|  |
| --- |
| function Foo() {      var i = 0;  **return** function() {          console.log(i++);      }  }    var f1 = Foo(),      f2 = Foo();  f1();0  f1();1  f2();0 |

1. 闭包例题

（1）function是引用类型：保存在堆中，变量f1,f2是保存在栈中；

（2）闭包：一般来说**函数执行完后它的局部变量就会随着函数调用结束被销毁**，但是此题Foo函数返回了一个匿名函数的引用（即一个闭包），它可以访问到Foo()被调用产生的环境，而局部变量i一直处在这个环境中，只要一个环境有可能被访问到，它就不会被销毁，所以说**闭包有延续变量作用域的功能**。这就好理解为什么：f1();//0 f1();//1

(3)   过程：