**js高级day02**

**js的预编译**

预编译分为全局预编译和局部预编译，全局预编译发生在页面加载完成时执行，而局部预编译发生在函数执行的前一刻。

**js运行三步曲**

1.语法分析

2.预编译

3.解释执行

**全局预编译的3个步骤：**

创建GO对象（Global Object）全局对象。

找变量声明，将变量名作为GO属性名，值为undefined

查找函数声明，作为GO属性，值赋予函数体

**局部预编译的4个步骤：**

创建AO对象（Activation Object）执行期上下文。

找形参和变量声明，将变量和形参名作为AO属性名，值为undefined

将实参值和形参统一。

在函数体里面找函数声明，值赋予函数体。

**call()和apply() bind()**

可以改变this的指向

区别：体现在传递参数上面

call（）里面的参数必须用，隔开

apply() 里面有参数的话，参数应该放到[]

求数组中的最大值

Math.max(1,2,3,55,89,520)

apply,call,bind的共同点和区别分别是什么？

共同点：都可以改变this的指向

区别：apply参数放到[] call参数用，隔开

bind不能执行这个函数

apply,call,bind底层代码？手写题。

**新的函数的调用方法**

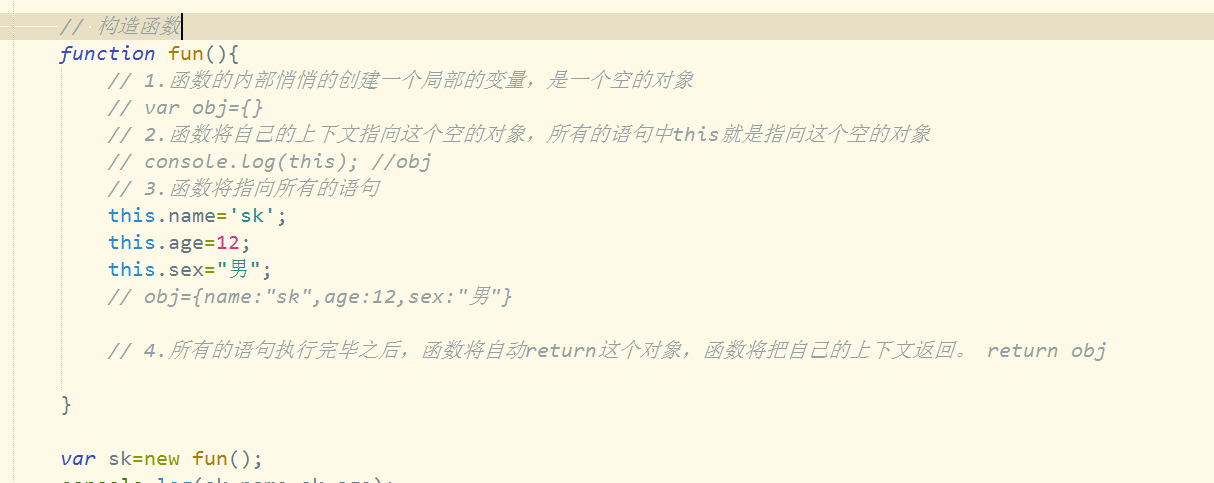
通过new的方式也可以调用函数

new可以调用函数

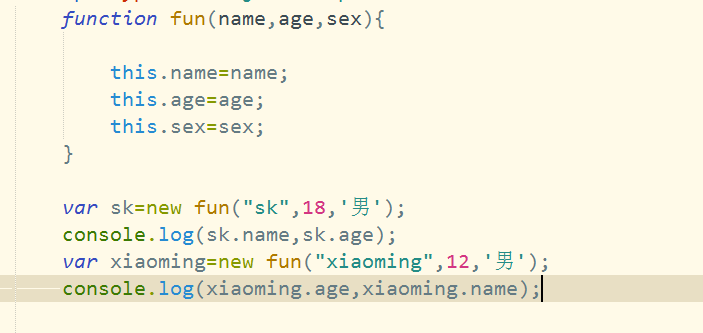
用new运算符调用一个函数的时候，会经历四个步骤

1. 函数的内部悄悄的创建一个局部的变量，是一个空的对象
2. 函数将自己的上下文指向这个空的对象，所有的语句中this就是指向这个空的对象
3. 函数将指向所有的语句
4. 所有的语句执行完毕之后，函数将自动return这个对象，函数将把自己的上下文返回。

所以，你只要看到new就一定要想到上述的四个步骤。



到目前为止，new可以调用函数，并且还可以返回一个对象，其实我们还可以加工一下



通过这个案例，我们知道sk 小明是不是拥有相同的属性

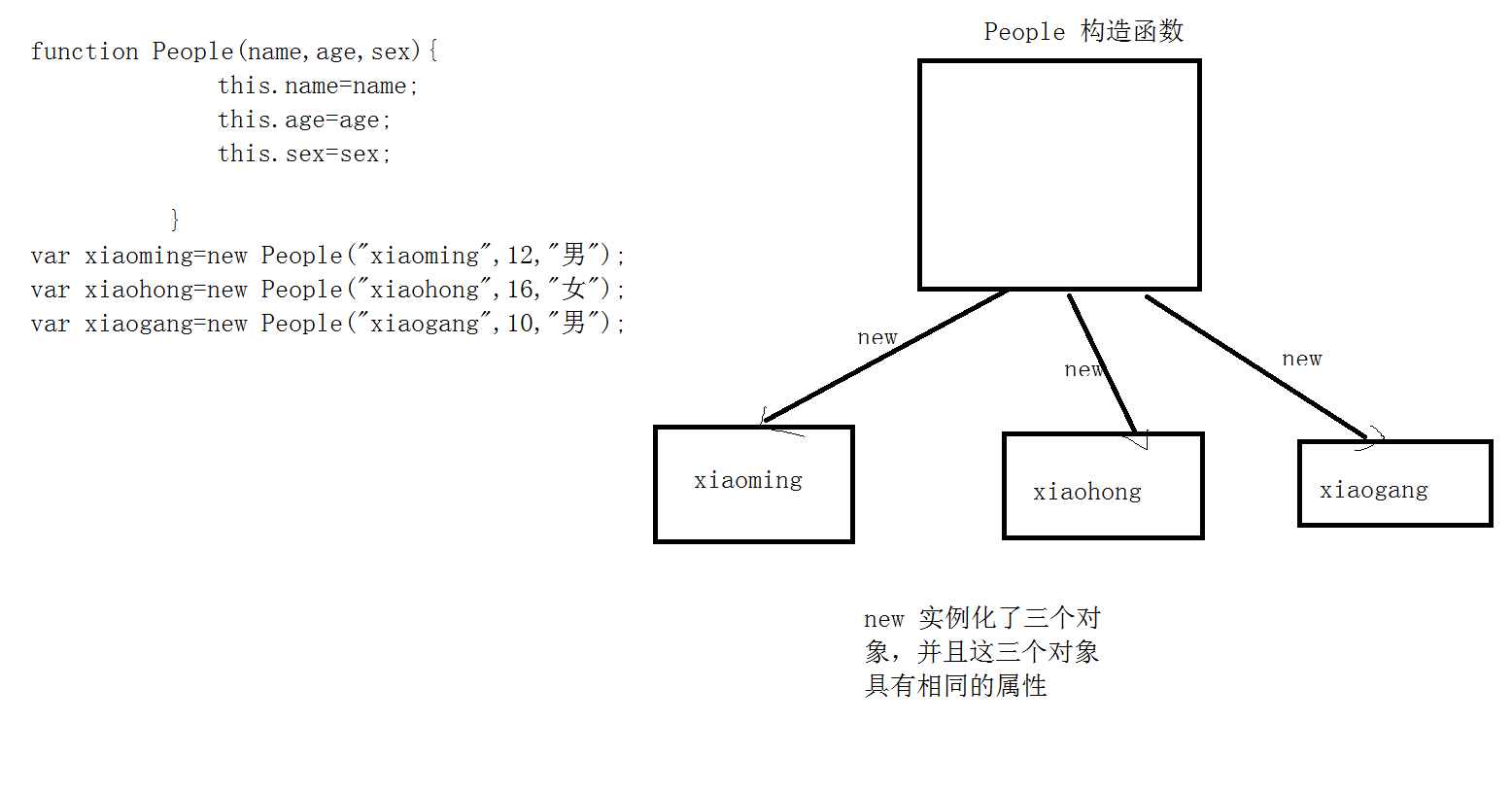
用new可以返回具有相同属性的对象，fun可以认为就是一个类，

sk 和小明 就是这个类的实例。

es5没有类的概念，只有构造函数。

当一个函数被new调用的时候，这个就是一个构造函数，它总能返回具有相同属性群的对象，所以这个函数很神奇，像一个模具一样，总在制造各种类似的产品。

构造函数首字母一般大写，区分普通函数



构造函数和其他函数并没有其他的区别，普通函数也可以通过new的方式来调用。

构造函数就是一个普通的函数，里面可以写任意的语句，只不过this指向的是一个空的对象。