

Feuille d'aide – TP : Validation de formulaire en JavaScript

Structure du formulaire

Le formulaire doit contenir au minimum cinq champs :

- Nom
- Prénom
- Email
- Mot de passe
- Âge

Chaque champ doit avoir un `id` unique. Il est conseillé de prévoir un élément HTML (par exemple un `<div class="erreur">`) sous chaque champ pour y afficher un éventuel message d'erreur.

Étapes de travail

1. Capter la soumission du formulaire

Utiliser un écouteur d'événement sur le formulaire (`submit`). Utiliser `preventDefault()` pour empêcher l'envoi tant que les données ne sont pas valides.

2. Récupérer les valeurs

Récupérer la valeur de chaque champ via JavaScript (`.value`). Pensez à enlever les espaces superflus (voir `trim()`).

3. Vérifier les règles de validation

Pour chaque champ :

- Vérifiez qu'il n'est pas vide.
- Vérifiez les contraintes spécifiques indiquées dans le sujet.

Vous pouvez créer une fonction de validation par champ pour organiser le code.

4. Afficher les erreurs

Prévoir une fonction ou une logique permettant d'afficher un message d'erreur sous un champ s'il est invalide. Ne pas oublier de **vider les anciens messages** à chaque validation.

5. Ne valider que si tout est correct

Ne permettre l'envoi (ou l'affichage d'un message de succès) **que si tous les champs sont valides**. Il est conseillé d'utiliser une variable booléenne (`formulaireValide`) que vous passez à `false` dès qu'une erreur est détectée.

Rappels utiles

Validation du mot de passe

Il doit comporter :

- Au moins 8 caractères
- Une majuscule
- Une minuscule
- Un chiffre

Pour cela, vous pouvez utiliser une **expression régulière** (regex).

Validation de l'email

Vérifiez que l'email correspond au format `texte@domaine.extension`. Là encore, une **regex** est adaptée.

Validation de l'âge

Vérifiez qu'il s'agit d'un **nombre**, compris entre **18 et 99**.

Conseils méthodologiques

- Validez un champ à la fois, puis testez.
- Organisez votre code avec des fonctions pour éviter la répétition.
- N'affichez les erreurs que si un champ est invalide, pour éviter d'encombrer l'interface.

- Ne vous précipitez pas à écrire tout le code : partez d'un formulaire simple, ajoutez une règle, testez, puis passez à la suivante.
-