css技巧

1、重置表单样式

```
1  <style>
2          /* 重置表单样式 */
3          input,button,select,textarea{
4              -webkit-appearance: none;
5              -moz-appearance: none;
6              appearance: none;
7              outline: none;
8              border: none;
9          }
10  </style>
11
12  <input type="text">
13  <input type="button" value="Button">
14  <input type="checkbox" name="" id="">
15  <input type="radio" name="" id="">
```

2、css变量

```
1  <style>
2          :root{
3              --theme_color:red;
4          }
5          body{
6              color: var(--theme_color,'#000');
7          }
8          /* 将变量声明到局部,只能在子节点中使用 */
9          .selector { --color: yellow }
10          .selector span { color: var(--color) }
11  </style>
12
13  <p class="selector">
14      <span>123</span>
15  </p>
```

```
16
17    <script>
18        // -> 操作全局变量
19        document.documentElement.style.setProperty('--theme_color','blu
20    </script>
```

3、移动端常用操作

```
1  <style>
2        /* 禁止保存图片 */
3
4        .no-callout {
5            -webkit-touch-callout: none
6        }
7
8        /* 禁止长按文字选择功能 */
9
10        .unselect {
11            -webkit-touch-callout: none;
12            -webkit-user-select: none;
13            -khtml-user-select: none;
14            -moz-user-select: none;
15            -ms-user-select: none;
16            user-select: none
17        }
18
19        /* 移动端顺畅滚动 */
20
21        .scroll-touch {
22            -webkit-overflow-scrolling: touch;
23        }
24
25        /* 移动端屏幕旋转时字体大小保持不变 */
26
27        html,
28        body,
29        form,
30        p,
31        div,
32        h1,
33        h2,
34        h3,
```

```css
35        h4,
36        h5,
37        h6 {
38            -webkit-text-size-adjust: 100%;
39            -ms-text-size-adjust: 100%;
40            text-size-adjust: 100%
41        }
42
43        /* 移动端横竖屏的适配 */
44
45        @media all and (orientation:portrait) {
46            body::after {
47                content: '竖屏'
48            }
49        }
50
51        @media all and (orientation:landscape) {
52            body::after {
53                content: '横屏'
54            }
55        }
56
57        /* 改变input光标的颜色 */
58
59        .input {
60            caret-color: red;
61        }
62
63        /* 移动端Android设备上去掉语音输入按钮 */
64
65        input::-webkit-input-speech-button {
66            display: none
67        }
68
69        /* 去掉为search自带的清空按钮 */
70
71        .search::-webkit-search-cancel-button {
72            display: none
73        }
74  </style>
```

4、文字模糊

```
1  <style>
2        .blur {
3            color: transparent;
4            text-shadow: 0 0 5px rgba(0, 0, 0, 0.5)
5        }
6    </style>
7
8    <p class="blur">Hello World</p>
```

5、换行

```
1  <style>
2        *{
3            margin: 0;
4            padding: 0;
5        }
6        .box{
7            width: 100px;
8            height: 100px;
9            border: 1px solid #000;
10        }
11        .box p{
12            /* 不换行 */
13            white-space: nowrap;
14
15            /* 自动换行 */
16            word-wrap: break-word;
17            word-break: normal;
18
19            /* 强制换行 */
20            word-break: break-all;
21        }
22    </style>
23
24    <div class="box">
25        <p>你可能就基本后看门狗就弄卡盟你今年可能你你你看看那就</p>
26    </div>
```

6、vertical-align

```
1  <style>
2      *{
3          margin: 0;
4          padding: 0;
5      }
6      .box{
7          border: 1px solid red;
8          font-size: 100px;
9      }
10     .wrap{
11         display: inline-block;
12         width: 100px;
13         height: 100px;
14         background-color: red;
15         /* baseline - 默认是基线,比如说是本实例x的下边缘 */
16         /* top - 元素的顶端与行中最高元素的顶端对齐 */
17         /* text-top - 元素的顶端与父元素字体的顶端对齐 */
18         /* bottom - 元素的低端与行中最低元素的顶端对齐 */
19         /* text-bottom - 元素的低端与父元素字体的低端对齐 */
20         /* middle - 相对于中线对齐（中线就是小写x的高度的一半）*/
21         vertical-align: middle;
22     }
23     </style>
24
25     <!--
26         vertical-align 垂直对齐
27             - 适用于inline inline-block
28     -->
29     <div class="box">
30         <div class="wrap"></div>xXay
31     </div>
```

7、cursor

```
1  <style>
2      *{
3          margin: 0;
4          padding: 0;
5      }
6      .box{
```

```
 7              width: 500px;
 8              height: 500px;
 9              background-color: red;
10              margin: 50px auto;
11              /*
12                  cursor:
13                      pointer     箭头
14                      wait        等待
15                      help        帮助
16                      pointer     小手
17                      text        文本
18                      move        鼠标移动
19              */
20              /* cursor: pointer; */
21
22              /* 自定义鼠标样式 */
23              cursor: url("https://p1.ssl.qhimg.com/t01a472755aac62783f.p
24          }
25      </style>
26
27      <div class="box"></div>
```

8、逗号分隔列表

```
 1  <style>
 2          p>span:not(:last-child)::after{
 3              content: ',';
 4          }
 5      </style>
 6
 7      <p>
 8          <span>1</span>
 9          <span>2</span>
10          <span>3</span>
11          <span>4</span>
12      </p>
```

9、checkbox改造成Switch

```
1  <style>
2          /* 背景层 */
3
4          .switch-component {
5              position: relative;
6              width: 60px;
7              height: 30px;
8              background-color: #dadada;
9              border-radius: 30px;
10             border: none;
11             outline: none;
12             -webkit-appearance: none;
13             transition: all .2s ease;
14         }
15
16         /* 按钮 */
17         .switch-component::after {
18             content: attr(data-content);
19             position: absolute;
20             top: 0;
21             left: 0;
22             width: 50%;
23             height: 100%;
24             text-align: center;
25             line-height: 30px;
26             color: #dadada;
27             background-color: #fff;
28             border-radius: 50%;
29             transition: all .2s ease;
30         }
31
32         /* 选中状态时，背景色切换 */
33
34         .switch-component:checked {
35             background-color: #86c0fa;
36         }
37
38         /* 选中状态时，按钮的位置移动 */
39
40         .switch-component:checked::after {
41             left: 50%;
42             color: #ccc;
43         }
44     </style>
45
46     <input class='switch-component' type='checkbox' data-content="关">
```

```
47
48    <script>
49        var needContributions;
50        document.querySelector(".switch-component").onclick=function()
51            if (this.checked) {
52                var needContributions = 0;
53                console.log(needContributions);
54                this.setAttribute('data-content', "开")
55            } else {
56                var needContributions = 1;
57                console.log(needContributions);
58                this.setAttribute('data-content', "关")
59            }
60        }
61    </script>
```

10、隐藏没有静音就自动播放的video

```
1 <style>
2        video[autoplay]:not([muted]) {
3            display: none
4        }
5    </style>
```

11、input获取焦点上浮效果

```
1 <style>
2        input{
3            appearance: none;
4            outline: none;
5            border: none;
6            padding: 1rem;
7            border-bottom: 1px solid #ebebeb;
8            transition: all .2s ease-in-out;
9        }
10        input:focus{
11            box-shadow: 0 3px 10px 1px rgba(0,0,0,.1);
12            transform: translateY(-.5rem);
13        }
```

```
14    </style>
15
16    <input type="text" placeholder="请输入">
```

12、菜单栏的弹性伸缩

```
1  <style>
2        body,
3        html {
4              height: 100%;
5              overflow-x: hidden;
6        }
7
8        .index {
9              height: 100%;
10       }
11
12       /* 菜单栏的初始样式 */
13
14       .menu {
15             position: fixed;
16             top: 0;
17             left: 0;
18             width: 10rem;
19             height: 100%;
20             background-color: darkgrey;
21             transform: translateX(-10rem);
22             z-index: 1;
23             transition: all .2s ease-in;
24       }
25
26       /* 内容区的初始样式 */
27
28       .content {
29             display: flex;
30             justify-content: center;
31             align-items: center;
32             width: 100%;
33             height: 100%;
34             background-color: cornsilk;
35             transform: translateX(0);
36             transition: all .3s ease-in;
```

```css
        }

    input {
        display: none;
    }

    /* 切换按钮的初始样式 */

    label {
        position: fixed;
        top: 1rem;
        left: 1rem;
        z-index: 2;
        transition: all .2s ease-in;
    }

    /* 切换按钮选中的样式 */

    input:checked~label {
        left: 12rem;
    }

    /* 切换按钮文字的切换样式 */

    input:not(:checked)~label::after {
        content: '拉出'
    }

    input:checked~label::after {
        content: '收起'
    }

    /* 菜单栏显示的样式 */

    input:checked~.menu {
        transform: translateX(0)
    }

    /* 内容区显示的样式 */

    input:checked~.content {
        transform: translateX(10rem)
    }
</style>

<div class="index">
```

```
83        <input type="checkbox" id="btn">
84        <label for="btn"></label>
85        <div class="menu"></div>
86        <div class="content">
87            <p>我是内容</p>
88        </div>
89    </div>
```

13、波浪效果

```
1  <style>
2      body {
3          background-color: #313131;
4      }
5
6      .wave {
7          position: absolute;
8          top: 0;
9          left: 0;
10         right: 0;
11         bottom: 0;
12         margin: auto;
13         width: 200px;
14         height: 200px;
15         background-color: #5291e0;
16         overflow: hidden;
17     }
18
19     .wave::before,
20     .wave::after {
21         content: '';
22         position: absolute;
23         left: 50%;
24         width: 500%;
25         height: 500%;
26         background-color: #fff;
27         border-radius: 48%;
28         animation-name: rotate;
29         animation-iteration-count: infinite;
30         animation-timing-function: linear;
31     }
32
```

```
33    .wave::before {
34        bottom: 15%;
35        border-radius: 45%;
36        animation-duration: 15s;
37    }
38
39    .wave::after {
40        bottom: 10%;
41        opacity: .5;
42        border-radius: 47%;
43        animation-duration: 15s;
44    }
45
46    @keyframes rotate {
47        0% {
48            transform: translate(-50%, 0) rotateZ(0deg);
49        }
50        50% {
51            transform: translate(-50%, 0) rotateZ(180deg);
52        }
53        100% {
54            transform: translate(-50%, 0) rotateZ(360deg);
55        }
56    }
57  </style>
58
59  <div class="wave"></div>
```

14、导航跟随效果

```
1  <style>
2      ul {
3          list-style: none;
4          font-size: 0;
5      }
6
7      li {
8          position: relative;
9          display: inline-block;
10         padding: 1rem;
11         cursor: pointer;
12         font-size: 16px;
```

```
13            }
14
15         li::after {
16             content: "";
17             position: absolute;
18             bottom: 0;
19             left: 100%;
20             width: 0;
21             height: 0;
22             border-bottom: 1px solid #000;
23             transition: .3s ease;
24         }
25
26         li:hover::after {
27             width: 100%;
28             left: 0;
29         }
30
31         li:hover~li::after {
32             left: 0;
33         }
34     </style>
35
36     <ul>
37         <li>1111</li>
38         <li>2222</li>
39         <li>3333</li>
40         <li>4444</li>
41     </ul>
```

15、左右滑动

```
1  <style>
2         .box {
3             width: 500px;
4             height: 200px;
5         }
6
7         .box .item {
8             width: 80px;
9             height: 200px;
10            margin-left: 20px;
```

```
11                 background: plum;
12             }
13
14        .scroll-x {
15             display: flex;
16             width: auto;
17             overflow-x: auto;
18             -webkit-overflow-scrolling: touch
19        }
20
21        .scroll-x>.scroll-x-item {
22             flex-shrink: 0;
23             margin-right: .5rem
24        }
25    </style>
26
27    <div class="scroll-x box">
28        <div class="scroll-x-item item">1</div>
29        <div class="scroll-x-item item">2</div>
30        <div class="scroll-x-item item">3</div>
31        <div class="scroll-x-item item">4</div>
32        <div class="scroll-x-item item">5</div>
33        <div class="scroll-x-item item">6</div>
34    </div>
```

16、水平居中方法

```
1 <style>
2        /*
3            方法1
4                优点：兼容性较好,甚至可以兼容ie6,ie7
5                缺点：child里的文字也会水平居中,可以在.child添加text-align:lef
6        */
7        /* .parent {
8             width: 200px;
9             height: 200px;
10            background-color: aquamarine;
11            text-align: center;
12        }
13        .child {
14            display: inline-block;
15        } */
```

```
16
17
18          /*
19                  方法2
20                          优点：只设置了child,ie8以上都支持
21                          缺点：ie6,7不支持将div换成table
22          */
23          /* .parent {
24              width: 200px;
25              height: 200px;
26              background-color: aquamarine;
27          }
28          .child {
29              display: table;
30              margin: 0 auto;
31          } */
32
33
34          /*
35                  方法3
36                          优点：居中元素不会对其他的产生影响
37                          缺点：transform属于css3内容,兼容性存在一定问题,高版本浏览器需要涉
38          */
39          /* .parent {
40              width: 200px;
41              height: 200px;
42              background-color: aquamarine;
43              position: relative;
44          }
45          .child {
46              position: absolute;
47              left: 50%;
48              transform: translateX(-50%);
49          } */
50
51
52          /*
53                  方法4
54                          优点：简单快捷
55                          缺点：低版本浏览器不支持
56          */
57          /* .parent {
58              width: 200px;
59              height: 200px;
60              background-color: aquamarine;
61              display: flex;
```

```
62            }
63        .child {
64            margin: 0 auto;
65        } */
66

67

68        /*
69            方法5
70                优点：简单,设置parent即可
71                缺点：低版本浏览器不支持
72        */
73        /* .parent {
74            width: 200px;
75            height: 200px;
76            background-color: aquamarine;
77            display: flex;
78            justify-content: center;
79        } */
80

81

82        /*
83            方法6
84                使用grid网格布局
85        */
86        .parent {
87            width: 200px;
88            height: 200px;
89            background-color: aquamarine;
90            display: grid;
91        }
92        .child {
93            justify-self: center;
94        }
95    </style>
96

97    <div class="parent">
98        <div class="child">Jerry</div>
99    </div>
```

17、水平垂直居中

```
1  <style>
```

```css
2        * {
3            margin: 0;
4            padding: 0;
5        }
6        /*
7            方法1
8                absolute + 负margin值
9                父元素设置相对定位,子元素设置绝对定位,margin值设置为负数
10               子元素需要定高
11       */
12       /* .parent {
13           width: 200px;
14           height: 200px;
15           border: 1px solid #000;
16           position: relative;
17       }
18       .child {
19           position: absolute;
20           top: 50%;
21           left: 50%;
22           margin-left: -50px;
23           margin-top: -50px;
24           width: 100px;
25           height: 100px;
26           background-color: #f00;
27       } */


30       /*
31           方法2
32               absolute + margin auto
33               父元素设置相对定位,子元素设置绝对定位,margin值设置auto,四个方向词
34               子元素需要定宽高
35       */
36       /* .parent {
37           width: 200px;
38           height: 200px;
39           border: 1px solid #000;
40           position: relative;
41       }
42       .child {
43           position: absolute;
44           left: 0;
45           right: 0;
46           top: 0;
47           bottom: 0;
```

```
48          margin: auto;
49          width: 100px;
50          height: 100px;
51          background-color: #f00;
52      } */
53
54
55      /*
56          方法3
57              absolute + calc
58              父元素设置为相对定位,子元素设置绝对定位,使用ralc
59              子元素需要定宽高
60      */
61      /* .parent {
62          width: 200px;
63          height: 200px;
64          border: 1px solid #000;
65          position: relative;
66      }
67      .child {
68          position: absolute;
69          top: calc(50% - 50px);
70          left: calc(50% - 50px);
71          width: 100px;
72          height: 100px;
73          background-color: #f00;
74      } */
75
76
77      /*
78          方法4
79              absolute + transform
80              父元素设置相对定位,子元素设置绝对定位,使用transform
81      */
82      /* .parent {
83          width: 200px;
84          height: 200px;
85          border: 1px solid #000;
86          position: relative;
87      }
88      .child {
89          position: absolute;
90          left: 50%;
91          top: 50%;
92          transform: translate(-50%, -50%);
93          background-color: #f00;
```

```css
            width: 100px;
            height: 100px;
        } */


        /*
            方法5
                使用flex
                子元素可以不用设置
        */
        /* .parent {
            width: 200px;
            height: 200px;
            border: 1px solid #000;
            display: flex;
            justify-content: center;
            align-items: center;
        }
        .child {
            width: 100px;
            height: 100px;
            background-color: #f00;
        } */


        /*
            方法6
                使用grid网格布局
                子元素可以不用设置宽高
        */
        .parent {
            width: 200px;
            height: 200px;
            border: 1px solid #000;
            display: grid;
        }
        .child {
            width: 100px;
            height: 100px;
            background-color: #f00;
            align-self: center;
            justify-self: center;
        }
    </style>

    <div class="parent">
```

```
140        <p class="child"></p>
141    </div>
```

18、自定义滚动条

```
1  <style>
2          /* Document scrollbar */
3          ::-webkit-scrollbar {
4              width: 8px;
5          }
6          ::-webkit-scrollbar-track {
7              box-shadow: inset 0 0 6px rgba(0, 0, 0, 0.3);
8              border-radius: 10px;
9          }
10         ::-webkit-scrollbar-thumb {
11             border-radius: 10px;
12             box-shadow: inset 0 0 6px rgba(0, 0, 0, 0.5);
13         }
14         /* Scrollable element */
15         .some-element::webkit-scrollbar {
16         }
17     </style>
18
19     <div class="custom-scrollbar">
20         <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Iur
21     </div>
```

19、缓冲动画 – animation–delay为负值

```
1  <style type="text/css">
2      /*
3       animation-delay
4          定义动画何时开始
5
6          默认为0 – 立即开始执行动画
7          正值 – 延迟指定时间后,开始执行动画
8          负值 – 立即执行,但跳过指定时间后进入动画
9      */
10     body{
```

```css
       color: #fff;
       background: #fff;
     }
     h1{
       font-size: 14px;
       font-family: "Microsoft Yahei";
       text-align: center;
     }
     .spinner{
       width: 800px;
       height: 50px;
       margin: 100px auto;
       text-align: center;
     }
     .spinner>div{
       width: 6px;
       height: 100%;
       display: inline-block;
       background: green;
       -webkit-animation:strechdelay 1.2s infinite ease-in-out;
     }
     .spinner .line2{
       -webkit-animation-delay:-1.1s;
     }
     .spinner .line3{
       -webkit-animation-delay:-1.0s;
     }

     .spinner .line4{
       -webkit-animation-delay:-0.9s;
     }

     .spinner .line5{
       -webkit-animation-delay:-0.8s;
     }
     @-webkit-keyframes strechdelay{
        0%,40%,100%{
         -webkit-transform:scaleY(0.4);
         }
        20%{
            -webkit-transform:scaleY(1);
       }

     }

 </style>
```

```
57
58        <!-- 缓冲动画 -->
59    <div class="spinner">
60      <div class="line1"></div>
61      <div class="line2"></div>
62      <div class="line3"></div>
63      <div class="line4"></div>
64      <div class="line5"></div>
65    </div>
```

20、缓冲动画 – 妙用border颜色

```
1  <style type="text/css">
2      body{
3        color: #fff;
4        background: #222;
5      }
6      h1{
7        font-size: 14px;
8        font-family: "Microsoft Yahei";
9        text-align: center;
10     }
11
12     .spinner{
13       width: 100px;
14       height: 100px;
15       border-radius: 100%;
16       margin: 100px auto;
17       border:10px solid rgba(255,255,255,0.2);
18       border-left-color:#fff;
19       -webkit-animation:start 1.2s infinite linear;
20     }
21
22     @-webkit-keyframes start{
23         from{
24           transform: rotate(0deg);
25         }
26         to{
27           transform: rotate(360deg);
28         }
29
30     }
```

```
31
32    </style>
33
34      <div class="spinner"></div>
```

21、缓冲动画 – 气泡loading效果

```
1  <style type="text/css">
2      body{
3        color: #222;
4      }
5      h1{
6        font-size: 14px;
7        font-family: "Microsoft Yahei";
8        text-align: center;
9      }
10     .spinner{
11       width: 40px;
12       height: 40px;
13       border-radius: 100%;
14       background-color: green;
15       margin: 100px auto;
16       -webkit-animation:fadeOut 1s infinite ease-in-out;
17     }
18     @-webkit-keyframes fadeOut{
19       from{
20         transform: scale(0);
21       }
22       to{
23         transform: scale(1);
24         opacity: 0;
25       }
26     }
27
28   </style>
29
30      <div class="spinner"></div>
```

22、2倍/3倍图css技巧

```
1  /*普通显示屏*/
2      .css{
3          background-image: url(img_1x.png);
4      }
5
6      /* 高清显示屏(设备像素比例大于等于2)使用2倍图   */
7      @media only screen and (-webkit-min-device-pixel-ratio:2){
8          .css{
9              background-image: url(img_2x.png);
10         }
11     }
12
13     /* 高清显示屏(设备像素比例大于等于3)使用3倍图   */
14     @media only screen and (-webkit-min-device-pixel-ratio:3){
15         .css{
16             background-image: url(img_3x.png);
17         }
18     }
```

23、css图形画法 – 扇形

```
1  <style>
2      #sx{
3          width: 0;
4          height: 0;
5          border-left: 70px solid transparent;
6          border-right: 70px solid transparent;
7          border-top: 100px solid red;
8          border-radius: 50%;
9      }
10 </style>
11
12 <div id="sx"></div>
```

24、css图形画法 – 梯形

```
1  <style>
2      #tx {
```

```
3        width: 100px;
4        height: 0;
5        border-left: 50px solid transparent;
6        border-right: 50px solid transparent;
7        border-bottom: 100px solid red;
8    }
9  </style>
10
11  <div id="tx"></div>
```

25、css图形画法 – 平行四边形

```
1  <style>
2      #sbx {
3          width: 150px;
4          height: 100px;
5          transform: skew(20deg);
6          background-color: red;
7      }
8  </style>
9
10  <div id="sbx"></div>
```

26、css图形画法 – 五边形

```
1  <style>
2      #wbx{
3          position: relative;
4          width: 54px;
5          box-sizing: content-box;
6          border-width: 50px 18px 0;
7          border-style: solid;
8          border-color: red transparent;
9          margin-top: 50px;
10     }
11     #wbx:before {
12         content: "";
13         position: absolute;
14         width: 0;
```

```
15              height: 0;
16              top: -85px;
17              left: -18px;
18              border-width: 0 45px 35px;
19              border-style: solid;
20              border-color: transparent transparent red;
21          }
22      </style>
23
24      <div id="wbx"></div>
```

27、css图形画法 – 六边形

```
1  <style>
2          #hexagon {
3              width: 100px;
4              height: 55px;
5              background: red;
6              position: relative;
7              margin-top: 50px;
8          }
9
10         #hexagon:before {
11             content: "";
12             position: absolute;
13             top: -25px;
14             left: 0;
15             width: 0;
16             height: 0;
17             border-left: 50px solid transparent;
18             border-right: 50px solid transparent;
19             border-bottom: 25px solid red;
20         }
21
22         #hexagon:after {
23             content: "";
24             position: absolute;
25             bottom: -25px;
26             left: 0;
27             width: 0;
28             height: 0;
29             border-left: 50px solid transparent;
```

```
30            border-right: 50px solid transparent;
31            border-top: 25px solid red;
32        }
33    </style>
34
35    <div id="hexagon"></div>
```

28、css图形画法 – 返回箭头

```
1  <style>
2        #curvedarrow {
3            position: relative;
4            width: 0;
5            height: 0;
6            border-top: 9px solid transparent;
7            border-right: 9px solid red;
8            transform: rotate(10deg);
9        }
10
11        #curvedarrow:after {
12            content: "";
13            position: absolute;
14            border: 0 solid transparent;
15            border-top: 3px solid red;
16            border-radius: 20px 0 0 0;
17            top: -12px;
18            left: -9px;
19            width: 12px;
20            height: 12px;
21            transform: rotate(45deg);
22        }
23    </style>
24
25    <div id="curvedarrow"></div>
```

29、css图形画法 – 五角星

```
1  <style>
2        #star-five {
```

```
          margin: 50px 0;
          position: relative;
          display: block;
          color: red;
          width: 0px;
          height: 0px;
          border-right: 100px solid transparent;
          border-bottom: 70px solid red;
          border-left: 100px solid transparent;
          transform: rotate(35deg);
      }

  #star-five:before {
          border-bottom: 80px solid red;
          border-left: 30px solid transparent;
          border-right: 30px solid transparent;
          position: absolute;
          height: 0;
          width: 0;
          top: -45px;
          left: -65px;
          display: block;
          content: '';
          transform: rotate(-35deg);
      }

  #star-five:after {
          position: absolute;
          display: block;
          color: red;
          top: 3px;
          left: -105px;
          width: 0px;
          height: 0px;
          border-right: 100px solid transparent;
          border-bottom: 70px solid red;
          border-left: 100px solid transparent;
          transform: rotate(-70deg);
          content: '';
      }
  </style>

  <div id="star-five"></div>
```

30、css图形画法 – 心形

```
1  <style>
2        #heart {
3              position: relative;
4              width: 100px;
5              height: 90px;
6        }
7
8        #heart:before,
9        #heart:after {
10             position: absolute;
11             content: "";
12             left: 50px;
13             top: 0;
14             width: 50px;
15             height: 80px;
16             background: red;
17             border-radius: 50px 50px 0 0;
18             transform: rotate(-45deg);
19             transform-origin: 0 100%;
20       }
21
22       #heart:after {
23             left: 0;
24             transform: rotate(45deg);
25             transform-origin: 100% 100%;
26       }
27   </style>
28
29   <div id="heart"></div>
```

31、css图形画法 – 结形

```
1  <style>
2        #infinity {
3              position: relative;
4              width: 212px;
5              height: 100px;
6              box-sizing: content-box;
```

```
 7            }
 8
 9        #infinity:before,
10        #infinity:after {
11            content: "";
12            box-sizing: content-box;
13            position: absolute;
14            top: 0;
15            left: 0;
16            width: 60px;
17            height: 60px;
18            border: 20px solid red;
19            border-radius: 50px 50px 0 50px;
20            transform: rotate(-45deg);
21        }
22
23        #infinity:after {
24            left: auto;
25            right: 0;
26            border-radius: 50px 50px 50px 0;
27            transform: rotate(45deg);
28        }
29    </style>
30
31    <div id="infinity"></div>
```

32、同级变淡效果

```
 1 <style>
 2        span {
 3            padding: 0 1rem;
 4            transition: opacity 0.2s;
 5        }
 6        .sibling-fade:hover span:not(:hover) {
 7            opacity: 0.5;
 8        }
 9    </style>
10
11    <div class="sibling-fade">
12        <span>Item 1</span>
13        <span>Item 2</span>
14        <span>Item 3</span>
```

```
15          <span>Item 4</span>
16          <span>Item 5</span>
17          <span>Item 6</span>
18      </div>
```

33、环形旋转

```
1  <style>
2      @keyframes donut-spin {
3          0% {
4              transform: rotate(0deg);
5          }
6          100% {
7              transform: rotate(360deg);
8          }
9      }
10     .donut {
11         display: inline-block;
12         border: 4px solid rgba(0, 0, 0, 0.1);
13         border-left-color: #7983ff;
14         border-radius: 50%;
15         width: 30px;
16         height: 30px;
17         animation: donut-spin 1.2s linear infinite;
18     }
19  </style>
20
21  <div class="donut"></div>
```

34、文本省略

```
1  /*文本-省略号 截断*/
2      .ellipsis{
3          word-wrap: normal;/* for IE */
4          -webkit-text-overflow: ellipsis;
5          -moz-text-overflow: ellipsis;
6          -o-text-overflow: ellipsis;
7          text-overflow: ellipsis;
```

```
 8        white-space: nowrap;
 9        overflow: hidden;
10     }
11   /*多行位置截断*/
12   .line-clamp {
13        overflow : hidden;
14        -webkit-text-overflow: ellipsis;
15        -moz-text-overflow: ellipsis;
16        -o-text-overflow: ellipsis;
17        text-overflow: ellipsis;
18        display: -webkit-box;
19        -webkit-line-clamp: 2;/*行数*/
20        -webkit-box-orient: vertical;
21     }
```

35、透明度优化

```
1 .opa{
2        opacity: .2;
3        -ms-filter: progid: DXImageTransform.Microsoft.Alpha(Opacity = 2
4        filter: alpha(opacity = 20);
5        -moz-opacity: .2;
6        -khtml-opacity: .2;
7     }
```

36、input-placeholder样式设置

```
1 /* WebKit browsers */
2     ::-webkit-input-placeholder {
3         color: #ababab;
4     }
5     /* Mozilla Firefox 4 to 18 */
6     :-moz-placeholder {
7         color: #ababab;
8         opacity: 1;
9     }
10    /* Mozilla Firefox 19+ */
11    ::-moz-placeholder {
12        color: #ababab;
```

```
13          opacity: 1;
14      }
15      /* Internet Explorer 10+ */
16      :-ms-input-placeholder {
17          color: #ababab;
18      }
```

37、1像素优化

```
1  .retainbt,.retainbb,.retainbl,.retainbr,.retainb { position:relative; }
2      .retainbt:before,.retainbb:after {pointer-events: none;position: ab
3      .retainbt:before {top: 0}
4      .retainbb:after {bottom: 0}
5      .retainbl:before,.retainbr:after {pointer-events: none;position: ab
6      .retainbl:before {left: 0}
7      .retainbr:after {right: 0}
8      .retainb:after {position: absolute;content: "";top: 0;left: 0; -web
9      @media (-webkit-min-device-pixel-ratio:1.5),(min-device-pixel-ratio
10         .retainbt:before,.retainbb:after {-webkit-transform:scaleY(.5);
11         .retainbl:before,.retainbr:after {-webkit-transform: scaleX(.5)
12         .retainb:after { width: 200%; height: 200%;-webkit-transform: s
13         .retainbt:before,.retainbl:before,.retainb:after {-webkit-trans
14         .retainbb:after,.retainbr:after { -webkit-transform-origin: 100
15      }
16      @media (-webkit-device-pixel-ratio:1.5) {
17         .retainbt:before,.retainbb:after { -webkit-transform: scaleY(.6
18         .retainbl:before,.retainbr:after {-webkit-transform: scaleX(.66
19         .retainb:after {width: 150%; height: 150%;-webkit-transform: sc
20      }
21      @media (-webkit-device-pixel-ratio:3) {
22         .retainbt:before,.retainbb:after { -webkit-transform: scaleY(.3
23         .retainbl:before,.retainbr:after { -webkit-transform: scaleX(.3
24         .retainb:after {width: 300%;height: 300%; -webkit-transform: sc
25      }
```

38、消除transition闪屏

```
1  .transition-css{
2          /*设置内嵌的元素在 3D 空间如何呈现: 保留 3D*/
```

```
3        -webkit-transform-style: preserve-3d;
4        /* (设置进行转换的元素的背面在面对用户时是否可见：隐藏）*/
5        -webkit-backface-visibility: hidden;
6    }
```

39、开启硬件加速 / 解决页面闪白 / 保证动画流畅

```
1 .run-css{
2        -webkit-transform: translate3d(0, 0, 0);
3        -moz-transform: translate3d(0, 0, 0);
4        -ms-transform: translate3d(0, 0, 0);
5        transform: translate3d(0, 0, 0);
6    }
```

40、图像灰度

```
1 img {
2    filter: gray;
3    -webkit-filter: grayscale(1);
4 }
```

41、长文本自动换行

```
1 pre {
2    white-space: pre-line;
3    word-wrap: break-word;
4 }
```

42、css实现省略号动画

```
1 .point:after {
2    overflow: hidden;
```

```
 3        display: inline-block;
 4        vertical-align: bottom;
 5        animation: ellipsis 2s infinite;
 6        content: "\2026";
 7 }
 8 @keyframes ellipsis {
 9     from {
10         width: 2px;
11     }
12     to {
13         width: 15px;
14     }
15 }
```

43、清除浮动

```
1 .clearfix:before, .container:after { content: ""; display: table; }
2 .clearfix:after { clear: both; }
3 .clearfix { zoom: 1; }
```

44、css元素透明

```
1 .transparent {
2     filter: alpha(opacity=50);
3     -khtml-opacity: 0.5;
4     -moz-opacity: 0.5;
5     opacity: 0.5;
6 }
```

45、通用媒体查询

```
1
2 @media only screen and (min-device-width : 320px) and (max-device-width
3   /* Styles */
4 }
```

```
 5 @media only screen and (min-width : 321px) {
 6   /* Styles */
 7 }
 8 @media only screen and (max-width : 320px) {
 9   /* Styles */
10 }
11 /* iPad */
12 @media only screen and (min-device-width : 768px) and (max-device-width
13   /* Styles */
14 }
15 @media only screen and (min-device-width : 768px) and (max-device-width
16   /* Styles */
17 }
18
19 @media only screen and (min-device-width : 768px) and (max-device-width
20   /* Styles */
21 }
22 /* 桌面 */
23 @media only screen and (min-width : 1224px) {
24   /* Styles */
25 }
26
27 @media only screen and (min-width : 1824px) {
28   /* Styles */
29 }
30
31 @media only screen and (-webkit-min-device-pixel-ratio:1.5), only scree
32   /* Styles */
33 }
```

46、全屏背景图

```
1 html {
2     background: url('bg.jpg') no-repeat center center fixed;
3     background-size: cover;
4 }
```

47、css3渐变模板

```css
.bg {
    background: #629721;
    background-image: -webkit-gradient(linear, left top, left bottom, fr
    background-image: linear-gradient(top, #83b842, #629721);
}
```

48、@font-face使用

```css
@font-face {
    font-family: 'MyWebFont';
    src: url('webfont.eot');
    src: url('webfont.eot?#iefix')
    url('webfont.woff') format('woff'),
    url('webfont.ttf')  format('truetype'),
    url('webfont.svg#svgFontName') format('svg');
}
body {
    font-family: 'MyWebFont', Arial, sans-serif;
}
```

49、css3阴影

```css
/* 内盒阴影 */
.div {
    box-shadow: inset 2px 0 4px #000;
}

/* 外盒阴影 */
.div {
    box-shadow: 0 2px 2px -2px rgba(0, 0, 0, 0.52);
}
```

50、css3对话气泡

```css
.line {
```

```css
    background-color: #ededed;
    border: 2px solid #666;
    font-size: 35px;
    line-height: 1.3em;
    margin: 10px auto;
    padding: 10px;
    position: relative;
    text-align: center;
    width: 300px;
    border-radius: 20px;
    box-shadow: 0 0 5px #888;
}
.chat-bubble-arrow-border {
    border-color: #666 transparent transparent transparent;
    border-style: solid;
    border-width: 20px;
    height: 0;
    width: 0;
    position: absolute;
    bottom: -42px;
    left: 30px;
}
.chat-bubble-arrow {
    border-color: #ededed transparent transparent transparent;
    border-style: solid;
    border-width: 20px;
    height: 0;
    width: 0;
    position: absolute;
    bottom: -39px;
    left: 30px;
}
```

51、css3悬浮提示文本

```css
a {
    border-bottom:1px solid #bbb;
    color:#666;
    display:inline-block;
    position:relative;
    text-decoration:none;
}
```

```css
  8 a:hover,
  9 a:focus {
 10     color:#36c;
 11 }
 12 a:active {
 13     top:1px;
 14 }
 15
 16 a[data-tooltip]:after {
 17     border-top: 8px solid #222;
 18     border-top: 8px solid hsla(0,0%,0%,.85);
 19     border-left: 8px solid transparent;
 20     border-right: 8px solid transparent;
 21     content: "";
 22     display: none;
 23     height: 0;
 24     width: 0;
 25     left: 25%;
 26     position: absolute;
 27 }
 28 a[data-tooltip]:before {
 29     background: #222;
 30     background: hsla(0,0%,0%,.85);
 31     color: #f6f6f6;
 32     content: attr(data-tooltip);
 33     display: none;
 34     font-family: sans-serif;
 35     font-size: 14px;
 36     height: 32px;
 37     left: 0;
 38     line-height: 32px;
 39     padding: 0 15px;
 40     position: absolute;
 41     text-shadow: 0 1px 1px hsla(0,0%,0%,1);
 42     white-space: nowrap;
 43     -webkit-border-radius: 5px;
 44     -moz-border-radius: 5px;
 45     -o-border-radius: 5px;
 46     border-radius: 5px;
 47 }
 48 a[data-tooltip]:hover:after {
 49     display: block;
 50     top: -9px;
 51 }
 52 a[data-tooltip]:hover:before {
 53     display: block;
```

```
54        top: -41px;
55    }
56    a[data-tooltip]:active:after {
57        top: -10px;
58    }
59    a[data-tooltip]:active:before {
60        top: -42px;
61    }
```

52、论文页面的卷曲效果

```
 1    ul.box {
 2        position: relative;
 3        z-index: 1;
 4        overflow: hidden;
 5        list-style: none;
 6        margin: 0;
 7        padding: 0;
 8    }
 9    ul.box li {
10        position: relative;
11        float: left;
12        width: 250px;
13        height: 150px;
14        padding: 0;
15        border: 1px solid #efefef;
16        margin: 0 30px 30px 0;
17        background: #fff;
18        -webkit-box-shadow: 0 1px 4px rgba(0, 0, 0, 0.27), 0 0 40px rgba(0,
19        -moz-box-shadow: 0 1px 4px rgba(0, 0, 0, 0.27), 0 0 40px rgba(0, 0,
20        box-shadow: 0 1px 4px rgba(0, 0, 0, 0.27), 0 0 40px rgba(0, 0, 0, 0
21    }
22    ul.box li:before,
23    ul.box li:after {
24        content: '';
25        z-index: -1;
26        position: absolute;
27        left: 10px;
28        bottom: 10px;
29        width: 70%;
30        max-width: 300px; /* avoid rotation causing ugly appearance at larg
31        max-height: 100px;
```

```
32      height: 55%;
33      -webkit-box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);
34      -moz-box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);
35      box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);
36      -webkit-transform: skew(-15deg) rotate(-6deg);
37      -moz-transform: skew(-15deg) rotate(-6deg);
38      -ms-transform: skew(-15deg) rotate(-6deg);
39      -o-transform: skew(-15deg) rotate(-6deg);
40      transform: skew(-15deg) rotate(-6deg);
41 }
42 ul.box li:after {
43      left: auto;
44      right: 10px;
45      -webkit-transform: skew(15deg) rotate(6deg);
46      -moz-transform: skew(15deg) rotate(6deg);
47      -ms-transform: skew(15deg) rotate(6deg);
48      -o-transform: skew(15deg) rotate(6deg);
49      transform: skew(15deg) rotate(6deg);
50 }
```

53、max-width防止图片撑破容器

```
1 img{
2      max-width: 100%;
3      display: inline-block;
4 }
```

54、css优化、提高性能的方法

加载性能

（1）css压缩：将写好的css进行打包压缩，可以减少很多体积

（2）css单一样式：margin-left，margin-bottom会比margin:12px 0 12px 0执行效率更高

（3）减少使用@import，建议使用link，因为后者是在页面加载时一起加载，前者是等待页面加载完成之后再进行加载

选择器性能