

Principes de base du JavaScript asynchrone

Code synchrone

Le code synchrone est exécuté séquentiellement du début d'un fichier à la fin d'un fichier – chaque ligne ne pouvant pas s'exécuter tant que la ligne précédente n'a pas fini de s'exécuter.

Code asynchrone

Le code asynchrone peut être exécuté en parallèle avec un autre code déjà en cours d'exécution.

Blocage

Le blocage se produit lorsque le code empêche un utilisateur d'interagir avec une application car le code d'arrière-plan ne termine pas son exécution.

Fils

Généralement, le nombre de tâches qu'un langage peut exécuter est limité par le nombre de threads auxquels le langage a accès.

async-javascript-single-thread

JavaScript est un langage monothread. Cependant, il peut gérer du code asynchrone à l'aide de la boucle d'événements.

async-javascript-syntaxes

En JavaScript, le code asynchrone peut être écrit de différentes manières, notamment :

- Rappels
- Promesses

- `async / await`

async-javascript-set-timeout

`setTimeout()` accepte deux arguments :

- A callback function that is executed asynchronously
- The minimal number of milliseconds a program waits before executing the callback function

async-javascript-set-interval

`setInterval()` accepts two arguments:

- A callback function that is executed asynchronously
- The number of milliseconds for how frequently the callback function executes