day04

十.表单（重点&难点\*\*\*\*\*\*\*\*\*\*\*\*）

8.新表单元素---h5新出的10个

①邮箱

|  |
| --- |
| <input type="email" name="email">  提交的时候，验证是不是有@符号，并且@前后有字符 |

②搜索类型

|  |
| --- |
| <input type="search" name="search">  提供了一个快速删除的小叉叉 |

③url类型

|  |
| --- |
| <input type="url" name="url">  判断必须有http://，后面还必须有字符 |

④电话号码类型

|  |
| --- |
| <input type="tel" name="tel">  移动端使用，获取焦点时，弹出虚拟数字键盘 |

⑤数字类型

|  |
| --- |
| <input type="number" name="number">  max="20" 设置最大值  min="10" 最小值  step="3" 步长 |

⑥范围类型

|  |
| --- |
| 提供了一个范围选择数字的滑块  <input type="range" name="range">  max="30"  min="20"  step="3" |

⑦颜色类型

|  |
| --- |
| <input type="color" name="color"> |

⑧日期选择控件

|  |
| --- |
| <input type="date" name="date"> |

⑨月份

|  |
| --- |
| <input type="month" name="month"> |

⑩星期

|  |
| --- |
| <input type="week" name="week"> |

**AJAX**

一.HTTP协议

1.url

|  |
| --- |
| 常用结构：协议+主机名称+目录结构+资源名称  URL完整的结构  <scheme>://<user>:<pwd>@<host>:<port>/<path>;<params>?<query>#<flag> |

①scheme

|  |
| --- |
| 方案，协议，允许以哪种方式获取服务器资源  不区分大小写。http/https/ftp |

②<user>:<pwd>

|  |
| --- |
| 直接在url中填写用户名和密码，登录管理界面 |

③<host>

|  |
| --- |
| 主机名称  表现形式两种 1.域名 标志性建筑，好记 www.baidu.com  2.IP地址 真正的网络上的门牌号码 |

④<port>

|  |
| --- |
| 端口号，一台计算中，对外提供服务器的柜台号。  享受不同的服务，需要去对应的柜台 |

⑤<params>

|  |
| --- |
| 参数 session/cookie 第三第四阶段学习，必须掌握  跟踪状态参数 |

⑥<query>

|  |
| --- |
| 查询字符串 |

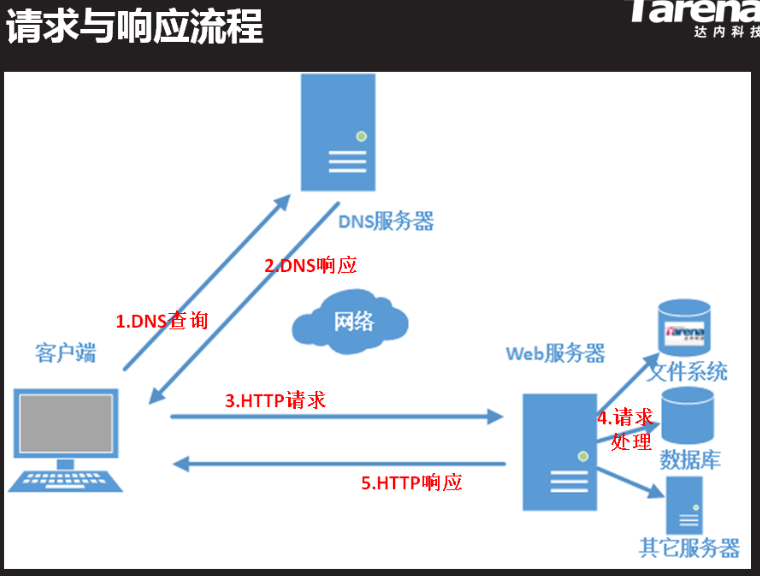
⑦<flag>

|  |
| --- |
| 锚点 |

2.http协议

|  |
| --- |
| HyperText Transfer Protocol 超文本传输协议  规范了数据是如何打包和传递的 |

3.web请求原理



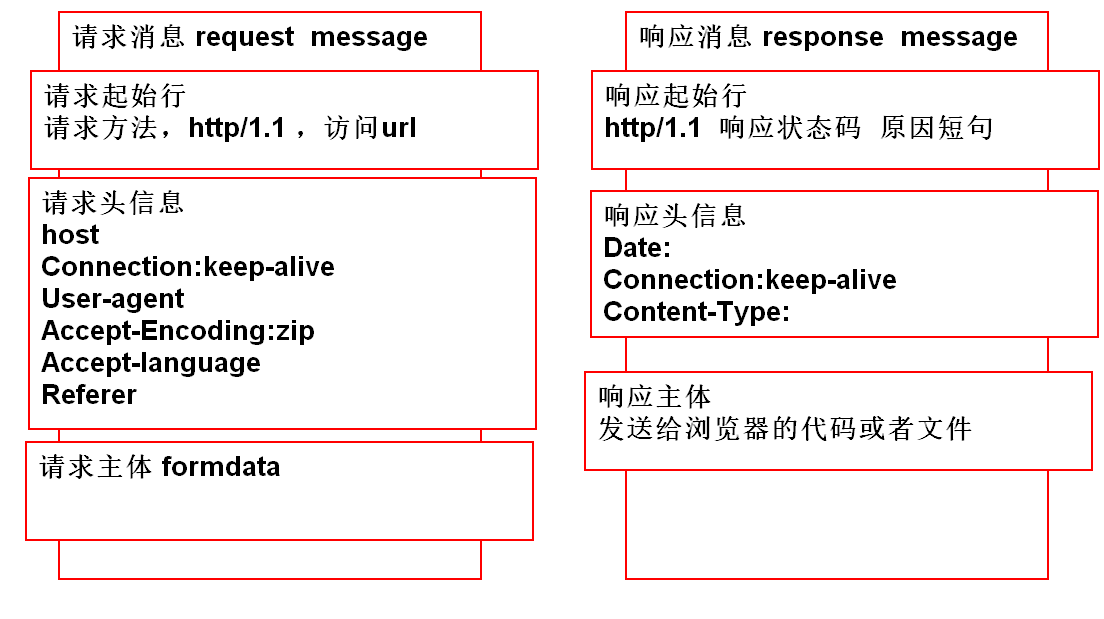
4.消息/报文

①请求消息request message

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 浏览器发送给服务器的文件块  1.请求起始行   |  | | --- | | 请求方法  http/1.1  请求的url |   请求方法   |  |  | | --- | --- | | http原生的请求方法 | Restful API接口 | | get 明文传参，上限2kb，查询字符串传参  向服务器要数据的时候用 | get 做查询模块的时候使用，没有请求主体  select | | post 隐式传参，给服务器传递数据的时候用  请求主体传递参数 | post 做新增模块的时候使用，有请求主体  insert | | put 往服务器上放资源的时候用 | put 做修改模块使用，有请求主体  update | | delete 从服务器删除资源的时候用 | delete 做删除模块时候，没有请求主体  delete | | option 预请求 |  |   2.请求头信息   |  | | --- | | 浏览器发送给服务器  1.Host: www.tmooc.cn  告诉服务器，要请求哪一台主机  2. Connection: keep-alive  告诉服务器，请开启持久连接  3. User-Agent:  浏览器告诉服务器，自己的信息  4. Accept-Encoding: gzip  告诉服务器，当前浏览器能够接收的压缩文件的类型  5. Accept-Language: zh-CN  告诉服务器，当前浏览器能够接收的自然语言的类型  6. Referer: http://www.tmooc.cn/  告诉服务器，我这个请求，来源于哪个网页 |   3.请求主体 form data. |

②响应消息 response message

|  |  |  |
| --- | --- | --- |
| 服务器发送给浏览器的文件块  1.响应起始行   |  | | --- | | http版本号 响应状态码 原因短句  响应状态码 200----成功  404---请求资源不存在，路径写错了  500---服务器代码错误 |   2.响应头信息   |  | | --- | | 1. Date: Mon, 01 Jun 2020 06:25:54 GMT  告诉浏览器，服务器响应的时间  2. Connection: keep-alive  服务器告诉浏览器，持久连接已经开启  3. Content-Type: text/html  告诉浏览器，响应数据的格式  取值 text/html html文件  text/css css文件  application/javascript js文件  image/png 图片  application/json json字符串  application/xml xml字符串 |   3.响应主体 |



5.缓存

|  |
| --- |
| 客户端将服务器响应回来的数据，进行自动的保存  当再次访问的时候，不再发送请求，直接使用保存的数据 |
| 缓存的优点  1.减少冗(rong)余的数据传输，节省流量  2.节省服务器带宽  3.降低了对服务器资源的消耗和运行的要求  4.降低了由于远距离传输而造成的加载延迟 |

6.HTTP的优化

①http的连接性能优化

|  |
| --- |
| 1.减少连接的创建次数（开启持久连接）  2.减少请求的次数 （缓存）  3.提高服务器端的运行速度  4.尽可能减少响应数据的长度 |

②安全的http协议，https

|  |
| --- |
| SSL：为数据通信提供的安全保障支持  1.客户端发送请求-->SSL层加密-->服务器获取到加密文件-->在SSL解密，得到请求明文  2.服务器发送响应-->SSL层加密-->浏览器获取到加密文件-->在SSL解密，得到请求明文 |

二.DOM操作（简易dom操作）

|  |
| --- |
| ajax提交请求，不需要使用form表单  不使用form表单，就失去自动收集整理数据的能力  所以需要使用js的dom操作，手动收集数据 |

1.完整的JavaScript有三个部分

|  |
| --- |
| 1.js核心，ES6  2.dom Document object model 文档对象模型  把所有的html文件，看成一个文档对象，获取其中标签对象  操作这个标签的数据，样式，方法  3.bom Browser Object model 浏览器对象模型  把浏览器中的对象获取到，使用js操作浏览器 |

2.使用dom操作对象的步骤

①写事件，调用js代码

②获取元素对象

③用对象点出属性，方法

3.使用dom操作input 元素

|  |
| --- |
| ①写事件，调用js代码，点击按钮，会调用show方法  <button onclick="show()">显示</button>  ②在show方法中，获取元素对象，n1是input的id值  var inp=document.getElementById("n1");  ③用对象点出value属性  3.1获取value值 var str=inp.value;  3.2修改value的值 inp.value="abc"; |

4.使用dom操作双标签元素

|  |
| --- |
| ①写事件，调用js代码，点击按钮，会调用show方法  <button onclick="show()">显示</button>  ②在show方法中，获取元素对象，n1是双标签的id值  var obj=document.getElementById("n1");  ③使用对象点出innerHTML来操作双标签的内容区域  obj.innerHTML  <div></div> |

5.innerHTML详解

|  |
| --- |
| 给双标签动态改变内部结构，用innerHTML  可以给双标签内部添加所有元素，建议使用模板字符的插值，处理html结构字符串  obj.innerHTML=html的结构字符串 |
| 04\_ex.html  做一个对象数组  点击按钮，把数组内容放到div中显示  准备 1.对象数组  2.div id="d1"  3.button |

作业：

1.使用for完成



2.完成用户列表的dom操作



作业：

1.nodejs接口必须熟练