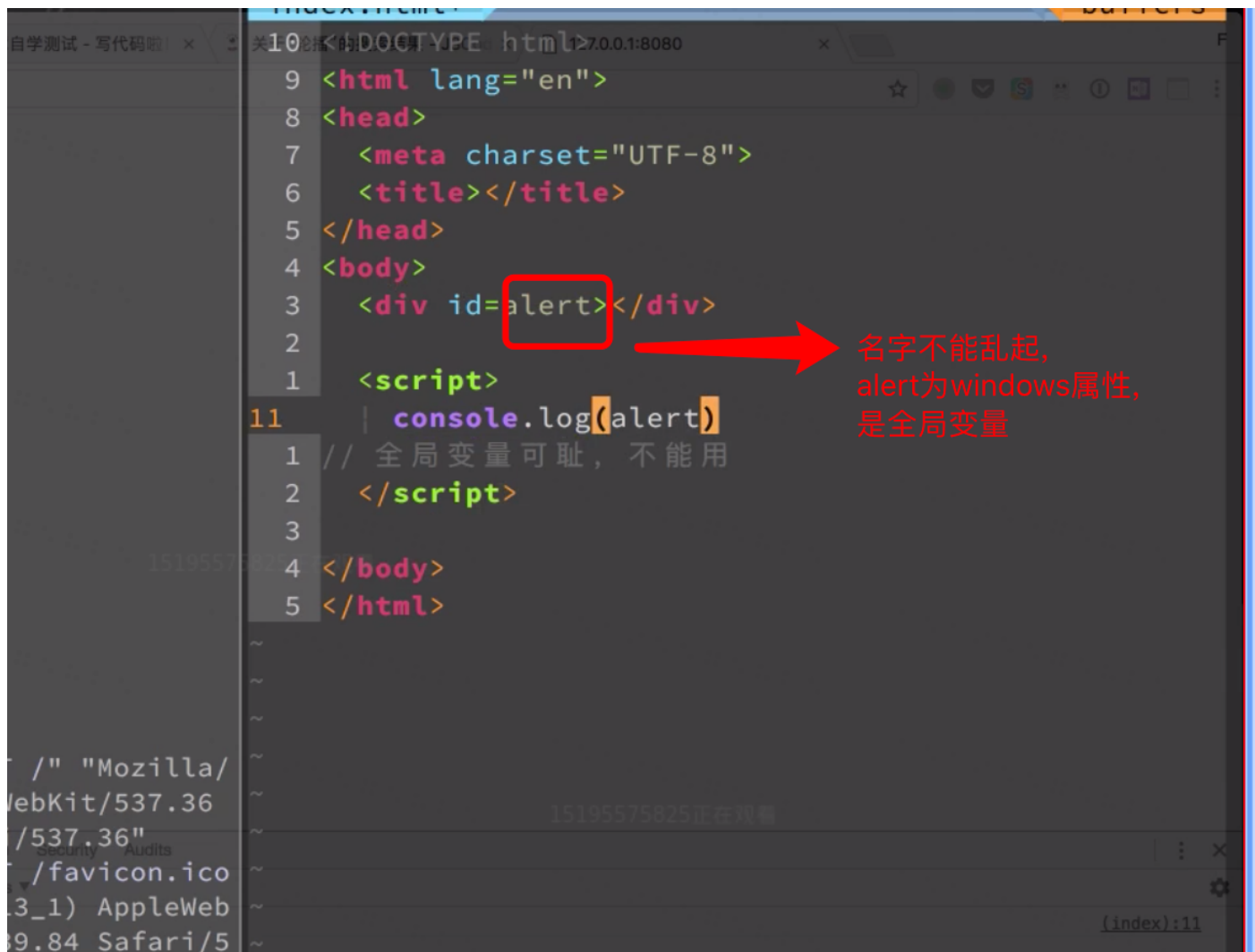


let

立即执行函数



```
10 <!DOCTYPE html>
9 <html lang="en">
8 <head>
7   <meta charset="UTF-8">
6   <title></title>
5 </head>
4 <body>
3   <div id=alert></div>
2
1   <script>
11     | console.log(alert)
1 // 全局变量可耻，不能用
2   </script>
3
4 </body>
5 </html>
```

名字不能乱起，
alert为windows属性，
是全局变量

正确:

```
10 <!DOCTYPE html>
9 <html lang="en">
8 <head>
7   <meta charset="UTF-8">
6   <title></title>
5 </head>
4 <body>
3   <div id=x></div>
2
1   <script>
11     console.log(x)
1 // 全局变量可耻，不能用
2   </script>
3
4 </body>
5 </html>
~
~
~
~
```

测试代码:(这里不要用全局变量,这里要么你把全局变量覆盖,要么全局变量把你覆盖!)

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Title</title>
6 </head>
7 <body>
8 <div id= self> </div>
9 <script>
10   var parent = document.querySelector('#self')
11   console.log(parent)
12 </script>
13 </body>
14 </html>
15
```

立即调用:

```
10 <!DOCTYPE html>
9 <html lang="en">
8 <head>
7   <meta charset="UTF-8">
6   <title></title>
5 </head>
4 <body>
3   <div id=self></div>
2
1   <script>
11  function x() {
1   |   var parent = document.querySelector('#self')
2   |   console.log(parent)
3   | }
4
5  x.call()
6  // 全局变量可耻，不能用
7  </script>
8
9 </body>
10 </html>
```

```
<script>
function () {
  |   var parent = document.querySelector('#self')
  |   console.log(parent)
}.call()
// 全局变量可耻，不能用
</script>
```

上面这种写法浏览器会报错;

用下面这种:

```
13<!DOCTYPE html>
12<html lang="en">
11<head>
10  <meta charset="UTF-8">
9    <title></title>
8</head>
7<body>
6  <div id=self></div>
5
4  <script>
3  (function(){
2    | var parent = document.querySelector('#self')
1    | console.log(parent)
14 } .call())
1 // 全局变量可耻，不能用
2  </script>
3
4</body>
5</html>
```

let

let让变量只在 `{ }` 里面有用!

```
14 <!DOCTYPE html>
13 <html lang="en">
12 <head>
11   <meta charset="UTF-8">
10   <title></title>
9 </head>
8 <body>
7   <div id=self></div>
6
5   <script>
4
3   {
2     let a = 1
1   }
15 console.log(a) // undefined
1
2 // 全局变量可耻，不能用
3   </script>
4
5 </body>
6 </html>
```

15195575825正在观看

(index):16

```
13 <!DOCTYPE html>
12 <html lang="en">
11 <head>
10   <meta charset="UTF-8">
9   <title></title>
8 </head>
7 <body>
6   <div id=self></div>
5
4   <script>
3     var a = 1
2     !function(){
1       var a = 2
4     }.call()
5
3   // 全局变量可耻，不能用
4   </script>
5
6 </body>
7 </html>
```

ES6之前
只能这样
声明局部
变量

立即执行函数

立即执行函数:(不想覆盖任何全局变量就用立即执行函数!)

```
7  *
8  * Copyright JS Foundation and other contributors
9  * Released under the MIT license
10 * https://jquery.org/license
11 *
12 * Date: 2017-03-20T18:59Z
13 */
14 (function( global, factory ) { ...
38
39 // Pass this if window is not defined yet
40 }( typeof window !== "undefined" ? window : this, function( window, noGlobal ) {
41
42 // Edge <= 12 - 13+, Firefox <=18 - 45+, IE 10 - 11, Safari 5.1 - 9+, iOS 6 - 9.1
43 // throw exceptions when non-strict code (e.g., ASP.NET 4.5) accesses strict mode
44 // arguments.callee.caller (trac-13335). But as of jQuery 3.0 (2016), strict mode should be common
45 // enough that all such attempts are guarded in a try block.
46 "use strict";
47
48 var arr = [];
49
50 var document = window.document;
51
52 var getProto = Object.getPrototypeOf;
```