

day09-前端基础CSS第七天

学习目标：

能够使用精灵图

能够使用字体图标

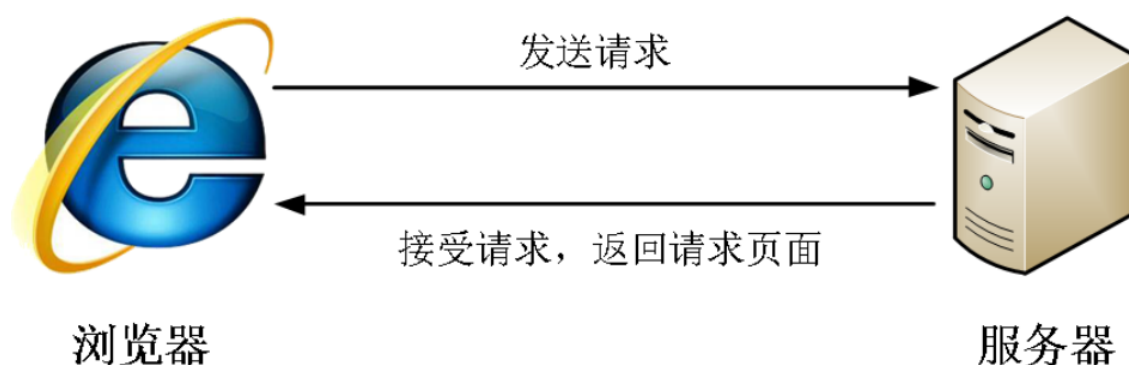
能够写出 CSS 三角

能够写出常见的 CSS 用户界面样式

能够说出常见的布局技巧

1. 精灵图（重点）

1.1 为什么需要精灵图



一个网页中往往会应用很多小的背景图像作为修饰，当网页中的图像过多时，服务器就会频繁地接收和发送请求图片，造成服务器请求压力过大，这将大大降低页面的加载速度。

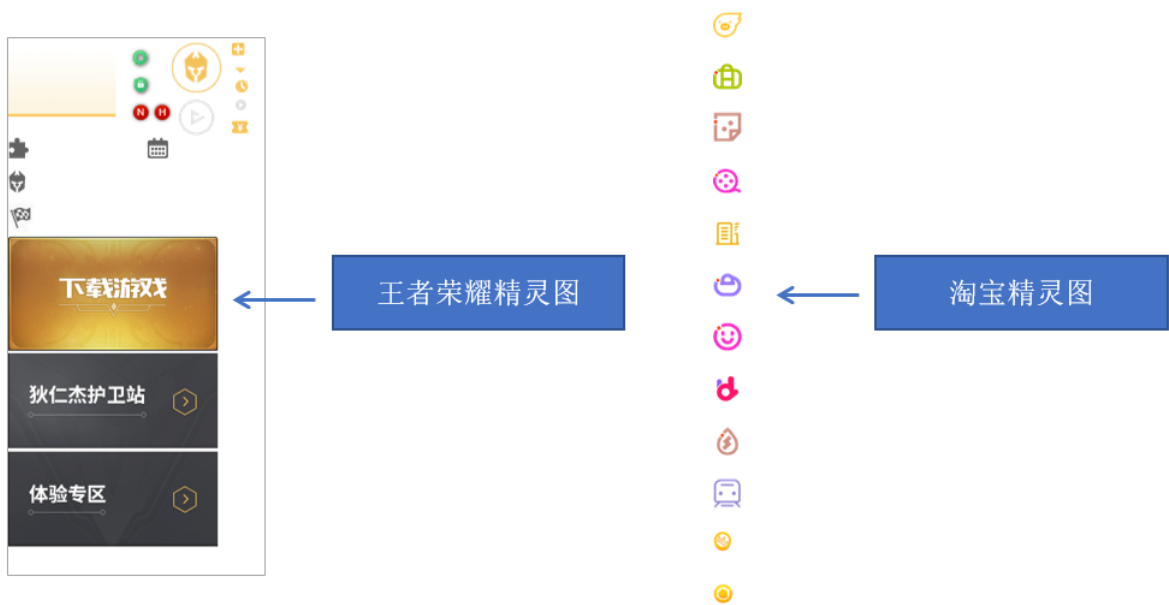
为什么使用精灵图（目的）：

为了有效地减少服务器接收和发送请求的次数，提高页面的加载速度，出现了 CSS 精灵技术（也称 CSS Sprites、CSS 雪碧）。

核心原理：

将网页中的一些小背景图像整合到一张大图中，这样服务器只需要一次请求就可以了。

精灵图举例：



1.2 精灵图 (sprites) 的使用

使用精灵图核心：

1. 精灵技术主要针对于背景图片使用。就是把多个小背景图片整合到一张大图片中。
2. 这个大图片也称为 sprites 精灵图 或者 雪碧图
3. 移动背景图片位置， 此时可以使用 background-position 。
4. 移动的距离就是这个目标图片的 x 和 y 坐标。注意网页中的坐标有所不同
5. 因为一般情况下都是往上往左移动，所以数值是负值。
6. 使用精灵图的时候需要精确测量，每个小背景图片的大小和位置。

使用精灵图核心总结：

1. 精灵图主要针对于小的背景图片使用。
2. 主要借助于背景位置来实现---background-position 。
3. 一般情况下精灵图都是负值。（千万注意网页中的坐标：x轴右边走是正值，左边走是负值，y轴同理。）

1.3 案例：拼出自己名字

1.3.1 案例效果



1.3.2 代码参考

结构

```
<span class="p">p</span>
<span class="i">i</span>
<span class="n">n</span>
<span class="k">k</span>
```

样式

```
span {
  display: inline-block;
  background: url(images/abcd.jpg) no-repeat;
}
.p {
  width: 100px;
  height: 112px;
  /* background-color: pink; */
  background-position: -493px -276px;
}
.i {
  width: 60px;
  height: 108px;
  /* background-color: pink; */
  background-position: -327px -142px;
}
.n {
  width: 108px;
  height: 109px;
  /* background-color: pink; */
  background-position: -215px -141px;
}
.k {
  width: 105px;
  height: 114px;
  /* background-color: pink; */
  background-position: -495px -142px;
}
```

2. 字体图标

2.1 字体图标的产生

字体图标使用场景： 主要用于显示网页中通用、常用的一些小图标。

精灵图是有诸多优点的，但是缺点很明显。

1. 图片文件还是比较大的。
2. 图片本身放大和缩小会失真。
3. 一旦图片制作完毕想要更换非常复杂。

此时，有一种技术的出现很好的解决了以上问题，就是**字体图标 iconfont**。

字体图标可以为前端工程师提供一种方便高效的图标使用方式，**展示的是图标，本质属于字体**。

2.2 字体图标的优点

轻量级：一个图标字体要比一系列的图像要小。一旦字体加载了，图标就会马上渲染出来，减少了服务器请求

- 灵活性：本质其实是文字，可以很随意的改变颜色、产生阴影、透明效果、旋转等
- 兼容性：几乎支持所有的浏览器，请放心使用
- 注意：字体图标不能替代精灵技术，只是对工作中图标部分技术的提升和优化。

总结：

1.如果遇到一些结构和样式比较简单的小图标，就用字体图标。



2.如果遇到一些结构和样式复杂一点的小图片，就用精灵图。



使用步骤

字体图标是一些网页常见的小图标，我们直接网上下载即可。因此使用可以分为：

- 1.字体图标的下载
- 2.字体图标的引入（引入到我们html页面中）
- 3.字体图标的追加（以后添加新的小图标）

2.3 字体图标的下载

推荐下载网站：

- **icomoon 字库** <http://icomoon.io> 推荐指数 ★★★★★

IcoMoon 成立于 2011 年，推出了第一个自定义图标字体生成器，它允许用户选择所需要的图标，使它们成一字型。该字库内容种类繁多，非常全面，唯一的遗憾是国外服务器，打开网速较慢。

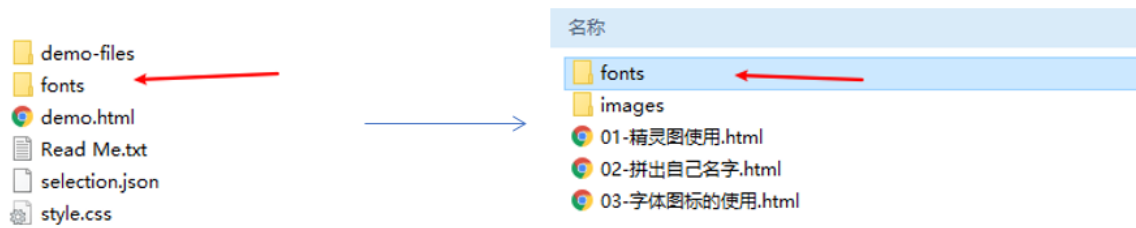
- **阿里 iconfont 字库** <http://www.iconfont.cn/> 推荐指数 ★★★★★

这个是阿里妈妈 M2UX 的一个 iconfont 字体图标字库，包含了淘宝图标库和阿里妈妈图标库。可以使用 AI 制作图标上传生成。重点是，免费！

2.4 字体图标的引入

下载完毕之后，注意原先的文件不要删，后面会用。

1. 把下载包里面的 **fonts** 文件夹放入页面根目录下



• 字体文件格式

不同浏览器所支持的字体格式是不一样的，字体图标之所以兼容，就是因为包含了主流浏览器支持的字体文件。

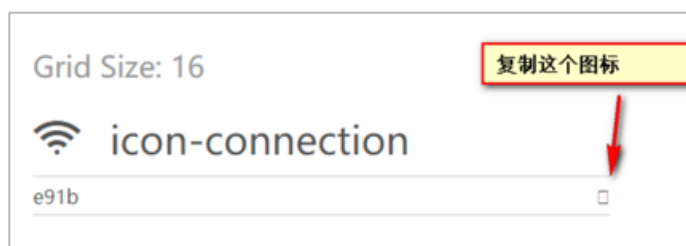
- 1).TureType(**.ttf**)格式.ttf字体是Windows和Mac的最常见的字体，支持这种字体的浏览器有IE9+、Firefox3.5+、Chrome4+、Safari3+、Opera10+、iOS Mobile、Safari4.2+;
- 2).Web Open Font Format(**.woff**)格式woff字体，支持这种字体的浏览器有IE9+、Firefox3.5+、Chrome6+、Safari3.6+、Opera11.1+;
- 3).Embedded Open Type(**.eot**)格式.eot字体是IE专用字体，支持这种字体的浏览器有IE4+;
- 4).SVG(**.svg**)格式.svg字体是基于SVG字体渲染的一种格式，支持这种字体的浏览器有Chrome4+、Safari3.1+、Opera10.0+、iOS Mobile Safari3.2+;

2.在 CSS 样式中全局声明字体：简单理解把这些字体文件通过css引入到我们页面中。

一定注意字体文件路径的问题

```
@font-face {
  font-family: 'icomoon';
  src: url('fonts/icomoon.eot?7kkyc2');
  src: url('fonts/icomoon.eot?7kkyc2#iefix') format('embedded-opentype'),
    url('fonts/icomoon.ttf?7kkyc2') format('truetype'),
    url('fonts/icomoon.woff?7kkyc2') format('woff'),
    url('fonts/icomoon.svg?7kkyc2#icomoon') format('svg');
  font-weight: normal;
  font-style: normal;
}
```

3. html 标签内添加小图标。



```
<span> · </span>
```

4. 给标签定义字体。

```
span {  
  font-family: "icomoon";  
}
```

注意：务必保证 这个字体和上面@font-face里面的字体保持一致

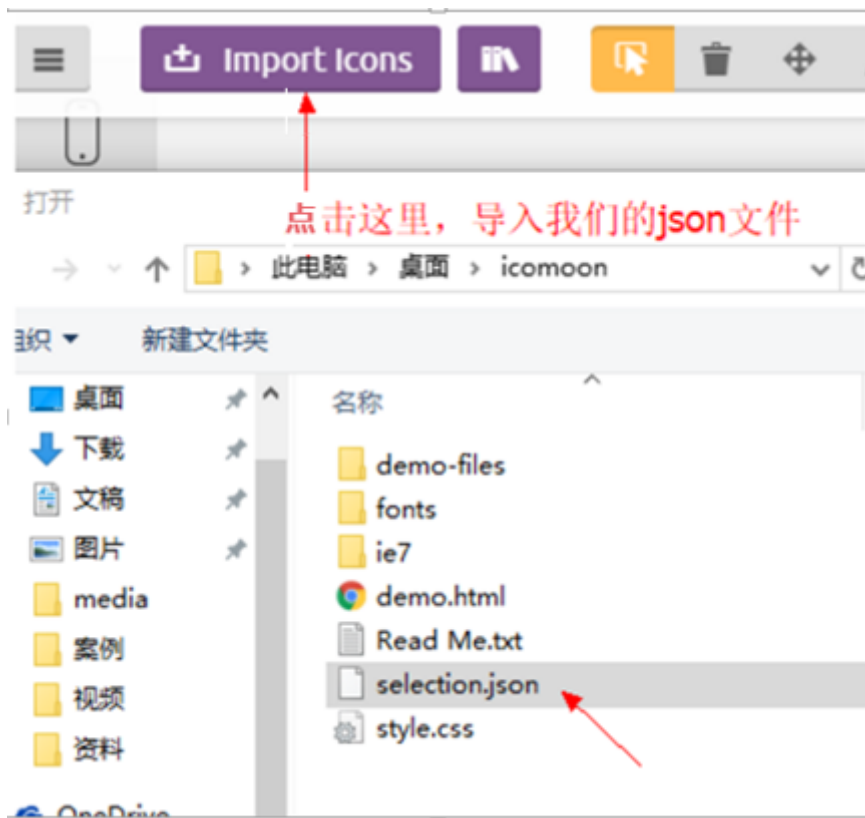


```
@font-face {  
  font-family: 'icomoon';  
  src: url('fonts/icomoon.eot?7kkyc2');  
  src: url('fonts/icomoon.eot?7kkyc2#ief')  
    url('fonts/icomoon.ttf?7kkyc2') format('truetype')  
    url('fonts/icomoon.woff?7kkyc2') format('woff')  
    url('fonts/icomoon.svg?7kkyc2#icomoon') format('svg');  
  font-weight: normal;  
  font-style: normal;  
}  
  
/* span使用字体图标 */  
span {  
  font-family: 'icomoon';  
  font-size: 100px;  
}
```

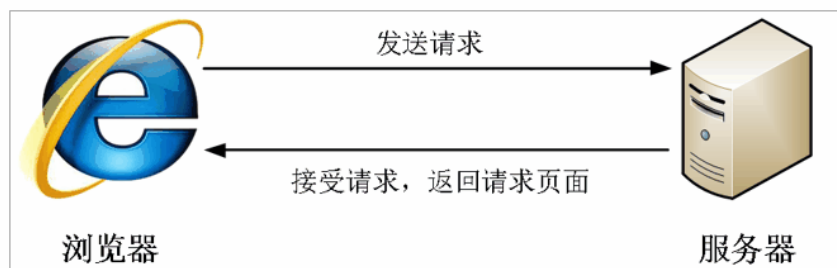
2.5 字体图标的追加

如果工作中，原来的字体图标不够用了，我们需要添加新的字体图标到原来的字体文件中。

把压缩包里面的 **selection.json** 从新上传，然后选中自己想要新的图标，从新下载压缩包，并替换原来的文件即可。



2.6 字体图标加载的原理：



icomoon.eot
icomoon.svg
icomoon.ttf
icomoon.woff

3. CSS 三角

3.1 介绍

网页中常见一些三角形，使用 CSS 直接画出来就可以，不必做成图片或者字体图标。

一张图，你就知道 CSS 三角是怎么来的了，做法如下：

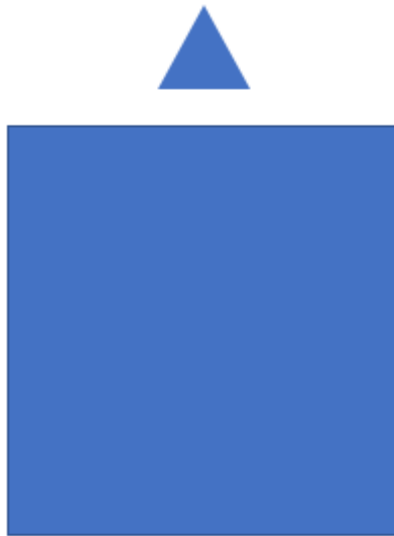


```
div {  
  width: 0;  
  height: 0;  
  border: 50px solid transparent;  
  border-color: red green blue black;  
  line-height: 0;  
  font-size: 0;  
}
```

1. 我们用css 边框可以模拟三角效果
2. 宽度高度为0
3. 我们4个边框都要写，只保留需要的边框颜色，其余的不能省略，都改为 transparent 透明就好了
4. 为了照顾兼容性 低版本的浏览器，加上 font-size: 0; line-height: 0;

3.2 案例：京东三角

3.2.1效果图



3.2.2 代码参考

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  <title>CSS 三角制作</title>  
  <style>  
    .box1 {  
      width: 0;  
      height: 0;  
      /* border: 10px solid pink; */  
      border-top: 10px solid pink;
```



```

        border-right: 10px solid red;
        border-bottom: 10px solid blue;
        border-left: 10px solid green;
    }
    .box2 {
        width: 0;
        height: 0;
        border: 50px solid transparent;
        border-left-color: pink;
        margin: 100px auto;
    }
    .jd {
        position: relative;
        width: 120px;
        height: 249px;
        background-color: pink;
    }
    .jd span {
        position: absolute;
        right: 15px;
        top: -10px;
        width: 0;
        height: 0;
        /* 为了照顾兼容性 */
        line-height: 0;
        font-size: 0;
        border: 5px solid transparent;
        border-bottom-color: pink;
    }
</style>
</head>
<body>
    <div class="box1"></div>
    <div class="box2"></div>
    <div class="jd">
        <span></span>
    </div>
</body>
</html>

```

4. CSS 用户界面样式

什么是界面样式

所谓的界面样式，就是更改一些用户操作样式，以便提高更好的用户体验。

- 更改用户的鼠标样式
- 表单轮廓
- 防止表单域拖拽

4.1 鼠标样式 cursor

```
li {  
  cursor: pointer;  
}
```

设置或检索在对象上移动的鼠标指针采用何种系统预定义的光标形状。

属性值	描述
default	小白 默认
pointer	小手
move	移动
text	文本
not-allowed	禁止

4.2 轮廓线 outline

给表单添加 outline: 0; 或者 outline: none; 样式之后，就可以去掉默认的蓝色边框。

```
input {  
  outline: none;  
}
```

4.3 防止拖拽文本域 resize

实际开发中，我们文本域右下角是不可以拖拽的。

```
textarea{  
  resize: none;  
}
```

vertical-align 属性应用

5. vertical-align 属性应用

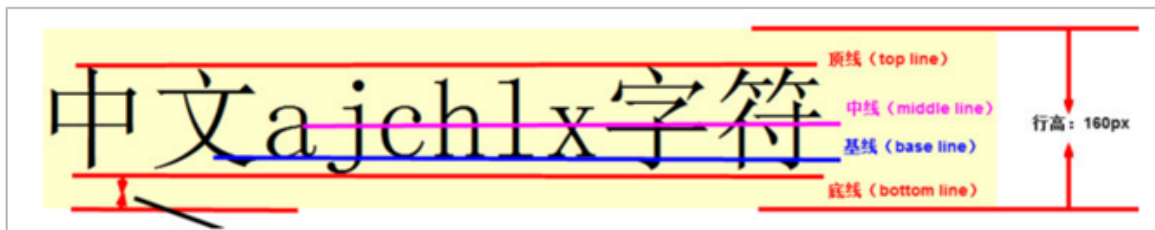
CSS 的 **vertical-align** 属性使用场景：经常用于设置图片或者表单(行内块元素) 和文字垂直对齐。

官方解释：用于设置一个元素的**垂直对齐方式**，但是它只针对于行内元素或者行内块元素有效。

语法：

```
vertical-align : baseline | top | middle | bottom
```

值	描述
baseline	默认。元素放置在父元素的基线上。
top	把元素的顶端与行中最高元素的顶端对齐
middle	把此元素放置在父元素的中部。
bottom	把元素的顶端与行中最低的元素的顶端对齐。



5.1 图片、表单和文字对齐

图片、表单都属于行内块元素，默认的 vertical-align 是基线对齐。



此时可以给图片、表单这些行内块元素的 vertical-align 属性设置为 middle 就可以让文字和图片垂直居中对齐了。

5.2 解决图片底部默认空白缝隙问题

bug：图片底侧会有一个空白缝隙，原因是行内块元素会和文字的基线对齐。

主要解决方法有两种：

1. 给图片添加 vertical-align: middle | top | bottom 等。（提倡使用的）
2. 把图片转换为块级元素 display: block;



6. 溢出的文字省略号显示

6.1 单行文本溢出显示省略号



单行文本溢出显示省略号--必须满足三个条件：

```
/*1. 先强制一行内显示文本*/
white-space: nowrap; （默认 normal 自动换行）

/*2. 超出的部分隐藏*/
overflow: hidden;

/*3. 文字用省略号替代超出的部分*/
text-overflow: ellipsis;
```

6.2 多行文本溢出显示省略号（了解）



日本84岁上皇后美智子癌症手术2天后出院，向医护人员...

多行文本溢出显示省略号，**有较大兼容性问题**，适合于webKit浏览器或移动端（移动端大部分是webkit内核）

```
/*1. 超出的部分隐藏 */
overflow: hidden;

/*2. 文字用省略号替代超出的部分 */
text-overflow: ellipsis;

/* 3. 弹性伸缩盒子模型显示 */
display: -webkit-box;

/* 4. 限制在一个块元素显示的文本的行数 */
-webkit-line-clamp: 2;

/* 5. 设置或检索伸缩盒对象的子元素的排列方式 */
-webkit-box-orient: vertical;
```

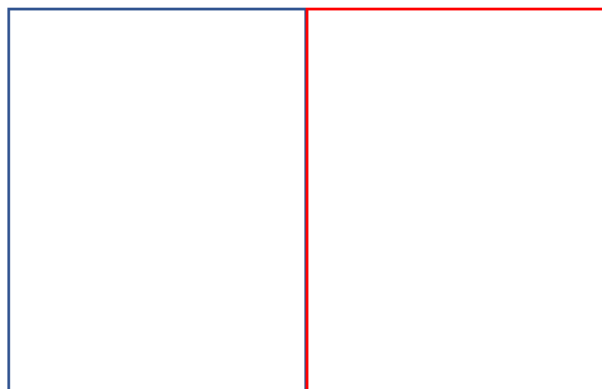
更推荐让后台人员来做这个效果，因为后台人员可以设置显示多少个字，操作更简单。

7. ☆ 常见布局技巧

巧妙利用一个技术更快更好的布局：

1. margin负值的运用
2. 文字围绕浮动元素
3. 行内块的巧妙运用
4. CSS三角强化

7.1. margin负值运用



- 1.让每个盒子margin 往左侧移动 -1px 正好压住相邻盒子边框
- 2.鼠标经过某个盒子的时候，提高当前盒子的层级即可（如果没有有定位，则加相对定位（保留位置），如果有定位，则加z-index）

7.2 文字围绕浮动元素

效果



【集锦】热身赛-巴西0-1秘
鲁 内马尔替补两人血染赛
场



【集锦】热身赛-阿根廷4-0
墨西哥 劳塔罗22分钟上演
帽子戏法

布局示意图



巧妙运用浮动元素不会压住文字的特性

7.3 行内块巧妙运用



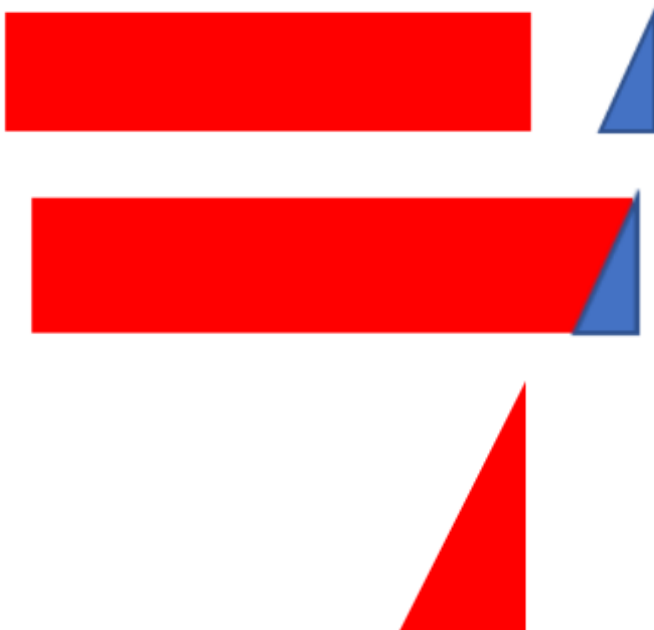
页码在页面中间显示:

1. 把这些链接盒子转换为行内块，之后给父级指定 `text-align:center`;
2. 利用行内块元素中间有缝隙，并且给父级添加 `text-align:center`; 行内块元素会水平会居中



7.4. CSS 三角强化 案例

7.4.1 原理



```
<style>
.box1 {
  width: 0;
```

```

height: 0;

/* 把上边框宽度调大 */
/* border-top: 100px solid transparent;
border-right: 50px solid skyblue; */
/* 左边和下边的边框宽度设置为0 */
/* border-bottom: 0 solid blue;
border-left: 0 solid green; */

/* 1.只保留右边的边框有颜色 */
border-color: transparent red transparent transparent;
/* 2. 样式都是solid */
border-style: solid;
/* 3. 上边框宽度要大, 右边框 宽度稍小, 其余的边框该为 0 */
border-width: 100px 50px 0 0 ;
}

</style>
</head>
<body>
    <div class="box1"></div>
</body>

```

7.4.2 案例效果



7.4.3 代码参考

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>CSS三角强化的巧妙运用</title>
    <style>
      .price {
        width: 160px;
        height: 24px;
        line-height: 24px;
        border: 1px solid red;
        margin: 0 auto;
      }
      .miaosha {
        position: relative;
        float: left;
        width: 90px;
        height: 100%;
        background-color: red;
        text-align: center;
        color: #fff;
        font-weight: 700;
        margin-right: 8px;
      }
    </style>
  </head>
  <body>
    <div>
      <div class="price">¥1650</div>
      <div class="miaosha">¥5650</div>
    </div>
  </body>
</html>

```

```

    }
    .miaosha i {
        position: absolute;
        right: 0;
        top: 0;
        width: 0;
        height: 0;
        border-color: transparent #fff transparent transparent;
        border-style: solid;
        border-width: 24px 10px 0 0;
    }
    .origin {
        font-size: 12px;
        color: gray;
        text-decoration: line-through;
    }
}
</style>
</head>
<body>
    <div class="price">
        <span class="miaosha">
            ¥1650
            <i></i>
        </span>
        <span class="origin">¥5650</span>
    </div>
</body>
</html>

```

8. CSS 初始化

不同浏览器对有些标签的默认值是不同的，为了消除不同浏览器对HTML文本呈现的差异，照顾浏览器的兼容，我们需要对CSS 初始化

简单理解：CSS初始化是指重设浏览器的样式。（也称为CSS reset）

每个网页都必须首先进行 CSS初始化。

这里我们以 京东CSS初始化代码为例。

Unicode编码字体：

把中文字体的名称用相应的Unicode编码来代替，这样就可以有效的避免浏览器解释CSS代码时候出现乱码的问题。

比如：

黑体 \9ED1\4F53

宋体 \5B8B\4F53

微软雅黑 \5FAE\8F6F\96C5\9ED1