day10

五.盒子模型

2.外边距的特殊效果

④外边距溢出

|  |
| --- |
| 在特殊情况下，为子元素添加上外边距，会作用到父元素  特殊情况：1.父元素没有上边框  2.设置上外边距的子元素的内容区域的上沿与父元素内容区域的上沿重合  （为第一个子元素设置上外边距时，这种说法不严谨）  解决方案：1.给父元素添加上边框，  弊端，会影响父元素实际占地高度  2.给父元素添加上内边距  弊端，会影响父元素实际占地高度  3.给父元素设置overflow:auto/hidden  弊端，为了解决外边距溢出，而增加了新的限制  4.在父元素中，第一个子元素位置，添加一个空的table标签  <div id="d2">  <table></table>  <div id="d3"></div>  </div> |

⑤外边距使用时机

|  |
| --- |
| 1.元素位置发生改变，可以使用外边距  2.元素与元素之间有空隙，这个空隙用外边距 |

3.内边距padding

|  |
| --- |
| 边框与内容区域之间的距离  改变内边距，感觉上是改变了元素的大小，  内边距有颜色，颜色同元素背景色  padding:value; 同时设置4个方向的内边距  padding-top  padding-right  padding-bottom  padding-left  取值数量  padding:v1;  padding:v1 v2; 上下 左右 但是padding没有auto  padding:v1 v2 v3; 上 左右 下  padding:v1 v2 v3 v4; 上右下左 |

4.box-sizing(重要)

|  |
| --- |
| 浏览器默认的盒子模型，元素实际占地宽度  左外边距+左边框+左内边距+内容区域宽度+右内边距+右边框+右外边距  box-sizing：设置盒子模型的计算公式  1.默认公式 content-box  2.border-box  设置width是border+padding+内容区域的总宽度  实际占地宽度：左外边距+width+右外边距  如果元素的尺寸，使用%设置，你要留心，是不是需要设置border-box |

六.背景相关样式

1.背景颜色

|  |
| --- |
| background-color:#ff0; |

2.背景图片

|  |
| --- |
| background-image:url(image/07.png); |

3.背景图片的平铺

|  |
| --- |
| background-repeat:  取值 1.repeat 默认值 平铺  2.no-repeat 不平铺  3.repeat-x 水平平铺  4.repeat-y 垂直平铺 |

4.背景图片的定位

|  |
| --- |
| background-position:  取值 1.x y 以px为单位的数字，定义x轴和y轴的位置  2.x% y%  3.x 或者 x%,不写y，默认y为50%  4.关键字 x：left/center/right y:top/center/bottom |

5.设置背景图片尺寸

|  |
| --- |
| background-size:  取值：1.x y 以px为单位的数字  2.x% y%  3.只取一个值，同时设置宽高  4.cover 要求图片把容器全部覆盖，图片会显示不完整  5.contain 要求容器包含完整的图片，图片显示完整，  但是容器会有空白区域 |

6.背景图片的固定

|  |
| --- |
| background-attachment:  取值：1.scroll 默认值，背景图定位以容器为准  背景图会随着页面滚动条滚动而变化  2. fixed 固定，背景图定位以body为准  页面的滚动条，滚动，背景图不会随着发生位置变化  但是，只能在本容器区域内显示 |

7.背景的简写方式

|  |
| --- |
| background:color image repeat attachment position;  最简方式 background:color/image;  background:#f00;  background:url(..png);  注意：如果想在已有的背景设置中，添加或修改属性  不能使用简写方式修改，不然会覆盖之前的样式 |

七.渐变 gradient

1.什么是渐变

|  |
| --- |
| 渐变，就是指多种颜色，平缓变化一种显示效果 |

2.渐变的主要因素

|  |
| --- |
| 色标：一种颜色，及其出现的位置  一个渐变，至少有两个色标 |

3.渐变的分类

①线性渐变 以直线的方式来填充渐变色

②径向渐变 以圆形的方式来填充渐变色

③重复渐变 将线性渐变，或者径向渐变，重复几次实现填充

4.线性渐变

|  |
| --- |
| background-image  background-image:linear-gradient(方向,色标1,色标2,........);  方向：取值 1. to top ↑  to right →  to bottom ↓  to left ←  2.角度 0deg == to top  90deg== to right  180deg== to bottom  270deg== to left  色标： 颜色 位置  位置的写法1. #f00 10px px为单位的数字  2. #f00 10% %  3.只有两种颜色，分别在0%和100%，可以省略位置 |

5.径向渐变

|  |
| --- |
| background-image:radial-gradient(半径 at 圆心,色标1..........);  半径：以px为单位的数字  圆心：x y 以px为单位的数字  x% y%  关键字 x:left/center/right y:top/center/bottom  关于色标，注意，如果色标的位置，使用px为单位的具体数字。  那么这个径向渐变的半径，就失去作用了 |

6.重复渐变

|  |
| --- |
| 重复的线性渐变  background-image:repeating-linear-gradient(方向,色标.....);  重复的径向渐变  background-image:repeating-radial-gradient(半径 at 圆心,色标.....); |

7.渐变的浏览器兼容性问题

|  |
| --- |
| 如果想兼容ie8以下，写兼容代码，写兼容的过程，css hack  第一步，添加浏览器内核  -webkit- chrome/safari  -moz- firefox  -ms- IE  -o- opera  注意：添加了浏览器内核的线性渐变，他的方向，只能写起点，不能加to  -webkit-linear-gradient(left,#f00 0%,#00f 100%);  -ms-linear-gradient(left,#f00 0%,#00f 100%);  -moz-linear-gradient(left,#f00 0%,#00f 100%);  -o-linear-gradient(left,#f00 0%,#00f 100%); |

八.文本格式化（重点）

1.字体属性

①字号大小

|  |
| --- |
| font-size: 以px/pt/em/rem为单位的数字 |

②字体系列

|  |
| --- |
| font-family：系列1，系列2，系列3......  取值：设备中字体库里有的字体，如果字体名称有空格，需要加""  用,隔开多个字体类型 |

③字体加粗

|  |
| --- |
| font-weight:  取值 1.关键字 lighter normal bold bolder  2.无单位的数字，必须是100的整倍数，最大值1000 |

④字体样式

|  |
| --- |
| font-style:  取值：1.normal 默认值 正常 无样式  2.italic 斜体 |

⑤小型大写字母

|  |
| --- |
| font-variant:small-caps; |

⑥字体属性的简写方式

|  |
| --- |
| font:style variant weigth size family;  最简方式：font:size family; |

样式编写步骤

|  |
| --- |
| 找目标元素，从上往下，从左往右， 先写外后写里  1.先写尺寸，大体位置  2.边框和背景相关  3.文本相关  4.微调 |

作业：

1.完成学子登陆页面

