1. HTTP协议、WebSocket协议、Socket协议之间的区别

HTTP协议：

是一个应用层协议，此协议是非持久化的，单向的网路协议，在建立连接之后只允许浏览器发出请求后，服务器才能返回相应的数据。当需要即时通讯时，通过轮询在特定的时间间隔（如1秒），由浏览器向服务器发送Request请求，然后将最新的数据返回给浏览器。这样的方法最明显的缺点就是需要不断的发送请求，而且通常HTTP request的Header是非常长的，为了传输一个很小的数据，需要付出巨大的代价，是很不合算的，占用了很多的宽带。

HTTP的地址格式如下：

http\_URL = "http:" "//" host [ ":" port ] [ abs\_path [ "?" query ]]

协议和host不分大小写

HTTP消息

一个HTTP消息可能是request或者response消息，两种类型的消息都是由开始行（start-line），零个或者多个header域，一个表示header域结束的空行（也就是，一个以CRLF为前缀的空行），一个可能为空的消息主体（message-body）。一个合格的HTTP客户端不应该在消息头或者尾添加多余的CRLF，服务端也会忽略这些字符。

Header的值不包括任何前导或后续的LWS（线性空白），线性空白可能会出现在域值（filed-value）的第一个非空白字符之前或最后一个非空白字符之后。前导或后续的LWS可能会被移除而不会改变域值的语意。任何出现在filed-content之间的LWS可能会被一个SP（空格）代替，header域的顺序不重要，但建议把常用的header放在前面。

Request消息

RFC2626中这样定义HTTP Request消息：

Request = Request-Line

\*(( general-header

| request-header（跟本次请求相关的一些header）

| entity-header ) CRLF)（跟本次请求相关的一些header）

CRLF

[ message-body ]