day04

十.表单(重点&难点\*\*\*\*\*\*\*\*\*\*)

4.h5新出的表单控件

|  |
| --- |
| h5根据需求尝试性的推出了一系列的新标签  目的是为了方便开发者操作，但是有很多的不完善  1.邮箱控件 提交的时候，验证有没有@。@前后有没有字符  <input type="email" name="email">  2.搜索类型，提供了一个可以快速删除的小叉叉  <input type="search" name="search">  3.范围类型。提供了一个0~100数字范围滑块  <input type="range" name="range">  4.url类型 验证用户输入的内容，以协议:开头  <input type="url" name="url">  5.电话号码类型，移动端获取焦点后，会弹出虚拟键盘  <input type="tel" name="tel">  6.数字类型  <input type="number" name="number" step="3" max="20" min="10">  step="3" 步长  max="20" 最大值  min="10" 最小值  7.颜色类型 提供了一个取色器  <input type="color" name="color">  8.日期类型,提供了日期选择模块  <input type="date" name="date">  9.月份类型,提供了月份选择模块  <input type="month" name="month">  10.星期类型  <input type="week" name="week"> |

**AJAX**

一.HTTP协议

1.URL

|  |
| --- |
| url结构：协议+主机名称+目录结构+资源名称  http://cdn.tmooc.cn/img/b9.jpg  URL完整的结构  <scheme>://<user>:<pwd>@<host>:<port>/<path>;<params>?<query>#<flag> |

2.URL完整的结构

①scheme

|  |
| --- |
| 方案，协议。定义以哪种方式获取服务器资源。  不区分大小写 http HTTP |

②<user>:<pwd>

|  |
| --- |
| 早期服务器管理登录网站管理页面时，会在这个位置填写用户名密码  这种方式现在已经过时了 |

③<host>

|  |
| --- |
| 服务器主机名称  1.IP地址，是服务器在互联网上官方的门牌号码  2.域名，是为了方便用户记忆和推广，起的标志性建筑 |

④<port>

|  |
| --- |
| 端口号，计算机提供各种服务的窗口柜台号 |

⑤<path>路径，资源在服务器上具体的存放位置

⑥<params>

|  |
| --- |
| 参数，跟踪状态参数 session/cookie |

⑦<query>

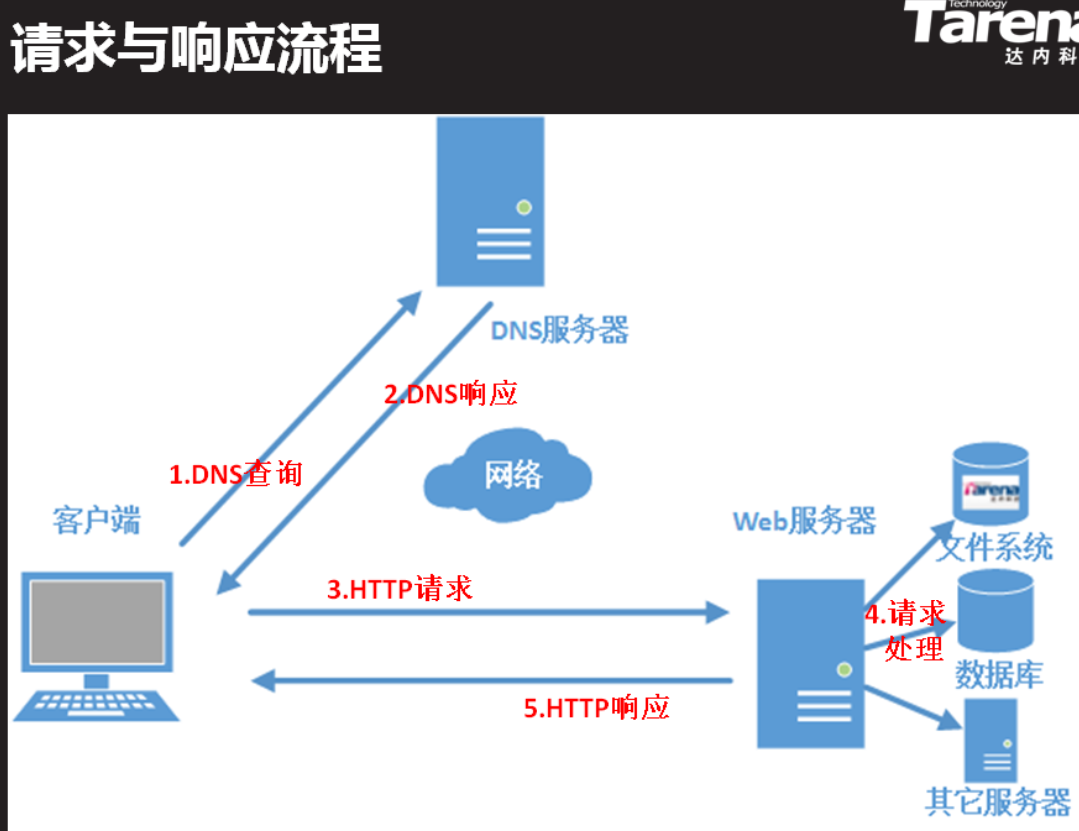
|  |
| --- |
| get/delete 请求使用的查询字符串 |

⑧<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 |
| http协议历史  http/1.1 支持持久连接 |

3.web请求原理的详细解释



4.消息/报文 Message

|  |
| --- |
| 1.请求消息 Request Message  2.响应消息 Response Message |

5.请求消息 Request Message

①请求起始行

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| GET / HTTP/1.1  Host: www.tmooc.cn  请求方法   |  |  | | --- | --- | | http原生的请求方法 | restful 请求方法 | | get 明文传参，使用查询字符串  上限2kb  向服务器要数据的时候使用  req.query | get 所有的查询都使用get ---select  没有请求主体  req.params | | post 隐式传参，使用请求主体formdata  把数据传递给服务器的时候  req.body | post 所有的新增 insert  有请求主体  req.body | | put 往服务器上放资源的时候使用 | put 所有的修改 update  有请求主体  req.body | | delete 从服务器上删除资源的使用 | delete 所有的删除 delete  没有请求主体  req.params | | option 预请求 |  |   协议版本号 http/1.1  请求的url |

②请求头信息

|  |
| --- |
| Host: www.tmooc.cn 浏览器告诉服务器，要访问的主机是哪一个  Connection: keep-alive 浏览器告诉服务器，请开启持久连接  User-Agent: 浏览器告诉服务器，我自己的信息  Accept-Encoding: gzip 浏览器告诉服务器，我能够接收压缩文件类型  Accept-Language: zh-CN 浏览器告诉服务器，我能够接收的自然语言的类型  Referer: http://www.tmooc.cn/ 浏览器告诉服务器，当前的请求来自于哪个页面 |

③请求主体

|  |
| --- |
| formdata |

6.响应消息 Response Message

①响应起始行

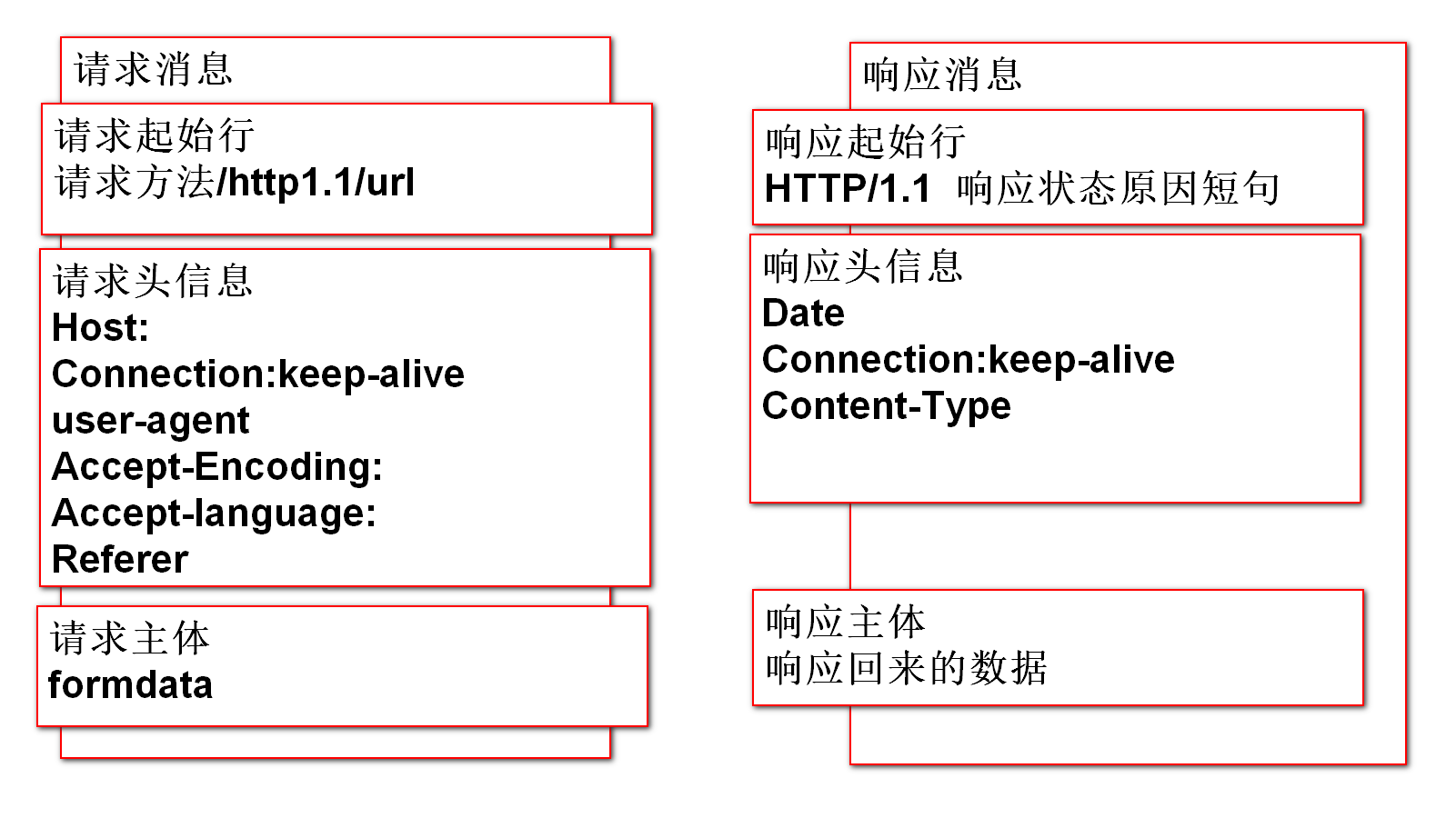
|  |
| --- |
| HTTP/1.1  响应状态200 404 500  原因短句 OK 对响应状态码的简短解释 |

②响应头信息

|  |
| --- |
| Date: Tue, 29 Sep 2020 06:06:56 GMT 响应的具体时间  Connection: keep-alive 服务器告诉浏览器，已经开启持久连接  Content-Type: text/html 服务器告诉浏览器，响应主体的格式  text/css  application/javascript  image/png  application/json json格式的响应  application/xml xml格式的响应  text/plain 普通文本 |

③响应主体

|  |
| --- |
| 真正响应回来的数据 |



6.缓存

|  |
| --- |
| ①客户端将服务器传递回来的响应数据，自动的在内存中保存一份  ②下一次客户端访问相同数据的时候，直接从内存中拿数据，不发送请求了  缓存优点  1.减少了冗rong余数据传输，节省客户端浏览  2.降低服务器资源损耗  3.在网络不好的时候，提升用户体验 |

7.安全的http协议https

|  |
| --- |
| HTTPS:安全版本的http协议  SSL：为数据通信提供安全支持  1.客户端发送请求-->SSL层加密-->服务器接收到加密文件-->SSL解密  2服务器发送响应-->SSL层加密-->客户端接收到加密文件-->SSL解密 |

二.DOM(简易的dom)

1.在ajax之前学dom原因

|  |
| --- |
| ajax负责发送请求，并接收响应  之前使用form提交请求  1.form可以发请求  2.form可以自动的收集整理用户填写的数据  使用ajax就不用form，就失去了自动收集整理数据的能力  我们需要使用dom手写代码收集数据  form表单提交数据，要求控件必须有name属性，  不使用form提交数据，控件就不要name属性，用id属性  特殊：单选按钮和多选按钮，要使用name属性---->做分组  总结：单选按钮和多选按钮，同时要有name和id属性  其它的控件，不要用name，用id |

2.DOM介绍

|  |
| --- |
| js包括3部分内容  1.js核心 ECMA script  2.DOM document object model 文档对象模型  把当前html页面，看成一个document的对象  document调用方法，获取内部元素的对象  通过这个元素对象，获取、修改用户填写的值/内容  3.BOM Browser object model 浏览器对象模型 |

3.操作dom的步骤

|  |
| --- |
| 1.写事件 (onclick="")  2.通过dom命令获取元素对象  var obj=document.getElementById("元素id");  3.通过元素对象，操作元素属性/值  ①获取/修改input对象的值 value  var r=obj.value;获取用户填入的值  obj.value="abc";修改input在页面中显示的值  ②获取/修改双标签的内容区域 innerHTML |

02\_ex.html

|  |
| --- |
| 点击按钮之后，把用户输入文本框中的数据，显示在div中 |

4.事件

|  |
| --- |
| 通过用户的操作，而调用js 方法  onclick="" 单击事件  onfocus="" 获取焦点事件  onblur="" 失去焦点事件  onload="" |

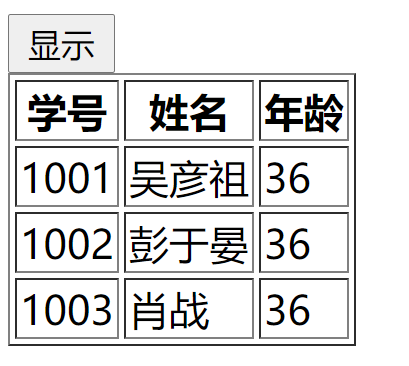
5.innerHTML详解

|  |
| --- |
| 由于双标签可以在内容区域嵌套其它标签  所有双标签对象的innerHTML="html结构"  d1.innerHTML=`<b><i>${n1.value}</i></b>`; |

作业

1.保证get方法登录熟练,各个文件之间的关系，app.js user.js public pool.js

2.使用dom完成



3.自己创建数组数据，完成下面表格



4.