1.HTTP 基础知识

1.1 概念

HTTP 协议是 HyperTextTransferProtocol（超文本传输协议）的缩写，是用于从万维网（WWW:WorldWideWeb）服务器传输超文本到本地浏览器的传送协议。

1.2 特点

- HTTP 基于 TCP/IP 协议：http 协议是基于 TCP/IP 协议之上的应用层协议。
- HTTP 是基于请求－响应模式：HTTP 协议规定，请求从客户端发出，最后服务器端响应该请求并返回。
- HTTP 是无状态保存：HTTP 协议是无状态协议。无状态是指协议对于事务处理没有记忆能力。缺少状态意味着如果后续处理需要前面的信息，则它必须重传，这样可能导致每次连接传送的数据量增大；另一方面，在服务器不需要先前信息时它的应答就较快。

1.3URL

HTTP 使用统一资源标识符（UniformResourceIdentifiers,URI）来传输数据和建立连接。URL 是一种特殊类型的 URI，包含了用于查找某个资源的足够的信息。

语法规则：scheme://host.domain:port/path/filename

- scheme：指定因特网服务的类型。最流行的类型是 HTTP。
- domain：指定因特网域名，比如：crazyit.org、fkjava.org 等。
- host：指定此域中的主机。如果被省略，HTTP 的默认主机是 www。
- port：指定主机的端口号。端口号通常可以被省略，HTTP 服务的默认端口号是 80。
- path：指定远程服务器上的路径，该路径也可以被省略，省略该路径则默认被定位到网站的根目录。
- filename：指定远程文档的名称。如果省略该文件名，通常会定位到 index.html、index.htm 等文件，或定位到 WEB 服务器设置的其他文件。

例子解析：

http://www.cufe.edu.cn/index.htm
http://news.cufe.edu.cn/info/1002/22816.htm
https://movie.douban.com/tag/#/

- 锚部分：从“#”开始到最后，都是锚部分。锚部分不是 URL 必须的部分。
- 参数部分：从“？”开始到“#”为止之间的部分为参数部分，又称搜索部分、查询部分。参数可以允许有多个参数，参数与参数之间用“&”作为分隔符。

1.4Request

1.4.1 常见的报文头

- Host：告知服务器，需要访问的域名和端口。如果没有给定端口号，会自动使用被请求服务的默认端口（比如请求一个 HTTP 的 URL 会自动使用 80 端口）
- Connection：告知服务器，当前的事务完成后，是否会关闭网络连接。
- 如果该值是“keep-alive”，网络连接就是持久的，不会关闭，使得对同一个服务器的请求可以继续在该连接上完成。
- Accept：告知服务器，客户端可以处理的内容类型。
- **User-Agent：**告知服务器，发起请求的用户代理软件的应用类型、操作系统、软件开发商以及版本号。
- Referer：告知服务器，当前请求页面的来源页面的地址，即表示当前页面是通过此来源页面里的链接进入的。
- Accept-Encoding：告知服务器，他支持的压缩格式。
- Accept-Language：告知诉服务器，他支持的语言。
- Cookie：含有先前由服务器通过 Set-Cookie 首部投放并存储到客户端的 HTTPcookies。

1.4.2 请求方法

- GET：最常见的方式，一般用于获取或者查询资源信息，也是大多数网站使用的方式，响应速度快。
- POST：相比 GET 方式，以表单形式上传参数的功能，因此除查询信息外，还可以修改信息。

1.5Response

1.5.1 常见的报文头

- Date：报文创建的日期和时间。
- Content-Type：告知浏览器，回送数据的类型。
- Transfer-Encoding：告知浏览器，将 entity 安全传递给用户所采用的编码形式。
  - chunked：数据以一系列分块的形式进行发送。
- Connection：告知浏览器，当前的事务完成后，是否会关闭网络连接。如果该值是“keep-alive”，网络连接就是持久的，不会关闭，使得对同一个服务器的请求可以继续在该连接上完成。
- Pragma：在 HTTP/1.0 中规定的通用首部，这个首部的效果依赖于不同的实现，所以在“请求-响应”链中可能会有不同的效果。
  - no-cache：强制要求缓存服务器在返回缓存的版本之前将请求提交到源头服务器进行验证。
- Cache-Control：指定指令来实现缓存机制
  - private：响应只能被单个用户缓存，不能作为共享缓存（即代理服务器不能缓存它）。私有缓存可以缓存响应内容。
  - no-cache：在发布缓存副本之前，强制要求缓存把请求提交给原始服务器进行验证。

- must-revalidate：一旦资源过期（比如已经超过 max-age），在成功向原始服务器验证之前，缓存不能用该资源响应后续请求。
- Server：告知浏览器，服务器的型号等软件相关信息。
- Content-Encoding：告知浏览器，数据压缩的格式。

1.5.2 响应状态码

- 1xx：请求收到，继续处理
- 2xx：操作成功收到，分析，接受
  - 200 交易成功
- 3xx：重定向
- 4xx：客户端错误
  - 403 禁止访问
  - 404 没有发现文件、查询或 URL
  - 418I'mateapot
- 5xx：服务器错误

1.6HTTPS 协议

HTTPS(SecureHypertextTransferProtocol)安全超文本传输协议，HTTPS 是在 HTTP 上建立 SSL 加密层，并对传输数据进行加密，是 HTTP 协议的安全版。