

## Python-HTTP 概况

### ◆前端知识

#### ■网页最主要由 3 部分组成:

- (1) HTML
- (2) CSS
- (3) JavaScript

#### ■比喻成建房子:

### ◆互联网

- 1、什么是互联网
- 2、互联网建立的目的?
- 3、什么是上网

### ◆Web 开发本质:

### ◆HTTP 协议

- 1、什么是 HTTP 协议:
- 2、为何要用 HTTP 协议
- 3、如何用 HTTP 协议:
  - 1、HTTP 是基于 TCP 协议
  - 2、HTTP 协议的特点:
  - 3、HTTP 之 request
  - 4、HTTP 之 response
- HTTP 协议完整工作流程

#### 1、客户端连接到 Web 服务器

#### 2、发送 HTTP 请求

#### 3、服务器接受请求并返回 HTTP 响应

#### 4、释放连接 TCP 连接

#### 5、客户端浏览器解析 HTML 内容

#### ■HTTP 协议关键性总结

## 前端知识

### 网页最主要由 3 部分组成:

结构、表现和行为。网页现在新的标准是 W3C，目前模式是 HTML、CSS 和 JavaScript。

#### (1) HTML

HTML，全称“Hyper Text Markup Language（超文本标记语言）”，简单来说，网页就是用 HTML 语言制作的。HTML 是一门描述性语言，是一门非常容易入门的语言。

## (2) CSS

CSS，全称“（层叠样式表）”。以后我们在别的地方看到“层叠样式表”、“CSS 样式”，指的就是 CSS。

## (3) JavaScript

JavaScript 是一门脚本语言。

HTML、CSS 和 JavaScript 的区别 我们都知道前端技术最核心的是 HTML、CSS 和 JavaScript 这三种。

### 比喻成建房子：

“HTML 是网页的结构，CSS 是网页的外观，而 JavaScript 是页面的行为  
如果我们把前端开发的过程比喻成“建房子”，做一个网页就像盖一栋房子

- 1、先把房子结构建好（HTML）
- 2、建好房子之后给房子装修（CSS），例如往窗户安上窗帘、往地板铺上漂亮的瓷砖
- 3、最后呢，装修完了之后，当夜幕降临的时候，我们要开灯（JavaScript），这样才能看得见里面。

## 互联网

### 1、什么是互联网

物理连接介质+互联网协议

### 2、互联网建立的目的？

用于打破地域限制来进行数据传输

### 3、什么是上网

用户通常使用浏览器发送请求给目标主机，然后将目标主机上的文件内容下载到本地，浏览器与服务端之间是基于 http 协议工作的

## Web 开发本质：

---

1. 浏览器输入网址回车都发生了什么?

1. 浏览器 给服务端 发送了一个消息

2. 服务端拿到消息

3. 服务端返回消息

4. 浏览器展示页面

C/S 架构 --> B/S 架构

## HTTP 协议

### 1、什么是 HTTP 协议：

全称 Hyper Text Transfer Protocol（超文本传输协议）

HTTP 协议是用于从（WWW:World Wide Web，简万维网）服务器传输超文本到本地浏览器的传送协议。

HTTP 协议工作于 B/S 架构上

浏览器作为 HTTP 客户端通过 URL 向 HTTP 服务端即 WEB 服务器发送请求 Request。

Web 服务器根据接收到的请求后，向客户端发送响应信息 Response。

HTTP 协议是基于 TCP/IP 通信协议来传递数据的（HTML 文件，图片文件等）

### 2、为何要用 HTTP 协议

是应用层测协议，用于浏览器与 web 服务端之间传输数据

### 3、如何用 HTTP 协议：

#### 1、HTTP 是基于 TCP 协议

#### 2、HTTP 协议的特点：

##### I：无连接

HTTP 协议本身是没有连接的，但没进行一次 HTTP 协议的通信

必先建立 TCP 连接，该 TCP 连接会在通信完毕后关闭，如果短时间内再次发起 HTTP 请求则必须重新建立连接，这么做的缺点是：1、对服务端造成过大的压力 2、数据传输速度慢  
解决方法：

Keepalive

II: 无状态:  
cookie 与 session

### 3、HTTP 之 request

GET url HTTP/1.1\r\n

k1:v1\r\n

k2:v2\r\n

....

\r\n

请求体（只有 post 方法才有请求体）

请求 URL:

URL 是一种 URI 的具体实现方式

https://www.cnblogs.com:80/linhaifeng/articles/8982675.html?a=1&b=1#anc1

https 比 http 多了一个 ssl 安全认证

域名+端口号

虚拟目录

文件

&请求参数

#页面锚点

请求方法:

GET

参数是存放 url 地址的? 后的

参数大小有限制

不安全

POST

参数是存放请求体中的

参数大小无限制

相对安全

请求头:

user-agent

referer
cookie

## 4、HTTP 之 response

```
HTTP/1.1 200 Ok\r\n
```

```
k1:v1\r\n
```

```
k2:v2\r\n
```

```
...
```

```
\r\n
```

相应体（即我们要从服务端下载的内容）

## HTTP 协议完整工作流程

HTTP 协议定义 Web 客户端如何从 Web 服务器请求 Web 页面，以及服务器如何把 Web 页面传送给客户端。HTTP 协议采用了请求/响应模型。客户端向服务器发送一个请求报文，请求报文包含请求的方法、URL、协议版本、请求头部和请求数据。服务器以一个状态行作为响应，响应的内容包括协议的版本、成功或者错误代码、服务器信息、响应头部和响应数据。

以下是 HTTP 请求/响应的步骤：

### 1、客户端连接到 Web 服务器

一个 HTTP 客户端，通常是浏览器，与 Web 服务器的 HTTP 端口（默认为 80）建立一个 TCP 套接字连接。例如，<http://www.oakcms.cn>。

### 2、发送 HTTP 请求

通过 TCP 套接字，客户端向 Web 服务器发送一个文本的请求报文，一个请求报文由请求行、请求头部、空行和请求数据 4 部分组成。

### 3、服务器接受请求并返回 HTTP 响应

Web 服务器解析请求，定位请求资源。服务器将资源复本写到 TCP 套接字，由客户端读取。一个响应由状态行、响应头部、空行和响应数据 4 部分组成。

#### 4、释放连接 TCP 连接

若 connection 模式为 close，则服务器主动关闭 TCP 连接，客户端被动关闭连接，释放 TCP 连接;若 connection 模式为 keepalive，则该连接会保持一段时间，在该时间内可以继续接收请求;

#### 5、客户端浏览器解析 HTML 内容

客户端浏览器首先解析状态行，查看表明请求是否成功的状态代码。然后解析每一个响应头，响应头告知以下为若干字节的 HTML 文档和文档的字符集。客户端浏览器读取响应数据 HTML，根据 HTML 的语法对其进行格式化，并在浏览器窗口中显示。

例如：在浏览器地址栏键入 URL，按下回车之后会经历以下流程：

- 1、浏览器向 DNS 服务器请求解析该 URL 中的域名所对应的 IP 地址;
- 2、解析出 IP 地址后，根据该 IP 地址和默认端口 80，和服务器建立 TCP 连接;
- 3、浏览器发出读取文件(URL 中域名后面部分对应的文件)的 HTTP 请求，该请求报文作为 TCP 三次握手的第三个报文的数据发送给服务器;
- 4、服务器对浏览器请求作出响应，并把对应的 html 文本发送给浏览器;
- 5、释放 TCP 连接;
- 6、浏览器将该 html 文本并显示内容;

## HTTP 协议关键性总结

### #1、简单快速

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。每种方法规定了客户与服务器联系的类型不同。由于 HTTP 协议简单，使得 HTTP 服务器的程序规模小，因而通信速度很快。

### #2、灵活

HTTP 允许传输任意类型的数据对象。正在传输的类型由 Content-Type 加以标记。

### #3、无连接

HTTP 无连接说的是：当某个客户机在短时间多次请求同一个资源，服务器并不能区别是否已经响应过用户的请求。

于是我们每次发送 http 请求，都需要事先发起一个到服务器的 TCP 请求，经历“三次握手”的过程。这针对大流量的服务器来说，开销是相当大的。这是 http 无链接带来的缺点

针对 http 无连接，人们设计了非持久连接和持久连接。实际上关于 http 协议非持久连接和持久连接是针对 tcp 协议的。当客户机/服务器的交互运行于 TCP 协议上时，应用程序的每个请求/响应对是经不同的 TCP 连接时，则该应用程序使用非持久连接，而当应用程序的每个请求/响应对是经相同的 TCP 连接发送，则该应用程序使用持久连接。

非持久连接

请求一个 HTTP 请求/响应需要的总时间=客户端发出建立连接+发生请求报文+服务器传输 HTML 文件的时间

持久连接

服务器在发送响应后，保持该 **TCP** 连接打开。在相同的客户机与服务器之间的后续请求和响应报文通过相同的连接进行传送。不需要再次建立 **tcp** 连接

#### #4、无状态

所谓 **http** 是无状态协议，言外之意是说 **http** 协议没法保存客户机信息，

无状态的优点是：

在服务器不需要先前信息时它的应答就较快。

无状态的缺点是：

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

关于 **http** 无状态阻碍了交互式应用程序的实现。比如记录用户浏览哪些网页、判断用户是否拥有权限访问等。于是，两种用于保持 **HTTP** 状态的技术就应运而生了，一个是 **Cookie**，而另一个则是 **Session**。

#### #5、支持 B/S 及 C/S 模式。