

• var、let 及 const 区别

■ 前置知识

■ 变量提升

```
console.log(a) // undefined
var a = 1
```

解析：上述代码中，输出了a，此时a还未声明，但是我们却可以正常输出这个未被声明的变量，这就是我们所说的变量提升，提升的仅仅是声明变量的操作

实际执行效果：

```
var a
console.log(a)
a = 1
```

不仅声明变量会提升，声明函数也会被提升

```
console.log(a) // f a() {}
function a() {}
var a = 1
```

并且函数声明优先于变量声明，这段代码实际的执行效果如下

```
var a
function a() {}
console.log(a) // f a() {}
a = 1
// 优先的意思就是函数声明会覆盖变量声明
```

说完了这些，想必大家也知道 var 存在的问题了，使用 var 声明的变量会被提升到作用域的顶部，接下来我们再来看 let 和 const。

```
var a = 1
let b = 1
const c = 1
console.log(window.a) // 1
console.log(window.b) // undefined
console.log(window.c) // undefined
function test(){
  console.log(a)
  let a
}
test() // 报错, a is not defined
```

■ 总结：

- 首先在全局作用域下使用 let 和 const 声明变量，变量并不会被挂载到 window 上，而 var 会
- let 和 const 会生成块级作用域，变量只能在块级作用域中使用。如在函数里使用 let 和 const，则块级作用域就是函数作用域
- let 和 const 有个特点，叫暂时性死区，变量在作用域内如未被声明，则不能使用
- let 和 const 不能重复声明，const 初始化时就要进行赋值，并且中途不允许修改值，可以修改对象的属性值，因为 const 声明变量时只是保证指向这个对象的指针不变而已