

上文([理解javascript原型和作用域系列\(1\)——一切都是对象](#))已经提到，函数就是对象的一种，因为通过instanceof函数可以判断。

```
var fn = function () { };  
console.log(fn instanceof Object); // true
```

对！函数是一种对象，但是函数却不像数组一样——你可以说数组是对象的一种，因为数组就像是对象的一个子集一样。但是函数与对象之间，却不仅仅是一种包含和被包含的关系，函数和对象之间的关系比较复杂，甚至有一点鸡生蛋蛋生鸡的逻辑，咱们这一节就缕一缕。

还是先看一个小例子吧。

```
function Fn() {  
    this.name = '王福朋';  
    this.year = 1988;  
}  
  
var fn1 = new Fn();
```

上面的这个例子很简单，它能说明：对象可以通过函数来创建。对！也只能说明这一点。但是我要说——**对象都是通过函数创建的**——有些人可能反驳：不对！因为：

```
var obj = { a: 10, b: 20 };  
var arr = [5, 'x', true];
```

但是不好意思，这个——真的——是一种——“快捷方式”，在编程语言中，一般叫做“语法糖”。

做“语法糖”做的最好的可谓是微软大哥，它把他们家C#那小子弄的不男不女从的，本想图个人见人爱，谁承想还得到处跟人解释——其实它是个男孩！

话归正传——其实以上代码的本质是：



复制代码

```
//var obj = { a: 10, b: 20 };  
//var arr = [5, 'x', true];
```

```
var obj = new Object();  
obj.a = 10;  
obj.b = 20;
```

```
var arr = new Array();  
arr[0] = 5;  
arr[1] = 'x';  
arr[2] = true;
```



复制代码

而其中的 Object 和 Array 都是函数：

```
console.log(typeof (Object)); // function  
console.log(typeof (Array)); // function
```

所以，可以很负责的说——**对象都是通过函数来创建的。**

现在是不是糊涂了——对象是函数创建的，而函数却又是一种对象——天哪！函数和对象到底是什么关系啊？

别着急！揭开这个谜底，还得先去了解一下另一位老朋友——prototype原型。

本系列文章不打算动辄几千字的长篇大论，咱们小步快跑，不至于看的太乏味。