

## ES6 Class

---

- class的基本语法
  - class的私有属性, #仅class可以调用, 实例不可
  - Generator方法
  - class的静态方法 static 不会被实例继承 类相当于实例的原型, 所有在类中定义的方法, 都会被实例继承。如果在一个方法前, 加上static关键字, 就表示该方法不会被实例继承, 而是直接通过类来调用, 这就称为“静态方法”
  - class的静态属性和实例属性
  - new.target
- class的继承
  - 通过extends 关键字实现继承
  - super关键字, 表示父类的构造函数, 用来新建父类的this对象。子类必须在constructor方法中调用super方法, 否则新建实例时会报错。这是因为子类自己的this对象, 必须先通过父类的构造函数完成塑造, 得到与父类同样的实例属性和方法, 然后再对其进行加工, 加上子类自己的实例属性和方法。如果不调用super方法, 子类就得不到this对象。
- ES5的继承, 实质是先创造子类的实例对象this, 然后再将父类的方法添加到this上面(Parent.apply(this)).ES6的继承机制完全不同, 实质是先将父类实例对象的属性和方法, 加到this上面 (所以必须先调用super方法), 然后再用子类的构造函数修改this。
  - 如果子类没有定义constructor方法, 这个方法会被默认添加, 代码如下。也就是说, 不管有没有显式定义, 任何一个子类都有constructor方法。