

Exercice 2 - Classes ES6 et modules (sommatif)

Devis (5)

Vous avez un fichier `index.html` et `styles.css` affichant un formulaire. Vous devez faire la factorisation en programmation orientée objet de la validation *custom* procédurale du dossier de base. Plus précisément :

- **(.5)** Toute la structure *HTML* et les styles sont déjà donnés, vous ne devez pas les modifier. Vous devrez toutefois lier le fichier `main.js`.
- **(1)** Vos scripts doivent suivre la structure et la syntaxe des modules *ES6* en programmation orientée objet (*POO*). Les fichiers `.js` nécessaires (`main.js`, `Formulaire.js` et `Validation.js`) sont déjà créés.
- **(.5)** Assurez-vous d'utiliser le préfixe `#` pour déclarer comme *private* toutes propriétés et méthodes qui n'ont pas à être publiques.
- **(.5)** Tous les champs (`inputs`) sont obligatoires.
- **(.5)** Le format du courriel doit également être validé et le message d'erreur doit être conséquent.
- **(.5)** Le label de chaque champ fautif ainsi que la couleur de sa bordure deviennent rouge (gestion de la classe CSS `.error`).
- **(.5)** Un message d'erreur est également injecté. Notez que ce message nomme le champ fautif. Tâchez d'éviter une structure conditionnelle lourde et non dynamique de `if - else`.
- **(.5)** Si un champ est fautif lorsque l'utilisateur entre une valeur valide mais que le formulaire n'est toujours pas valide à la soumission, ce champ perd ses styles ainsi que son message d'erreur.
- **(.5)** Si tous les champs sont saisis, le formulaire est supprimé du *DOM* et le message 'Merci !' est affiché.
- L'objectif est de développer une validation générique qui pourra prendre en charge n'importe quel formulaire.