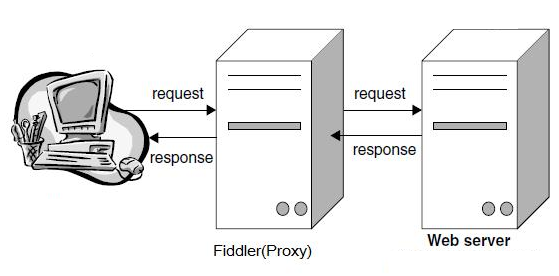
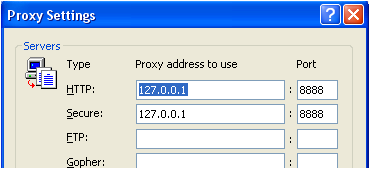
**Fiddler的工作原理**

Fiddler 是以代理web服务器的形式工作的,它使用代理地址:127.0.0.1, 端口:8888. 当Fiddler会自动设置代理， 退出的时候它会自动注销代理，这样就不会影响别的程序。不过如果Fiddler非正常退出，这时候因为Fiddler没有自动注销，会造成网页无法访问。解决的办法是重新启动下Fiddler.





**同类的其它工具**

同类的工具有: httpwatch, firebug, wireshark

**Fiddler 如何捕获Firefox的会话**

能支持HTTP代理的任意程序的数据包都能被Fiddler嗅探到，Fiddler的运行机制其实就是本机上监听8888端口的HTTP代理。 Fiddler2启动的时候默认IE的代理设为了127.0.0.1:8888，而其他浏览器是需要手动设置的，所以将Firefox的代理改为127.0.0.1:8888就可以监听数据了。

Firefox 上通过如下步骤设置代理

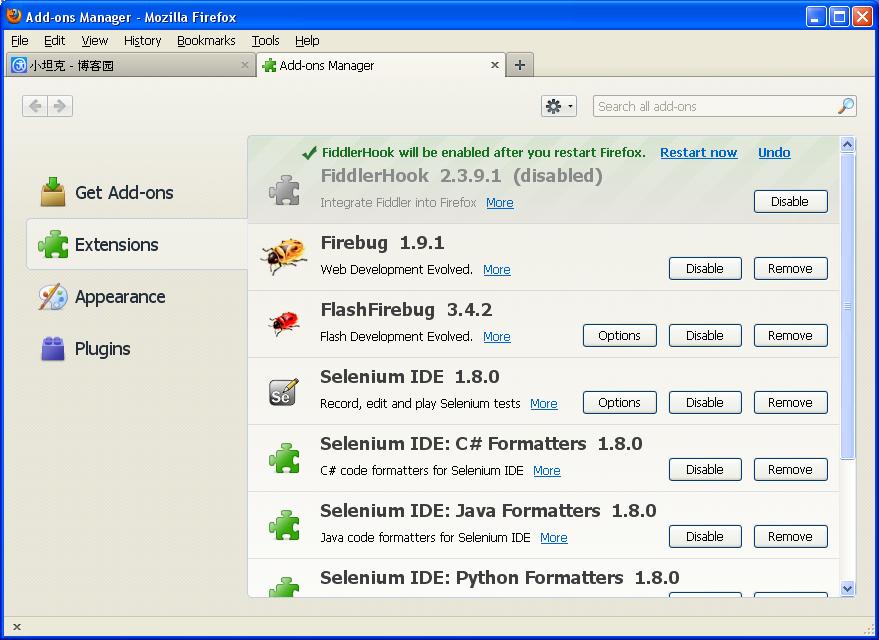
点击: Tools -> Options,  在Options 对话框上点击Advanced tab - > network tab -> setting.

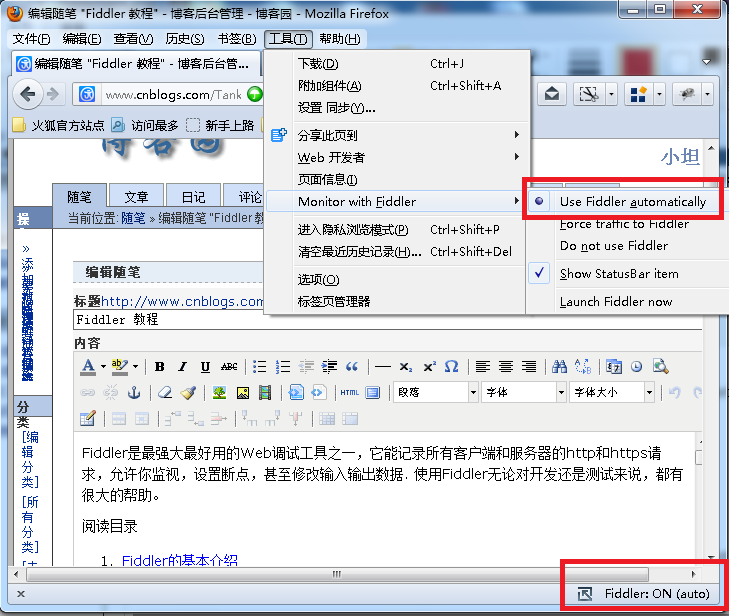
**Firefox 中安装Fiddler插件**

修改Firefox 中的代理比较麻烦， 不用fiddler的时候还要去掉代理。 麻烦

推荐你在firefox中使用fiddler hook 插件, 这样你非常方便的使用Fiddler获取firefox中的request 和response

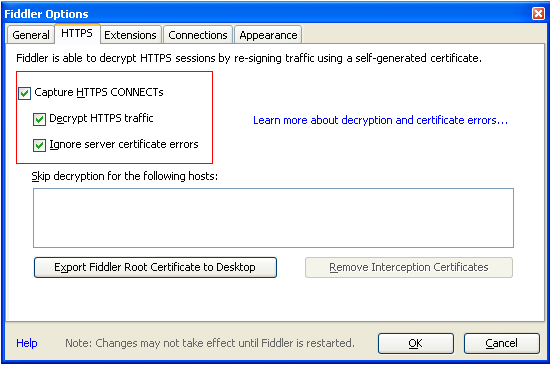
当你安装fiddler后， 就已经装好了Fiddler hook插件， 你需要到firefox中去启用这个插件  
打开firefox   tools->Add ons -> Extensions 启动 FiddlerHook



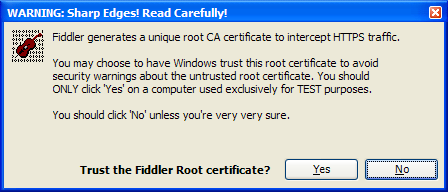


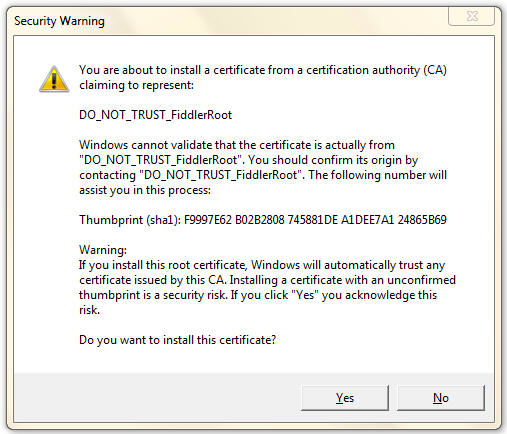
**Fiddler如何捕获HTTPS会话**

默认下，Fiddler不会捕获HTTPS会话，需要你设置下, 打开Fiddler  Tool->Fiddler Options->HTTPS tab



 选中checkbox, 弹出如下的对话框，点击"YES"

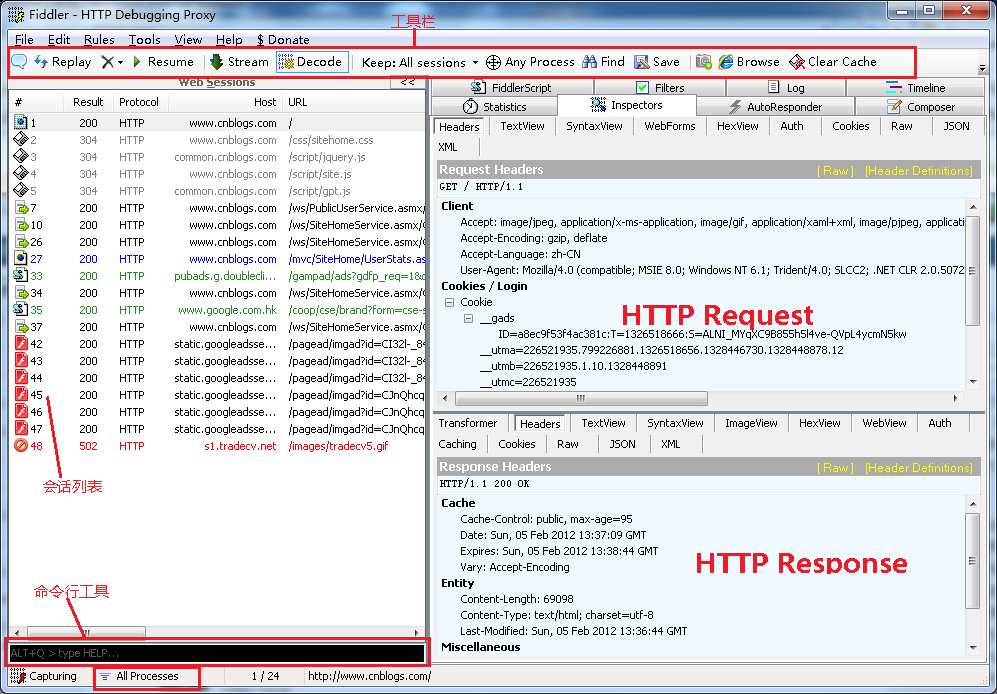




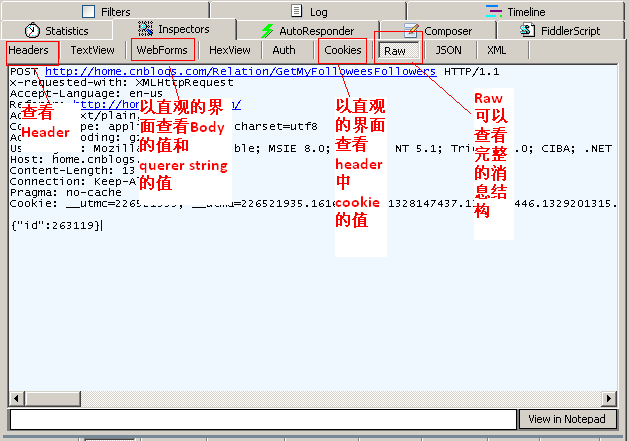
点击"Yes" 后，就设置好了

**Fiddler的基本界面**

 看看Fiddler的基本界面



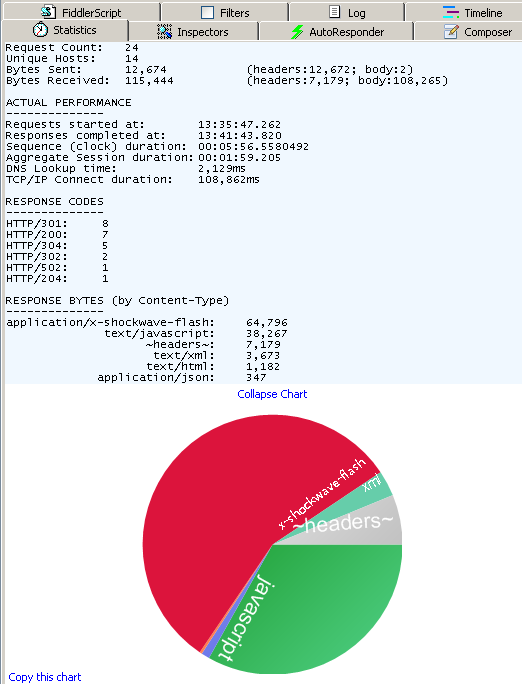
Inspectors tab下有很多查看Request或者Response的消息。 其中Raw Tab可以查看完整的消息，Headers tab 只查看消息中的header. 如下图



**Fiddler的HTTP统计视图**

通过陈列出所有的HTTP通信量，Fiddler可以很容易的向您展示哪些文件生成了您当前请求的页面。使用Statistics页签，用户可以通过选择多个会话来得来这几个会话的总的信息统计，比如多个请求和传输的字节数。

选择第一个请求和最后一个请求，可获得整个页面加载所消耗的总体时间。从条形图表中还可以分别出哪些请求耗时最多，从而对页面的访问进行访问速度优化



**QuickExec命令行的使用**

Fiddler的左下角有一个命令行工具叫做QuickExec,允许你直接输入命令。

常见得命令有

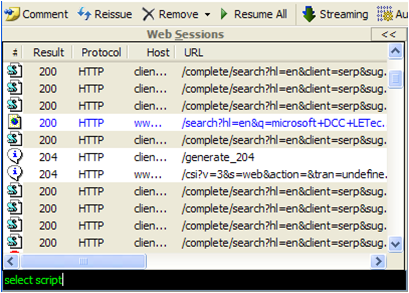
help  打开官方的使用页面介绍，所有的命令都会列出来

cls    清屏  (Ctrl+x 也可以清屏)

select  选择会话的命令

?.png  用来选择png后缀的图片

bpu  截获request



**Fiddler中设置断点修改Request**

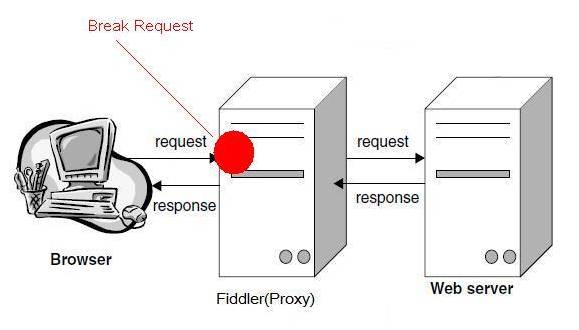
[[作者：小坦克]](http://www.cnblogs.com/TankXiao/admin/%5b%E4%BD%9C%E8%80%85%EF%BC%9A%E5%B0%8F%E5%9D%A6%E5%85%8B%5d%20http:/www.cnblogs.com/TankXiao/)  Fiddler最强大的功能莫过于设置断点了，设置好断点后，你可以修改httpRequest 的任何信息包括host, cookie或者表单中的数据。设置断点有两种方法

第一种：打开Fiddler 点击Rules-> Automatic Breakpoint  ->Before Requests(这种方法会中断所有的会话)

如何消除命令呢？  点击Rules-> Automatic Breakpoint  ->Disabled

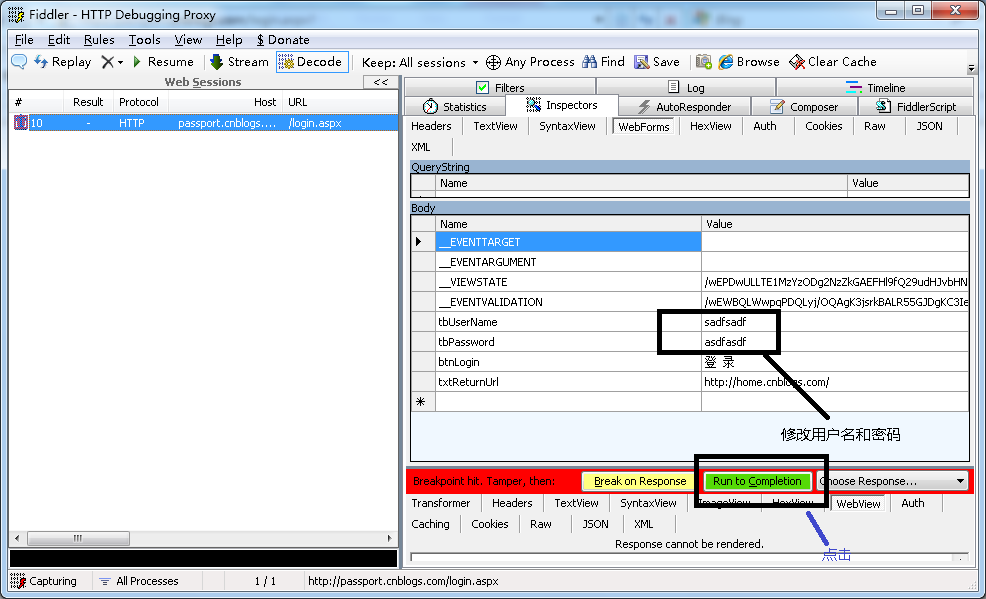
第二种:  在命令行中输入命令:  bpu www.baidu.com   (这种方法只会中断www.baidu.com)

如何消除命令呢？  在命令行中输入命令 bpu



看个实例，模拟博客园的登录， 在IE中打开博客园的登录页面，输入错误的用户名和密码，用Fiddler中断会话，修改成正确的用户名密码。这样就能成功登录

1. 用IE 打开博客园的登录界面  http://passport.cnblogs.com/login.aspx  
2. 打开Fiddler,  在命令行中输入bpu http://passport.cnblogs.com/login.aspx  
3. 输入错误的用户名和密码 点击登录  
4. Fiddler 能中断这次会话，选择被中断的会话，点击Inspectors tab下的WebForms tab 修改用户名密码，然后点击Run to Completion 如下图所示。  
5. 结果是正确地登录了博客园



**Fiddler中设置断点修改Response**

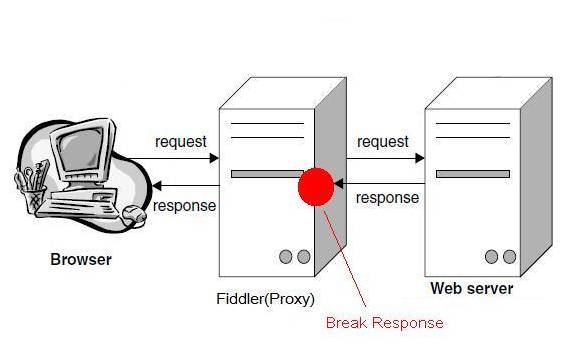
当然Fiddler中也能修改Response

第一种：打开Fiddler 点击Rules-> Automatic Breakpoint  ->After Response  (这种方法会中断所有的会话)

如何消除命令呢？  点击Rules-> Automatic Breakpoint  ->Disabled

第二种:  在命令行中输入命令:  bpafter www.baidu.com   (这种方法只会中断www.baidu.com)

如何消除命令呢？  在命令行中输入命令 bpafter,



具体用法和上节差不多，就不多说了。

**Fiddler中创建AutoResponder规则**

Fiddler 的AutoResponder tab允许你从本地返回文件，而不用将http request 发送到服务器上。

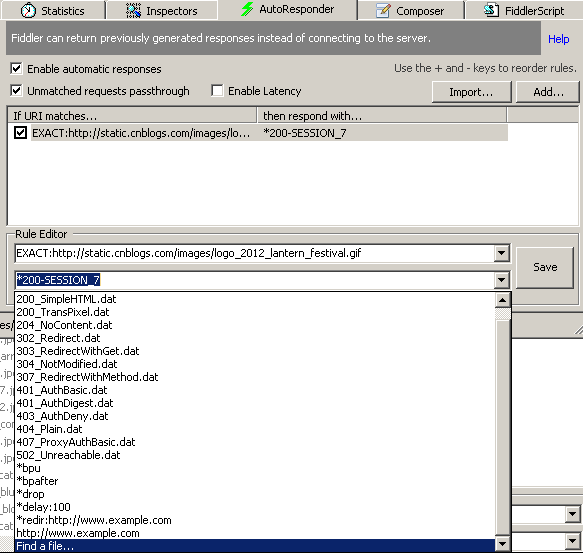
看个实例. 1. 打开博客园首页，把博客园的logo图片保存到本地，并且对图片做些修改。

2. 打开Fiddler 找到logo图片的会话， http://static.cnblogs.com/images/logo\_2012\_lantern\_festival.gif，  把这个会话拖到AutoResponer Tab下

3. 选择Enable automatic reaponses 和Unmatched requests passthrough

4. 在下面的Rule Editor 下面选择 Find a file... 选择本地保存的图片.  最后点击Save 保存下。

5.  再用IE博客园首页, 你会看到首页的图片用的是本地的。



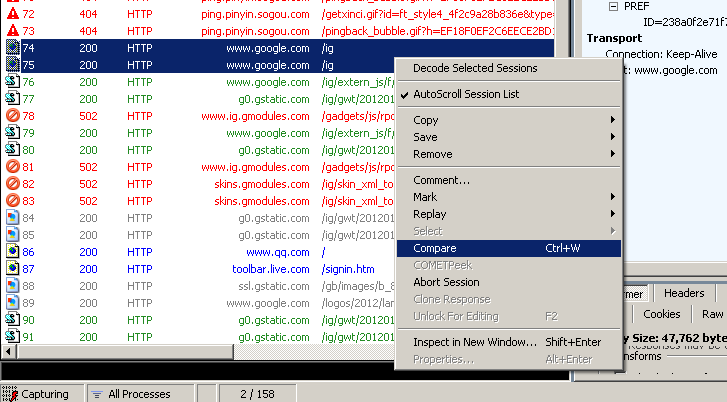


**Fiddler中如何过滤会话**

每次使用Fiddler, 打开一个网站，都能在Fiddler中看到几十个会话，看得眼花缭乱。最好的办法是过滤掉一些会话，比如过滤掉图片的会话. Fiddler中有过滤的功能, 在右边的Filters tab中，里面有很多选项, 稍微研究下，就知道怎么用。

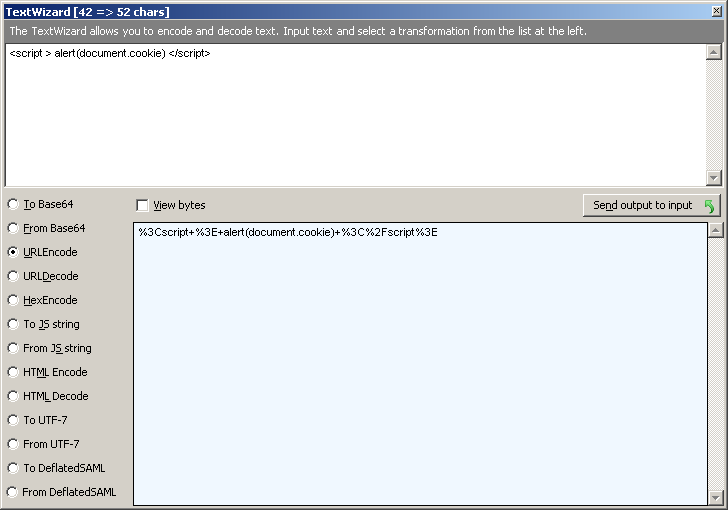
**Fiddler中会话比较功能**

选中2个会话，右键然后点击Compare，就可以用WinDiff来比较两个会话的不同了 (当然需要你安装WinDiff)



**Fiddler中提供的编码小工具**

点击Fiddler 工具栏上的TextWizard,  这个工具可以Encode和Decode string.



**Fiddler中查询会话**

用快捷键Ctrl+F 打开 Find Sessions的对话框，输入关键字查询你要的会话。 查询到的会话会用黄色显示



**Fiddler中保存会话**

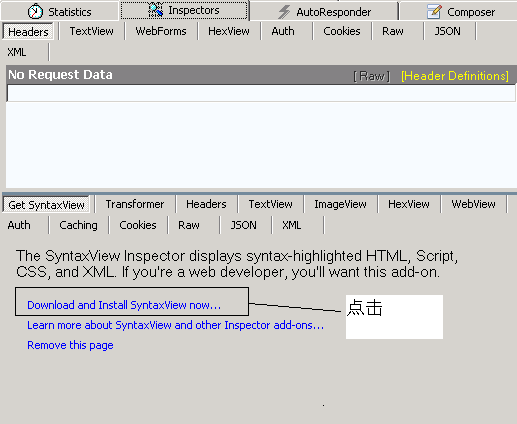
有些时候我们需要把会话保存下来，以便发给别人或者以后去分析。  保存会话的步骤如下：

选择你想保存的会话，然后点击File->Save->Selected Sessions

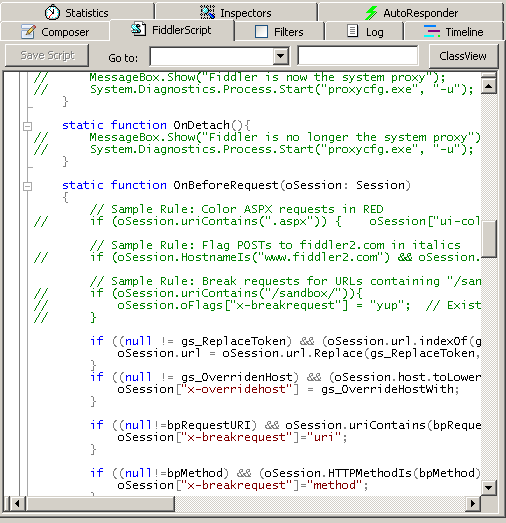
**Fiddler的script系统**

Fiddler最复杂的莫过于script系统了 官方的帮助文档: <http://www.fiddler2.com/Fiddler/dev/ScriptSamples.asp>

首先先安装SyntaxView插件，Inspectors tab->Get SyntaxView tab->Download and Install SyntaxView Now... 如下图



安装成功后Fiddler 就会多了一个Fiddler Script tab, 如下图



在里面我们就可以编写脚本了， 看个实例 让所有cnblogs的会话都显示红色。

把这段脚本放在OnBeforeRequest(oSession: Session) 方法下，并且点击"Save script"

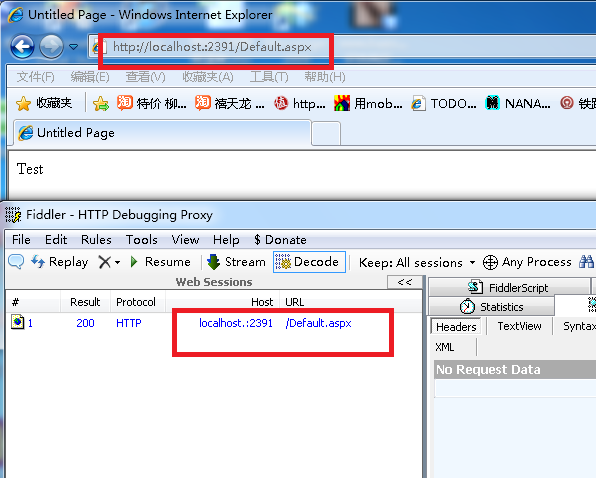
if (oSession.HostnameIs("www.cnblogs.com")) {  
 oSession["ui-color"] = "red";  
 }

这样所有的cnblogs的会话都会显示红色

**如何在VS调试网站的时候使用Fiddler**

我们在用visual stuido 开发ASP.NET网站的时候也需要用Fiddler来分析HTTP， 默认的时候Fiddler是不能嗅探到localhost的网站。不过只要在localhost后面加个点号，Fiddler就能嗅探到。

例如：原本ASP.NET的地址是 http://localhost:2391/Default.aspx，  加个点号后，变成 http://localhost.:2391/Default.aspx 就可以了



第二个办法就是在hosts文件中加入  127.0.0.1  localsite

如何你访问http://localsite:端口号   。  这样Fiddler也能截取到了。

**Response 是乱码的**

有时候我们看到Response中的HTML是乱码的， 这是因为HTML被压缩了， 我们可以通过两种方法去解压缩。

1. 点击Response Raw上方的"Response is encoded any may need to be decoded before inspection. click here to transform"

2. 选中工具栏中的"Decode"。  这样会自动解压缩。

