--------------------第一天 HTML -------------------------

Ie trident

sarfari webkit

Firefox Gecko

Chrome blink

Opera blink

--------------------第二天 HTML -------------------------

**1.表格 table**

**格式：**

<table>

<caption>标题</caption>

<thead>

<tr>

<th></th>

</tr>

</thead>

<tbody>

<tr>

<td></td>

</tr>

</tbody>

</table>

**属性：**

表格边框（默认无边框）

<table border = "1">

单元格与大表格边框的距离

<tr cellspacing = "0">

控制单元格内部文字与单元格边框的距离

<tr cellpadding = "10"

控制表格的位置

<table align = "center/left/right">

**合并单元格：**

**注意：**

不能跨thead tbody合并

th 不能和 td 合并

不能同时跨行 跨列

从上到下 从左到右 合并

跨行合并 rowspan

跨列合并 colspan

1. **表单元素**

input标签

属性：

radio 单选框

checkbox 复选框

设置name相同

默认选中 checked

按钮：

普通按钮：button

提交按钮：submit

重置按钮：reset

图像按钮：image

文件按钮：file

属性：

maxlength : 可以输入的字符个数

文本框的值 ：value

label 标签 包裹 input标签 点击 label input 被选中

<label><input type=”text”/></label>

label 包裹两个 input 标签 处理方法

<label for=”this”> <input /><input id=”this” /></label>

下拉菜单

<select>

<option></option>

<option></option>

</select>

默认选中属性：selected=”selected”

**form**

get 方法 通过浏览器地址栏进行提交

Post 通过浏览器后台提交

HTML5

标签：

header 头部

footer 定义底部

nav 定义导航栏

article 定义文章

section 定义区域

aside 定义侧边

datalist:

<input type=”text” list=”#id名”/>

<datalist id=”id名”>

<option></option>

<option></option>

<option></option>

</datalist>

<fieldset>

<legend></legend>

</fieldset>

H5新增input type属性：

Email 输入电子邮箱

Tel 输入手机号

Number 输入数字

Url 输入网址

Search

Range 滑块

Time

Date

Month

Week

Color

新增属性：

Placeholder 占位符

Autofocus 自动获得焦点(百度搜索框)

Multiple 上传多个文件

Autocomplete 启动自动完成功能 form提交之后、 必须有name

Required 输入框不能为空

Accesskey=“s” 设置快捷获取焦点

新增标签：

Audio

<audio autoplay loop controls>

<source src=””/>

<source src=””/>

</audio>

------------------------------第三天 css-------------------------

Css：层叠样式表

font-weight:

normal : 默认值，浏览器会显示标准的字体样式。

Font-style: italic

Font：斜体 加粗 大小 字体 (必须遵循该顺序)

可以省略前两个 后两个一定要写

伪类选择器：

1. 链接伪类选择器：

Link 没有点击过的样式

Visited 点击过的样式

hover

Active 鼠标按下不松开的样式

Css3 结构伪类选择器

:first-child

:last-child

:nth-child(2)

:nth-last-child(2)

Text-indent：2em 段落首缩进 1em= 一个汉字的大小

Letter-spacing 字间距（中英文都可以）

Word-spacing 单词间距

Text-shadow:水平距离 垂直距离 模糊 颜色

--------------------- 第四天 css：---------------------------

块级元素：

<div> <p> <ul> <li><h1>

独占一行

可以设置宽高

默认宽度是父元素的

可以容纳其他块元素

文字类的块级元素内部不能嵌套 块级元素（p h1 h2 h3 h4 h5）

行内元素：

和相邻行内元素在一行

不能设置宽高

宽高是内容的宽高

只能容纳文本和其他行内元素

行内块元素

<input /> <img /> <td>

可以同行显示 ，会有一段小的距离

可以设置宽高

默认是大小是内容大小

行内元素 块级元素 行内块元素之间的相互转换

Display：inline

Display：block

Display：inline-block

交集选择器

Div.class

并集选择器（分组选择器）

p,h1,div {}

后代选择器

选择器 空格 子孙

子元素选择器 只选择 儿子 不选 孙子

父元素 > 子元素

孙子元素会继承 文字的属性（颜色 大小）

属性选择器

选择器[属性名]

选择器[属性名=“属性值”]

[^=] 在开头

[$=] 在结尾

[\*=] 有就可以

伪元素选择器

设置第一个字的样式

：：first-letter{

}

设置第一行的样式

：：first-line{

}

设置文字被选中的样式

：：selection{

}

在之前加入

:: before{

Content：内容

}

在之后加入

::after{

Content：内容

}

背景颜色 图片

Background-color：

Background-repeat：repeat-x repeat-y no-repeat

background-img：

background-position：left top right bottom center

Background-attachment：scroll fixed

简写

Background：color url repeat fixed position

Background：rgba（0，0,0，0.5）

背景缩放

background-size：100px；

background-size：cover；等比缩放 溢出部分隐藏

Background-size：contain；等比缩放 宽度或高度任何一个到达盒子边 不再缩放

Text-decoration： underline none

----------------- 第五天 css--------------------

层叠性：相同的选择器 相同的样式属性 相同的权重。

继承性：text- font- line- color 与文字样式有关的一般都可以继承

与盒子有关的属性 width height 内外边距是不可以继承的

优先级（权重）

！important > 行内样式 > id > class > 标签 > 继承样式 /通配符选择器

无穷 1,0,0,0 0,1,0,0 0,0,1,0 0,0,0,1

a:hover 0,0,1,1 (有类 有标签)

ul li 0,0,0,2

继承的权重是 0

border-color: red；

border-width: 1px;

Border-style: solid dashed虚线 double dotted点线

border ：solid #red 1px；

table-collapse：collapse；

Fw：移动工具v

按住空格 移动

行内元素 不要指定上下的内外边距。

外边距合并：垂直相邻的两个盒子 分别设置margin 中间的margin大小不变 按照大的值显示

避免方式 给一个盒子设置外边距

嵌套的两个盒子 垂直向同一方向添加外边距问题 取最大值

解决方案：给父盒子设置1px border

设置1px padding

给父盒子设置 overflow：hidden

块级元素不设置宽度高度 添加padding不会增加盒子的大小

-------------------第六天 css--------------------------

宽度剩余法： 不用padding 直接给div 一个宽度 把空白位置的宽度算进去。

Box-sizing：content-box；

box-sizing： border-box；

Box-shadow： 水平位置 垂直位置 模糊距离 阴影大小 阴影颜色 内外阴影 inset

标准流 正常的显示样式 没有浮动和 position 定位的。

转换为行内块元素可以将块元素变为在一行显示 但是有空隙。

Float 浮动

浮动的元素 不占位置 脱离标准流。

为了防止浮动元素覆盖：

<div>

<div>浮动</div>

<div>浮动</div>

</div>

浮动元素不会 影响父元素的 padding border

两个都浮动 并排显示

第一个浮动第二个不浮动 第二个占据第一个的位置

第一个不浮动 第二个浮动 第二个与第一个底边对齐

行内元素/块级元素 添加浮动之后 会有一个显示模式变成行内块的操作。

标准流父元素的高度 会被孩子 的高度撑起来

浮动了 孩子不能撑起父元素的高度 下面的元素会移动位置。

清除浮动：：：：：

1.在浮动的标签下面添加一个空标签

<div>浮动的盒子</div>

<div class=”clear”>空标签</div>

选择器 {clear：both；}

缺点： 增加无意义标签

1. 给浮动的元素的 父元素添加 overflow： hidden；

触发BFC

优点： 代码简洁

缺点：

1. after伪元素清除

谁浮动给 浮动元素的父元素

Clearfix：after {

Content：”.”;

Height:0;

Visibility:hidden;

Display:block;

Clear:both;

}

兼容Ie 6-7

Clearfix {

\*zoom:1;

}

原理跟添加额外标签一样。

1. after before
2. .Clearfix：after，.clearfix：before {

Content：””;

Display:table;

}

.Clearfix:after {

Clear:both;

}

.clearfix {

\*zoom :1;

}

----------------------第七天 css----------------------------

----------------------第八天 css----------------------------

Position

定位模式 Static（默认） 可以去除定位 不能设置边偏移。

相对定位 相对于 他本来的标准流位置的左上角位置移动。不脱离标准流 还占有原来的位置。

absolute绝对定位 相对于定义了 定位的 最近的父元素 移动。脱离标准流，在标准流不占位置。（变成行内快） 随滚动条 滚动

（在父元素没有定位的时候 如果浏览器有滚动条 子元素绝对定位 会跟着滚动条移动。）

Fixed 固定定位

脱离标准流 不会随滚动条的滚动而滚动 （转换为行内快）

边偏移

子元素 绝对定位 父元素 相对定位 ：

如果父元素设置绝对定位 脱离标准流 影响布局。

z-index： 只有设置了 定位的才有

设置了 定位的 比浮动 或者标准流的高。

浮动 和 固定定位 绝对定位 都能进行显示模式的转换 转换成 inlin block

Display visibility

Display： none； 隐藏不占位置。

Visibility： hidden； 占位置

Overflow ： hidden 移除隐藏

----------------------第九天 css----------------------------

Cursor ：default（默认白色箭头） pointer 手 move 十字 text

Outline： 0；取消轮廓线

防止拖拽文本框 resize：none；

图片和文字的 居中对齐 （默认是基线对齐）

Vertical-align: Baseline 基线。

Vertical-align：middle 中线对齐 。 （行内块）

vertical-align: bottom

因为默认是 基线对齐 图片与父元素之间有一个白色的间隙 。

解决办法：不再基线对齐 改为 其他的对齐方式 / 图片改为块元素

Word-break：Break-all 允许拆开单词显示 / Keep-all 不允许拆开

White-space: normal;

White-space: nowrap; 强制同行显示

超出部分 ... 显示的方法：先添加强制一行 white-space：nowrap；overflow:hidden；text-overflow: ellipsis /clip

设置盒子的大小。

精灵图的position 都是负值

特殊字体

下载字体 引入样式 @font-face‘

使用伪元素添加

伪类 类 选择器 选择的是对象

伪元素是插入一个元素 行内的盒子

动画效果

Transition： width 1s ease 2s

----------------------第九天 css----------------------------

Css 变形

Transform ： translate（x，y）

： translateX（x）

： translateY （y）

和 transition 搭配使用。

Transform ：Translate（50%） 为百分比的时候 是以本身的大小

Transform：scale（2）如果只写一个参数 宽度高度 都缩放

Transform ： rotate （xdeg） 更改旋转的中心点： transform： origin （x，y）

Transform ： skew（deg）x: 正值向左倾斜 负值向右倾斜

3d

Transform ： rotateX rotateY rotateZ

视距 perspective 视距越小 效果越明显 给父元素

谁做动画谁加过度。

Backface-visibility：hidden；

Animation ：name 2s；

@keframes name {

}

伸缩布局

给父元素一个 display ：flex

子元素 一个 flex： 1；

min-widht: 最小值

Max-width： 最大值

有固定宽度的盒子 要用 总的宽度 减去 固定的宽度 剩下的 宽度 再给其他盒子平均划分。

垂直排列 flex-direction： column

水平排列 flex-direction： row

Seo 搜索引擎优化





BFC

一块特殊的渲染区域 ， 区域内的盒子 不会受到外部的影响 也不会影响外部盒子。

正常的元素是按照浏览器规范统一渲染的，只有触发了BFC条件的才独立渲染。

Block list-item table 具有 BFC

触发BFC属性 float：不为none 固定定位和绝对定位

Overflow：hidden 不为默认

Javascript第1天

在html中获取的任何数据都是字符串 包括写的任何数据

最初目的：处理表单的验证操作

Javascript 组成

ECMAscript: ECMA:欧洲计算机制造联合会

DOM:文档（document）对象（object）模型（model）

BOM:浏览器（briwser）对象（object）模型（model）

路径 ./ 当前目录。

1. 相对路径

- ./assets/bootstrap.css

- ../assets/bootstrap.css

- assets/bootstrap.css

- 网页中使用（a,link,img,script,location）

- 代码读写服务端文件时可以用

2. 绝对路径

- /assets/bootstrap.css

- 网页中使用（a,link,img,script,location）

- 切记不能在文件读写时使用

3. 物理路径

- D:\itbetter\03-server-side\archives\day-08\code\songs\bootstrap.css

- 不可能在网页中使用（a,link,img,script,location）

- 代码读写服务端文件时可以用

使用外链引入js文件 写在script 标签中的内容会被忽略。

Script标签 写在 body闭合标签的上面。

变量是计算机内存中存储数据的标识符。根据变量名可以获取内存中存储的数据。

当声明了一个变量没有给值 值是默认的undefined。

声明变量注意事项：

尽量不要声明没有值得变量。

尽量把变量放在script标签的最上面。

声明多个变量使用 var name = ”lm” , age = 20 ;

变量的命名规范：

由 字母 数字 下划线 $ 组成 但是不能由数字开头。

不能使关键字和保留字：

关键字：编程语言中规定不能使用的字符。

保留字：编程语言未来要使用的的特殊字符。

命名要有意义。

驼峰命名法 首个单词的首字母小写后面每个单词的首字母大写。

数据类型：

简单数据类型：

string（字符串）

number（数字） 16进制0x开头 8进制0开头(如果内容超出了限定的大小会忽略0)

boolean（布尔）

undefined（未定义）

null（空）typeof 是 对象

复杂数据类型：

Object（对象）

检测数据类型

typeof（运算符不是函数） ： console.log（typeof 变量）

类型转换：

tostring 方法。

var a = 13;

null undefined 没有 tostring 方法

string方法 可以将 undefined null 变成字符串类型

Var a = null;

console.log（string(a)）

单行注释 //

注释 多行注释 /\*\*/

Js中的最大数字和最小数字

Number.MAX\_VALUE

Number.MIN\_VALUE

console.log(0.1+0.2 = 0.3)

返回值不是 true；(不要判断浮点数相等）

NaN是一个数字 但是 是一个特殊的数字

Not a number

isNaN () 判断不是一个数字 如果是一个数字返回false。

Number()方法:

如果变量中有不是数字的字符 不能转化成数字

如果是布尔值可以转化成数字 false -> 0 true -> 1

parseInt()方法:

从的第一个数字开始到碰到的第一个不是数字的字符结束。

不能把布尔值变成 0 1

Parsefloat()

可以转换成浮点数。

不能把布尔值变成 0 1

var num1 = “30”;

Var num2 = 20;

Console.log(num1 - num2); 等于10

Var num1 = “abc”

Var num2 = “123”

Console.log(num1 - num2); 等于 NaN

console.log(NaN == NaN) false

转换成布尔值Boolean()：

NaN 空字符串（str=””） Null undefined 0 转换成boolean 是false。

Console.log( “20” == 20); true 双等号 只判断值是否相等 不管类型是否相等

Console.log (“20” === 20);false 三等号 判断 类型和值都要相等

!= 不用判断类型相等

!== 要判断类型是否相等

（自变量是写死的）

负值运算符

numA numB

numA+=numB

numA-=numB

numA\*=numB

numA/=numB

numA+=numB

numA%=numB

Switch 语句

Switch(num) {

Case 1:

Console.log(‘星期一’);

Break;

Case 2:

Console.log(‘星期一’);

Break;

Case 3:

Console.log(‘星期一’);

Break;

Case 4:

Console.log(‘星期一’);

Break;

Case 5:

Console.log(‘星期一’);

Break;

Case 6:

Console.log(‘星期一’);

Break;

Case 7:

Console.log(‘星期天’);

Break;

Default:

Console.log(‘erro’);

Break;

Ver num=10;

If(num%2){

Console.log(‘偶数’);

}else {

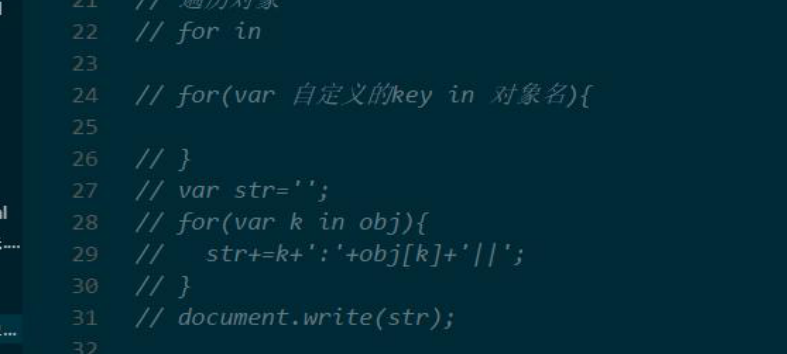
Console.log(‘奇数’);

}

判断语句后的条件中会发生隐式转换 num%2 会转换成false



遍历 对象



堆栈

简单类型在内存中的存储

对象：内置对象 浏览器对象 自定义对象

# 数组和字符串的方法

数组的方法

concat连接两个或更多的数组,返回结果. 问题: 如何连接多个数组

arr1.concat(arr2);

Join() 把数组的所有元素放入一个字符串.元素通过指定的分隔符进行分割.

arr.join(‘ ’)

Pop() 删除并返回数组的最后一个元素

arr.pop() 改变原数组的值

push() 向数组的末尾加入一个或更多元素并返回新的长度.

arr.push(‘a’). 返回新的数组的长度.

Reverse() 颠倒数组中的顺序.

arr.reverse().

Shift() 删除并返回数组的第一个元素.

arr.shift().(改变原数组的值)

概念: Slice()从某个已有的数组返回选定的元素.

var arr = [1,2,3,4,5,6];

有两个参数 第一个参数为截取的起始位置 第二个参数为结束位置 但不包括结束位置.

数组的最后一个的索引为 -1 依次类推.

1.如果第一个参数为正数 第二个参数为负数

a.正数没有超过负数位置 arr.slice(1,-1) 即索引为1的值在索引为-1的值之前:[2,3,4,5].

b.正数超过了负数的位置 splice(5,-6) 则取不到值.

1. 如果第一个参数为负数 第二个参数为正数
2. 负数的位置没有超过正数的位置 arr.slice(-5,4) 即索引为-5的值在索引是4的值前面:[2,3,4]

b.负数的位置超过了正数的位置 arr.slice(-1 ,1) 取不到值.

1. 如果不写第二个参数

会取到唯一参数之后的所有值

Sort() 对数组的元素进行排序

arr.sort();

改变了原来arr的值.

Splice(a,b,1) 删除元素,并向数组添加新元素

三个参数 第一个参数 从数组中的第a个下标开始 删除b个元素 ,插入1

toString() 把数组转换成字符串,并返回结果.

arr.tostring

Unshift 向数组的开头添加一个或者多个元素,并返回新的长度.

arr.unshift().

字符串的方法

CharAt() 返回在指定位置的字符.

Str.charAt();

Concat() 连接字符串

Str.concat(str1);

IndexOf()检索字符串

Str.indexOf(‘a’) 寻找a所在位置的下标(会在找到第一个a的时候返回值)

Lastindexof() 从后向前检索字符串

Str.lastindexof(‘a’)

Slice() 提取字符串的片段并在新的字符串中返回被提取的部分

可以写负数 写负数 会把负数加上字符串的长度的值作为参数 运算结束之后 如果前面的参数比后面的参数大 则不会有返回值.

Substring()

两个参数 如果有负数直接转换成0 并且 会自动把两个参数中小的放在前面.

如果只写了 一个参数则从一个参数直接截取到最后

Substr() 从起始索引号提取字符串字符串中指定数目的字符

两个参数 第一个为起始位置 第二个参数为 截取的字符串的个数

Split() str.split(‘b’)将字符串根据括号中的字符分割成数组.

toLowerCase() ;

toUpperCase();

StartsWith() 以...开头

PHP

$str = ‘http://abc’;

Strpos($str,’abc’) 查找是否存在字符串;

找到了 返回0 没有找到返回false;

API 应用程序编程接口 application programming interface

接口都是提供特定能力的事物 都是有输入有输出.

$str = strlen($str);

Isset 方法

$arr = array(

‘key1’=>’value1’,

‘key2’=>’value2’,

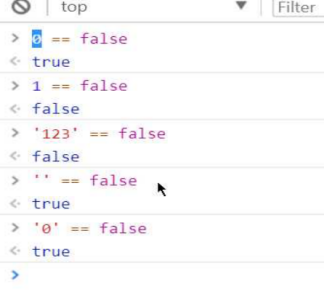
‘key3’=>’value3’

)

将等式两边转换成数字看是否相等;

例: ‘0’ == false;

将’0’ 转换成 0 false 转换成0 所以相等.



Time()时间戳获取的事件是格林威治时间.

修改时区

Date\_default\_timezone\_set(‘PRC’);

Date(’Y-m-d H:i:s’,time())

Strtotime() 可以将一个有格式的时间字符串 转换成一个 事件戳

Date.timezone (配置文件)

-- SQL 是由一套公共的标准的，不同数据库之间都可以执行，只不过有一些有特性的点

-- 新增

-- insert into users value (null, 'CEO', '麻子', 18, 0);

-- insert into users values (null, 'CEO', '麻子', 18, 0);

-- insert into users (name, title, age, gender) value ('CEO', '麻子', 18, 0);

-- 删除

-- 删除数据一定要配合数据筛选

-- delete from users where title = 'ufo' and id > 1;

-- delete form users where id in (1, 2, 3, 4)

-- 修改

update users set title = 'CEO', `name` = '张三' where id = 1

-- 查询

-- select `id`, `title`, `name` from `users`;

-- 如果使用通配符会产生全表扫描，建议指定列名

-- select \* from users;

-- select id, 'foo' from users;

-- select 语句可以选择列或者一个具体的值

-- 查询数据语句得到的是结果集

-- 增删改语句得到的是执行这个语句受影响的行数