# 网络

**http协议**

1.什么是Http协议

浏览器客户端和服务器端之间数据传输的格式规范,客户向服务器请求服务时，只需传送请求方法和路径。请求方法常用的有GET、HEAD、POST.

2.URI和URL的不同

URI： 统一资源标记符。/text/hello。用于标记任何资源。可以是本地文件系统，局域网的资源,可以是互联网。

URL: 统一资源定位符。如：http://localhost:8080/text/hello.html。只能是互联网资源。是URI的子集。

3. POST请求例子

第一部分：请求行， post请求，http协议版本号

第二部分：请求头， 浏览器类型，连接状态等

第三部分：空行，第七行空行

第四部分：请求数据

POST / HTTP1.1

Host:www.wrox.com

User-Agent:Mozilla/4.0

Connection: Keep-Alive

name=Professional%20Ajax&publisher=Wiley

4.响应Response

第一部分：状态行。 协议版本号， 状态码，状态消息

第二部分：消息头， 日期和时间, Type , 编码类型UTF-8

第三部分：空行

第四部分：响应正文，<html>

HTTP/1.1 200 OK

Date: Fri, 22 May 2009 06:07:21 GMT

Content-Type: text/html; charset=UTF-8

<html>

5.常见的状态码 状态描述

1xx：指示信息--表示请求已接收，继续处理

2xx：成功 200

3xx：重定向--要完成请求必须进行更进一步的操作 302

4xx：客户端错误 404

5xx：服务器端错误 500

6.GET和POST请求的区别

GET，数据会附在URL之后,地址栏中显示出来

POST, 数据放置在是HTTP包的包体中

GET有大小限制

post 比get更安全

**OSI 网络七层模型**

物理层 ：传输介质

->数据链路层 ：局域网MAC层

->网络层 ：外网IP层

->传输层 ：TCP UDP

->会话层 ：定义如何开始、控制和结束一个会话。

->表示层 ：定义数据格式及加密

->应用层 ：http，FTP，pop3邮件

**tcp的三次握手？**

第一次握手：客户端发送syn包到服务器。（同步请求）

第二次握手：服务器收到syn包，必须确认客户的SYN，同时自己也发送一个AcK包（回复SYN+ACK）（同步请求应答）。

第三次握手：客户端收到服务器的SYN＋ACK包，向服务器发送确认包ACK。

**TCP和UDP的区别?**

TCP是确认包的到达，所以不丢失包(数据)，顺序也是好的。

UDP不会确认包的到达，所以可能会丢失数据包，当然顺序也有可能会乱。

除了这些以外：

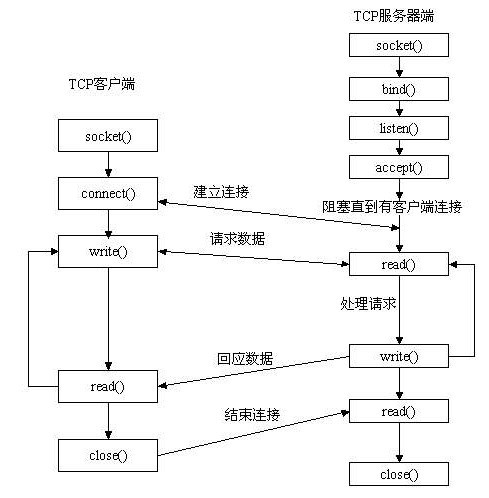
1. TCP是面向连接(三次握手)的，稳定、可靠，慢（文件传输，http）

2. UDP是面向无连接的，不稳定，快，实时性高（视频通话）

QQ就是UDP协议（快），MSN用的TCP（慢）

# Socket

网络上的两个程序通过一个双向的通信连接实现数据的交换，这个连接的一端称为一个socket。(ip 端口)



# 各种ip

1、127.0.0.1是回送地址，指本地机，一般用来测试使用。

2、localhost是本地DNS解析的127.0.0.1的域名，hosts文件最后一行：127.0.0.1 localhost

3、本机IP则指你连到网络上的IP地址，可以是内网地址，当然也可能是公网IP，这个就是你实际利用TCP/IP协议与网上计算机通信时使用的IP了。

4、0.0.0.0（指的是绑定所有本机所有ip）已经不是一个真正意义上的IP地址了。它表示的是这样一个集合：所有不清楚的主机和目的网络。 0.0.0.0才是真正表示“本网络中的本机”。

在实际应用中，一般我们在服务端绑定端口的时候可以选择绑定到0.0.0.0，这样我的服务访问方就可以通过我的多个ip地址访问我的服务。

比如我有一台服务器，一个外放地址A,一个内网地址B，如果我绑定的端口指定了0.0.0.0，那么通过内网地址或外网地址都可以访问我的应用。但是如果我只绑定了内网地址，那么通过外网地址就不能访问。

# 针对性翻墙方法

https://www.ipaddress.com

输入需要的网址 github.com

找到ttl最小的

github.com A 192.30.253.112 60

hosts:

192.30.253.112 github.com