

Prototype e Heranca em JavaScript

1. prototype

O 'prototype' é um objeto do qual outros objetos herdam métodos. Usado em funções construtoras e classes nativas. Exemplo:

```
function Pessoa(nome) {  
  this.nome = nome;  
}  
Pessoa.prototype.dizerOi = function() {  
  return `Oi, eu sou ${this.nome}`;  
};
```

2. __proto__

É uma referência interna ao prototipo de um objeto. Exemplo:

```
let obj = {};  
console.log(obj.__proto__ === Object.prototype); // true
```

Uso desaconselhado em produção.

3. Object.getPrototypeOf()

Forma moderna de acessar o prototipo de um objeto. Exemplo:

```
const carro = { marca: 'Fiat' };  
console.log(Object.getPrototypeOf(carro));
```

4. Object.setPrototypeOf()

Permite alterar o prototipo de um objeto. Exemplo:

```
let animal = { som: 'rugido' };  
let leao = {};  
Object.setPrototypeOf(leao, animal);  
console.log(leao.som);
```

5. Object.create()

Cria um novo objeto com prototipo definido. Exemplo:

```
const pessoa = {  
  falar() { return 'Ola!'; }  
};  
  
const cliente = Object.create(pessoa);  
console.log(cliente.falar());
```