

socket编程---实现get post 向http发送请求

原创2014年11月01日 21:31:36

标签：socket / http协议 / GET / POST

4555

一、HTTP

Http定义了与服务器交互的不同方法，最基本的方法有4种，分别是GET，POST，PUT，DELETE。URL全称是资源描述符，我们可以这样认为：URL地址，它用于描述一个网络上的资源，而HTTP中的GET，POST，PUT，DELETE就对应着资源的查，改，增，删4个操作。到这里，大家应该有个大概的了解了，GET一般用于获取/查询资源信息，而POST一般用于更新资源信息。

1.根据HTTP规范，GET用于信息获取，而且应该是安全的和幂等的。

2.根据HTTP规范，POST表示可能修改服务器上的资源的请求。

二、GET

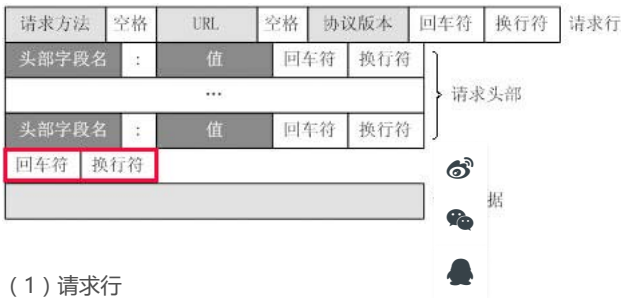
GET：当客户端要从服务器中读取文档时，使用GET方法。
GET方法要求服务器将URL定位的资源放在响应报文的数据部分，回送给客户端。
使用GET方法时，请求参数和对应的值附加在URL后面，利用一个问号（“?”）代表URL的结尾与请求参数的开始，传递参数长度受限制。
例如，/index.jsp?id=100&op=bind。

三、POST

POST：当客户端给服务器提供信息较多时可以使用POST方法。
POST方法将请求参数封装在HTTP请求数据中，以名称/值的形式出现，可以传输大量数据，可用来传送文件。

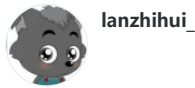
四、HTTP请求

HTTP报文是面向文本的，报文中的每一个字段都是一些ASCII码串，各个字段的长度是不确定的。
HTTP有两类报文：请求报文和响应报文。
请求报文一个HTTP请求报文由请求行（request line）、请求头部（header）、空行和请求数据4个部分组成，下图给出了请求报文的一般格式。



（1）请求行
请求行由请求方法字段、URL字段和HTTP协议版本字段3个字段组成，它们用空格分隔。例如，GET /index.html HTTP/1.1。
HTTP协议的请求方法有GET、POST、HEAD、PUT、DELETE、OPTIONS、TRACE、CONNECT。这里介绍最常用的GET方法和POST方法。
GET：当客户端要从服务器中读取文档时，使用GET方法。GET方法要求服务器将URL定位的资源放在响应报文的数据部分，回送给客户端。使用GET方法时，请求参数和对应的值附加在URL后面，利用一个问号（“?”）代表URL的结尾与请求参数的开始，传递参数长度受限制。例如，/index.jsp?id=100&op=bind。
POST：当客户端给服务器提供信息较多时可以使用POST方法。POST方法将请求参数封装在HTTP请求数据中，以名称/值的形式出现，可以传输大量数据，可用来传送文件。

（2）请求头部
请求头部由关键字/值对组成，每行一对，关键字和值用英文冒号“:”分隔。请求头部通知服务器有关于客户端请求的信息，典型的请求头有：
User-Agent：产生请求的浏览器类型。



原创203

粉丝129

喜欢200

评论64

等级：博客5

访问量：36万+

积分：5170

排名：6453

他的最新文章 更多文章

- tkinter--计算器
- 位图法查找算法
- 位图法排序
- bitset应用---产生1万个不重复的随机数
- 指针的指针

文章分类

善用指针	9篇
Linux学习	30篇
字符串函数实现	12篇
笔试题	19篇
C++标准库	52篇
C++ 排序	10篇

博主专栏

- Linux-C编程

26528

29篇
- C++标准库

50103

41篇

文章存档

2018年1月	1篇
2015年12月	1篇
2015年11月	2篇
2015年10月	1篇
2015年5月	1篇
2015年4月	8篇

(3) 空行

最后一个请求头之后是一个空行，发送回车符和换行符，通知服务器以下不再有请求头。
对于一个完整的http请求来说空行是必须的，否则服务器会认为本次请求的数据尚未完全发送到服务器，处于等待状态。

(4) 请求数据

请求数据不在GET方法中使用，而是在POST方法中使用。POST方法适用于需要客户填写表单的场合。与请求数据相关的最常使用的请求头是Content-Type和Content-Length。

五、代码示例

post

```
[cpp]
1. //post
2. #include<stdio.h>
3. #include<winsock2.h>
4. #include<windows.h>
5. #pragma comment(lib, "ws2_32.lib")
6.
7. int main()
8. {
9.     int num;
10.
11.     SOCKET s;
12.     WSADATA wsa;
13.     struct sockaddr_in serv;
14.
15.     char sndBuf[1024], rcvBuf[2048];
16.
17.     WSASStartup(MAKEWORD(2, 1), &wsa);
18.
19.
20.     if ((s = socket(AF_INET, SOCK_STREAM, 0))<0)
21.     {
22.         perror("socket error!");
23.         exit(1);
24.     }
25.
26.     memset(&serv,0,sizeof(serv));
27.     serv.sin_family = AF_INET;
28.     serv.sin_port = htons(80);
29.     serv.sin_addr.S_un.S_addr = inet_addr("115.239.210.26");
30.
31.     if ((connect(s, (struct sockaddr *)&serv, sizeof(serv)))<0)
32.     {
33.         perror("connet error!");
34.         exit(1);
35.     }
36.
37.     memset(sndBuf, 0, 1024);
38.     memset(rcvBuf, 0, 2048);
39.
40.     //头信息
41.     strcat(sndBuf, "POST ");
42.     strcat(sndBuf, "http://www.baidu.com");
43.     strcat(sndBuf, " HTTP/1.0\r\n");
44.     strcat(sndBuf, "Host: ");
45.     strcat(sndBuf, "baidu.com");
46.     strcat(sndBuf, "\r\n");
47.     strcat(sndBuf, "Connection: keep-alive\r\n");
48.
49.     strcat(sndBuf, "\r\n");
50.
51.     puts(sndBuf);
52.
53.     if ((num = send(s,sndBuf,1024, 0))<0)
54.     {
55.         perror("send error!");
56.         exit(1);
57.     }
58.
59.     puts("send success!\n");
60.
61.     do{
62.
63.         if ((num = recv(s, rcvBuf, 2048, 0))<0)
64.         {
65.             perror("recv error!");
66.             system("pause");
```

字符串函数---atoi()函数详解及实现 (完整版)

36353

字符串函数---itoa()函数详解及实现

27202

socket编程---send函数&recv函数详解

19874

字符串函数---atof()函数详解及实现 (完整版)

17207

华为研发面试总结

13348

C语言字节对齐、结构体对齐最详细的解释

11207

字符串函数---strcat()与strncat详解及实现

9135

字符串函数---strcpy()与strncpy()详解及实现

6967

C++实现to_string函数--int to string

5247

C++实现统计字符串中各字符的个数

4857

联系我们



请扫描二维码联系客服
webmaster@csdn.net
400-660-0108
QQ客服 客服论坛

关于 招聘 广告服务 百度
©1999-2018 CSDN版权所有
京ICP证09002463号

经营性网站备案信息
网络110报警服务
中国互联网举报中心
北京互联网违法和不良信息举报中心

```

70.         {
71.             printf("%s", rcvBuf);
72.             memset(rcvBuf, 0, 2048);
73.         }
74.     } while (num>0);
75.
76.     puts("\nread success!\n");
77.
78.     closesocket(s);
79.
80.     system("pause");
81.     WSACleanup();
82.
83.     return 0;
84.
85. }

```



3



```

POST http://www.baidu.com HTTP/1.0
Host: baidu.com
Connection: keep-alive

send success!

HTTP/1.1 200 OK
Date: Sat, 01 Nov 2014 13:53:28 GMT
Server: Apache
P3P: CP=" OTI DSP COR IVA OUR IND COM "
P3P: CP=" OTI DSP COR IVA OUR IND COM "
Set-Cookie: BAIDUID=D64BB5D1F01B46EFFBE4CBA80DC658E2;FG=1; expires=Sun, 01-Nov-1
5 13:53:28 GMT; max-age=31536000; path=/; domain=.baidu.com; version=1
Set-Cookie: BAIDUID=D64BB5D1F01B46EFBD7F2CEC3C2D05665;FG=1; expires=Sun, 01-Nov-1
5 13:53:28 GMT; max-age=31536000; path=/; domain=.baidu.com; version=1
Last-Modified: Tue, 31 Dec 2013 11:13:44 GMT
ETag: "1cdb-4eed2a7498a00"
Accept-Ranges: bytes
Content-Length: 7387
Cache-Control: max-age=1
Expires: Sat, 01 Nov 2014 13:53:29 GMT
Vary: Accept-Encoding, User-Agent
Connection: Close
Content-Type: text/html

```

http://blog.csdn.net/lanzhihui_10086

get

```

[cpp]
1. //http get
2. #include<stdio.h>
3. #include<winsock2.h>
4. #include<windows.h>
5. #pragma comment(lib, "ws2_32.lib")
6.
7. int main()
8. {
9.     int num;
10.
11.     SOCKET s;
12.     WSADATA wsa;
13.     struct sockaddr_in serv;
14.
15.     char sndBuf[1024], rcvBuf[2048];
16.
17.     WSStartup(MAKEWORD(2, 1), &wsa);
18.
19.
20.     if ((s = socket(AF_INET, SOCK_STREAM, 0))<0)
21.     {
22.         perror("socket error!");
23.         exit(1);
24.     }
25.
26.     memset(&serv, 0, sizeof(serv));
27.     serv.sin_family = AF_INET;
28.     serv.sin_port = htons(80);
29.     serv.sin_addr.S_un.S_addr = inet_addr("115.239.210.26");
30.
31.     //BING 70.37.92.127
32.     //BAIDU 115.239.210.26
33.     //baidu api 180.149.132.118
34.     //google 74.125.31.99
35.     //sina 59.175.132.113
36.     //163 222.243.110.164
37.     //taobao 218.75.155.151
38.     //jd 116.211.94.1

```

```

42.     perror("connet error!");
43.     exit(1);
44. }
45.
46. memset(sndBuf, 0, 1024);
47. memset(rcvBuf, 0, 2048);
48.
49. strcat(sndBuf, "GET /index.html HTTP/1.0\r\n");
50.
51. strcat(sndBuf, "\r\n");
52.
53. puts(sndBuf);
54.
55. if ((num = send(s, sndBuf, 1024, 0)) < 0)
56. {
57.     perror("send error!");
58.     exit(1);
59. }
60.
61. puts("send success!\n");
62.
63. do{
64.
65.     if ((num = recv(s, rcvBuf, 2048, 0)) < 0)
66.     {
67.         perror("recv error!");
68.         exit(1);
69.     }
70.     else if (num > 0)
71.     {
72.         printf("%s", rcvBuf);
73.         memset(rcvBuf, 0, 2048);
74.     }
75. } while (num > 0);
76.
77. puts("\nread success!\n");
78.
79. closesocket(s);
80.
81. system("pause");
82. WSACleanup();
83.
84. return 0;
85. }

```

```

GET /index.html HTTP/1.0

send success!

HTTP/1.1 200 OK
Date: Sat, 01 Nov 2014 13:54:31 GMT
Content-Type: text/html
Content-Length: 14613
Last-Modified: Wed, 03 Sep 2014 02:48:32 GMT
Connection: Close
Vary: Accept-Encoding
Set-Cookie: BAIDUID=CAA7EE958F22A2CECBE0596E9DEC1C98;FG=1; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
Set-Cookie: BAIDUPSID=CAA7EE958F22A2CECBE0596E9DEC1C98; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
Set-Cookie: BIDSVRTM=0; path=/
P3P: CP=" OTI DSP COR IVA OUR IND COM "
Server: BWS/1.1
Pragma: no-cache
Cache-control: no-cache
BDPAGETYPE: 1
BDQID: 0x8536465e003b068f
BDUSERID: 0
Accept-Ranges: bytes

```

目前您尚未登录，请 [登录](#) 或 [注册](#) 后进行评论

SOCKET模仿HTTP POST请求



ppt0501

2014年06月21日 17:11

 4281

http最常用的两种请求方法，get和post，现在用socket模仿post请求，

如何使用SOCKET 发送HTTP1.1 GET POST请求包



liujiaoyage

2016年09月12日 15:59

 7234

是面向文本的，报文中的每一个字段都是一些ASCII码串，各个字段的长度是不确定的。HTTP有两类报文：请求报文和响应报文。请...



yc0188

2009年10月29日 10:58

 31752





C 语言Socket 实现http 带参数的POST请求



liujiaoyage

2016年09月12日 15:59

 7234

C 语言Socket 实现带参数的 http post请求

如何使用SOCKET 发送HTTP1.1 GET POST请求包




tyshatang

2015年02月09日 01:27

 748


原文：如何使用SOCKET 发送HTTP1.1 GET POST请求包 <http://blog.csdn.net/yc0188/article/details/4741871> HTTP报...

如何利用socket进行HTTP访问




tanqiuwei

2014年04月09日 10:54

 24458


如何利用socket进行HTTP访问 <http://blog.csdn.net/wbczyh/article/details/2117094> 平常我们要访问某个URL一般都是通过浏览器...

httplib使用socket发送请求




YAOQINGGG

2016年03月27日 23:34

 1199


socket Socket的英文原义是“孔”或“插座”。作为BSD UNIX的进程机制，取后一种意思。通常也称作“套接字”，用于描述IP地址和端口，是一个通信链的句柄，可以用来实现不...

使用Socket发送Http请求




sddxqlrjxr

2016年05月05日 13:45

 3923


1、建立TCP连接。SocketAddress addr = new InetSocketAddress("www.baidu.com",80); Socket sock = new Socket...

socket发送请求



xuxu198899223

2015年03月03日 10:40

 364

/** */ package com.saicfc.pmpf.common.utils; import java.io.BufferedReader; import java.io.Byt...

Java Socket 实现HTTP与HTTPS协议发送POST/GET请求




zmx729618

2016年05月11日 15:00

 6714


JAVA Socket 实现HTTP与HTTPS客户端发送POST与GET方式请求 哇,一看标题怎么这么长啊,其实意思很简单,哥讨厌用HTTP Client做POST与GET提交觉...

使用c发送GET请求



x_i_y_u_e

2016年07月26日 15:08

 1002

参考地址：http://www.wangafu.net/~nickm/libevent-book/01_intro.html#_footnote_1 在阅读libevent文档时发现这个例子，文档列...

1、在B/S架构中，Web浏览器与Web服务器之间的一次HTTP请求与响应：需要完成以下步骤: http://localhost:8089/index.html 1) 、浏览器端根据所在的HTTP协议解...



用WinSock实现HTTP的GET

yihucha166 2007年10月06日 15:53 3330

以前用WinInet的类写过网络爬虫，所以也想自己试着用socket来写一个类似程序，看了很多的例程以及文章，自己用WinSock写了一个http下get功能的程序，比较基本体现了这类程序的...

winsock编程使用HTTP代理（一）

ciaos 2012年07月03日 19:24 1770

代理服务我们比较熟悉了，常见的代理方式有HTTP，sock4，sock5。到底浏览器是怎么通过代理服务器访问目标资源的呢，我这里使用最简单的http代理服务来写个程序测试一下。 首先用socket编...

使用socket post 数据到http和https

2012年11月12日 16:06 46KB [下载](#)



C++ Socket 发送 http post请求

isfirst 2016年07月20日 23:06 258

#include "stdafx.h" #include #include #include #include #include #include //dont forget to add wsock...

C++ 实现模拟登录---发送post表单信息包

交流学习！

lanzhihui_10086 2014年11月13日 22:29 2591

Socket实现发送HTTP请求

2014年06月01日 22:41 5KB [下载](#)



socket编程实现http GET请求

SP_daiyq 2014年12月04日 15:03 2807

// 如果host不是点分十进制格式，则将其转换成点分十进制格式 BOOL CTestDlg::GetRealIP(string& host, string & retip) { retip = ho...

一个简单的HTTP请求与应答基于socket

MOVIE14 2017年08月03日 18:47 452

tomcat是一个web容器，网络请求基于HTTP，HTTP底层基于socket抽象层，直接上图（盗的图） 我就简单实现了一个socket来模拟HTTP请求与应答 package Sock...

JAVA代码:通过Socket执行HTTP的POST方法

package net.java2000.net;import java.io.BufferedReader;import java.io.BufferedWriter;import java.io....

java2000_net 2008年08月31日 08:41 6314