```
需求:客户端或者浏览器等
                                 解决:为了将数据准确的送达给对
           与 服务器 之间进行数据传输
                                 方,我们需要给通信双方规定一个协
           需要, 而对服务器来说,请求方
                                 议,只有遵循这个协议的双方,才能
           不是单一客户端,而是多端.
                                 正常通信(正常情况下)
                                 超文本:
                                          传输的数据,图文,视频,音频 等等
                                                          请求行: 地址,端口等
                                                          请求头: 亲请求相关设置,如是
                                        前端向服务器请求数
                                                          否缓存等等
                                 请求:
                                        据(后面详细解释)
                                                          请求体: 请求数据
                                                          换行:
           http为什么叫超文本传输协议?
                                                          响应头:
                                        服务器返回请求结果或
                                 响应:
                                                          响应体:
                                        者返回数据
                                                          响应
                                        数据传输判断,身份验证等等,只
                                 协议:
                                        有双方遵循这个规则,才能完成
                                        请求和响应
                        浏览器输入 http://132.232.16.94 网址,回车后返回了
                        一个网页,就完成了(至少)一次请求和响应
           完成一次请求:
                        注意:协议是根据ip地址才能对应请求是对应哪一个服务器的
                        而现实中常常是这样的 https://www.baidu.com,也就是域名网址(ip
                        地址不利于记忆), 因此多了一层 ip地址 和 域名 之间的绑定关系
                                         1. 客户端: http://132.232.16.94
                               ip地址请求:
                                         2. 服务器: 根据ip地址请求到服务器,进行进一步数据响应
                                        1. 客户端: http://niexiaobo.com
           http是如何请求数据的呢?
                                        2. DNS域名解析服务器(不对外可见): 顾名思义,就
                               域名请求:
                                        是将域名对应的ip地址找到,并返回
                                        3. 服务器: 根据ip地址请求到服务器,进行进一步数据响应
                            1.协议先向对应的DNS域名解析服务器请求对应的ip地
                            址,如果找到了,就根据返回的ip地址后,先是存储其对应
                            关系,然后再请求公司的服务器
                            2.如果对应的DNS域名解析服务器没有找到呢? 那么就会向
                            该DNS服务器的上一级服务器请求,一直找下去,直到找到
                            3.根服务器: 所有DNS域名解析服务器最终都是指向root
                            根服务器的, 它包含全世界的ip地址与域名的对应关系
                                   1. 为什么不直接向根服务器请求ip地址?
                                   答: 全世界都向根服务器请求,响应会很慢,甚至超时. 设置
                                   不同的节点服务器来处理最近的请求,并进行一个存储
                                   2. 根据域名是如何查找lp地址的?
                            问题:
                                   答:域名与ip地址是类似key-value一样存取(哈希值对应表)
           DNS域名解析服务器
                                   3. DNS服务器域名缓存策略?
           是如何解析的?
                                   答: 没有找到请求域名对应的ip地址时, 说明没有存储过, 此时需
                                   要向父结点甚至根结点请求ip地址,返回后先是存储对应关系,再
                                   返回给协议ip地址
                                        Wi-Fi
                                             Wi-Fi TCP/IP DNS WINS 802.1X 代理 硬件
                                         DNS服务器:
                            查看分配的
                                         打开wifi设置,默认都有分配一个DNS服务器
                            DNS服务器:
                                                                    可以添加多个DNS地址
                                           IPv4或IPv6地址
                                       lp地址 + 端口号
                    两个计算机通信必须知道:
                    比如 http://132.232.16.94:80/
                                          http默认端口80
                    Ip地址是一个32位的二进制数,通常被分割为8位二进制数
                    Ip地址显示通常以十进制加点 分割显示
                                          通过命令 ping 127.0.0.1 回车,如果返回数据信
                    本机地址(回环地址)localhost:
                                          息,说明网卡是通的,不然就是网卡有问题,或配
           lp地址:
                    127.0.0.1
                                          置问题
                            <sup>一</sup>播地址(发给当前网络所有计算机发消息):
                    其他:
                                                         任何: 0.0.0.0
                           255.255.255.255
                    一个ip地址唯一标识一台计算机,但是
                                              比如两块网卡,一块集成网卡,
                    一台计算机可以有多个ip地址对应:
                                              一块无线网卡
                    内网: 内网的计算机都是通过路由器上网, 对外公开的是路由器的ip地址
                    标识不同进程作用
                    http默认端口80, https默认端口443, FTP 默认21,
                    远程服务器默认ssh端口为22
                    取值返回 0-65535
                    通过ip地址找到计算机,再通过端口号找
                                               而服务或软件通过监听端口号,
           端口号:
                    到对应的服务或进程, 向其发送数据
                                               来确保获得自己的数据
                                           常见默认端口, 以及有服务功能
                            公认端口: 0-1024
                                           的软件:比如邮件服务,
                    分类:
                            注册端口:1025-49151
                                             分配给用户进程和应用程序
                                                  不固定支配,可以动态使用,比如
                            动态/私有端口: 49152-65535
                                                  设置一个个人端口并用来监听
                          1.地址空间不同,IPv4中规定IP地址长度为32,而IPv6中IP地址的长度为128
                          2.路由表大小不同,IPv6的路由表相比IPv4的更小。
                          3.IPv6的组播支持以及对流的支持要强于IPv4。
           IPv4和IPv6 是啥?
                          4.安全性不同,IPv6的安全性更高,在使用IPv6的网络时,用
                          户可对网络层的数据进行加密。
                         5.协议扩充不同,IPv6允许协议进行扩充而IPv4不允许
                               答案是肯定的,一旦分配的DNS出现异常,那么域名请求就
                               会失败. (有时发现连接不上网络,但是可以QQ发消息)
                               异常可能是服务器自身的原因,也可能是被攻击(DNS被劫持)
           DNS服务器会出现异常吗?
                                                   不使用域名,直接使用ip地址请求,绕过域名解析
                               如果是被劫持,如何能请求?
                                                   或者手动设置其他DNS服务器地址, 比如谷歌
                                                   的 8.8.8.8, 或者国内的 14.14.14.14 等等
                   HTTP 请求报文由请求行、请求头部、空行 和 请求包体 4 个部分组
                   成,如下图所示:
                                                      请求行
                    请求方法
                                     协议版本
http理解2
                              URL
                                              换行符
                          空格
                                  空格
                                          回车符
                   头部字段名称
                          : (冒号)
                                          回车符
                                   值
                                              换行符
                                                   请求头部
           请求:
                                          回车符 换行符
                   头部字段名称
                          : (冒号)
                             空行(回车符 或 换行符)
                                                      请求包体
                                请求包体
                   注意空格间隔
                  HTTP 响应报文由状态行、响应头部、空行 和 响应包体 4 个部分组成,
                  如下图所示:
                                                     状态行
                                  空格 状态码描述
                                          回车符
                                              换行符
                    协议版本
                         空格
                             状态码
                   头部字段名称
                          : (冒号)
                                          回车符
                                              换行符
                                   值
                                                    响应头部
           响应
                                          回车符
                                              换行符
                   头部字段名称
                          :(冒号)
                             空行(回车符 或 换行符)
                                                      响应包体
                                响应包体
                  注意: POST上传大数据时,响应包体前后有字符标识,类似 – – xxxx
                                      ASCⅡ码:数字,字符,英文字母等 127个编码,
                         欧美(ASCII 码):
                                      ASCII 码 256个足够,比如 a-97, b-98
                                      除了支持ASCII 码的那些,还要支持简体中文,这时需
                         中国(GB2312):
                                      要自己的编码格式, 此时支持汉字较少,几千个
                                                          如果不指定编码格式,有中文就
                                    需求不断扩大,要支持更多汉
                         中国(GBK):
           数据存储及编码:
                                    字,此时需要扩编,于是有了GBK
                                                          会乱码,一般设置UTF-8
                         其他,比如BIG5 等等
                                     各个地区和国家都定义了自己的编码格式,为了统一编码格式,创
                                     建了Unicode编码,其支持全世界大部分信息编码格式
                         Unicode编码:
                                     只兼容ASCII 码,不兼容其他编码格式
                      http1.0 版本协议默认设置connection = close,请求响应-立即断开链接
           http短连接
                      每次请求都需要重新连接,三次握手,4次分手,消耗资源
                      三次请求:
                                请求1-连接1-响应1-关闭1 - 请求2-连接2-响应2-关闭2 - 请求3-连接3-响应3-关闭3
                      现在大部分是基于http1.1版本,默认connection = keep alive;
                      关闭需要手动设置close
                      客户端和服务器建立链接后,不会立即断开链接,而是等待一会,如果
                      这个客户端还有新的数据请求,就不需要再次链接了,节约资源
           http长连接
                      如果一段时间后(keep alive timeout 超时),依然没有发起请求,那么
                      服务器认为客户端暂时不需要请求, 断开链接
                      总结: http长连接就是让多个请求复用, 避免重复建立链接
                      三次请求:
                                请求1-连接1-响应1 - 请求2-响应2 - 请求3-响应3- 关闭1
                      与http长连接是两个概念, tcp也有一个长连接概念, 两者到底是啥区别呢?
                             1. http是基于tcp/ip的协议(不是udp/ip)
                                                 1. 请求(ip+端口)- 2. tcp连接 - 3. 响应 - 4. 关闭tcp连接
                                                 条件: tcp连接必须有一端来断开,这样才能完成一次短连接请求
                             2. http1.0的一次简单请求
                             包括(默认http短连接):
                                                    如果tcp连接没有被服务
                                                                       (服务器能建立的tcp连接数越多,
                                                 器主动释放, 总不能一直持有吧
                                                                       越消耗资源,更何况无效连接)
                                                 (客户端是被动关闭的)
                                                  1. 上面三次请求中只有一次连接和一次关闭tcp连接, 说明http长连接
                                                  就是多次请求下对tcp连接的复用
                             3. http1.1后默认的长连接?
                                                  条件: tcp连接也必须有一端来断开,这样才能完成一次长连接
                                                     如果tcp连接没有被主动释放呢?
                                                         ,去检测两端是否运行正常,是否还在线
           tcp长连接
                                          默认不开启
                                                     tcp Keep-Alive
                      区别:
                                                        两端之间没有正常关闭tcp连接(比如服
                                                        务器挂了),导致长时间没法通信
                                          开启后触发条件:
                                                        Tcp连接后,长时间不发消息,就这样闲着
                             4. tcp长连接:
                                                   TCP协议会过一段时间(可以设置)后去发送一个空报文给对方,若
                                          做什么?
                                                   有响应则对方在线,保持连接;若没有反应,则进行重试,达到一定
                                                   次数之后则认为连接已经丢失,没有必要再保持了。
                                                 tcp长连接: TCP的Keepalive机制意图在于探测连接的对端是否存活
                                                 说白了就是, 你们这么久不发信息,到底还在不在线呢,?
                                          总结:
                                                 tcp连接后,超时时长内没有发生请求,就会去做一个两端的是否在线状态检查
                                                 是tcp连接在规定时间内没
                                                                     可能是忘了关闭tcp连接,也可能
                                                 有通信情况下触发的
                                                                     是某端出现异常,无法响应
                    http -> socket -> tcp/ip
                    socket -> tcp/ip
           socket
                    socket -> udp/ip
                    Socket 是系统底层tcp/udp操
                    作对外开放的API 抽象接口
                  在即时通讯这块, 其使用的是基于socket通信, 协议有tcp或者udp, 腾讯等大厂都是udp+自己
                  私有协议保证数据传输完整和安全性,一般小公司都是tcp,相对简单,但是限制多
                  为了实时获得信息,我们需要一个tcp链接是长久保持的状态
                  Tcp长连接其底层默认是2小时进行一次保活请求,实际上等不了这么
           心跳
                  久,就会被各种外界因素强制关闭连接
                  我国的移动网络环境一般5分钟后就会删除路由ip地址对应信息,也就间接失效了
                  因此,我们设置socket的超时为3-5分钟以内,来保证tcp长连接,通过不
                  断的发送0/1等简单信息请求来保持连接,这就是心跳
                                    https://www.jianshu.com/p/2dbb360886a8
           iOS即时通讯,从入门到"放弃"?
                       0:默认使用http缓存策略
                                           iOS默认超时时长60s
                                         对数据及时显示要求高是,比如
                        1: 总是加载最新的数据
                                         股票,比如直播
           请求缓存策略
                                                  数据更新慢,缓存有效性比较长,
                        2:优先加载缓存,没有就加载新数据
                                                  此时可以节省时间和流量
                                    1. 断网的情况下只查询缓存
                       3:只加载缓存
                                    2. 或者缓存页面请求新数据之前先加载缓存
                     默认60s
           请求超时
                     自定义设置15-30s
```