

Les modules de base

Les modules se répartissent en trois catégories : les modules de base, les modules de fichiers et les modules npm.



Les modules de base ont des noms réservés qui sont fournis par Node lui-même, comme 'fs'. Pour utiliser un module de base, il suffit d'écrire `require('nom_du_module')`. Le type de l'objet renvoyé dépendra du module.

Nous allons examiner le [module event](#).


1. Module events

Créer un fichier  app.js

 app.js


1. `const EventEmitter = require('events');` 
2. `const emitter = new EventEmitter();`
- 3.
4. `emitter.on('messageLogged', ({msg}) => {`
5. `console.log(`listener called ${msg}`)`
6. `})`
- 7.
8. `const log = require('./hello');` 
9. `log("test")`

Lig. 4 : Ajout d'un écouteur. Les noms des événements sont des chaînes de caractères en majuscules, mais toute clé de propriété JavaScript valide peut être utilisée.

Créer un fichier  hello.js.

 hello.js

```

1. const EventEmitter = require('events');
2. const emitter = new EventEmitter();
3.
4. const urgence = "absolue"
5. function affiche(message) {
6.   console.log(`${urgence} : ${message}`);
7.   emitter.emit('messageLogged', {msg:"Hello!"})
8. }
9.
10. module.exports = affiche; 

```

Lancez votre application `node app.js`

`C:\Users\DD\Test>node app.js`

 
absolue : test

L'affichage de l'événement est absent !

Comment corriger ce code pour avoir les deux messages affichés ?



Aide : Il est rare dans les applications d'utiliser directement un objet émetteur. On a l'habitude de créer des classes héritant de la classe event.

Correction :

La correction est classique, elle passe par la création d'une classe [qui hérite](#) de la classe EventEmitter.

Tous les objets qui émettent des événements sont des instances de la classe EventEmitter. Ces objets exposent une fonction `eventEmitter.on()` qui permet de joindre une ou plusieurs fonctions à des événements nommés émis par l'objet.

app.js

```

1. const Logger = require('./logger');
2.
3. const logger = new Logger();
4.
5. logger.on('messageLogged', ( {date} ) => {
6.   console.log(`listener called at ${date.toLocaleString()}`)
7. })
8.
9. logger.log("Bonjour !");

```

logger.js

```

1. const EventEmitter = require('events');
2.

```

```
3. const level = "absolute";
4.
5. class Logger extends EventEmitter {
6.   log(message){
7.     console.log(`level : ${level} - ${message}`);
8.     this.emit('messageLogged', {date:new Date()})
9.   }
10.}
11. module.exports = Logger;
```