day08

一.CSS概述

1.什么是css

|  |
| --- |
| Cascading style sheets  层叠样式表，级联样式表，简称样式表 |

2.css的作用

|  |
| --- |
| 修饰html，亚洲四大邪术  html：负责网页结构的搭建，内容的承载  css:负责网页的修饰，美化页面 |

3.html属性和css的使用原则

|  |
| --- |
| html属性，代码没有可重用性  css有一定的可重用性，但是没有达到极致(scss帮助css的重用性达到了极致)  W3C建议我们使用css的方式来取代html的属性 |

二.css的语法规范

1.使用css样式

①内联样式（行内样式）

|  |
| --- |
| 跟html属性一样，只能作用在一个html标签，不可重用  使用元素的style属性来定义样式  <any style="样式声明"> any代表任意元素  内联样式没有可重用性，  默认优先级最高，  项目中基本不用，只在学习和测试的时候使用 |

②内部样式

|  |
| --- |
| 在同一个html页面中可以重用  在<head></head>中编写<style></style>标签  在style标签中编写样式  选择器{样式声明1;样式声明2.。。。。。。。。。。。。。。}  项目中使用较少，学习和测试使用较多 |

③外部样式

|  |
| --- |
| 单独创建一个css文件，  在需要使用的html中  head>link来引用这个css文件  <link rel="stylesheet" href="my.css"> |

练习：01\_ex.html

|  |
| --- |
| h1 lorem 内联样式 背景pink 字体yellow  h2 lorem 内部样式 背景yellow 字体red  h3 lorem 外部样式 背景 red 字体pink |

2.css样式的特性

①继承性

|  |
| --- |
| 一些样式可以被后代元素继承  但是有一些样式是不能继承，尺寸，内外边距，定位，动画，过渡  一般，关于字体的设置都可以继承  例外，a标签不继承字体颜色 |

②层叠性

|  |
| --- |
| 可以给一个元素设置多个样式，如果设置的样式不发生冲突  那么所有样式都会应用这个元素上 |

③优先级

|  |
| --- |
| 默认优先级  高 内联  就近原则 内部和外部  低 浏览器默认样式 |

④调整优先级

|  |
| --- |
| color:yellow !important;  !important会提升样式的优先级，比内联都高  如果多个冲突的样式属性，都设置了!important。那么按照就近原则  内联style属性中不允许加!important |

练习

|  |
| --- |
| 02\_ex.html  一个p标签，内容lorem  内部 文本blue 字号24px  外部 文本red 字号40px  在style标签前引入外部样式，f12观察样式  在style标签后引入外部样式，f12观察样式  尝试使用!important改变优先级 |

三.基础选择器（重点，8种选择器）

1.选择器的作用

|  |
| --- |
| 规范了页面中哪些元素能够使用定义好的样式  选择器就是一个条件，页面中符合这个条件的元素都会应用这个样式 |

2.选择器详解

①通用选择器

|  |
| --- |
| 让页面中所有的元素都应用这个样式  \*{样式声明}  项目中很少使用  \*{margin:0;padding:0} 所有元素内外边距清0，做样式的统一，css reset  把浏览器给的默认样式，统一或者清除  不同的浏览器厂商，定义的默认样式不同  在真正的开发之前，都要做css reset |

②.元素选择器，标签选择器

|  |
| --- |
| 页面中所有对应元素应用这个样式，设置某元素的公共样式  div{样式声明}  p{样式声明}  元素的关键字{样式声明} |

③ID选择器，私人订制

|  |
| --- |
| 匹配使用这个id的元素，同一个页面中只匹配一个元素  <any id="id值">  #id值{样式声明} |

练习

|  |
| --- |
| 03\_ex.html  h2标签，内容假文，id为text1  使用id选择器设置文本为紫色，背景为灰色 purple gray  字体为斜体 font-style:italic;  查看页面效果后  再使用元素选择器，设置文本为红色，背景为粉色 red pink  再查看效果---对就近原则产生了怀疑----选择器的权值 |

④类选择器

|  |
| --- |
| 定义页面上某一列元素使用的样式，谁想用就用class引用  .类名{样式声明;样式声明;.....}  <any class="类名1 类名2....">  分类选择器  .类名1.类名2{同时引用了这个两个类的元素应用此样式}  元素选择器.类名{引用了这个类名的这个元素才能应用此样式}  类名的命名规范 1.不能以数字开头  2.只能包含字母，数字，- \_  3.尽量见名知意 |

练习

|  |
| --- |
| 03\_ex.html  添加div和p 内容假文  用类选择设置字体颜色为红色，两个元素都引用这个类  使用分类选择器为p标签设置背景颜色为蓝色  .text-red{}  p.text-red{} |

⑤群组选择器

|  |
| --- |
| 多个选择器之间使用逗号分隔，同时定义一个样式  选择器1，选择器2，选择器3.....{样式声明}  相当于  选择器1{样式声明}  选择器2{样式声明}  选择器3{样式声明} |

⑥后代选择器

|  |
| --- |
| 通过多个元素的后代关系，匹配元素  后代关系：一级以及大于一级的嵌套关系  选择器1 选择器2 选择器3....{样式声明} |

⑦子代选择器

|  |
| --- |
| 通过元素的子代关系，匹配元素  子代关系：一级嵌套关系  选择器1 > 选择器2 >选择器3.....{样式声明}  子代选择器和后代选择器可以混写  div p > span{background-color:pink}  div >p span{background-color:pink} |

⑧伪类选择器

|  |
| --- |
| 匹配元素的某种状态下，应用的样式  :link{} 匹配链接未被访问过的状态  :visited{} 匹配链接已被访问过的状态  :hover{} 匹配元素鼠标悬停的状态  :active{} 匹配元素被激活时的状态 |

练习

|  |
| --- |
| 04\_ex.html  一个a标签，内容晚上吃大餐  1.访问后，文本用颜色orange  2.被激活时，文本颜色green  3.鼠标悬停时，文本颜色红色  4.未被访问，文本颜色pink  坑：如果4种状态作用到同一个元素上时，需要按照一定顺序编写，不然会冲突  爱恨原则 love & hate  :link :visited :hover :active |
| 匹配元素获取焦点时的状态  :focus{} |

3.选择器的权值

|  |
| --- |
| 权值：标识当前选择器的重要程度，权值越大，优先级越高  默认权值，高--->低  !important >1000  内联样式 1000  id选择器 100  类选择器 10  伪类选择器 10  元素选择器 1  \*通用选择器 0  继承的样式 无  权值的总结：  1.当一个选择器中包含多个选择器的时候，需要将所有选择器的权值进行相加，然后比较，权值大的优先显示  2.权值相同时，遵循就近原则  3.群组选择器的权值要单独计算，不能无脑相加  4.添加!important直接获取最高权值，内联样式不能写important  5.选择器的权值计算，不能超过本身的最大数量级  （100个1加一起也小于10） |

四.尺寸和边框

1.尺寸

|  |
| --- |
| 改变元素的宽度和高度  width：  取值：1.px为单位的数字  2.% 父元素的宽度的百分比  height：  取值：1.px为单位的数字  2.% 父元素的高度的百分比  但是如果父元素是body，就没有高度  min-width 最小宽度  max-width 最大宽度  取值1.px为单位的数字  2.% 元素本身尺寸的100%； max-width:100%经常在图片上使用  这张图片可以变小显示  但是最大不能超过本身原始尺寸  min-height  max-height |

作业：

1.熟读今天的笔记

①css的使用

②css的特性

③选择器的使用

2.重新完成根据uid修改用户信息

3.重新完成登录模块