

##HTML

##HTML

*HTML:超文本标记语言

*乱码问题:文本和内容编码不一致

*作用

制作静态网页

*使用地方

在设计网站的页面的时候使用到HTML

*使用

HTML的文件的创建

先创建TXT文件,然后改后缀名

一个HTML文件,扩展名为.html或者.htm结尾

*结构标签

*根标签:<html></html>

*<head></head> 头标签

*<body></body> 体标签

*HTML的字体标签

 :字体标签

文字

标签的属性:

<标签 属性名="属性值" 属性名="属性值"></标签>

color属性:字体颜色

英文单词设置:black,red

使用16进制数设置#FF0000 红色 #00FF00 绿色 #0000FF 蓝色

size属性:字体的大小

值是1-7

face属性:字体的字体

*HTML的排版标签

*标题标签 h标签 <h1>...<h6>

*段落标签 p标签

*粗体标签 b标签

*斜体标签 i标签

*下划线标签 u标签

*居中标签 center标签(已过时)

*标签的嵌套

换行

<hr>横线

*HTML标准

HTML 4

HTML 5 <!DOCTYPE html> 是对Javascript.CSS的增强

<pre> 原生标签

##网站图片页面显示

*HTML图片标签

*图片标签

*属性:

* src :图片的来源.

* width :图片的宽度. 100px px代表像素点

* height:图片的高度.

* alt :图片找不到显示的内容.

*图片引入的路径

*路径:相对路径

- *如果一如的图片和html文件是同一级路径
直接写文件名或者./文件名
- *如果引入的图片是html的上一级路径,..../文件名
- *如果引入的图片是html的上两级路径,..../文件名
- *如果引入的图片是html的下一级路径./文件名

##网站列表页面的显示

*HTML的列表标签

*有序列表

```
<ol>  
  <li></li>  
</ol>
```

*无序列表

```
<ul>  
  <li></li>  
</ul>
```

<ul style="list-style: none;">去掉小圆点

*HTML的超链接标签

*HTML的超链接标签:<a>

*属性:

*href:链接的路径

*target:打开的方式

*_self:在自身页面打开

*_blank:新打开一个窗口

*无序列表的属性

*type的属性

*disc:实心点

*circle:空心圆

*square:方块

有序列表的属性:

* type属性:

* 1 : 数字类型

* a : 英文类型

* i : 罗马字符

超链接的target属性:

*_self:本页面打开

*_blank: 在一个新的窗口打开

*_parent:

##HTML的表单标签

*为什么要加name属性:为了传递参数

*HTML的表单标签: <form>

* 常用属性:

* action属性:提交的路径.默认提交到当前页面

* method属性:请求的方式.GET和POST.默认是GET.

* get和post区别

* get:会显示在地址栏上,有大小限制

* post:不会显示在地址栏上,没有大小限制

* post更加安全

* post没有大小限制

*HTML中表单元素:

*<input type="text"> 文本框

*常用属性:

*name :表单元素的名称,必须有name属性,然后后台才可以接收数据

*value:文本框的默认值

*size:文本框的长度

*maxlength:文本框输入的最大长度

*readonly:只读文本框

*<input type="password"> 密码框

*

*<input type="radio"> 单选按钮

*<input type="CheckBox"> 复选按钮

*<input type="button"> 普通按钮:没有任何功能的按钮

*<input type="submit"> 提交按钮

*<input type="reset"> 重置按钮

*<input type="file"> 文件上传的表单项

*<input type="hidden"> 隐藏字段(类似于文本框,可以上传,但是不能看到, 一般储藏ID)

*<input type="image"> 图片按钮(把图片设置成链接)

*<select> 下拉列表

*<textarea> 文本域

*html的框架标签

*框架标签:

<frameset>,标签与body标签是冲突,有frameset就没有body.

*属性:

rows

cols

*使用<frame>标签,frame代表切分的每个部分.

*特殊字符的标签

*空格:

*大于:>

*小于:<

*常用标签

<div><a><div>

*div 是html的块标签

块级元素:独占一行,可以修改高度和宽度 div

块级元素转行内元素添加属性display:inline;

行内元素:不独占一行,不可以修改高度和宽度 span i u

行内元素转块级元素添加属性display:block;

<div></div>默认一个div独占一行

默认在同一行

*table和div布局的区别

div更加灵活,并且解析加载速度快

*css概述

*html相当于网站的骨架!css对骨架进行美化!

*什么是css:

层叠样式表(Cascading Style Sheets)

样式定义如何显示html元素

*css的使用地方

任何网站都会使用css去美化页面

*如何使用css

*css 的基本语法:

一个选择器,一个声明

选择器(属性:属性值,属性:属性值,)

*css的引入方式

行内样式:直接在html元素上使用style属性设置css(只有某个标签自己使用)

```
<h1 style="color:red;font-size:200px;">标题</h1>
```

页面内样式:在html的<head>标签中使用<style>标签设置css

```
<style>
    h1 {
        color:blue;
        font-size: 40px;
    }
</style>
```

外部样式:单独定义一个.css文件,在html中引入该.css文件

```
<link href="../../css/demo1.css" rel="stylesheet" type="text/css" />
```

*css的选择器

*元素选择器(标签选择器):

*ID选择器

*类选择器

*选择器的优先级:

ID选择器>类选择器>标签选择器

*css的浮动

为了让某些标签(块级元素,div,li)横着显示,所以使用浮动

*使用float属性完成div的浮动

*常用left,right

*清除浮动 clear:both

1.在浮动的元素后面添加<div style = "clear:both"> 不建议使用

2.给浮动元素的父元素添加一个样式:clearfix

*CSS中的继承问题

在css中,宽度可以继承,高度不可以

浮动以后的元素是不会继承宽度的

*居中

margin:0 auto;

*函数(百分数和px之间运算关系)

calc(70% - 2px) -号两端要有空格

```
<div style="width: 100%;height: 300px;background-color: aqua;">
  <div class="f-left" style="width: 30%-2px;border: 1px solid black;
    background-color: red;height: 100%;"></div>
  <div class="f-left" style="width: 70%-2px;border: 1px solid black;
    background-color: hotpink;height: 100%;"></div>
</div>
```

*选择器

*属性选择器

*后代选择器(子孙元素选择器)

div span 查找所有div中所有的span元素

*子代选择器

div>span(选择第一个span)

*并列选择器

选择器, 选择器 {}

*盒子模型

内边距:padding(顺时针)

外边距:margin

宽度计算方式:padding-left+padding-right+border-left+border-right+content

*定位

*position属性设置定位

relative:相对定位,性低于父标签

absolute:绝对定位,是相对于body

使用的时候:子绝父相(里面是绝对定位)

使用另外两个属性:left,top

*fixed:广告制作,滚动时其他滚动,这个固定.

*伪类

*link visited hover active 顺序不能乱,href必须有值

*样式总结

背景

背景颜色 background-color:red;

背景图片 background-image:url(xx.png);

背景重复 background-repeat:no-repeat;

背景位置 background-position: center center;

背景缩写 background : red url(x.png) no-repeat center center;

背景大小 background-size:

边框

border : 1px solid black;

文本/字体

大小 font-size : 12px;

颜色 color : red;

去除超链接下划线 text-decoration : none;

文本上下居中 line-height:100px;height:100px;/height==line-height

文本左右居中 text-align:center;

无序列表:去除小圆点

<ul style="list-style:none">

外边距:margin

内边距:padding

#javascript

*概述:

是运行在浏览器端的脚本语言

*组成:

ECMAScript DOM BOM

ECMAScript: JavaScript的基本的语法

BOM: Browser Object Model :

DOM: Document Object Model :

*其他的脚本语言:

JavaScript,ActionScript,Flex

*用途:

是页面跟家丰富,使页面动起来

*基本语法:

*区分大小写

*是一种弱语言 (Java是强语言)

*分号可有可无

*变量命名和Java一样

*注释和代码块与Java一样

*数据类型

*JS将数据类型分成两类

原始类型:

undefined:未定义类型

boolean:布尔类型

number:数字类型

string:字符或字符串

null:空

引用类型:

对象类型,默认值是null

*JS运算符

JS中运算符和Java基本一致

JS中有个===全等于,是类型和值都一致,才为true

*语句和Java一致

*JS通常开发的步骤

JS通常由一个事件触发

出发一个函数,定义一个函数

获取操作对象的控制权

修改要操作的对象的属性或值

*定义函数:

```
function 函数名称(){}
```

```
window.onload = function(){};
```

*常用事件:onclick ondblclick onmouseover onmouseout(onmouseenter onmouseleave)

onload:html文本加载完毕后,加载资源请求,加载资源请求完毕后,加载onload

*JS引入方式

*通常两种方式:

一种: 页面内直接编写JS代码,JS代码需要使用<script></script>.

二种: 将JS的代码编写到一个.js的文件中,在HTML中引入该JS代码即可.

*正则的匹配:

*JS中有两种匹配正则的方式:

* 使用String对象中的match方法.

* 使用正则对象中的test方法.

*HTML的window对象

*设置定时

*setInterval("JS代码",时间):每隔多少毫秒执行某个表达式

*setTimeout:过多少毫秒执行下一个表达式

*清除定时

*clearInterval();

*clearTimeout();

*css的显示和隐藏两种方式

display:block,none 隐藏后不占位

visibility:visible,hidden 隐藏后占位

*基本语法

定义变量 var 变量名 = 值;

变量名: 局部变量(定义在方法内部的变量)我们以_开头。

给变量赋予默认的值

如果是基本类型 var num = undefined;

如果是引用类型 var person = null;

在控制台打印变量 console.log(变量名);

定义对象 var person = {

name:"张三",

age:14,

gender:'女'

}

定义方法(函数) function 方法名(){}

function 方法名(name,id){}

function (){} //匿名方法, 当作为方法的实际参数的时候使用

方法的调用

方法名();

null和undefined报错问题

Cannot read property 'name' of undefined 原因是调用了undefined.name

Cannot read property 'name' of null 原因是调用了null.name

*debug:

- 1.打开控制台，查看console是否有错误信息
- 2.如果有错误信息，可以直接定位哪里有问题
- 3.如果没有错误信息，我们需要打开source，查看源代码，加断点
- 4.找事件，根据事件名字找到对应的方法 比如 onload = "show()" 那我们就可以找到方法是show方法
- 5.在show方法的第一行加断点
- 6.仔细观察代码是走到哪一行报错的，并记录下来
- 7.通过watch查看出问题的那一行的变量

【JS 的输出】

```
* document.getElementById("").innerHTML="HTML的代码";
* document.write("");
```

【JS 的事件】

```
* onfocus :获得焦点.
* onblur :失去焦点.
* onsubmit :提交的时候.
```

*onchange:下拉列表改变事件

*ondblclick:双击

*键盘事件

*onkeyup:

*onkeydown:

*onkeypress:

*鼠标事件

*onmousemove:

*onmouseout:

*onmouseover:

*onmouseup:

*onmousedown:

*DOM

*概述

Document Object model:文档对象模型

将一个HTML的文档加载带内存形成一个树形结构,从而操作树形结构改变HTML的样子

*DOM的使用

知道document,element,attribute的属性和方法

*常用操作

获得元素:

```
> 获得元素:
* document.getElementById(); -- 通过 ID 获得元素.
* document.getElementsByName(); -- 通过 name 属性获得元素.
* document.getElementsByTagName(); -- 通过标签名获得元素.
> 创建元素:
* document.createElement(); -- 创建元素
* document.createTextNode(); -- 创建文本
> 添加节点:
* element.appendChild(); -- 在最后添加一个节点.
* element.insertBefore(); -- 在某个元素之前插入.
> 删除节点:
* element.removeChild(); -- 删除元素
```

所有的标签都是Element对象

所有的属性是Attribute

所有的文本的内容是Text

Document,Element,Attribute,Text统称为Node

*事件名称不加on;事件处理加on.

#jQuery

*概述

是一个JS的框架,对传统的JS,对传统的JS进行了封装.

现在企业的开发中不会使用传统的JS进行开发,通常对使用JS框架进行开发

*JS常用框架

jQuery, ExtJs, DWR, Prototype...

*JQ的使用

学习jQuery的语法

*jQuery介绍

工具类 EXTJS

*jQuery的入门

首先要引入js

*JQ的显示和隐藏

*JQ的效果操作

*show();

使用一:JQ对象.show();

使用二:JQ对象.show("slow");

使用三:JQ对象.show(毫秒值);

使用四:JQ对象.show(毫秒值,function(){});

*hide();

*slideDown();向上滑动

*slideUp();向下滑动

*fadeIn();淡入

*fadeOut();淡出

*animate();自定义动画

*toggle:单机切换函数

JQ对象.toggle

*JQ选择器

*基本选择器

*id选择器:

`$("#id")`

*类选择器

`$(".class")`

*元素选择器

`$("元素")`

*通配符选择器

`$("*")`

*并列选择器

`$("#id, class")`

*层级选择器

*后代选择器:使用空格

*自元素选择器:使用>

*下一个元素:使用+

*兄弟元素:使用~

*基本过滤选择器

*prop:非自定义属性,可以获取所有的属性值

*attr:自定义属性,可以获取所有的属性值

*val():可以获取,设置对象的value值

#jQuery

*常用事件

事件

```
blur([[data],fn))
change([[data],fn))
click([[data],fn))
dblclick([[data],fn))
error([[data],fn))
focus([[data],fn))
focusin([data],fn)
focusout([data],fn)
keydown([[data],fn))
keypress([[data],fn))
keyup([[data],fn))
mousedown([[data],fn))
mouseenter([[data],fn))
mouseleave([[data],fn))
mousemove([[data],fn))
mouseout([[data],fn))
mouseover([[data],fn))
mouseup([[data],fn))
resize([[data],fn))
scroll([[data],fn))
select([[data],fn))
submit([[data],fn))
unload([[data],fn))
```

*

*jQuery的查找

```
children();
find();
parent();
```

*JQ时间的处理

trigger和triggerhandler的区别:

trigger: 支持事件冒泡

triggerhandler: 不支持

#Bootstrap:前端响应式页面

*一套能够在不同设备上能够通用的网页，不影响正常浏览的方式。

*概述

是目前很受欢迎的前端框架。

*使用场景

bs设计出响应式页面，由他设计网页

*全局CSS

bootstrap的框架提供了一系列CSS样式，样式可以直接使用

*布局容器:

container: 根据设备给定宽度

container-fluid: 类似于100%的宽度，占据全部视口（viewport）的容器

*栅格系统:

栅格样式:

* 设备的分辨率大于 1200 使用lg样式

* 设备的分辨率大于 992 < 1200 使用md样式

* 设备的分辨率大于768 < 992 使用sm样式

* 设备的分辨率小于 768 使用xs样式

将一行分成12列.定义div元素 样式的和 加一起等于12 即可.