

[←](#) 2-17 Class

## 类 - Class



## Class



Javascript是一种基于对象（object-based）的语言，你遇到的所有东西几乎都是对象。但是，它又不是一种真正的面向对象编程（OOP）语言，因为它的语法中没有class（类）。



这是摘自阮一峰老师的博客，这句话放在 ES5 可以说不为过，然而到了 ES6 这么说就已经不严谨了。因为 ES6 中已经有了专属的 class 语法了

有的同学喜欢函数式的编程方式，有的同学喜欢面向对象的编程思维，我们今天不论长短，重点讲述怎么使用 ES6 的 class 语法完成面向对象的发。

## Intro

对于面向对象编程而言，更关注类的声明、属性、方法、静态方法、继承、多态、私有属性。

## Basic Syntax

首先我们要先来说明在 JavaScript 世界里如何声明一个“类”。在 ES6 之前大家都是这么做的：

```
let Animal = function (type) {  
  this.type = type  
  this.walk = function () {  
    console.log(`I am walking`)  
  }  
}  
  
let dog = new Animal('dog')  
let monkey = new Animal('monkey')
```

在上述代码中，我们定义了一个叫 Animal 的类，类中声明了一个属性 type、一个方法 walk；然后通过 new Animal 这个类生成实例，完成了类定义和实例化。当然你也可以这样写：

```
let Animal = function (type) {  
  this.type = type  
}  
  
Animal.prototype.walk = function () {  
  console.log(`I am walking`)  
}  
  
let dog = new Animal('dog')  
let monkey = new Animal('monkey')
```

在 ES6 中把类的声明专业化了，不在用 function 的方式了，请看：

```
class Animal {  
  constructor (type) {  
    this.type = type  
  }  
  walk () {  
    console.log(`I am walking`)  
  }  
}  
  
let dog = new Animal('dog')  
let monkey = new Animal('monkey')
```

很明显，从定义上就很专业了，有构造函数、方法，但是 ES6 增加了新的数据类型 class 吗？

```
console.log(typeof Animal); //function
```