基本概念

1. 严格模式：在整个脚本的最上方添加 “use strict”
2. 可以在函数内部定义全局变量，不加修饰符，如 a = 1; 不推荐
3. 数据类型：：Undefined、Null、Boolean、Number、String和object
4. undefined 值是派生自 null 值的，因此 ECMA-262规定对它们的相等性测试要返回 true
5. 只要意在保存对象的变量还没有真正保存对象，就应该明确地让该变量保存 null 值。
6. 字符串转boolean：var messageAsBoolean = Boolean(message);空为false
7. NAN:译为不是数值，判断是否为NAN使用isNAN(), 只有 0除以0才会返回 NaN，正数除以 0返回 Infinity，负数除以 0返回-Infinity
8. ===和==，===数据类型和数值必须一样，==会转换成相同类型，推荐使用全等===
9. ECMAScript不存在块级作用域
10. for-in 可以用于迭代对象属性
11. js可以给代码添加标签如

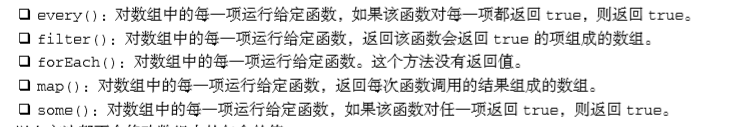
start: for (var i=0; i < count; i++) {

alert(i);

}

通常配合break和continue使用

1. js参数本质是一个数组，在函数内可以通过 arguments 关键字进行访问，如arguments[1], arguments[2],.., arguments.length获取数组长度
2. 判断是否为数组 Array.isArray(value)
3. 输出数组实际上输出的是数组中的每个toString()方法,可以这样 arr.toLocalString()
4. 可以用join()来替换，进行输出 ，如 arr.join(“||”)
5. 可以对数组进行排序，如 arr.sort([compose]),compose可有可无，书写规则跟java一样。
6. splice()方法，可以用于增删，splice()第一个参数代表位置，第二个代表删除个数，第三个以后代表插入数量‘
7. 迭代方法



reduce() 能把数组中的各个值相加到一起

19．arguments.callee()这个等价于 函数名(),使用场景，在递归调用中消除函数名和函数体的耦合

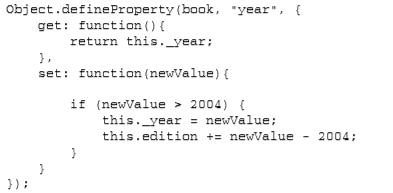
20. toFixed()方法会按照指定的小数位返回数值的字符串表示,可以有参数，表示保留几位小数

21. eval()方法会调用解析，李米娜的字符串跟正常的js代码一样，在严格模式下失效。

22. Object.defineProperty(person, "name", { writable: false, value: "Nicholas" });

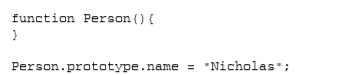
可以修改对象的属性，如某个字段是否可读等等，表示能否通过 delete 删除属性从而重新定义属性；Enumerable表示能否通过 for-in 循环返回属性；Writable表示能否修改属性的值。

 [[Get]]：在读取属性时调用的函数。默认值为 undefined。  [[Set]]：在写入属性时调用的函数。默认值为 undefined。



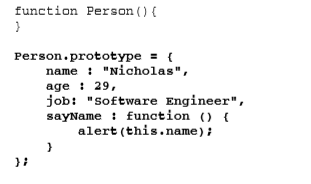
23. 工厂模式，要领：在工厂函数中new 一个对象，然后返回new这个对象

24．原型模式：跟java中的static关键字很像，所有对象共享，不过函数内部的属性要优于通过 函数名.prototype.属性名 设置的优先级。



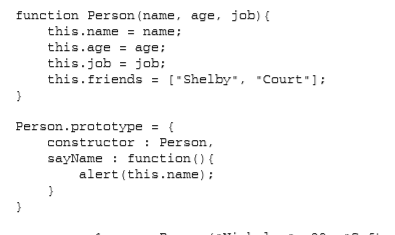
25. person1.hasOwnProperty("name"));判断是否拥有原型属性

26. 更简单的原型语法,这种方法会改变原型链的指向为自己,可以添加constructor: Person

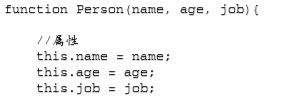


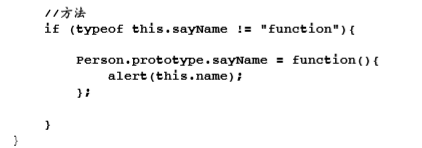
27. constructor: Person, 可以设置指向的原型，默认为原始原型，可以设置为自定义函数，如this.调用的时候

28. 组合使用构造函数模式和原型模式：



29. 动态原型模式





30. 稳妥构造函数模式，就是方法内不定义任何成员属性，方法实现类似于工厂模式，然后通过类似与get方法返回。

31. 原型链：主要作为实现继承的主要方法

如何修改原型链：SubType.prototype = new SuperType();

存在问题：任意继承SuperType()的对象修改类父方法属性，都会改变

1. 闭包注意事项

this 和 arguments。内部函 数在搜索这两个变量时，只会搜索到其活动对象为止，因此永远不可能直接访问外部函数中的这两个变量

1. 超时调用

设置超时调用 var timeoutId = setTimeout(function() {

alert("Hello world!"); }, 1000);

注意：把它取消 clearTimeout(timeoutId);

1. 循环调用