**Advantage**

Avoid re-created instance method to this inside of the constructor.

**method() implementation**

The method() takes two parameters:

• The name of the new method

• The implementation of the method

[复制代码](javascript:void(0);)

if (typeof Function.prototype.method !== "function") {

Function.prototype.method = function (name, implementation) {

this.prototype[name] = implementation;

return this;

};

}

[复制代码](javascript:void(0);)

**How to use this pattern**

[复制代码](javascript:void(0);)

var Person = function (name) {

this.name = name;

}.method('getName', function () {

return this.name;

}).method('setName', function (name) {

this.name = name;

return this;

});

var a = new Person('Adam');

a.getName(); // 'Adam'

a.setName('Eve').getName(); // 'Eve'

[复制代码](javascript:void(0);)

**References*:***

*JavaScript Patterns -*by Stoyan Stefanov (O`Reilly)