ES6新增了块级作用域，总结一句话大致就是：大括号{}包起来的代码块基本山都可以当做块级作用域。函数体大括号，if-else大括号，for循环大括号，switch大括号，try-catch大括号等。需要注意的是，for循环，每一次循环时的{}都是一个独立的块级作用域。

使用let或const声明的变量只在当前块级作用域生效，出了这个代码块，就无法访问。日常开发中，块级作用域中使用的变量，尽量使用let或者const声明。需要注意的问题:let和const变量一定要先声明，再使用，避免出错。不要试图利用变量提升的特性。  
const声明变量时，一定要初始化，否则会报错。let建议也在声明时初始化。  
const声明的变量一旦初始化，以后就不可以在进行赋值操作，但可以对其引用的对象进行更改。

let和const声明的变量不能再重复声明。虽然var可以无限次重复声明，但是并不适用于这两个新的声明方式。、

箭头函数，它提供了一种极致优雅和简洁的方式来定义匿名函数。

简而言之，箭头函数就是丢掉了关键字 function，然后用一个箭头 => 来分离一个匿名函数的参数部分和函数体。在使用传统的匿名函数时，箭头函数还消除了一个关键的错误源，即函数内的 this 对象的值。使用箭头函数，this 是基于词法绑定，这仅仅是意味着它的值被绑定到父级作用域的一种奇特的方式，并且永远不会改变。如果一个箭头函数定义在一个自定义对象 countup 中，this 值毫无疑问地指向 countup。

对比传统匿名函数，this 的值在哪变化取决于它被定义的上下文环境。当在上面的例子中尝试引用 this.counter，结果将返回 undefined，这种行为可能会把很多不熟悉动态绑定的复杂性的人搞糊涂。使用箭头函数，this 的值总是可预测并且容易推断的。