

# HTML编码规范

## 1. 写在前面

本次的软件工程实践，前端由3个人共同开发，为了保证 html 的可读性，提高协作效率。故做以下编码约定。

本规范还未经过实践的检验，后期还会对本规范的条目进行增删，以适应实际开发

## 2. 代码风格

### 2.1 .缩进与换行

- 使用 `4` 个空格做为一个缩进层级，不允许使用 `2` 个空格 或 `tab` 字符。

示例：

```
<ul>
  <li>first</li>
  <li>second</li>
</ul>
```

- 每行不得超过 `120` 个字符。

### 2.2 .命名

- `class` 必须单词全字母小写，单词间以 `-` 分隔。
- `class` 必须代表相应模块或部件的内容或功能，不得以样式信息进行命名。

示例：

```
<!-- good -->
<div class="sidebar"></div>

<!-- bad -->
<div class="left"></div>
```

- 元素 `id` 必须保证页面唯一。
- `id` 建议单词全字母小写，单词间以 `-` 分隔。同项目必须保持风格一致。
- `id`、`class` 命名，在避免冲突并描述清楚的前提下尽可能短。

示例：

```
<!-- good -->
<p class="comment"></p>
<!-- bad -->
<p class="com"></p>
```

- 同一页面，应避免使用相同的 `name` 与 `id`

示例：

```
<input name="foo">
<div id="foo"></div>
<script>
// IE6 将显示 INPUT
alert(document.getElementById('foo').tagName);
</script>
....
```

## 2.3 .标签

- 标签名必须使用小写字母。

示例：

```
<!-- good -->
<p>Hello StyleGuide!</p>

<!-- bad -->
<P>Hello StyleGuide!</P>
```

- 标签的使用必须符合标签嵌套规则。
- `HTML` 标签的使用应该遵循标签的语义。
- 使用应尽量简洁，减少不必要的标签。

示例：

```
<!-- good -->


<!-- bad -->
<span class="avatar">
  
</span>
```

## 2.4 属性

- 属性名必须使用小写字母。

示例：

```
<!-- good -->
<table cellpadding="0">...</table>

<!-- bad -->
<table cellspacing="0">...</table>
```

- 属性值必须用双引号包围。
- 自定义属性建议以 `xxx-` 为前缀，推荐使用 `data-`。

示例：

```
<ol data-ui-type="Select"></ol>
```

## 3. 通用

---

### 3.1. 编码

- 页面必须使用精简形式，明确指定字符编码。指定字符编码的 `meta` 必须是 `head` 的第一个直接子元素。
- `HTML` 文件使用无 `BOM` 的 `UTF-8` 编码。

### 3.2. CSS和JavaScript引入

- 引入 `CSS` 时必须指明 `rel="stylesheet"`。

示例：

```
<link rel="stylesheet" src="page.css">
```

- 引入 `CSS` 和 `JavaScript` 时无须指明 `type` 属性。
- 在 `head` 中引入页面需要的所有 `CSS` 资源。
- `JavaScript` 应当放在页面末尾，或采用异步加载。

## 4. head

---

### 4.1. title

- 页面必须包含 `title` 标签声明标题。
- `title` 必须作为 `head` 的直接子元素，并紧随 `charset` 声明之后。

## 5. 表单

---

### 5.1. 控件标题

- 有文本标题的控件必须使用 `label` 标签将其与其标题相关联。

解释：

有两种方式：

1. 将控件置于 `label` 内。
2. `label` 的 `for` 属性指向控件的 `id`。

推荐使用第一种，减少不必要的 `id`。如果 `DOM` 结构不允许直接嵌套，则应使用第二种。

示例：

```
<label><input type="checkbox" name="confirm" value="on"> 我已确认上述条款</label>

<label for="username">用户名: </label> <input type="text" name="username" id="username">
```

## 5.2. 按钮

- 使用 `button` 元素时必须指明 `type` 属性值。

示例:

```
<button type="submit">提交</button>
<button type="button">取消</button>
```

- 尽量不要使用按钮类元素的 `name` 属性。

## 6. 模板中的 HTML

由于后端采用 thinkphp5，mcv 模式，需要使用到模板语言，故有以下规范

- 模板代码的缩进优先保证 `HTML` 代码的缩进规则。

示例:

```
<!-- good --><!-- good -->
{if $display == true}
<div>
    <ul>
        {foreach $item_list as $item}
            <li>{$item.name}</li>
        {/foreach}
    </ul>
</div>
{/if}
```

- 模板代码应以保证 `HTML` 单个标签语法的正确性为基本原则。

示例:

```
<!-- good -->
<li class="{if $item.type_id == $current_type}focus{/if}">{ $item.type_name }</li>

<!-- bad -->
<li {if $item.type_id == $current_type} class="focus" />{ $item.type_name }</li>
```