

2.5 HTML5 文档大纲

HTML5 对如何处理位于 `article`、`aside`、`nav` 和 `section` 等元素中的 `h1`~`h6` 有一套算法。该算法通常称为 HTML5 文档大纲。不过，目前还没有浏览器实现这套算法，在屏幕阅读器中只有 JAWS（一款运行于 Windows 下的屏幕阅读器）支持，而它的实现还存在问题。鉴于此，W3C 已经将文档大纲列入可能从最终定稿的规范中移除的特性。即便文档大纲最终留在规范中，或者浏览器将其实现了，我们还是可以按照本节方法对分级标题进行标记。这种方法不仅适用于当前环境，还足以应对未来，因为文档大纲不会对页面造成破坏。

2.5.1 定义文档节段

在 HTML5 文档中，节段是一个很重要的概念，它表示一个语义独立的内容块。能够定义节段的元素包括：`body`、`section`、`article`、`aside`、`nav`、`header`、`footer`。

节段可以相互嵌套，形成嵌套的结构层次关系。每个节段都必须有自己的标题，即使是嵌套的内层节段，标题使用 `h1`、`h2`、`h3`、`h4`、`h5`、`h6` 元素之一标识。

【示例 1】 下面示例定义了两个顶级节段，第一个节段有三个子节段。

```
<body>
<h1>[网页标题]</h1>
<section>
  <h1>[区块标题]</h1>
  <section>
    <h1>[子区块 1 标题]</h1>
    <p>[正文内容]</p>
  </section>
  <section>
    <h1>[子区块 2 标题]</h1>
    <p>[正文内容]</p>
  </section>
  <aside>
    <h1>[侧栏标题]</h1>
    <p>[侧栏内容]</p>
  </aside>
</section>
<footer>
  <h1>[页脚标题]</h1>
  <p>[版权信息]</p>
</footer>
</body>
```

在 <http://gsnedders.html5.org/outliner/> 页面在线提交本示例文档，梳理文档的层次结构，则上面代码片段形成如图 E2.1 所示的大纲。



图 E2.1 提取示例文档纲要信息

在节段内，第一个 HTML 标题元素定义了当前节段的标题。在同一个节段内，h1 表示最高级别，h6 表示最低级别。标题的级别只在节段内有参考价值，不同节段之间没有可比性。

【示例 2】下面示例演示了不同节段结构中，标题元素的级别并不重要。

```
<body>
<h1>[网页标题]</h1>
<section>
  <h1>[区块标题]</h1>
  <section>
    <h1>[子区块 1 标题]</h1>
    <p>[正文内容]</p>
  </section>
</section>
<footer>
  <h3>[页脚标题]</h3>
  <p>[版权信息]</p>
</footer>
</body>
```

在上面示例中的第一个顶层节段的 h1，子节段中的 h2，以及第二个顶层节段中的 h3，就没有可比性，这时 h1 和 h3 属于同一个层级，重要性平等，如图 E2.2 所示。



图 E2.2 不同级别标题在不同节段中的地位比较

注意，任何级别标题元素都可以用作显示定义的节段的标题，不过不推荐这种无主次的做法。

2.5.2 隐式分节

HTML5 分节元素不会强制性定义大纲，为了与 HTML4 保持兼容，有一种方式来定义节段，而不需要分节元素，这种方式就是隐式分节。

当标题元素（h1~h6）不是父节段的第一个标题时，它会隐式定义一个新的节段。这种隐式节段通过在父节点中与之前标题的相对级别来确定。如果比之前的标题级别更低，那么就会定义一个新的子节段。

【示例 1】下面示例使用标题元素进行分节，演示效果如图 E2.3 所示。

```
<body>
<h1>[网页标题]</h1>
<section>
  <h1>[区块标题 1]</h1>
  <p>[正文内容]</p>
  <h2>[区块标题 2]</h2>
  <p>[正文内容]</p>
  <h3>[区块标题 3]</h3>
  <p>[正文内容]</p>
</section>
</body>
```



图 E2.3 不同级别标题在同一节段中的地位比较

【示例 2】如果与之前标题的级别相同，那么就会开始一个新的同级隐式节段，效果如图 E2.4 所示。

```
<body>
<h1>[网页标题]</h1>
<section>
  <h1>[区块标题 1]</h1>
  <p>[正文内容]</p>
  <h1>[区块标题 2]</h1>
  <p>[正文内容]</p>
  <h3>[区块标题 3]</h3>
  <p>[正文内容]</p>
</section>
</body>
```



图 E2.4 同级别标题在同一节段中的层次比较

【示例 3】如果比之前标题的级别更高，那么就会开始一个更高级别的隐式节段，效果如图 E2.5 所示。

```
<h1>[网页标题]</h1>
<section>
  <h2>[区块标题 1]</h2>
  <p>[正文内容]</p>
  <h3>[区块标题 2]</h3>
  <p>[正文内容]</p>
  <h1>[区块标题 3]</h1>
  <p>[正文内容]</p>
</section>
</body>
```



图 E2.5 通过高级别标题隐式提升节段层次

【注意】

作为经验法则，标题级别应该与节段嵌套级别相匹配，但为了方便节段在多个文档中的重用，也存在例外的情况。例如，一个节段可能会存储在内容管理系统中并在运行时组装为完整的文档。在这种情况下，建议使用 `h1` 作为可重用部分的最高标题级别。

2.5.3 特殊分节

1. 分节根

分节根拥有独立的大纲体系，其内的节段与外部大纲没有联系。使用 `blockquote`、`details`、`fieldset`、`figure`、`td` 可以定义分节根元素。

【示例】下面示例设计一个多层嵌套的大纲。

```
<body>
<h1>[网页标题]</h1>
<section>
  <h1>[区块标题]</h1>
  <section>
    <h2>[子区块 1 标题]</h2>
    <p>[正文内容]</p>
    <figure>
      <h1>[流媒体标题]</h1>
      
    </figure>
  </section>
  <section>
    <h2>[子区块 2 标题]</h2>
    <p>[正文内容]</p>
    <blockquote>
      <h1>[块引用标题]</h1>
      <p>[正文内容]</p>
    </blockquote>
  </section>
</section>
</body>
```

这个大纲并不包含 `blockquote` 元素的内部大纲，也不包含 `figure` 元素的内部大纲，如图 E2.6 所示。`blockquote` 元素是一个外部引用，是一个分节根并隔离了内部的大纲。`figure` 元素是一个流媒体容器，内部内容与外部内容类型不同，不能够相提并论，因此也被隔离为一个独立的纲要系统。



图 E2.6 文档大纲不包含分节根内大纲

2. 主纲之外的节段

下面 4 个元素用来定义不属于文档主要大纲中的节段。

- **aside**: 虽然它是主要的分节元素, 但并不属于主要的文档流, 类似解释栏或广告栏。**aside** 元素内部有自己的大纲, 但并不计入文档主要大纲中。
- **nav**: 是 HTML 导航分节元素, 包含了很多导航链接。文档中可以有多于一个这样的元素, 如文档内部的链接, 类似目录, 以及链接到其他站点的导航链接。这些链接并不是文档主纲中的一部分。
- **header**: 是 HTML 头部分节元素, 定义页眉, 通常会包含 LOGO、站点名称, 以及菜单等。也可以是一个节段的头部, 包含节段的标题等。**article**、**section**、**aside** 和 **nav** 可以拥有自己的 **header**。
- **footer**: 是 HTML 脚部分节元素, 定义页脚, 通常会包含版权信息、法律声明, 以及一些其他链接。也可以是节段的页脚, 可能包含了节段的发布数据、许可声明等。**article**、**section**、**aside** 和 **nav** 可以拥有自己的 **footer**。