Ajouter une méthode à un objet littéral, le mot clé this dans une méthode ...

Comment utiliser l'opérateur this en Javascript

Dans la plupart des langages de programmation, le mot-clé **this** est utilisé dans le code non-statique d'une classe pour se référer à l'objet courant.
Par exemple :

```
let cat={
    name:"cat",
    type:"félin",
    eat:"carnivore",
    afficher(){
        console.log(this.eat)
        console.log(this.type)
        console.log(this.name)
    },
};
cat.afficher();// renvoi "carnivore" , "félin", "cat"
```

Qu'est-ce que this ?

this est un opérateur et comme tout opérateur il retourne une valeur.

1. **D'où vient cette valeur ?** Cette valeur provient d'un contexte d'exécution. Il faut savoir qu'au lancement du script et ensuite à chaque appel d'une fonction un contexte d'exécution est placé en pile d'exécution.
2. **Qu'est qu'il y a dans un contexte d'exécution ?** Il y a la valeur de this ça je viens de le dire et il y a aussi des informations qui vont permettre à la fonction de s'exécuter par exemple les arguments qui lui sont passés. Je ne détaille pas plus. Mais vous pouvez déjà comprendre qu'à chaque appel d'une fonction étant donné que l'on change les arguments et bien on va créer un nouveau contexte. Ce qu'il faut retenir ici c'est l'idée car en fait il y a beaucoup plus d'informations que ça dans un contexte d'exécution.
3. **Qui gère cette valeur de this ? Qui l'affecte ?** Et bien c'est le moteur JavaScript qui le fait et qui le fait pour nous.

4. **Alors c'est gentil mais à quoi ça va me servir ?** Par exemple quand vous instanciez une classe pour créer un nouvel objet et bien dans la fonction constructeur this va désigner ce nouvel objet. Autre exemple : quand vous passez un callback à un gestionnaire d'événement et bien dans ce callback this va désigner l'élément du document sur lequel est posé ce gestionnaire.
5. **Et si la valeur de this ne vous convient pas ?** il faut que vous sachiez que vous pouvez la changer en utilisant les méthodes call, apply ou bind du constructeur Function. On voit ça dans un autre tuto.
6. **Que vaut this ?** Pour pouvoir utiliser la valeur de this vous devez être capable de prévoir ce que va faire le moteur JavaScript et pour cela on va étudier dans ce tuto **les situations les plus courantes dans lesquelles les fonctions vont être invoquées**. Il faut aussi savoir que la valeur de this peut dépendre **dans un très petit nombre de cas** de l'utilisation ou pas du mode strict et aussi de l'environnement d'exécution de JavaScript à savoir le navigateur ou un serveur.

Dans une méthode this renvoi l'objet dans lequel est contenu la méthode.

Qu'est-ce qu'une méthode :

Les objets peuvent contenir des propriétés et des méthodes. Les propriétés sont des valeurs dont est constitué notre objet et qui peuvent changer tandis que les méthodes représentent des tâches que nous aurons associé à notre objet. Il existe deux notations pour créer et manipuler un objet, de manière littérale et par l'intermédiaire d'un constructeur.

Une méthode est une fonction contenue dans un objet.

Plusieurs méthodes peuvent être contenu dans un objet, une méthode peu en appelé un autre.

```
let cat={
    name:"cat",
    type:"félin",
    eat:"carnivore",
    afficher(){
        console.log(this.eat)
        console.log(this.type)
        console.log(this.name)
    },
    test(){
        this.afficher()
    }
};
cat.test(); // renvoi "carnivore" , "félin", "cat"
```