

Property getters and setters

Có 2 loại thuộc tính object:

- Đầu tiên là thuộc tính dữ liệu. Chúng ta đều biết cách làm việc với chúng, 99% những thuộc tính chúng ta làm việc là thuộc tính dữ liệu
- Loại thứ 2 là thuộc tính bộ truy cập (accessor properties). Chúng là những function đặc biệt thực thi khi get và set một giá trị.

Getter và setter

Phương thức Getter, setter được coi là thuộc tính bộ truy cập

```
let user = {
  name: 'John',
  surname: 'Smith',

  get fullName() {
    return `${this.name} ${this.surname}`
  },

  set fullName(value) {
    ;[this.name, this.surname] = value.split(' ')
  }
}

// set fullName được thực thi với giá trị được gán phía dưới
user.fullName = 'Alice Cooper'

alert(user.name) // Alice
alert(user.surname) // Cooper
```

Và bây giờ ta có một thuộc tính "ảo" là `fullName`. Nó có thể đọc và viết

Accessor descriptors - Bộ mô tả của bộ truy cập

Bộ mô tả cho thuộc tính bộ truy cập thì khác một chút so với thuộc tính dữ liệu. Với thuộc tính bộ truy cập, nó không có `value` và `writable`, nhưng lại có thêm `get` và `set` function. Đây là những gì mà một bộ mô tả của bộ truy cập có

- `get`: một function mà không có tham số, nó hoạt động khi thuộc tính được đọc
- `set`: một function mà có một tham số, nó được gọi khi thuộc tính được set
- `enumerable`: tương tự thuộc tính dữ liệu
- `configurable`: tương tự thuộc tính dữ liệu

Với một object được tạo sẵn, chúng ta có thể tạo getter và setter thông qua `defineProperty`

```
let user = {
  name: 'John',
  surname: 'Smith'
}

Object.defineProperty(user, 'fullName', {
  get() {
    return `${this.name} ${this.surname}`
  },

  set(value) {
    ;[this.name, this.surname] = value.split(' ')
  }
})

alert(user.fullName) // John Smith

for (let key in user) alert(key) // name, surname
```

getters/setters thông minh

Nếu chúng ta muốn cấm đặt một cái tên quá dài cho `user`, chúng ta có thể dùng cách này

```
let user = {
  get name() {
    return this._name
  },
  set name(value) {
    if (value.length < 4) {
      alert('Name is too short, need at least 4 characters')
      return
    }
    this._name = value
  }
}

user.name = 'Pete'
alert(user.name) // Pete

user.name = '' // Name is too short...
```

Vậy nên tên sẽ được lưu trữ trong thuộc tính `_name`, và truy cập nó thông qua getter và setter