

PIZZA-PROMISE

Les promesses en JavaScript sont une façon de travailler avec des opérations asynchrones. Une promesse est un objet qui peut être dans l'un des trois états suivants : en attente, rempli et rejeté. Les promesses vous permettent d'éviter ce que l'on appelle « l'enfer des rappels » (callback hell) et de rendre votre code plus lisible et gérable.

Imaginez que vous voulez commander une pizza. Vous appelez la pizzeria et passez votre commande. Ce processus peut être comparé à la création d'une promesse.

1. Création de la promesse: Lorsque vous passez votre commande, vous ne savez pas exactement quand la pizza sera prête, mais on vous promet de vous la livrer. C'est une promesse - l'engagement de réaliser une action à l'avenir.

2. En attente (Pending): Pendant que la pizza est préparée, vous êtes en attente. La promesse est dans un état "pending".

3. Réalisation (Fulfilled): Lorsque la pizza est prête et livrée, la promesse est tenue et passe à l'état "fulfilled". Vous pouvez alors déguster votre pizza. Cela correspond à l'exécution réussie de la promesse.

4. Rejet (Rejected): Si quelque chose ne se passe pas comme prévu - par exemple, s'il n'y a plus d'ingrédients ou si le livreur ne trouve pas votre adresse - votre commande est annulée. La promesse est rejetée et vous recevez un message d'erreur.

Dans le code, au lieu de pizza, il peut s'agir d'une requête au serveur, d'un calcul en cours ou d'autres opérations nécessitant du temps. Les promesses aident à gérer ces actions asynchrones en fournissant des méthodes pratiques `.then()` et `.catch()` pour traiter l'exécution réussie ou les erreurs, respectivement.