

Sortie JavaScript

Possibilités d'affichage JavaScript

JavaScript peut « afficher » les données de différentes manières :

Écrire dans un élément HTML, à l'aide de `innerHTML`.

Écriture dans la sortie HTML à l'aide de `document.write()`.

Écriture dans une boîte d'alerte, à l'aide de `window.alert()`.

Écriture dans la console du navigateur, à l'aide de `console.log()`.

Utiliser innerHTML

Pour accéder à un élément HTML, JavaScript peut utiliser la méthode `document.getElementById(id)`

L'attribut `id` définit l'élément HTML. La propriété `innerHTML` définit le contenu HTML

```
<script>  
document.getElementById("demo").innerHTML = 5 + 6;  
</script>
```

La modification de la propriété `innerHTML` d'un élément HTML est un moyen courant d'afficher des données en HTML.

Utilisation de document.write()

A des fins de test, il est pratique d'utiliser `document.write()`:

```
<script>  
document.write(5 + 6);  
</script>
```

L'utilisation de `document.write()` après le chargement d'un document HTML **supprimera tout le HTML existant**

```
<button type="button" onclick="document.write(5 + 6)">Cliquez-ici</button>
```

La méthode `document.write()` ne doit être utilisée qu'à des fins de test.

Utilisation de window.alert()

Vous pouvez utiliser une boîte d'alerte pour afficher des données

```
<script>  
window.alert(5 + 6);  
</script>
```

Vous pouvez ignorer le mot - clé **window**.

En JavaScript, l'objet **window** est l'objet de portée globale, ce qui signifie que les variables, propriétés et méthodes appartiennent par défaut à l'objet **window**. Cela signifie également que la spécification du mot-clé **window** est facultative.

```
<script>alert(5 + 6);</script>
```

Utilisation de console.log()

À des fins de débogage, vous pouvez appeler la méthode `console.log()` dans le navigateur pour afficher les données.

```
<script>  
console.log(5 + 6);  
</script>
```

JavaScript Print

JavaScript n'a pas d'objet d'impression ni de méthode d'impression.

Vous ne pouvez pas accéder aux périphériques de sortie à partir de JavaScript.

La seule exception est que vous pouvez appeler la méthode `window.print()` dans le navigateur pour imprimer le contenu de la fenêtre en cours.

```
<p>Cliquez sur le bouton pour imprimer la page actuelle.</p>  
<button onclick="window.print()">Imprimer</button>
```