

第八天笔记

css3选择器

- nth-child 这个代表的是选中 **所有儿子元素** 中的第几个;
- nth-of-type 是选中**儿子元素中特定类型**中的第几个

```
/* .main>p:nth-child(4){
    background:cadetblue;
} */
/* 正数所有儿子中的第几个 */
/* ul>li:nth-child(3){
    background:green;
} */
/* 最后一个 */
/* ul>li:last-child{
    background:coral;
} */
/* 第一个 */
/* ul>li:first-child{
    background:cornflowerblue;
} */
/* 倒数第几个 */
/* ul>li:nth-last-child(2){
    font-size:100px;
} */
/* 偶数 even */
ul>li:nth-child(2n){
    background:cornflowerblue;
}
/* 奇数 odd*/
ul>li:nth-child(2n+1){
    background:lightgreen;
}
.main>p:nth-child(4){
    background:cadetblue;
}

</style>
<ul>
<li>1</li>
<li>1</li>
<li>1</li>
<li>1</li>
<li>1</li>
<li>1</li>
<li>1</li>
</ul>
```

```
<div class="main">
<h1>1</h1>
<h2>2</h2>
<p>p1</p>
<h4>h4</h4>
<h3>3</h3>
<h4>4</h4>
<h5>5</h5>
<h6>6</h6>
<p>p3</p>
<p>p4</p>
</div>
```

```
/* 所有儿子中的第几个 */
/* 所有儿子中的第三个，但是必须是p元素，但是找不到，所以不起作用 */
.main>p:nth-child(3){
    background: red;
}
/* 特定类型的第几个 */
.main>p:nth-of-type(3){
    background: yellow;
}
/* 倒数 */
.main>p:nth-last-of-type(1){
    background: lawngreen;
}
.main>p:nth-of-type(2n){
    background: lightcoral;
}
/* 第一个 */
.main>p:first-of-type{
    background: lightpink;
}
/* 最后一个 */
.main>p:last-of-type{
    background: lightyellow;
}
```

iconfont使用

font-class 引用

font-class 是 Unicode 使用方式的一种变种，主要是解决 Unicode 书写不直观，语意不明确的问题。

与 Unicode 使用方式相比，具有如下特点：

- 兼容性良好，支持 IE8+，及所有现代浏览器。
- 相比于 Unicode 语意明确，书写更直观。可以很容易分辨这个 icon 是什么。
- 因为使用 class 来定义图标，所以当要替换图标时，只需要修改 class 里面的 Unicode 引用。
- 不过因为本质上还是使用的字体，所以多色图标还是不支持的。

使用步骤如下：

第一步：引入项目下面生成的 fontclass 代码：

```
<link rel="stylesheet" href="./iconfont.css">
```

第二步：挑选相应图标并获取类名，应用于页面：

```
<span class="iconfont icon-xxx"></span>
```

"iconfont" 是你项目下的 font-family。可以通过编辑项目查看，默认是 "iconfont"。

1、刚开始使用iconfont的时候，可以先使用生成的在线地址，在项目中外链入地址，然后在页面中

```
<span class="iconfont icon-xxx"></span>
```

2、整体使用完icon之后，下载下来，把其中的iconfont.css文件放入项目的css文件夹中，在项目中外链入iconfont.css文件，应在index.css之前引入

```
<title>Document</title>
| <!-- 当地址改变的时候需要重新生成 -->
<!-- <link rel="stylesheet" href="http://at.alicdn.com/t/
font_2162895_jx2sfpmayye.css"> -->
<!-- 下载完成之后 -->
<link rel="stylesheet" href="../iconfont.css">
<style>
|   .icon-arrow-right{
|       color: red;
|   }
|   .icon-gouwuche-01{
|       font-size: 33px;
|   }
| </style>
</head>
<body>
|   <span class="iconfont icon-arrow-right"></span>
|   <span class="iconfont icon-gouwuche-01"></span>
</body>
```

定位position

- 值：
 - static 静态定位
 - relative 相对定位
 - absolute 绝对定位
 - fixed 固定定位
- 相对定位relative
 - 不脱离普通文档流
 - 相对定位的元素层级高于普通文档流
 - 给绝对定位的元素做参照物
 - 相对定位，相对自身原有位置
- 绝对定位：
 - 参照物是别人，但是参照物在设置的时候有一个规则，必须是祖先级元素
 - 如果有多个参照物，浏览器按照“就近原则”，如果都没有设置参照物，那参照物就是body
 - 在设置参照物的时候，参照物本身已经是绝对定位或者固定定位，就不用再写position:relative
 - 绝对定位的元素不占位置，脱离文档流
 - 层级也高于普通文档流的
 - 给绝对定位的元素宽度用百分比来表示的时候，指的是参照物的百分比（如果参照物和父级不是一个）
 - 行内元素写宽高本来不起作用，一旦设置了定位，写宽高也可以起作用
 - 块级元素不写宽度会自动充满父级，但是一旦给此元素设置了定位，它的宽度就是由自身内容决定

面试题：如何让一个元素在整个屏幕或者是一个盒子中水平垂直居中

让这个元素

`position: absolute;`

`left: 50%;`

`top: 50%;`

`margin-left:` 负的盒子宽度的一半

`margin-top:` 负的盒子高度的一半

```
<style>
  .box{
    width:200px;
    height:200px;
    background:green;
    position: absolute;
    left:50%;
    top:50%;
    margin-left:-100px;
    margin-top:-100px;
  }
</style>
```

当只让其垂直居中显示的时候，可以给这个元素设置

`position: absolute;`

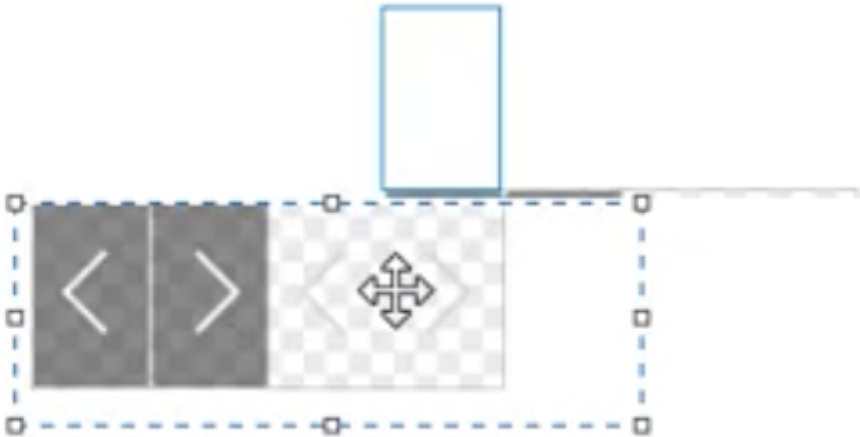
`top: 50%;`

`margin-top:` 负的高度的一半

精灵图

在做项目的时候，有时候咱们为了优化，减少http请求，会把多张图片合并到一张图片上面，这种图就是“精灵图”，也称“雪碧图”。(只针对小图)

精灵图的原理：通过改变background-position。再记住一点，在写精灵图的时候，background-position的x轴，y轴的值一定的负的。



```
.arrowLeft,.arrowRight{
  width: 41px;
  height: 69px;
  background: url(../images/icon-slides.png) no-repeat;
  position: absolute;
  top: 50%;
  margin-top: -35px;
}
.arrowLeft{
  left: 234px;
  background-position: -84px 0;
}
.arrowLeft:hover{
  background-position: 0 0;
}
.arrowRight{
  right: 0;
  background-position: -124px 0;
}
.arrowRight:hover{
  background-position: -42px 0;
}
```

border-radius: 50%(圆)