

# extends

**extends** 关键词被用在类声明或者类表达式上，以创建一个类是另一个类的子类。

## 语法

```
class ChildClass extends ParentClass { ... }
```

## 描述

**extends** 关键词用来创建一个普通类或者内建对象的子类。

扩展的`.prototype`必须是一个 `Object` 或者 `null`。

## 示例

### 使用 extends

第一个例子是根据名为 `Polygon` 类 创建一个名为 `Square` 的类。 你可以从[实战演示](#)看到这个例子。

```
1  class Square extends Polygon {  
2      constructor(length) {  
3          // 这里把length传参给父类的构造方法  
4          // 作为父类Polygon的宽和高  
5          super(length, length);  
6          // 备注：在衍生类中使用this前必须先调用super()方法  
7          // 忽视这一点将会导致一个引用错误  
8          this.name = 'Square';  
9      }  
10  
11     get area() {  
12         return this.height * this.width;  
13     }  
14  
15     set area(value) {  
16         this.area = value;  
17     }  
18 }
```

```
17 |     }  
18 | }
```

## 使用 extends 扩展内建对象

这个示例继承了 `Date` 对象。你可以从[实战演示](#)看到这个例子。

```
1  class myDate extends Date {  
2      constructor() {  
3          super();  
4      }  
5  
6      getFormattedDate() {  
7          var months = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'];  
8          return this.getDate() + "-" + months[this.getMonth()] + "-" + this.getFullYear();  
9      }  
10 }
```

## 扩展 null

可以像扩展普通类一样扩展 `null`，但是新对象的原型将不会继承 `Object.prototype`。

```
class nullExtends extends null {  
    constructor() {}  
  
    //don't work in Chrome  
    //in Chrome, supper() method must be invoke in constructor()  
    //but null can't be invoked as constructor  
}  
  
Object.getPrototypeOf(nullExtends); // Function.prototype  
Object.getPrototypeOf(nullExtends.prototype) // null
```

## 详述

详述	状态	注释
<a href="#">ECMAScript 2015 (6th Edition, ECMA-262)</a> extends	ST Standard	最初的定义
<a href="#">ECMAScript Latest Draft (ECMA-262)</a> extends	LS Living Standard	

# 浏览器兼容性

	Desktop	Mobile			
特性	Chrome	Firefox (Gecko)	Internet Explorer	Opera	Safari
基本支持	42.0	仅在夜间频道（Nightly channel）可用（自2015年3月）	?	?	?
数组的子类	43.0	未实现	?	?	?

## 扩展阅读

- [类](#)
- [super](#)