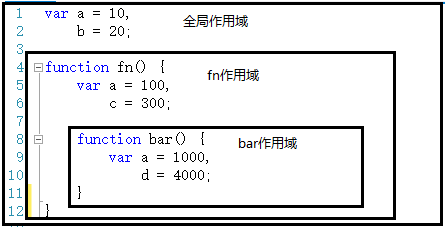
JavaScript的作用域

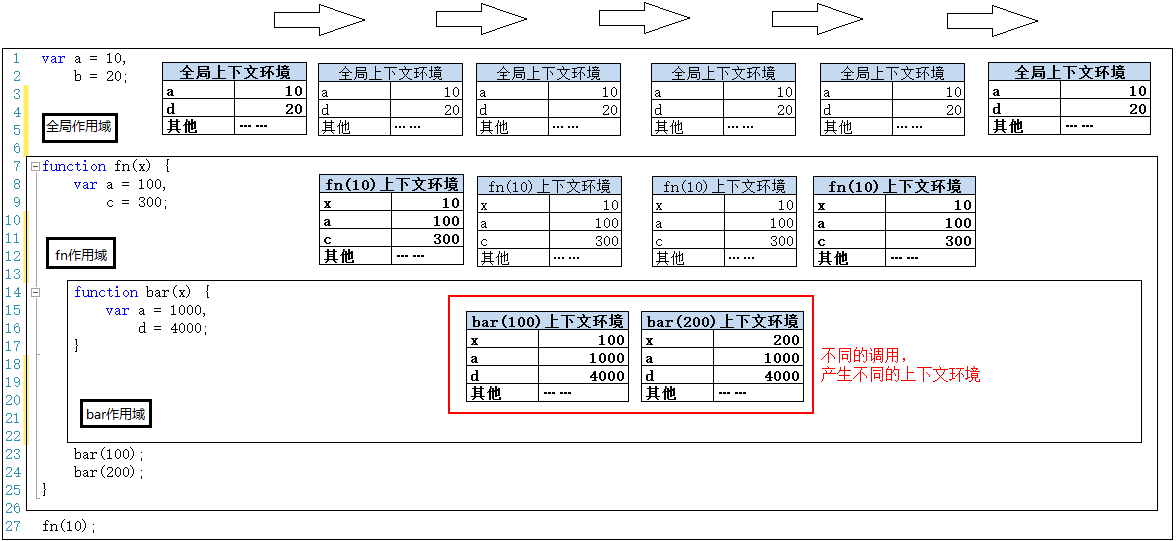
1、JS没有块级作用域，即在for、if、while等控制语句中不能声明变量。所谓“块”，就是大括号“{}”中间的语句。

2、JS除了全局作用域之外，只有函数中可以创建作用域。声明变量时，全局代码要在代码前端声明，函数中要在函数体一开始就声明好。建议用“var”形式。

作用域的用处：隔离变量，不同作用域下同名变量不会有冲突。



作用域有上下级的关系，上下级关系的确定就看函数是在哪个作用域下创建。



静态作用域：取变量的值要到**创建**这个函数的那个作用域中取。即在创建这个函数的时候，这个函数的作用域就已经决定了，而不是在调用它的时候。所以在调用函数时要使用的变量要去创建函数的作用域中查找

作用域是针对变量的，例：创建函数f1，在f1里创建函数f2，在f2里创建函数f3

全局作用域 f1作用域 f2作用域 f3作用域

查找所需用到的变量先是在f3中查找；找不到，向上到f2中查找；找不到，向上到f1中查找；找不到，向上到全局中查找。

跨作用域去寻找变量的过程所形成的线路，称为：作用域链

自由变量：凡是跨了自己的作用域的变量

取自由变量时的过程：

一、先在当前作用域查找a，有获取并结束，没有继续查找；

二、如果当前是全局作用域，则证明a未定义，结束

三、如果当前不是全局，那就是函数作用域，将创建该函数的作用域作为当前作用域；

四、跳转第一步

var x = 10; //声明全局变量

function fn(){ //x是自由变量

console.log(x); //函数没有声明x,转到父作用域--全局中查找

}

function show(f){

var x = 20;

(function(){ //匿名函数

f(); //调用参数fn函数，变量x由创建fn()函数的作用域决定，因为fn函数的作用域在创建的时候已决定了

})();

}

show(fn); //运行函数，参数为函数

