1、面试题

说一下http的工作流程？http 1.0、http 1.1、http 2.0具体有哪些区别？

2、面试官心里分析

这个就是让你聊聊http，说白了http工作原理，你都知道了，发起个http，底层都是tcp、ip、以太网那块再走，一层一层包裹数据包。所以http的关键就是让你聊聊http请求和http响应的规范。

3、面试题剖析

http发起请求的底层原理，大家其实都知道了，理解了那个原理，就一通百通了。那么来聊下http请求和响应的规范吧。其实请求的报文，就是请求头、请求方法、请求正文，GET/POST啥的，应该都知道；请求头，自己百度一下吧，作为一个工程师必须知道。响应，状态行，响应头，响应正文，状态行，200,400,500，实在不想讲了；响应头，自己查一下。

http请求封装到应用层数据包，封装在tcp数据包，封装在ip数据包，封装在以太网数据包，如果过大，可能会拆成几个包，走以太网协议+交换机 -> 广播 -> 网关 -> 多个网关 -> 目标的机器 -> 一层一层拆包 -> http请求报文 -> 传递给tomcat -> spring mvc -> http响应 -> 一样的路径会去

最最底层，这个数据如何传输？走的是物理层，网线、光缆，所有数据都是0/1电路信号

http协议，其实是每个搞java必须会的基础。

互联网初期，一般一个网页几乎都没什么图片，当时就是挂一些文字，一个网页里就是一大坨的文字。http 1.0版本。

浏览器 -> 网站，互相之间是先要通过tcp三次握手，建立一个连接，浏览器和网站互相都给对方留出一份资源，浏览器发起http请求 -> tcp -> ip -> 以太网，网站上面去，网站返回一个响应，连接关闭，tcp四次挥手。释放掉浏览器和网站各自给对方保持的一份资源。

http 1.0要指定keep-alive来开启持久连接，默认是短连接，就是浏览器每次请求都要重新建立一次tcp连接，完事儿了就释放tcp连接。早期的网页都很low，没啥东西，就一点文字，就用这个没问题。但是现在，一个网页打开之后，还要加载大量的图片、css、js，这就坑爹了，发送多次请求。

早期，2000年之前，那个时候网页，都很low，当时你打开一个网页，就是说现场底层tcp三次握手，跟网站建立一个tcp连接，然后通过这个tcp连接，发送一次http请求，网站返回一个http响应（网页的html，里面有一大段文字），浏览器收到html渲染成网页，浏览器就走tcp四次挥手，跟网站断开连接了

到了后面，发现说2000之后，2010之后更不用说了，网页发展很迅猛，一个网页包含着大量的css、js、图片等资源。比如你请求一个网页，这个网页的html先过来，过来之后，浏览器再次发起大量的请求去加载css、js、图片，打开一个网页可能浏览器要对网站服务器发送几十次请求。

http 1.0，疯了，刚开始请求网页的html，tcp三次握手建立连接 -> 请求/响应 -> tcp四次挥手断开连接，接着再次要加载css、js、图片，要发送30个请求，上面的过程来30次，30次频繁的建立tcp连接以及释放tcp连接。很慢很慢。

其实最慢的不是说发送请求和获取响应，打开和释放连接，这都是很重的过程

http 1.1默认支持长连接，就是说，浏览器打开一个网页之后，底层的tcp连接就保持着，不会立马断开，之后加载css、js之类的请求，都会基于这个tcp连接来走。http 1.1还支持host头，也就可以支持虚拟主机；而且对断点续传有支持。

浏览器，第一次请求去一个网站的一个页面的时候，就会打开一个tcp连接，接着就在一段时间内都不关闭了，然后接下来这个网页加载css、js、图片大量的请求全部走同一个tcp连接，频繁的发送请求获取响应，最后过了一段时间，这些事儿都完了，然后才会去释放那一个tcp连接。大幅度的提升复杂网页的打开的速度，性能。

http 2.0，支持多路复用，基于一个tcp连接并行发送多个请求以及接收响应，解决了http 1.1对同一时间同一个域名的请求有限制的问题。二进制分帧，将传输数据拆分为更小的帧（数据包），frame（数据包，帧），提高了性能，实现低延迟高吞吐。

853769620