**一， 基础知识**

* Css 全称 （Cascading Style Sheets）
* 功能：主要用于定义HTML在浏览器内的显示样式，如文字大小，颜色，字体等
* 语法：

一个css样式由选择符和声明组成，

eg. div { font-size:20px; }

* 注释：

单行： /\* \*/

多行： /\*\*

\*/

**\*三种方式的优先级**

外部引入 < 嵌入式 < 内联式

**二， 选择器**

选择器分类

1. 标签选择器

最基本的选择器。就是html代码中的标签名称

eg. <span>123</span> 🡺 span{letter-spacing:0.5px;}

1. 类选择器

最常用到，对应html标签中的class属性。语法： .选择器名称{样式代码}

eg. <div class=”box”></div> 🡺 .box{position:relative;}

1. ID选择器

语法：#选择器名称{样式代码}

eg. <div id=”container”></div> 🡺 #container {margin:0;}

1. 子选择器

匹配父元素下第一代符合子选择器的子元素。语法： 父选择器名称 > 子选择器名称

eg. <span id=”content”><i class=”box”></i></span> 🡺 #content > .box{border:1px solid #fff}

1. 包含选择器

匹配该元素下的所有符合子选择器的后辈元素 。 语法：父选择器名称 子选择器名称

eg. <span id=”content”><i class=”box”></i></span> 🡺 #content .i{color: #000;}

1. 通用选择器

无条件匹配所有元素。 语法： \*

eg. <span class=”food”></span> 🡺 .food \*{padding-left: 5px;}

1. 伪类选择器

伪类？

用于向某些选择器添加特殊的效果。如： a:active a:hover等

常用的伪类选择符：

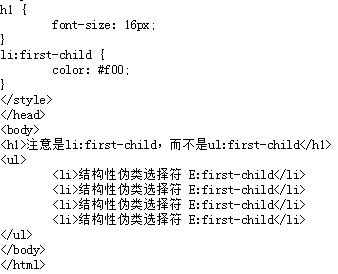
1. 语法：选择器名称 : hover

元素被hover上去的样式。

eg. <div class=”food”></ div> 🡺 .food:hover{text-decoration:underline;}

1. 语法：选择器名称 : first-child

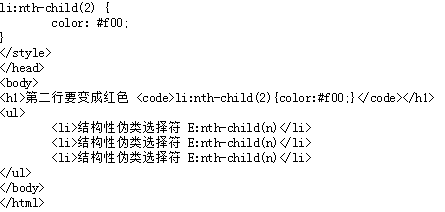
匹配满足条件的第一个元素。

eg. 下面的例子匹配的是第一个li

1. 语法：选择器名称 : nth-child(n)

匹配第n个元素，括号内可以写数字，可以写乘法因子表达式，可以写参数（even，odd）

eg. 下面的例子匹配的是第二个li



1. 语法：选择器名称 : checked

匹配界面上处于选中状态的元素

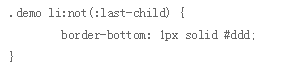
eg. 下面的例子中被选中的元素会加上红色背景



1. 语法：选择器名称 : not

匹配不包含该选择器的元素

eg. 下面的表示选择不包含最后一个li的元素的其他li元素



**四，选择器优先级**

* **权重：标签 🡺 权值为1**

**类选择符 🡺 权值为10**

**ID选择符 🡺 权值最高为100**

* **相同权重：处于最后面的css样式会被应用**
* **设置最高权重：有些特殊情况需要为某些样式设置最高权值，可以使用!important**

**Eg. p{color:red!important;}**

**P{color:green;} 该p元素的颜色是红色**

**二， 单位，值**

* **颜色**

**Css中，字体，背景，边框等需要用到颜色，设置颜色的方法**

**1.英文颜色名称 i{color:aliceblue}**

**2.rgb i{color:rgb(227,205,11)}**

**3.rgba i{color:rgba(227,205,11,0.5)}**

**4.十六进制 i{color:#00b1ff}**

**\*有些可以缩写成三位，如：p{color:#000;background-color:#369;}**

* **长度**

**常用到的长度单位：px % 像素px(pixel)**

**Css规范中假设 90像素 = 1英寸**

1. **%**

**相对于父元素的百分比值**

**三，常用排版样式**

* **字体**

1. **直接设置：body{font-family:”Microsoft YaHei”;}**
2. **引入特殊字体：**

**@font-face{**

**font-famlily: ‘newFont’;**

**src: url(‘./asset/font/newFont.ttf’)**

**}**

**body{font-family:”newFont”;}**

* **字号，颜色**

**body{font-size:12px;color:#ddd;}**

* **加粗**

**p{font-weight:400;}**

* **斜体**

**p{font-style:italic;}**

**\*字体系列样式可以简写在一个标签里面**

**P{ font-style:italic;**

**font-weight:400;**

**font-size:12px;**

**font-family:”Microsoft YaHei”;**

**}**

* **下划线, 删除线**

**p{text-decoration:underline;}**

**p{text-decoration:line-through;}**

* **缩进**

**p{text-indent:2em;}**

* **行高**

**div{line-height:30px;}**

* **字间距，字母间距**

**p{letter-spacing: 5px;word-spacing:3px;}**

* **对齐**

**span{text-align:center;}**

**三，盒模型**

**网页设计中经常用到的一种思维模型。网页分为很多块，就像盒子一样，包含内容(content), 填充（padding），边框（border），边界（margin）的属性**

* **边框**

**div{border:1px solid #000;}**

**样式分开写:**

**div{border-width:1px;border-style:solid;border-color:#000;}**

**四个边框分开写：**

**Border-top, border-right, border-bottom, border-left**

* **宽高**

**Css中定义的宽高，指的是填充的宽和高，所以一个元素即一个盒子的实际宽高度=margin + border + padding +width/height**

* **Padding(内边距)**

**div{padding:20px 10px 15px 30px;}也可以分开写**

**div{padding-top:20px;}**

**div{padding-right:10px;}**

**div{padding-top:15px;}**

**div{padding-top:30px;}**

**\*上下左右值都一样，可以直接写成**

**div{padding:30px;}**

**\*上下一样，左右一样，可以直接写成**

**div{padding:30px 20px;}**

**\*上下不一样，左右一样，可以直接写成**

**div{padding:30px 10px 20px;}**

* **Margin（外边距）**

**用法同上，区别：padding在border内部，margin在border外部**

**三，布局模型**

* 流动，浮动和层
* 层模型

1.相对定位position:relative;

相对于没设置之前的位置的移动，移动的方向和幅度由left，right，top，bottom属性决定

2.绝对定位position:absolute;

相对于其最接近的一个具有定位属性的父包含块进行绝对定位，如果不存在这样的包含块，则想对于body父元素进行定位，移动的方向和幅度由left，right，top，bottom属性决定

3.固定定位position:fixed;

与position:absolute相似，相对于移动的坐标是屏幕内的网页窗口，不随着浏览器窗口的滚动儿变化，移动的方向和幅度由left，right，top，bottom属性决定

**五，css样式设置小技巧**

* **在chrome页面上直接调试**
* **水平居中**

1. **行内元素：被设置的元素为行内元素，给父元素设置text-align:center；**
2. **块状元素：定宽的时候用margin：0 auto;不定宽的情况下可以给元素外层加上table标签，让表格自动去居中；还可以给内部元素设置inline，使内部元素变成行内，从而给父元素设置text-algin:ceneter；给父元素设置float，然后给父元素设置position:relative;和left：50%，子元素设置position:relative;和left：-50%**

* **垂直居中**

1. **父元素高度确定：设置height和line-height**
2. **高度确定的多行文本，图片：**
3. **使用table，同时设置vertical-algin:middle**
4. **设置块级元素的display为table-cell，然后设置vertical-algin:middle**

* **Display类型的隐性改变**

**被设置position：absolute或者float：left/right;后，元素的display自动变成display：inline-block；（块状元素），且默认宽度不占满父元素。**

**五，css框架-bootstrap**

**中文官网：** <http://www.bootcss.com/>

**下载， 安装， 引入**

**主要使用的组件：**

* 布局容器

1. **.container 类用于固定宽度并支持响应式布局的容器。**
2. **.container-fluid 类用于 100% 宽度，占据全部视口（viewport）的容器。**

* 栅格系统

什么是栅格系统？ 就是一个通过设置行row和列col创建好的布局。最多一行有12列，超出就在下一行显示

各种类型适用的尺寸，

* button

btn ，

btn-default, btn-info…

btn-lg, btn-sm, btn-xs

* 表单

一定要设置label属性

* 组件

1. awesome for bootstrap 图标哭
2. 导航
3. 分页
4. 进度条
5. 面板

* bootstrap.js

六．代码规范讲解

https://github.com/suansuancu/web-Standard

* 缩进4个空格
* 命名遵循驼峰