

# NSD Devweb DAY01

1. [案例1：HTML 文档片段](#)
2. [案例2：创建 HTML 文档并设置头元素](#)
3. [案例3：使用文本标记为 HTML 页面添加内容](#)
4. [案例4：为HTML页面添加图像和链接](#)
5. [案例5：为 HTML文档添加表格](#)
6. [案例6：为 HTML 页面添加导航目录](#)
7. [案例7：创建表单](#)

## 1 案例1：HTML 文档片段

### 1.1 问题

1. 【参见COOKBOOK】

### 1.2 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：HTML标题

HTML 标题是通过<h1> - <h6> 标签来定义的

01.   <h1>这是标题 1</h1>
02.   <h2>这是标题 2</h2>
03.   <h3>这是标题 3</h3>
04.   <h4>这是标题 4</h4>
05.   <h5>这是标题 5</h5>
06.   <h6>这是标题 6</h6>

实例演示如图-1所示：

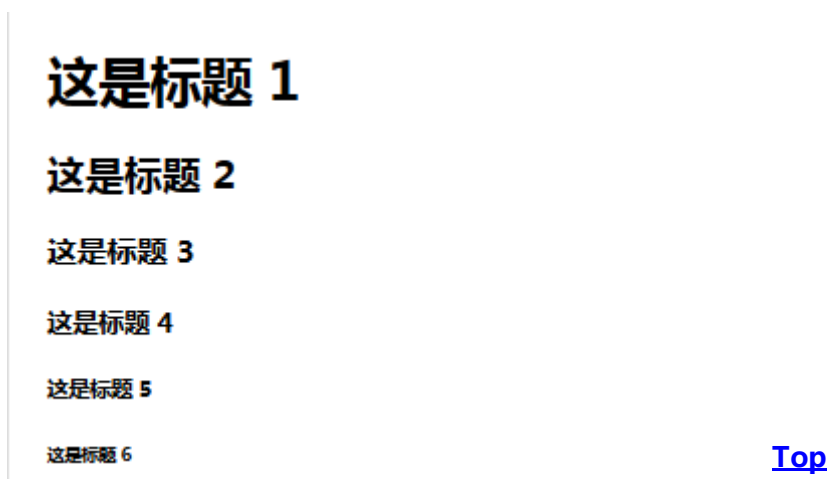


图-1

## 步骤二：HTML段落

HTML 段落是通过标签 `<p>` 来定义的

```
01  <p>这是一个段落。</p>
```

实例演示如图-2所示：

这是一个段落。

图-2

## 步骤三：HTML链接

HTML 链接是通过标签 `<a>` 来定义的.

```
01  <a href="http://www.baidu.com">这是一个链接使用了 href 属性</a>
```

实例演示如图-3所示：

[这是一个链接使用了 href 属性](#)

图-3

## 2 案例2：创建 HTML 文档并设置头元素

### 2.1 问题

1. 创建标准结构的 HTML 文档
2. 为文档添加头元素 `<head>`
3. 为页面设置标题、编码格式

### 2.2 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：HTML实例

```
01  <!DOCTYPE html>
02. <html>
03. <head>
04. <meta charset="utf-8">
05. <title>hello world</title>
06. </head>
```

[Top](#)

```
07.  <body>
08.
09.  <h1>我的第一个标题</h1>
10.
11.  <p>我的第一个段落</p>
12.
13.  </body>
14.  </html>
```

步骤二：实例演示如图-4所示：

# 我的第一个标题

我的第一个段落。

图-4

步骤三：HTML实例解析

<!DOCTYPE html> 声明为 HTML5 文档

<html> 元素是 HTML 页面的根元素

<head> 元素包含了文档的元（meta）数据，如 <meta charset="utf-8"> 定义网页编码格式为 utf-8。

<title> 元素描述了文档的标题

<body> 元素包含了可见的页面内容

<h1> 元素定义一个大标题

<p> 元素定义一个段落s

## 3 案例3：使用文本标记为 HTML 页面添加内容

### 3.1 问题

1. 创建 html 文档
2. 使用文本标记为页面添加内容

### 3.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：HTML实例

[Top](#)

```
01  <!DOCTYPE html>
```

```
02.  <html>
03.
04.  <head>
05.  <meta charset="utf-8">
06.  <title> hello world </title>
07.  </head>
08.
09.  <body>
10.  <b>这个文本是加粗的</b>
11.  <br />
12.  <big>这个文本字体放大</big>
13.  <br />
14.  <i>这个文本是斜体的</i>
15.  <br />
16.  <small>这个文本是缩小的</small>
17.  <br />
18.  这个文本包含
19.  <sub>下标</sub>
20.  <br />
21.  这个文本包含
22.  <sup>上标</sup>
23.  <br />
24.  <u>下划线</u>
25.  <br />
26.  <s>删除线</s>
27.  </body>
28.
29.  </html>
```

## 步骤二：实例演示如图-5所示：

这个文本是加粗的  
这个文本字体放大  
这个文本是斜体的  
这个文本是缩小的  
这个文本包含 下标  
这个文本包含 上标  
下划线  
删除线

[Top](#)

图-5

### 步骤三：HTML实例解析

文本样式的作用是对文本进行修饰

`<b>...</b>`：加粗

`<i>...</i>`：倾斜

`<u>...</u>`：下划线

`<s>...</s>`：删除线

`<sup>...</sup>`：上标

`<sub>...</sub>`：下标

`<big>...</big>`：放大

`<small>...</small>`：缩小

## 4 案例4：为HTML页面添加图像和链接

### 4.1 问题

1. 创建 html 文档
2. 使用图像和链接元素为页面添加内容

### 4.2 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：HTML实例

##### 1)图像链接实例

```
01.  <!DOCTYPE html>
02.  <html>
03.  <head>
04.  <meta charset="utf-8">
05.  <title>hello world</title>
06.  </head>
07.  <body>
08.
09.  <p>创建图片链接:
10.  <a href="http://www.baidu.com">
11.  </a></p>
12.
13.  <p>无边框的图片链接:
14.  <a href="http://www.baidu.com">
15.  </a></p>
16.
17.  </body>
18.  </html>
```

[Top](#)

## 2)锚点实例：

```

01.  <p>
02.  <a href="#C4">查看章节 4</a>
03.  </p>
04.
05.  <h2>章节 1</h2>
06.  <p>这边显示该章节的内容……</p>
07.
08.  <h2>章节 2</h2>
09.  <p>这边显示该章节的内容……</p>
10.
11.  <h2>章节 3</h2>
12.  <p>这边显示该章节的内容……</p>
13.
14.  <h2><a name="C4">章节 4</a></h2>
15.  <p>这边显示该章节的内容……</p>
16.
17.  <h2>章节 5</h2>
18.  <p>这边显示该章节的内容……</p>

```

## 步骤二：实例演示

## 1)图像链接如图-6所示：

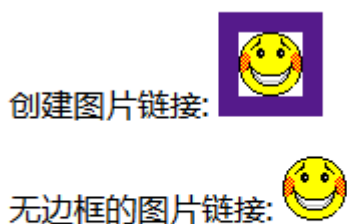


图-6

此时，点击图片超链接会把用户带到百度的首页。

## 2)锚点实例如图-7所示：

[Top](#)

[查看章节 4](#)

## 章节 1

[这边显示该章节的内容.....](#)

## 章节 2

[这边显示该章节的内容.....](#)

## 章节 3

[这边显示该章节的内容.....](#)

## 章节 4

[这边显示该章节的内容.....](#)

## 章节 5

[这边显示该章节的内容.....](#)

题-7

此时点击查看章节4，会把用户带到章节4位置

### 步骤三：HTML实例解析

1)HTML链接语法：

01. `<a href="url">链接文本</a, target="_blank">`

使用<a>元素创建超级链接

href 属性描述了链接的目标。

使用 target 属性，可以定义被链接的文档在何处显示。

2)HTML链接锚点使用方式：

01. 定义锚点:

02. `<a name=" anchorname1">锚点一</a>`

03. 链接到锚点，在锚点名前加上#：

04. `<a href="# anchorname1">...</a>`

锚点是文档中某行的一个记号，用于链接到文档中的某个位置

3)HTML图像语法：

[Top](#)

```
01. 
```

使用<img>元素创建图像

在页面上显示图像，需要使用源属性（src）。源属性的值是图像的 URL 地址，即存储图像的位置。

alt 属性用来为图像定义一串预备的可替换的文本

height（高度）与 width（宽度）属性用于设置图像的高度与宽度。

## 5 案例5：为 HTML文档添加表格

### 5.1 问题

1. 【参见COOKBOOK】

### 5.2 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一：HTML实例

```
01. <table border="1">
02.   <tr>
03.     <td>第一行, 第一列</td>
04.     <td>第一行, 第二列</td>
05.   </tr>
06.   <tr>
07.     <td>第二行, 第一列</td>
08.     <td>第二行, 第二列</td>
09.   </tr>
10. </table>
```

步骤二：实例演示如图-8所示：

第1行, 第1列	第1行, 第2列
第2行, 第1列	第2行, 第2列

图-8

#### 步骤三：HTML实例解析

1)创建表格：

表格由 <table> 标签来定义。

[Top](#)



每个表格均有若干行（由 <tr> 标签定义），

每行被分割为若干单元格（由 <td> 标签定义）。字母 td 指表格数据（table data），即数据单元格的内容。数据单元格可以包含文本、图片、列表、段落、表单、水平线、表格等等

## 2)<table>元素常用属性

width，设置表格宽度

height，设置表格高度

align，设置表格对齐方式(left|center|right)

border，设置表格边框宽度

cellpadding，设置内边距(单元格边框与内容之间的距离)

cellspacing，设置外边距(单元格之间的距离)

bgcolor，设置表格背景颜色

## 3)<tr>元素常用属性

align，设置水平对齐方式(left|center|right)

valign，设置垂直对齐方式(top|middle|bottom)

## 4)<td>元素常用属性

align，设置水平对齐方式(left|center|right)

valign，设置垂直对齐方式(top|middle|bottom)

width，设置宽度

height，设置高度

colspan，设置单元格跨列

rowspan，设置单元格跨行

## 步骤四：不规则表格HTML实例

```
01.  <body>
02.
03.  <h4>单元格跨两格:</h4>
04.  <table border="1">
05.    <tr>
06.      <td>Name</td>
07.      <td colspan="2">Telephone</td>
08.    </tr>
09.    <tr>
10.      <td>Bill Gates</td>
11.      <td>555 77 854</td>
12.      <td>555 77 855</td>
13.    </tr>
14.  </table>
15.
16.  <h4>单元格跨两列:</h4>
```

[Top](#)

```

17.  <table border="1">
18.  <tr>
19.    <td>First Name: </td>
20.    <td>Bill Gates</td>
21.  </tr>
22.  <tr>
23.    <td rowspan="2">Telephone: </td>
24.    <td>555 77 854</td>
25.  </tr>
26.  <tr>
27.    <td>555 77 855</td>
28.  </tr>
29. </table>
30.
31. </body>

```

**步骤五：实例演示如图-9所示：**

**单元格跨两格:**

Name	Telephone	
Bill Gates	555 77 854	555 77 855

**单元格跨两列:**

First Name:	Bill Gates
Telephone:	555 77 854
	555 77 855

图-9

### 步骤六：HTML实例解析

设置单元格 <td> 的跨行或者跨列属性

1)跨列：colspan，水平方向延伸单元格，值为一正整数，代表此单元格水平延伸的单元格数

2)跨行：rowspan，垂直方向延伸单元格，值为一正整数，代表此单元格垂直延伸的单元格数。

## 6 案例6：为 HTML 页面添加导航目录

[Top](#)

### 6.1 问题

## 1. 使用列表元素为 html 文档添加导航目录

## 6.2 步骤

实现此案例需要按照如下步骤进行。

### 步骤一：HTML实例

#### 1)有序列表

```
01. <h4>编号列表：</h4>
02. <ol>
03. <li>Apples</li>
04. <li>Bananas</li>
05. <li>Lemons</li>
06. <li>Oranges</li>
07. </ol>
08.
09. <h4>大写字母列表：</h4>
10. <ol type="A">
11. <li>Apples</li>
12. <li>Bananas</li>
13. <li>Lemons</li>
14. <li>Oranges</li>
15. </ol>
```

#### 2)无序列表

```
01. <h4>圆圈列表：</h4>
02. <ul style="list-style-type: circle">
03. <li>Apples</li>
04. <li>Bananas</li>
05. <li>Lemons</li>
06. <li>Oranges</li>
07. </ul>
08.
09. <h4>正方形列表：</h4>
10. <ul style="list-style-type: square">
11. <li>Apples</li>
12. <li>Bananas</li>
13. <li>Lemons</li>
14. <li>Oranges</li>
```

[Top](#)

15. `</ul>`

### 3)嵌套列表

```
01. <ul>
02.   <li>Coffee</li>
03.   <li>Tea
04.     <ul>
05.       <li>Black tea</li>
06.       <li>Green tea</li>
07.     </ul>
08.   </li>
09.   <li>Milk</li>
10. </ul>
```

### 步骤二：实例演示：

1)有序列表如图-10所示：

#### 编号列表：

1. Apples
2. Bananas
3. Lemons
4. Oranges

#### 大写字母列表：

- A. Apples
- B. Bananas
- C. Lemons
- D. Oranges

图-10

2)无序列表如图-11所示：

[Top](#)

### 圆圈列表：

- Apples
- Bananas
- Lemons
- Oranges

### 正方形列表：

- Apples
- Bananas
- Lemons
- Oranges

图-11

3)嵌套列表如图-12所示：

- Coffee
- Tea
  - Black tea
  - Green tea
- Milk

图-12

## 步骤三：HTML实例解析

1)有序列表：

<ol>元素编写有序列表，<ol> 元素中只能包含具体的列表项元素 <li>

有序列表常用属性type：

作用：指定列表项前面标识的类型

取值：

1：按数字方式排列，默认值

A：按照大写英文字母方式排列

a：按照小写英文字母排列

I：按大写罗马字符方式排列

i：按小写的罗马字符排列

2)无序列表：

<ul>元素表示无序列表，<ul> 元素中只能包含具体的列表项元素 <li>

无序列表常用属性type：

作用：指定列表项前面标识的类型

取值：

disc 默认值，实心原点

circle：空心原点

[Top](#)

square : 实心方块

none : 不显示任何标识

3)嵌套列表 :

列表中允许再出现列表

被嵌套的列表只能出现在<li></li>中

无序列表中嵌套有序列表,有序列表中也嵌套无序列表

## 7 案例7 : 创建表单

### 7.1 问题

1. 【参见COOKBOOK】

### 7.2 步骤

实现此案例需要按照如下步骤进行。

#### 步骤一 : form元素解析

1)表单作用 : 用于接收用户的数据并提交给服务器

2)表单中的两个要素 : form 元素、表单控件 ( 用于与用户进行交互的元素 )

3)表单元素标记 : 使用<form></form>元素创建表单

4 ) 表单常用属性 :

action : 指定提交给服务器的处理程序的地址

method : 指出表单数据提交的方式,取值为get或post

enctype : 指定表单数据进行编码的方式,即允许将什么样的数据提交给服务器

name : 表单名称

01. 表单使用表单标签 <form> 来设置:
02. <form>
03. .
04. input 元素
05. .
06. </form>

#### 步骤二 : HTML表单控件-文本域

1)文本域通过<input type="text"> 标签来设定,当用户要在表单中键入字母、数字等内容时,就会用到文本域。

01. <form>
02. First name: <input type="text" name="firstname"><br>
03. Last name: <input type="text" name="lastname">

[Top](#)

04. `</form>`

2)浏览器显示如图-13所示：



图-13

3)常用属性如下：

border-collap

name：定义控件名称

value：值

maxlength 最大输入字符

readonly 只读

placeholder 提示占位符

2)密码字段

### 步骤三：HTML表单控件-密码字段

1)密码字段通过标签`<input type="password">` 来定义:

01. `<form>`

02. Password: `<input type="password" name="pwd">`

03. `</form>`

2)浏览器显示如图-14所示：



图-14

3)常用属性如下：

border-collap

name：定义控件名称

value：值

maxlength 最大输入字符

readonly 只读

placeholder 提示占位符

### 步骤四：HTML表单控件-单选框、复选框

[Top](#)

1)`<input type="radio">` 定义了单选框. 用户需要从若干给定的选择中选取一个选项。

```

01.  <form>
02.  <input type="radio" name="vehicle" value="Bike">I have a bike<br>
03.  <input type="radio" name="vehicle" value="Car">I have a car
04.  </form>

```

2)浏览器显示如图-15所示：

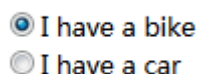


图-15

3)<input type="checkbox"> 定义了复选框. 用户需要从若干给定的选择中选取一个或若干选项。

```

01.  <form>
02.  <input type="checkbox" name="vehicle" value="Bike">I have a bike<br>
03.  <input type="checkbox" name="vehicle" value="Car">I have a car
04.  </form>

```

4)浏览器显示如图-16所示：

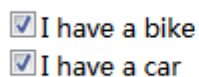


图-16

5)常用属性如下：

name:除定义名称之外，还有分组作用，一组的单元按钮和复选框，name属性必须一致，提交时会把复选框中的内容按照数组进行提交

value值，提交前定义，当前用户选择的时候，会将该控件value值提交给服务器

checked设置预选中，该属性无值

### 步骤五：HTML表单控件-按钮

1)<input type="submit"> 定义了提交按钮

当用户单击确认按钮时，表单的内容会被传送到另一个文件。表单的动作属性定义了目的文件的文件名。由动作属性定义的这个文件通常会对接收到的输入数据进行相关的处理。

```

01.  <form name="input" action="html_form_action.php" method="get">
02.  Username: <input type="text" name="user">
03.  <input type="submit" value="Submit">
04.  </form>

```

[Top](#)



2)浏览器显示如图-17所示：



图-17

假如您在上面的文本框内键入几个字母，然后点击确认按钮，那么输入数据会传送到 "html\_form\_action.php" 的页面。该页面将显示出输入的结果。

3)<input type="reset">定义了重置按钮

当用户单击reset重置按钮时，表单的内容会被清空

```
01. <form name="input" action="html_form_action.php" method="get">
02. Username: <input type="text" name="user">
03. <input type="submit" value="Submit">
04. <input type="reset" value="reset">
05. </form>
```

4)浏览器显示如图-18所示：

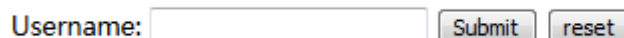


图-18

5)<input type="button">定义了普通按钮：

```
01. <form action="">
02. <input type="button" value="Hello world! ">
03. </form>
```

6)浏览器显示如图-19所示：

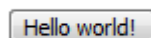


图-19

属性：value：按钮上的文本

## 步骤六：HTML表单控件-隐藏域和文件选择框

1)隐藏域

作用：想提交给服务器但不想给用户看的数据要放在隐藏域中

[Top](#)

语法：<input type=" hidden" >

属性：

name：控件名称

value：控件的值

6)文件选择框：

作用：提交文件

语法：<input type=" file" >

属性：

name:控件名称

[Top](#)