

## 第十二课 继承

### 学习目录

- 什么是继承
- 构造函数进阶应用
- 继承的方式

### 一 . 什么是继承

许多支持面向对象编程的语言都支持继承，继承分别有两种方式，一种是接口继承，另一种是实现继承，在 js 中只有实现继承，我们主要讨论的也是实现继承，实现继承的方式主要是依靠原型链来做到的。

其实继承简单来理解就是利用原型链让对象 a 拥有对象 b 的属性和方法。

```
var obj1 = {  
  
  a:100,  
  
  getNum:function(){  
  
    console.log(this.a);  
  
  }  
  
};  
  
var obj2 = obj1;  
  
function Fn1(){  
  
  this.num = 100;
```

```
}  
  
Fn1.prototype.getNum = function(){  
    console.log(this.num);  
}
```

```
var f1 = new Fn1();  
  
f1.getNum();
```

```
function Fn2(){  
    this.job = 'web 培训'  
}
```

```
Fn2.prototype = new Fn1();  
  
Fn2.prototype.getJob = function(){  
    console.log(this.job);  
}
```

```
var f2 = new Fn2();  
  
f2.getNum();  
  
f2.getJob();
```

1. 每个构造函数都有一个原型对象 Fn1.prototype。
2. 每个原型对象都包含一个指向该构造函数的指针 Fn1.prototype.constructor。
3. 每个 new 出来的实例都包含一个指向原型对象的隐式原型 f1.\_\_proto\_\_

如果让一个构造函数的原型对象等于另一个构造函数的实例，根据原型链的这种关系，那么这个构造函数的 `Fn2.prototype = new Fn1()`，`var f1 = new Fn1()` 实际上是 `Fn2.prototype` 根据 `__proto__` 找到了 `f1.__proto__`，`f1.__proto__` 又找到了 `Fn1.prototype`，也就是 `Fn2.prototype.__proto__` 通过 `f1.__proto__` 隐式原型拥有了 `Fn1.prototype` 的属性和方法。

```
console.log(Fn1.prototype == f1.__proto__); // true
```

```
console.log(Fn2.prototype.__proto__ == f1.__proto__); // true
```

```
console.log(Fn2.prototype.__proto__ == Fn1.prototype); // true
```

## 二．构造函数进阶应用

问题：为什么我们要在构造函数里定义属性，在构造函数的原型对象上写方法？

1. 从实例化的过程来看，每个实例对象的属性大多数情况下都是不同的，所以大多数属性是可变属性，定义在构造函数里的属性优先级较高，所有的实例对象可以快速访问使用构造函数中的属性。

2. 我们可以通过实例对象来访问保存在构造函数原型对象上的值，但是不能通过实例对象来重写原型中的值，如果一个实例对象修改了一个原型上的属性，其他实例对象在去访问这个实例属性的时候这个实例属性就不是以前的属性了。

3. 方法大多数情况下是固定的，因此不必每次实例化的时候都把所有的方法实例化复制一遍。

4. 构造函数的继承是依托于原型链来完成，而在原型链中，构造函数的原型对象大多数情况下都是用来共享方法的，当然也可以共享一些很少改变的共享公用属性。

5. 每次实例化都可以得到一个构造函数实例属性的副本，同时又可以共享着公用的方

法，这样也最大限度的节省了内存，同时构造函数中的属性也支持传递参数来动态修改。

这种组合使用构造函数与原型组合使用的方式，在 js 组件化开发过程中最通用用到的组件化开发方式。

```
function Fn1(name,num){  
  
    this.name = name;  
  
    this.num = num;  
  
}  
  
Fn1.prototype.getTearcher = function(){  
  
    console.log(this.name + this.num);  
  
}  
  
var f1 = new Fn1('阿凯老师',100);  
  
var f2 = new Fn1('小燕老师',200);  
  
f1.getTearcher();  
  
f2.getTearcher();
```

**谢谢观看！**

我是星星课堂老师：周小周