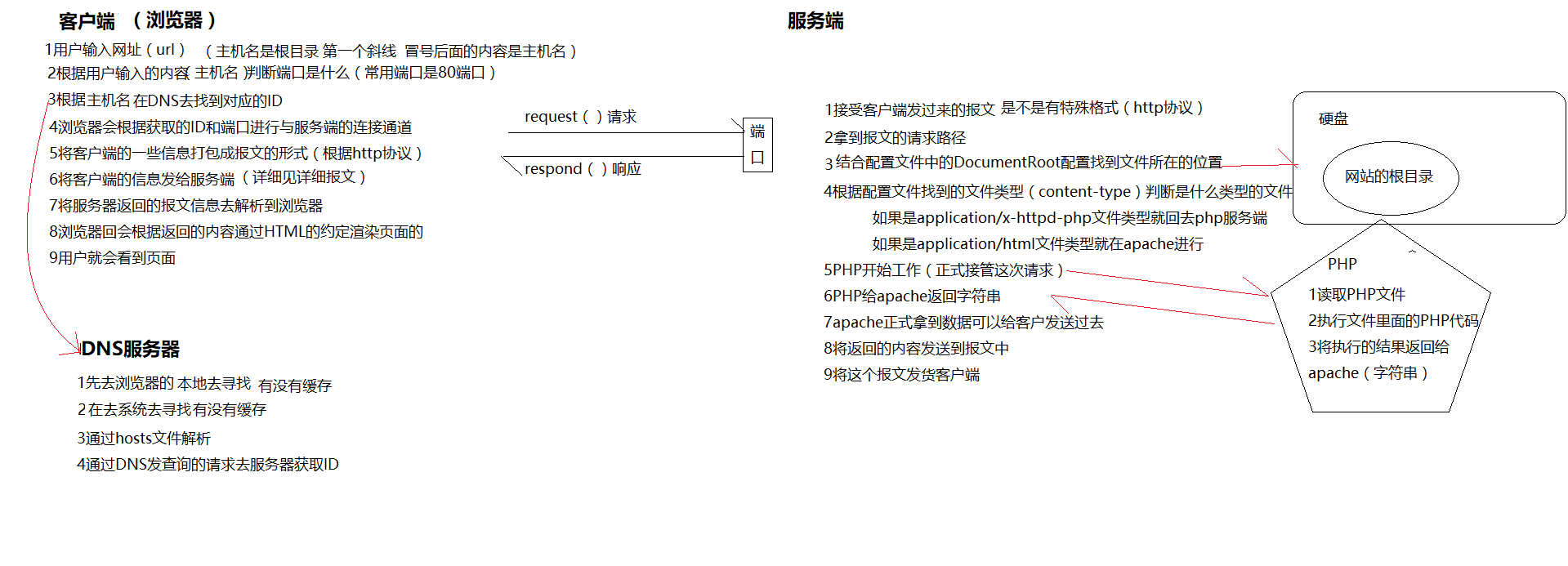
Content-type 请求体里面的一个属性 Content-Type: application/x-www-form-urlencoded

如果服务端响应的 Content-Type 为 application/json ajax可以不用设置

application/text

dataType: 'json',（ajax里面设置这个属性）

header('Content-Type: application/json'); 再头部设置文件类型为json（响应类型）



DNS注意：

只能影响本机的DNS查找

必须是管理员的权限运行的编辑器才有权利更改hosts文件

（在服务器中，apache相应的只是html代码 响应给客户端之后 浏览器就自动渲染界面 php知识把结果返回给apache）

启动apache

先请求

进入服务器之后先判断是静态页面还是动态页面

静态页面 自己处理文件，将结果直接响应到客户端

动态页面 读取php文件里面的东西，执行代码 将代码结果返回到apache apache之后后在响应到客户端

http协议 超文本传输协议

http 是有请求和响应构成，还是一个无状态的协议 每次的请求都是重新请求

HTTP协议永远都是客户端发起请求，服务器回送响应

工作流程

一次HTTP操作称为一个事务，其工作过程可分为四步：

1）首先客户端与服务器需要建立连接。只要单击某个链接，HTTP的工作开始。

2）建立连接后，客户端发送一个请求给服务器，请求方式的格式为：统一资源标识符（URL）、协议版本号，后边是MIME信息包括请求修饰符、客户端信息和可能的内容。

3）服务器接到请求后，给予相应的响应信息，其格式为一个状态行，包括信息的协议版本号、一个成功或错误的代码，后边是MIME信息包括服务器信息、实体信息和可能的内容。

4）客户端接收服务器所返回的信息通过浏览器显示在用户的显示屏上，然后客户机与服务器断开连接。

如果在以上过程中的某一步出现错误，那么产生错误的信息将返回到客户端，有显示屏输出。对于用户来说，这些过程是由HTTP自己完成的，用户只要用鼠标点击，等待信息显示就可以了。

Cookie和Session有以下明显的不同点：

1）Cookie将状态保存在客户端，Session将状态保存在服务器端；

2）Cookies是服务器在本地机器上存储的小段文本并随每一个请求发送至同一个服务器。Cookie最早在RFC2109中实现，后续RFC2965做了增强。网络服务器用HTTP头向客户端发送cookies，在客户终端，浏览器解析这些cookies并将它们保存为一个本地文件，它会自动将同一服务器的任何请求缚上这些cookies。Session并没有在HTTP的协议中定义；

3）Session是针对每一个用户的，变量的值保存在服务器上，用一个sessionID来区分是哪个用户session变量,这个值是通过用户的浏览器在访问的时候返回给服务器，当客户禁用cookie时，这个值也可能设置为由get来返回给服务器；

4）就安全性来说：当你访问一个使用session 的站点，同时在自己机子上建立一个cookie，建议在服务器端的SESSION机制更安全些.因为它不会任意读取客户存储的信息。

Session机制

Session机制是一种服务器端的机制，服务器使用一种类似于散列表的结构（也可能就是使用散列表）来保存信息。

当程序需要为某个客户端的请求创建一个session的时候，服务器首先检查这个客户端的请求里是否已包含了一个session标识 - 称为 session id，如果已包含一个session id则说明以前已经为此客户端创建过session，服务器就按照session id把这个 session检索出来使用（如果检索不到，可能会新建一个），如果客户端请求不包含session id，则为此客户端创建一个session并且生成一个与此session相关联的session id，session id的值应该是一个既不会重复，又不容易被找到规律以仿造的字符串，这个 session id将被在本次响应中返回给客户端保存。

有两种基本的加解密算法类型：

1）对称加密：密钥只有一个，加密解密为同一个密码，且加解密速度快，典型的对称加密算法有DES、AES等；

2）非对称加密：密钥成对出现（且根据公钥无法推知私钥，根据私钥也无法推知公钥），加密解密使用不同密钥（公钥加密需要私钥解密，私钥加密需要公钥解密），相对对称加密速度较慢，典型的非对称加密算法有RSA、DSA等。

https通信的优点：

1）客户端产生的密钥只有客户端和服务器端能得到；

2）加密的数据只有客户端和服务器端才能得到明文；

3）客户端到服务端的通信是安全的。

3.5 http代理

虚拟主机：是在网络服务器上划分出一定的磁盘空间供用户放置站点、应用组件等，提供必要的站点功能与数据存放、传输功能。

所谓虚拟主机，也叫“网站空间”就是把一台运行在互联网上的服务器划分成多个“虚拟”的服务器，每一个虚拟主机都具有独立的域名和完整的Internet服务器（支持WWW、FTP、E-mail等）功能。一台服务器上的不同虚拟主机是各自独立的，并由用户自行管理。但一台服务器主机只能够支持一定数量的虚拟主机，当超过这个数量时，用户将会感到性能急剧下降。

3.6.2虚拟主机的实现原理

虚拟主机是用同一个WEB服务器，为不同域名网站提供服务的技术。Apache、Tomcat等均可通过配置实现这个功能。

相关的HTTP消息头：Host。

例如：Host: www.baidu.com

客户端发送HTTP请求的时候，会携带Host头，Host头记录的是客户端输入的域名。这样服务器可以根据Host头确认客户要访问的是哪一个域名。

Onclick 里面有 moseuu mousedown

From

设置这次请求的请求体格式是 urlencoded，目的是让服务端认识请求体中的数据