

【前端面试】使用过粘连布局吗？如何实现？

前言

这个问题考察的是大家对 CSS 基础知识的掌握程度。粘连布局是众多 CSS 中布局方式中的一种，比如常见的还有单列布局、三栏布局等等，今天我们主要将的是粘连布局。因为在我们平常的页面布局中，粘连布局使用的地方还是挺多的，有些小伙伴明明都在项目中用过这种布局方式了，但是却不知它叫粘连布局，如果面试的时候没有回答上来就很亏。

所以今天我们就讲一讲 CSS 众多布局中的一种：**粘连布局**。

1.什么是粘连布局？

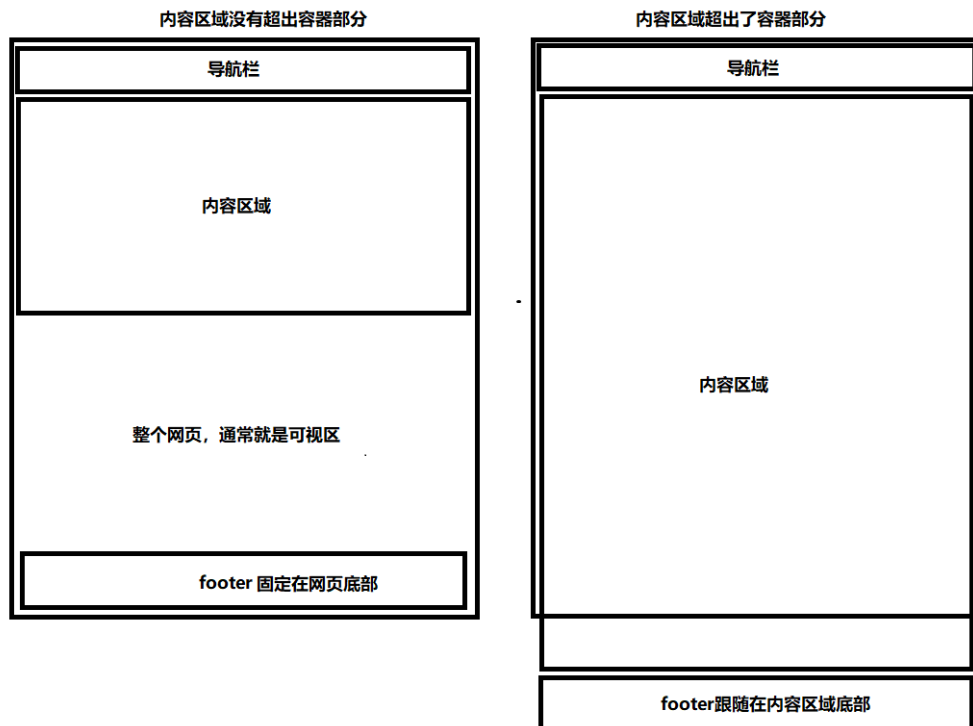
粘连布局可能有些小伙伴没有听过，但是我相信很多小伙伴即使没有听过粘连布局，但是在项目中可能已经用到过了。

我们可以给粘连布局一个**简单的定义**：

所谓粘连布局（也可以叫做 sticky footer 布局），就是当内容区域元素没有超出容器时，网页底部 footer 紧贴在网页底部，当内容区域超长时，底部 footer 就会紧跟在内容区域底部，而不是紧贴在网页底部。

上面的定义中有好几个关键词：内容区域、footer、超长等等。可能小伙伴们只看定义还不一定能理解粘连布局是什么，接下来我们看一张图就能够很好理解了。

粘连布局示意图：



看了上面那张图大家应该就能理解什么叫做粘连布局，其实说白了粘连布局就是解决网页 footer 的问题。

上面图中所示的问题在我们项目开发中很容易遇到，比如我们下面的网页：



上图中我们发现 footer 跟在内容区域后面，这个时候网页留下了一大片空白，这一般不是我们想要的，我们希望 footer 此时紧贴在网页的底部。

想要实现这种效果，就用到了我们的粘连布局。

2.实现粘连布局的方式

既然我们知道了粘连布局主要解决了什么问题，那么我们接下来就该想一想如何实现粘连布局。

实现粘连布局的方式不只一种，我们一起来学习一下。

我们先来理一理思路，粘连布局主要就是解决两个问题：

- 第一个问题：内容区域不够时，footer 紧贴底部
- 第二个问题：内容区域超长时，footer 紧贴内容区

2.1 负 margin-top

我们可以想一想第一个问题出现的原因是什么？无非就是内容区域的高度不够，那么我们把内容区域的高度变大一点岂不是就解决了这个问题了，至于第二个问题，内容超长后，在我们不做任何处理的情况下，footer 本来就是更在内容区域后面的。

实现原理：

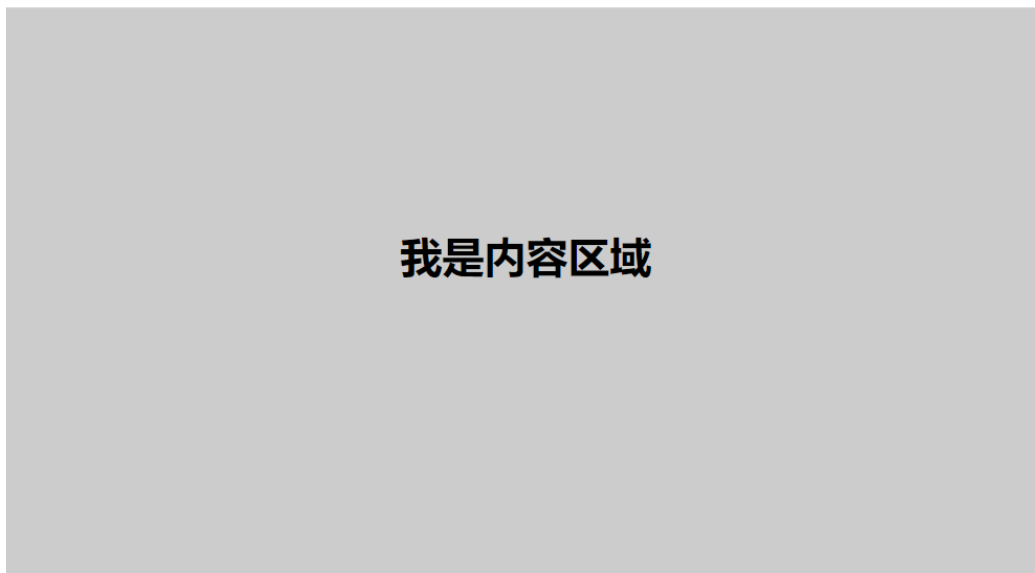
- 给内容区添加额外的包裹元素
- 设置包裹元素的 padding-bottom
- 设置 footer 元素的 margin-top

代码如下：

```
1 <head>
2   <style>
3     * {
4       padding: 0;
5       margin: 0;
6     }
7     html,
8     body {
9       height: 100%;
10    }
11    .content-wrap {
12      min-height: 100%;
13    }
14  }
15  .content {
16    height: 50vh;
17    width: 100%;
18    background-color: #ccc;
19    display: flex;
20    align-items: center;
21    justify-content: center;
22    padding-bottom: 60px;
23  }
24  .footer {
25    height: 60px;
26    width: 100%;
27    background-color: rgb(47, 46, 46);
28    display: flex;
29    align-items: center;
30    justify-content: center;
31    color: #fff;
32    margin-top: -60px;
```

```
33     }
34     </style>
35 </head>
36
37 <body>
38     <!-- 内容区域 -->
39     <div class="content-wrap">
40         <div class="content">
41             <h1>我是内容区域</h1>
42         </div>
43     </div>
44     <!-- footer -->
45     <div class="footer">
46         <h1>我是 footer</h1>
47     </div>
48 </body>
```

实现效果：



我是footer

上段代码主要分为了两部分：内容区域、footer 区域。我们给内容区域外层设置了包裹元素，并且将该包裹元素的最小高度设置为了 100%，这样就相当于内容区域就填充满了整个屏幕，footer 区域自然就跟随在内容区域后面。然后将 footer 的上外边距设置为了负的 footer 高度，这样就可以让 footer 感很好出现在屏幕的底部，然后我们给 content 设置了一个 padding，避免 footer 遮挡内容。

我们试试当内容区域超长是什么效果，效果如下：



2.2 负 margin-bottom

上面我们利用了 margin-top 来实现了粘连布局，那么同理，我们也可以利用 margin-bottom 来实现粘连布局。

实现原理：

- 将所有内容元素添加一个包裹元素
- 将包裹元素高度设置为 100%
- 通过设置包裹元素的 margin-bottom 让 footer 显现出来

代码如下：

```
1 <head>
2   <style>
3     * {
4       padding: 0;
5       margin: 0;
6     }
7
8     html,
9     body {
10      height: 100%;
11    }
12
13    .content-wrap {
```

```
14     min-height: 100%;
15     margin-bottom: -60px;
16 }
17
18 .content {
19     height: 50vh;
20     width: 100%;
21     background-color: #ccc;
22     display: flex;
23     align-items: center;
24     justify-content: center;
25 }
26
27 .footer {
28     height: 60px;
29     width: 100%;
30     background-color: rgb(47, 46, 46);
31     display: flex;
32     align-items: center;
33     justify-content: center;
34     color: #fff;
35 }
36 </style>
37 </head>
38
39 <body>
40     <!-- 内容区域 -->
41     <div class="content-wrap">
42         <div class="content">
43             <h1>我是内容区域</h1>
44         </div>
45     </div>
46     <!-- footer -->
47     <div class="footer">
48         <h1>我是 footer</h1>
49     </div>
50 </body>
```

实现效果:

我是内容区域

我是footer

上段代码和我们前面使用负 margin-top 的方式非常的类似，其实实现原理都是一样的，总之就是为了让 footer 出现在页面底部。

2.3 flex 布局

flex 布局基本上可以解决网页布局中 90% 以上的问题，这其中当然包括粘连布局了。我们前面两种方法虽然可以实现粘连布局，但是还是有一个弊端的：我们需要固定底部高度。而使用了 flex 布局，这些就都不是问题了，而且实现起来也很简单。

代码如下：

```
1 <head>
2   <style>
3     * {
4       padding: 0;
5       margin: 0;
6     }
7
8     html,
9     body {
10      height: 100%;
11      display: flex;
12      flex-direction: column;
13    }
14    .content {
15      flex: 1;
16    }
```

```
17     height: 50vh;
18     width: 100%;
19     background-color: #ccc;
20     display: flex;
21     align-items: center;
22     justify-content: center;
23 }
24 .footer {
25     width: 100%;
26     background-color: rgb(47, 46, 46);
27     display: flex;
28     align-items: center;
29     justify-content: center;
30     color: #fff;
31 }
32 </style>
33 </head>
34
35
36 <body>
37     <!-- 内容区域 -->
38     <div class="content">
39         <h1>我是内容区域</h1>
40     </div>
41     <!-- footer -->
42     <div class="footer">
43         <h1>我是 footer</h1>
44     </div>
45 </body>
```

实现效果：

我是内容区域

我是footer

上段代码中我们没有给 footer 设置高度，也没有设置任何的 margin，我们只是利用了 flex:1 这个属性就让内容区域填满了页面。不熟悉 flex 布局的小伙伴还需要自己下去好好学学。

总结

网页的布局有很多种，我们每天都会浏览各种各样的网页，你不妨静下心来，仔细分析一下该网页的布局采用的时哪种方式，我们今天讲的粘连布局只是冰山一下，更为丰富的布局方式还需要小伙伴们自行下来开采。