom, l'âge, et le CIN (Carte d'Identité Nationale).
- **La méthode sort()** : sort() est une méthode intégrée de JavaScript qui permet de trier les éléments d'un tableau . on l'utilise pour trier les voitures

Les différentes classes :

**Classe Etudiant** : Représente un étudiant avec des propriétés telles que le nom, le prénom, l'âge et le CNE (Code National de l'Étudiant).



```
CS. Sélection Invite de commandes
Microsoft Windows [version 10.0.19045.5555]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\youuss>node "D:\atelier 1\exo 2\exercice2.js"
Ali moussa - 22 ans
Martin Charles - 23 ans
omar fati - 21 ans
paul Jean - 20 ans
Le professeur mohamed enseigne.
```

## Exercice 3 :

Les classes et les fonctions implémentées :

**Classe Vecteur2D** : Représente un vecteur en deux dimensions avec des propriétés x et y.

- **Méthode afficher** : Affiche les coordonnées du vecteur dans la console.
- **Méthode additionner** : Additionne les coordonnées du vecteur actuel avec celles d'un autre vecteur et retourne un nouveau vecteur résultant de cette addition.

**Classe Rectangle** : Représente un rectangle avec des propriétés longueur, largeur, et nom.

- **Méthode afficher** : Affiche les dimensions du rectangle dans la console.
- **Méthode surface** : Calcule et retourne la surface du rectangle

**Classe Carre (héritant de Rectangle)** : Représente un carré, qui est un type de rectangle où longueur et largeur sont égales.

**Classe Point** : Représente un point dans un espace à deux dimensions avec des propriétés `x` et `y`.

**Classe Segment** : Représente un segment de ligne défini par deux points d'origine (`orig`) et d'extrémité (`extrem`)

- **Méthode afficher** : Affiche les coordonnées du segment dans la console.

```
C:\> Invite de commandes

Microsoft Windows [version 10.0.19045.5555]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\youuss>node "D:\atelier 1\exo 2\exercice3.js"
Vecteur2D(0, 0)
Vecteur2D(3, 4)
Vecteur2D(3, 4)
rectangle - Longueur: 5, Largeur: 3
Surface: 15
carré - Longueur: 4, Largeur: 4
Surface: 16
Segment de (1, 2) à (4, 6)
```

## Exercice 4 :

**Les différents fichiers utilisés :**

**Le fichier blog.html** : Ce fichier HTML est conçu pour créer une interface utilisateur pour un blog avec des fonctionnalités de connexion, d'inscription et de publication de messages . il sera en interaction avec le fichier blog.js et styles.css pour le rendre plus dynamique et plus stylé .

**Le fichier styles .css** : Contient les différents styles pour embellir l'interface utilisateur .

**Le fichier BLOG.js** : Ce fichier JavaScript gère la logique d'un simple système de blog avec des fonctionnalités d'inscription, de connexion et de publication de messages

Les différentes classes implémentées :

- **Classe User** : Représente un utilisateur avec un nom d'utilisateur, un email et un mot de passe.
- **Classe Post** : Représente un message de blog avec un titre, une description et un auteur.
- **Classe Blog** : Représente un blog contenant une liste d'utilisateurs et une liste de messages. Gère les opérations d'ajout d'utilisateurs, d'ajout de messages et de récupération de messages par utilisateur ou de tous les messages.

Les différentes fonctions :

**Fonction login** : Permet à un utilisateur de se connecter. Vérifie les champs de connexion et les informations d'identification. Si les informations sont correctes.

**Fonction submitPost** : Permet à l'utilisateur connecté de publier un message. Si les champs de titre ou de description sont vides, affiche une alerte. Si aucun utilisateur n'est connecté, affiche une alerte demandant de se connecter. Après la soumission, vide les champs du formulaire de publication.

**Fonction showAllPosts** : Affiche tous les messages publiés sur le blog.

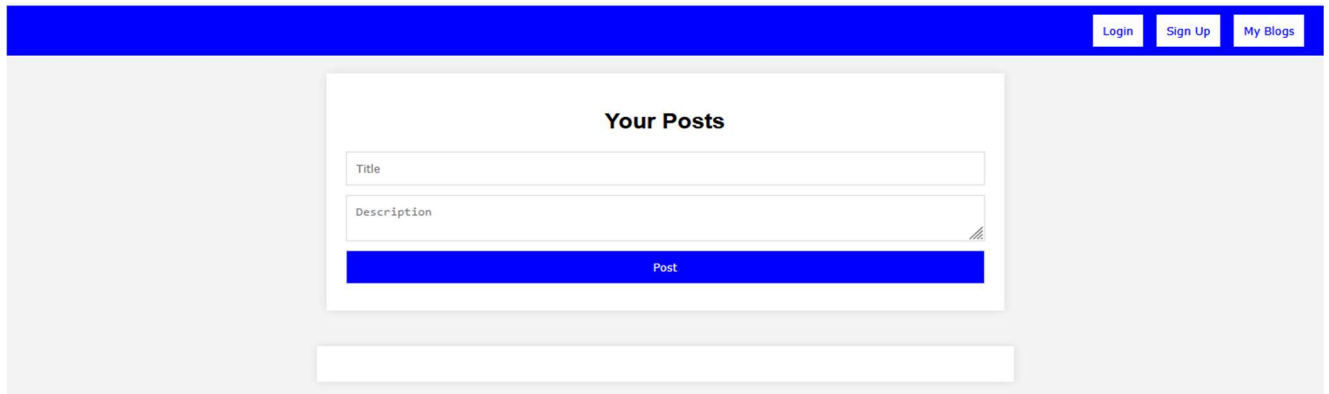
Fonction **showMyBlogs** : Affiche les messages publiés par l'utilisateur actuellement connecté. Si aucun utilisateur n'est connecté, affiche une alerte demandant de se connecter.

**Fonction showSignup** : Affiche le formulaire d'inscription et masque les formulaires de connexion et de publication de messages.

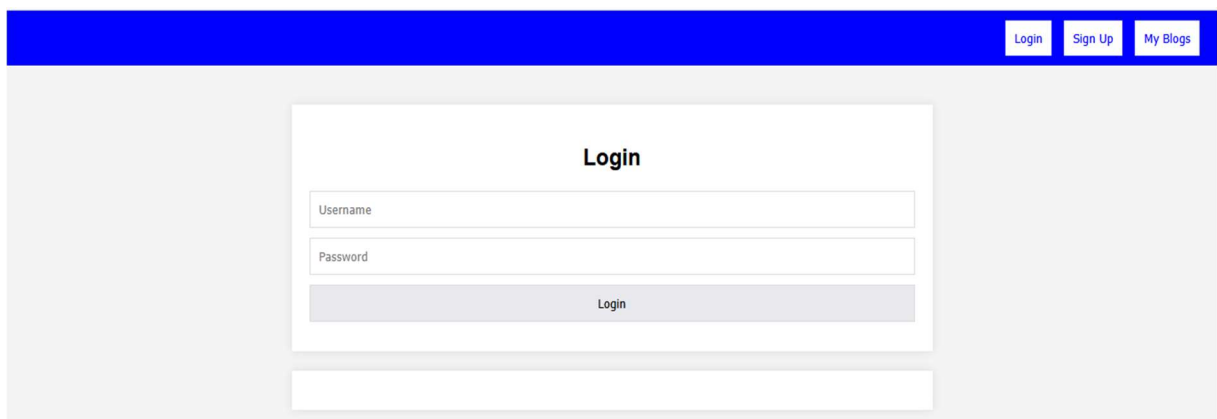
**Fonction showLogin** : Affiche le formulaire de connexion et masque les formulaires d'inscription et de publication de messages.

**Méthode toJSON** : La méthode **toJSON** convertit l'objet `Post` en une chaîne JSON. La méthode **JSON.stringify** prend l'objet actuel (représenté par `this`) et le convertit en une chaîne JSON. Cela permet de sérialiser l'objet pour qu'il puisse être stocké ou transmis facilement.

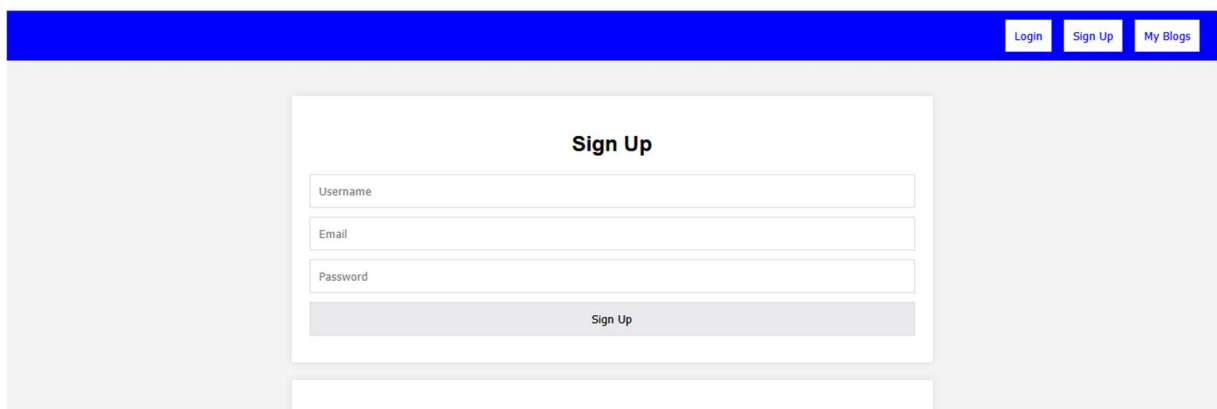
**Méthode statique fromJSON** : La méthode **fromJSON** est une méthode statique qui prend une chaîne JSON comme paramètre et crée une nouvelle instance de la classe `Post` à partir de cette chaîne.



The screenshot shows a web application interface with a blue header bar containing three buttons: "Login", "Sign Up", and "My Blogs". The main content area has a light gray background. In the center, there is a white box titled "Your Posts". Inside this box, there are two input fields: "Title" and "Description". Below these fields is a blue button labeled "Post". Below the white box, there is a white rectangular area.



The screenshot shows a web application interface with a blue header bar containing three buttons: "Login", "Sign Up", and "My Blogs". The main content area has a light gray background. In the center, there is a white box titled "Login". Inside this box, there are two input fields: "Username" and "Password". Below these fields is a gray button labeled "Login". Below the white box, there is a white rectangular area.



The screenshot shows a web application interface with a blue header bar containing three buttons: "Login", "Sign Up", and "My Blogs". The main content area has a light gray background. In the center, there is a white box titled "Sign Up". Inside this box, there are three input fields: "Username", "Email", and "Password". Below these fields is a gray button labeled "Sign Up". Below the white box, there is a white rectangular area.

[Login](#)[Sign Up](#)[My Blogs](#)

### Your Posts

### Mes Posts

#### LA NOUVELLE DU JOUR

NOUS DEVONS FAIRE UNE RECHERCHE AVANCEE SUR L'EVOLUTION DE L'IA en Afrique