

# HÀM FIRST-CLASS VÀ HIGHER-ORDER

## HÀM FIRST-CLASS

- 👉 JavaScript coi các hàm này là các first-class citizen
- 👉 Điều này có nghĩa là các hàm **chỉ đơn giản là các giá trị**
- 👉 Hàm chỉ là một "loại" đối tượng khác

👉 Lưu trữ hàm trong các biến hoặc thuộc tính:

```
const add = (a, b) => a + b;  
  
const counter = {  
  value: 23,  
  inc: function() { this.value++; }  
};
```

👉 Chuyển hàm dưới dạng đối số vào các hàm KHÁC:

```
const greet = () => console.log('Hey Jonas');  
btnClose.addEventListener('click', greet);
```

👉 Trả hàm về từ hàm

👉 Phương thức gọi trên hàm:

```
counter.inc.bind(someOtherObject);
```

## HÀM HIGHER-ORDER

- 👉 Một hàm **nhận** một hàm khác làm đối số, **trả về** một hàm mới, hoặc **cả hai**
- 👉 Điều này chỉ có thể thực hiện nhờ hàm first-class

👉 Hàm nhận một hàm khác

```
const greet = () => console.log('Hey Jonas');  
btnClose.addEventListener('click', greet);
```

Hàm higher-order

Hàm callback



👉 Hàm trả về hàm mới

```
function count() {  
  let counter = 0;  
  return function() {  
    counter++;  
  };  
}
```

Higher-order function

Returned function