W1D2

JavaScript包含以下几个部分：

核心（ECMAScript）

文档对象模型（DOM)

* 是一个API, 用于在HTML中使用扩展的XML

浏览器对象模型（BOM）

* 用于支持访问和操作浏览器的窗口

Script: 可以直接在网页中嵌入JavaScript代码，通过他在网页中包含外部JS文件

<script>

Function sayHi(){

Console.log(“Hi!”).

}

</script>

<script src = “example.js”></script>

标签位置：将js引用放在<body>元素中的页面内容后面，这样页面会在处理JS代码之前完全渲染页面。

推迟执行脚本：defer, 脚本会被延迟到整个页面都解析完毕后再运行，只对外部脚本文件有效

异步执行脚本： async, 也只适用于外部脚本, 并不能保证能按照他们出现的次序执行.

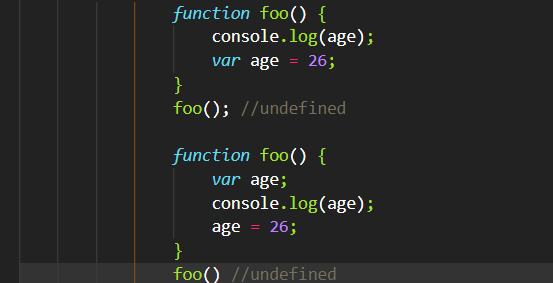
尽可能将JS放在外部文件中：可维护性，缓存，适应未来

<nonscript>用来给不支持JS或禁用JS的浏览器提供替代内容

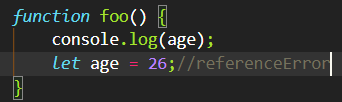
ES6中，标识符尽量使用驼峰大小写形式

变量: var let const

Var声明的变量作用域是在函数内部，let声明的范围是块作用域，块作用域是函数作用域的子集。Var会把所有变量声明都拉到函数作用域的顶部。下面两个function是等价的。



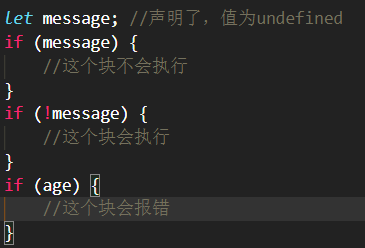
Let不会提升。



Const和let基本相同，唯一的区别是声明变量的同时必须初始化变量，且尝试修改const声明的变量时会导致运行错误。

大多数开发者不再需要var，const优先，let次之

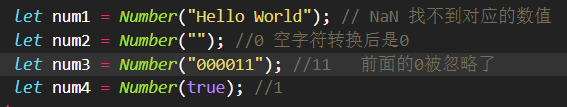
ES6有6种简单数据类型：Undefined, Null,Boolean, Number, String, Symbol, 复杂数据类型：Object.

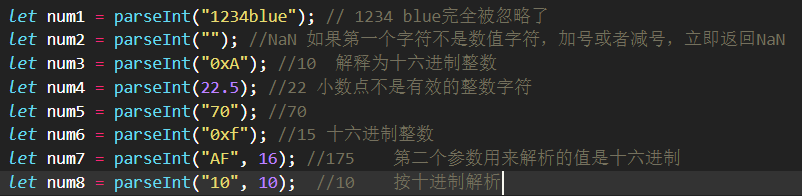


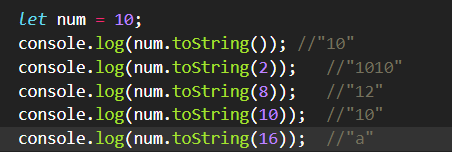
可以用Boolean()转型函数将其他类型的值转换为布尔值

let message = “Hello world”;

Let messageAsBoolean = Boolean(message);







Symbol:

Let fooSysbol = Symbol(‘foo’);

Console.log(fooSymbol); //Symbol(foo)

Symbol()函数不能和new关键字一起使用,避免创建符号包装对象

Let mySymbol = new Symbol(); //TypeError: Symbol is not a constructor

可以借用Object()函数

Let mySymbol = Symbol();

Let myWrappedSymbol = Object(mySymbol);