

FLORIAN BÊME

BUILDER PATTERN

CONSTRUIS DES OBJETS COMPLEXES
PROPREMENT !



01

C'est quoi ?

Un design pattern qui te permet de construire tes objets de façon lisible et claire, étape par étape.



02

Le problème

Ton constructeur devient illisible avec trop de paramètres...

- ✗ Constructeur avec plein de paramètres
- ✗ Ordre des paramètres impossible à retenir
- ✗ Impossible de savoir qui fait quoi



03

La solution : Builder

Construis ton objet brique par brique !

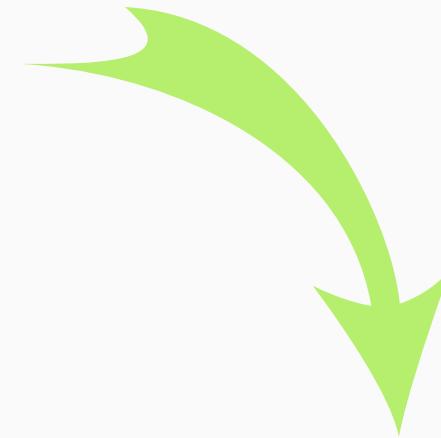
- Syntaxe fluide et lisible
- Paramètres optionnels gérés proprement
- Validation centralisée dans build()



Implémentation

```
// SANS BUILDER ✘  
// Impossible de savoir ce que ça fait !
```

```
const user = new User(  
  'John',  
  'Doe',  
  'john.doe@mail.com',  
  undefined,  
  undefined,  
  undefined,  
  'admin',  
  true  
)
```



```
// AVEC BUILDER ✓  
class UserBuilder {  
  setFirstName(firstName) {  
    this.firstName = firstName;  
    return this; // ← Retourne 'this' pour permettre le chaînage  
  }  
  // ... autres setters  
  build() {  
    return new User(this);  
  }  
}  
  
// Utilisation  
const user = new UserBuilder()  
  .setFirstName('John')  
  .setLastName('Doe')  
  .setEmail('john.doe@mail.com')  
  .setRole('admin')  
  . setActive(true)  
  .build();
```