

INTRODUCTION À JAVASCRIPT

Histoire de Javascript :

JavaScript a été créé en 1995 par Brendan Eich alors qu'il travaillait à Netscape Communications. Le langage a été initialement développé en dix jours seulement, dans le but de permettre aux développeurs de créer des sites web dynamiques. Netscape l'a d'abord appelé Mocha, puis LiveScript, et finalement JavaScript, principalement pour profiter de la popularité de Java, un langage de programmation de Sun Microsystems, bien que les deux langages soient très différents.



INTRODUCTION À JAVASCRIPT

Pourquoi utiliser Javascript ?:

JavaScript est utilisé pour rendre les sites web interactifs. Il permet aux développeurs de manipuler le contenu d'une page web en temps réel. Cela peut signifier des choses comme changer les couleurs, les styles ou les positions des éléments, générer de nouveaux éléments en réponse à des actions de l'utilisateur, valider des formulaires avant leur soumission, créer des jeux, etc. De plus, avec l'arrivée de Node.js, JavaScript peut être utilisé pour créer des serveurs web et développer des applications back-end, ce qui en fait un langage de programmation complet pour le développement web full-stack.

INTRODUCTION À JAVASCRIPT

Où est-ce que JavaScript s'exécute ?

Traditionnellement, JavaScript est un langage de programmation côté client, ce qui signifie qu'il est exécuté dans le navigateur de l'utilisateur et non sur le serveur qui héberge le site web. Cela signifie que lorsque vous visitez un site web qui utilise JavaScript, le code est envoyé à votre navigateur et exécuté là-bas. Cependant, avec l'arrivée de Node.js, JavaScript peut maintenant être exécuté côté serveur. Cela signifie que JavaScript peut être utilisé pour écrire le code qui s'exécute sur le serveur, en plus du code qui s'exécute dans le navigateur de l'utilisateur.

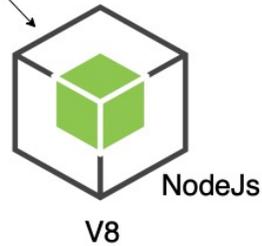
Exécution du Javascript



MOTEUR
JAVASCRIPT

Coté client

Coté serveur



JavaScriptCore ,également appelé quirrelfish ou nitro



V8



IonMonkey



Chakra

Un moteur JavaScript est un programme ou un interpréteur qui convertit le code JavaScript en code machine pour qu'il puisse être exécuté par votre ordinateur. Il est généralement intégré dans un navigateur web, mais peut aussi être utilisé de manière autonome pour exécuter du JavaScript en dehors d'un navigateur, comme dans le cas de Node.js.

Voici quelques exemples de moteurs JavaScript :

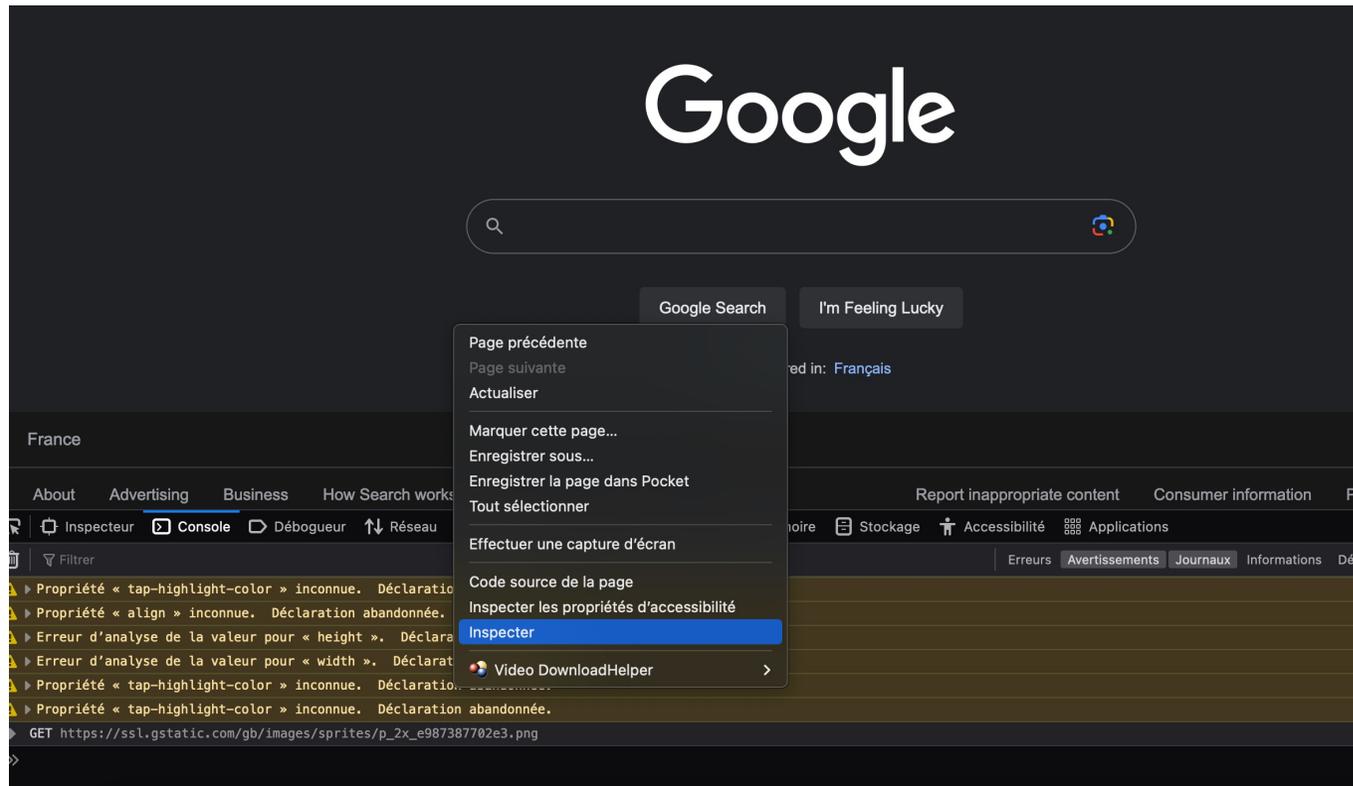
1.V8 : Développé par Google, V8 est probablement le moteur JavaScript le plus connu. Il est utilisé dans Chrome et Chromium, ainsi que dans beaucoup d'autres projets open source, y compris Node.js.

2.SpiderMonkey : C'est le premier moteur JavaScript, développé par Netscape, et c'est celui qui est actuellement utilisé par Firefox.

3.JavaScriptCore : Aussi connu sous le nom de Nitro, c'est le moteur JavaScript utilisé par Safari. Il est développé par Apple.

4.Chakra : C'est le moteur JavaScript développé par Microsoft pour son navigateur Edge. Cependant, depuis que Microsoft a adopté Chromium pour Edge, ils ont aussi commencé à utiliser V8.

Javascript côté client : La console



On peut dire que la console du navigateur est l'interface entre l'utilisateur (ou le développeur) et le moteur JavaScript. Elle permet d'interagir directement avec le moteur JavaScript pour exécuter du code, afficher les erreurs, et inspecter les objets JavaScript.

La console dans un navigateur est un outil de développement très utile qui sert à plusieurs fins. Voici quelques-unes de ses principales utilisations :

1. Exécution de code JavaScript : Vous pouvez taper du code JavaScript directement dans la console, et il sera exécuté par le navigateur. C'est un moyen rapide et facile de tester des morceaux de code JavaScript.

2. Débogage : La console affiche les messages d'erreur JavaScript. Lorsqu'une erreur se produit, le navigateur indique souvent le fichier et le numéro de ligne où l'erreur a eu lieu, ce qui peut aider à localiser et à résoudre le problème.

3. Journalisation : Les développeurs peuvent utiliser la fonction `console.log()` pour afficher des messages dans la console. Cela peut être utile pour le débogage, pour suivre l'exécution du code, ou pour afficher des valeurs de variables à des fins de test.

La console : afficher un texte dans la console :

