day05

|  |  |
| --- | --- |
| HTTP原生的请求方法 | Restful API 规则 |
| get:客户端向服务器要数据  明文，有大小限制2KB  使用查询字符串传递明文  req.query | get:查询 select  无请求主体  req.params |
| post：客户端要传递给服务器数据的时候，使用  隐式，没有大小限制  使用请求主体传递参数  formdata就是请求主体  req.body | post:新建 insert  有请求主体  req.body |
| put:客户端往服务器放文件的时候使用，一般被禁用 | put:修改 update  有请求主体  req.body |
| delete:客户端从服务器删除文件的时候使用，一般被禁用 | delete：删除  无请求主体  req.params |

二.DOM操作

2.使用js获取元素

|  |
| --- |
| <div id="d1"></div>  var $div=document.getElementById("d1")  ES6对通过id获取元素对象，做了简化  在js中，使用id值直接代表对象  <input type="text" id="uname">  <div id="d1"></div>  d1.innerHTML=uname.value; |

写dom步骤总结

|  |
| --- |
| 1.写事件  2.写事件中的方法  3.方法中获取元素对象  4.修改对象的属性 |

3.input对象的值

|  |
| --- |
| var msg=对象.value;获取值  对象.value="...." 修改值 |

4.双标签的内容

|  |
| --- |
| var msg=对象.innerHTML; 获取  对象.innerHTML="..." 修改  使用  对象.innerHTML="<a></a>" 来动态的修改页面布局 |

5.事件

|  |
| --- |
| 通过用户的行为来激发的操作  onclick="fn()" 用户单击之后，fn()被调用  onblur="fn()" 焦点移除  onfocus="fn()" 获取焦点 |

练习

|  |
| --- |
| 03\_ex.html 11:00~11:15休息  1.文本框---用户名：uname  2.密码框--密码：upwd  3.密码框--重复密码：cpwd  上面三个input，每个后面都有一个span msg1,msg2,msg3  需求：  1.用户名，获取焦点时，msg1提示用户名6~12  失去焦点时，判断用户名是否为空，并msg1提示  判断用户名格式是否正确，msg1提示  2.密码框获取焦点，msg2提示密码6~8位  失去焦点，非空判断，格式判断，msg2中给提示  3.重复密码框 失去焦点时，非空判断， 密码相同的判断，msg3 |

三.Ajax

1.同步Synchronous

|  |
| --- |
| 在一个任务执行的过程中，不能执行其他任务  一次只能做一件事  同步访问：浏览器在向服务器发送请求的过程中，浏览器只能等待服务器的响应，不能做别的事  出现的场合  1.地址栏输入url  2.a标签跳转  3.form表单提交 |

2.异步Asynchronous

|  |
| --- |
| 一个任务进行的过程中，不影响其他任务  异步访问：浏览器在发送请求的时候，用户可以在页面做其他操作  出现场合  1.用户名重复的验证 codeboy  2.聊天室  3.股票走势图  4.百度的搜索建议 |

3.什么是ajax

|  |
| --- |
| Asynchronous JavaScript and XML  异步的 js 和 XML  本质：使用js提供的异步对象，向服务器发送请求，  并接收服务器响应回来的数据  异步对象：XMLHttpRequest |

4.使用ajax

|  |
| --- |
| 1.创建异步对象  2.创建请求  3.发送请求  4.接收响应 |

①创建异步对象

|  |  |
| --- | --- |
| var xhr=new XMLHttpRequest();  上面代码不支持IE8.0以下的版本  如果想支持ie8以下(以下代码了解即可，不需要记忆)   |  | | --- | | var xhr="";  //如果你的浏览器有XMLHttpRequest属性，  //就说明，你的浏览器是ie8以上的版本  if(window.XMLHttpRequest){  xhr=new XMLHttpRequest();  }else{  xhr=new ActiveXObject("Microsoft.XMLHttp");  } | |

②创建请求

|  |
| --- |
| xhr.open(method,url,isAsyn);  method:请求方法，必须是string类型  url:请求的接口，必须是string类型  isAsyn：是否使用异步访问 true/false |

③发送请求

|  |
| --- |
| xhr.send(请求主体);  如果使用的请求方法，没有请求主体。  那么参数就可以不写，或者写null  xhr.send(null)/ xhr.send(); |

④接收服务器的响应

|  |
| --- |
| 1.xhr.readyState属性  有4个值 0 请求尚未初始化  1 已经打开服务器连接，正在发送请求  2 开始接收响应头消息  3 开始接收响应主体  4 响应数据接收完毕  只要请求一创建，readyState的值就自动开始改变  2.xhr.status属性，服务器响应状态码  当值为200的时候，表示服务器响应传输成功  3.xhr.onreadystatechange=function(){}  绑定监听，监听xhr.readyState的值的改变  function会执行4次。  xhr.readyState==4是我们需要的  同时，还要求xhr.status==200  4.xhr.responseText属性  响应数据就保存在这个属性中 |

练习

|  |
| --- |
| 04\_ex.html---->创建在静态托管文件夹中  按钮，点击按钮发送请求  得到响应后，把响应放在按钮之前的div中显示  创建一个接口ajax.js中 "/ajaxdemo1"  响应一句话"这是我第一个自己完成ajax" 17：05~17：15休息 |

5.使用http的get方法带参数发送请求

常见错误

1.跨域错误

|  |
| --- |
| 你使用本地打开的html网页  而不是使用靠服务器打开的网页  127.0.0.1/04\_ajax.html  <http://127.0.0.1:8080/04_ajax.html> |

2.方法名未定义

|  |
| --- |
| 事件中调用的方法名和你定义的方法名是否一致 |

3.出乎意料的结尾

|  |
| --- |
| 结尾少括号 } ) |

4.数据库没有开启

|  |
| --- |
|  |

作业：

1.先把课上的登录模块写完

2.重新写一遍http的get方法的登录

3.提高题 模仿学子商城注册页，使用失去焦点事件

做用户名的注册验证

4.使用get方法，获取用户列表，把获得的响应放在div中显示