



Formulaires JavaScript

[< Précédent](#)[Suivant >](#)

Validation de formulaire JavaScript

La validation du formulaire HTML peut être effectuée par JavaScript.

Si un champ de formulaire (fname) est vide, cette fonction alerte un message et renvoie false pour empêcher la soumission du formulaire:

Exemple JavaScript

```
function validateForm() {  
    var x = document.forms["myForm"]["fname"].value;  
    if (x == "") {  
        alert("Name must be filled out");  
        return false;  
    }  
}
```

La fonction peut être appelée lorsque le formulaire est soumis:

Exemple de formulaire HTML

```
<form name="myForm" action="/action_page.php" onsubmit="return  
validateForm()" method="post">  
Name: <input type="text" name="fname">
```



Essayez vous-même »

JavaScript peut valider une entrée numérique

JavaScript est souvent utilisé pour valider une saisie numérique:

Veuillez saisir un nombre entre 1 et 10

Soumettre

Essayez vous-même »

Validation automatique du formulaire HTML

La validation du formulaire HTML peut être effectuée automatiquement par le navigateur:

Si un champ de formulaire (fname) est vide, l'attribut **requis** empêche l'envoi de ce formulaire:

Exemple de formulaire HTML

```
<form action="/action_page.php" method="post">
  <input type="text" name="fname" requis>
  <input type="submit" value="Submit">
</form>
```

Essayez vous-même »



La validation des données

La validation des données est le processus permettant de s'assurer que les entrées des utilisateurs sont propres, correctes et utiles.

Les tâches de validation typiques sont les suivantes:

- l'utilisateur a-t-il rempli tous les champs obligatoires?
- l'utilisateur a-t-il entré une date valide?
- l'utilisateur a-t-il saisi du texte dans un champ numérique?

Le plus souvent, le but de la validation des données est d'assurer une saisie correcte de l'utilisateur.

La validation peut être définie par différentes méthodes et déployée de différentes manières.

La validation côté serveur est effectuée par un serveur Web, une fois que les entrées ont été envoyées au serveur.

La validation côté client est effectuée par un navigateur Web avant que la saisie ne soit envoyée à un serveur Web.

Validation de la contrainte HTML

HTML5 a introduit un nouveau concept de validation HTML appelé **validation des contraintes**.

La validation des contraintes HTML repose sur:

- Validation des contraintes **Attributs d'entrée HTML**
- Validation des contraintes **CSS Pseudo-sélecteurs**
- Validation des contraintes **Propriétés et méthodes DOM**

Attributs d'entrée de validation de contrainte HTML

Attribute	Description
disabled	Specifies that the input element should be disabled
max	Specifies the maximum value of an input element



pattern	Specifies the value pattern of an input element
required	Specifies that the input field requires an element
type	Specifies the type of an input element

Pour une liste complète, accédez à [Attributs de saisie HTML](#) .

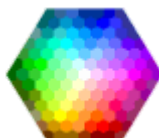
Sélecteurs CSS de validation de contraintes

Selector	Description
:disabled	Selects input elements with the "disabled" attribute specified
:invalid	Selects input elements with invalid values
:optional	Selects input elements with no "required" attribute specified
:required	Selects input elements with the "required" attribute specified
:valid	Selects input elements with valid values

Pour une liste complète, accédez à [CSS Pseudo Classes](#) .

[< Précédent](#)[Suivant >](#)

PIPETTE À COULEURS



COMMENT

Onglets
Dropdowns

[HTML](#)[CSS](#)[PLUS](#)[RÉFÉRENCES](#)[Top Navigation](#)[Boîtes modales](#)[Barres de progression](#)[Formulaire de connexion](#)[Parallax Inclut la plage de](#)[Google Maps Curseurs](#)[Infobulles Diaporama](#)[Liste de tri Liste de tri](#)

PARTAGER



CERTIFICATS

[HTML](#) , [CSS](#) , [JavaScript](#) , [PHP](#) , [jQuery](#) , [Bootstrap](#) et [XML](#) .

[Lire la suite "](#)

[HTML](#)[CSS](#)[PLUS](#)[RÉFÉRENCES](#)[Signaler une erreur](#)[IMPRIMER LA PAGE](#)[FORUM](#)[SUR](#)

Top 10 des tutoriels

[HTML Tutorial](#)[CSS Tutorial](#)[JavaScript Tutorial](#)[How To Tutorial](#)[W3.CSS Tutorial](#)[Bootstrap Tutorial](#)[SQL Tutorial](#)[PHP Tutorial](#)[jQuery Tutorial](#)[Python Tutorial](#)

Top 10 References

[HTML Reference](#)[CSS Reference](#)[JavaScript Reference](#)[W3.CSS Reference](#)[Bootstrap Reference](#)[SQL Reference](#)[PHP Reference](#)[HTML Colors](#)[jQuery Reference](#)[Python Reference](#)

Top 10 Examples

[HTML Examples](#)[CSS Examples](#)[JavaScript Examples](#)[How To Examples](#)[W3.CSS Examples](#)[Bootstrap Examples](#)[PHP Examples](#)[jQuery Examples](#)[Angular Examples](#)[XML Examples](#)

Web Certificates

[Certificat HTML Certificat](#)[CSS Certificat](#)[JavaScript Certificat](#)[jQuery Certificat](#)[PHP Certificat](#)[Bootstrap Certificat](#)[XML](#)

[HTML](#)[CSS](#)[PLUS](#)[RÉFÉRENCES](#)

pour améliorer la lecture et la compréhension de base. Les didacticiels, les références et les exemples sont constamment révisés pour éviter les erreurs, mais nous ne pouvons garantir l'exactitude de tous les contenus.

En utilisant ce site, vous acceptez d'avoir lu et accepté nos conditions d'utilisation , nos cookies et notre politique de confidentialité . Copyright 1999-2018 par Refsnes Data. Tous les droits sont réservés.

Propulsé par W3.CSS .

