

**캄보디아 고용서비스 전산망 구축 지원 사업**

**개발부문**

**퍼블리싱 스타일 가이드**

**개정이력**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **개정번호** | **개정일자** | **개정내용** | **작성자** | **검토자** | **승인자** |
| 0.2 | 2019.09.02 | 최초 작성 | 이영호 |  |  |
| 0.7 | 2019.09.13 | 작성 완성 | 이영호 |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**1 HTML 코드 작성 규칙**

* 1. **기본 규칙**

**1.1.1 W3C Validation**

- HTML은 해당 DOCTYPE(문서형 정의) 명세에 맞게 작성하며, W3C validation을 통과해야 한다.

**1.1.2 영문 소문자 사용**

- HTML5 에서는 대, 소문자를 구분하지 않는다.

- XHTML에서는 DOCTYPE(문서형 정의)을 제외한 모든 요소와 속성은 소문자로 작성한다.

- 본 가이드에서는 **소문자 작성을 원칙**으로 하는 것으로 정의 한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <SPAN Class="desc">contents</SPAN> | <span class="desc">contents</span> |

**1.1.3 속성 표기**

- 속성 값은 큰따옴표(" ")로 묶는다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <img src='test.gif' style='width:100px;height:100px;' alt='text'> | <img src="test.gif" style="width:100px;height:100px;" alt="text" /> |

**1.1.4 entity 코드 사용**

- 특수 기호는 entity name을 사용하여 entity 코드로 변환한다. entity number는 사용하지 않는다.

- entity 코드는 ISO-8859-1을 기준으로 하며, 아래 경로에서 확인할 수 있다.

HTML ISO-8859-1 Reference: <http://www.w3schools.com/charsets/ref_html_entities_4.asp>

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <a href="...&nid=2"> | <a href="... &amp;nid=2"> |

**참고**

<, >, ", ', & 등의 특수기호를 entity 코드로 변환하지 않으면, 브라우저가 이를 시작/종료 요소나

속성으로 잘못 해석할 수 있다. 자동으로 생성되는 링크의 경로나 이미지의 alt값에도 entity 코드가 바르게 적용되도록 한다.

**1.1.5 종료 태그 사용**

- 종료 태그를 반드시 선언한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <a href=””>Go to page | <a href=””>Go to page</a> |

**1.1.6 Void 요소의 종료 태그**

<area>, <base>, <br>, <col>, <embed>, <hr>, <img>, <input>, <keygen>, <link>, <meta>,

<param>, <source>, <track>, <wbr>

- HTML5 DTD에서는 다음의 요소에 대해서는 / 문자가 올 수 있다.

- XHTML DTD에서는 다음의 요소에 대해서 시작 태그가 />로 끝나야 한다. .

- 본 가이드는 다음의 요소에 대해서는 **시작 태그에 />로 포함 하는 것으로 정의 한다.**

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <img src="test.gif" alt="text"></img>  <img src="test.gif" alt="text"> | <img src="test.gif” alt="text" **/**> |

**2.2 들여쓰기 규칙**

- 들여쓰기를 하면 코드의 가독성이 높아지고 전체 HTML 구조를 쉽게 파악할 수 있다. 다음과 같은 들여쓰기 규칙을 준수한다.

- 마크업의 중첩이 깊어질 때마다 자식 요소는 1탭 들여 쓰고, 1탭의 크기는 공백 4칸으로 설정한다.

- 단, 다음의 경우에는 들여 쓰지 않는다.

\* <html>의 자식 요소인 <head>, <body>

\* <head>의 자식 요소

\* <body>의 자식 요소

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <table>  <caption>...</caption>  <colgroup>  <col>…</col>  </colgroup>  <thead>  <tr>  <th>...</th>  <th>...</th>  </tr>  </thead>  <tbody>  <tr>  <td>...</td>  <td>...</td>  </tr>  </tbody>  </table> | <table>  <caption>...</caption>  <colgroup>  <col />  </colgroup>  <thead>  <tr>  <th>...</th>  <th>...</th>  </tr>  </thead>  <tbody>  <tr>  <td>...</td>  <td>...</td>  </tr>  </tbody>  </table> |

**1.3 빈 줄**

- 빈 줄을 사용하려면 다음과 같은 사용 규칙을 준수한다.

- 의미 있는 객체를 구분하기 위하여 코드 그룹 간 1줄씩 빈 줄을 만드는 것은 허용한다.

- 빈 줄의 간격은 1줄을 초과하지 않는다.

- 빈 줄을 사용하는 것은 선택 사항이다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <head>  contents...  </head>  <!-- Blank line -->  <!-- Blank line -->  <body>  contents...  </body> | <head>  contents...  </head>  <!-- Blank line -->  <body>  contents...  </body> |

**1.4 DTD 선언**

HTML 문서는 반드시 DTD를 선언한다.

**1.4.1 기본 DTD**

- 신규 HTML 문서를 작성할 때 DTD를 사용 한다.

- 본 가이드에서는 **HTML5 DTD를 원칙**으로 하는 것으로 정의 한다.

- HTML5 문서

|  |
| --- |
| <!DOCTYPE html> |

- XHTML 문서

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> |

**1.4.2 기타 DTD**

- HTML 4.01

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  "http://www.w3.org/TR/html4/strict.dtd"> |

- Quirks Mode

|  |
| --- |
| <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  "http://www.w3.org/TR/html4/loose.dtd"> |

**1.5 인코딩 선언**

- HTML 문서는 반드시 인코딩 정보를 선언한다. 인코딩 설정은 DB의 인코딩 방식과도 관련이 있으므로 반드시 프로젝트의 담당 개발자와 협의하여 정해야 한다.

- 본 가이드에서는 **utf-8 인코딩을 원칙**으로 하는 것으로 정의 한다.

**1.5.1 기본 인코딩**

다음은 인코딩 방식으로 utf-8을 선언한 예이다.

- HTML5 문서

|  |
| --- |
| <meta charset="utf-8" /> |

- XHTML 문서

|  |
| --- |
| <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> |

**1.5.2 기타 인코딩**

utf-8 인코딩을 사용할 수 없으면 euc-kr을 사용한다. 다음은 인코딩 방식으로 euc-kr을 선언한

예이다.

- HTML5 문서

|  |
| --- |
| <meta charset=" euc-kr" /> |

- XHTML 문서

|  |
| --- |
| <meta http-equiv="Content-Type" content="text/html; charset=euc-kr" /> |

HTML, CSS 파일을 저장할 때 반드시 설정한 인코딩을 선택하여 저장한다.

**1.6 주석**

**1.6.1 기본 형식**

- HTML 주석의 시작과 종료는 아래와 같이 표기하며, 기본 형식에 맞게 작성한다.

|  |
| --- |
| Start an annotation <!-- content of annotation -->  End an annotation <!-- //content of annotation --> |

- 주석 기호와 주석 내용 사이에는 반드시 공백 한 칸이 있어야 한다.

- 주석 기호와 주석 내용 사이에는 다른 기호가 올 수 없다.

- 주석 기호와 주석 내용 사이의 줄 바꿈은 허용하지 않는다. 단, 주석 내용 간 줄 바꿈은 허용한다.

- 시작과 종료 주석의 주석 내용은 동일해야 한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| <!--gnb--> | <!-- gnb -->  …  <!-- // gnb --> |
| <!--\* gnb \*--> |
| <!-- gnb  --> |
| <!-- gnb start-->  …  <!-- // gnb end --> |

**1.6.2 레이아웃 및 콘텐츠 영역의 주석 표기**

- wrap을 제외한 레이아웃과 독립된 콘텐츠 영역의 시작과 끝에 주석을 표기하며, 레이아웃은 id 또는 class 이름과 동일하게 주석을 넣는다. 독립된 콘텐츠 영역의 주석 표기는 선택 사항이다.

- 15줄 이내 영역은 주석을 생략 할 수 있다.

- HTML 코드와 주석은 줄로 분리해야 한다.

|  |
| --- |
| <div id="content">  <div class="namecard"> ... </div>  </div>  <!-- //content --> |

**1.6.3 수정된 부분의 주석 표기**

- 수정된 부분의 시작과 끝에 주석을 넣는다. 주석 내용 앞에는 DDMMYY의 형식으로 수정된 날짜 정보를 입력한다. 필요에 따라서 수정된 내용을 한 줄 이내로 입력할 수 있다.

- 수정된 부분의 주석 표기는 선택 사항이다.

|  |
| --- |
| <!-- 16092019 -->  ...  <!-- //16092019 -->  <!-- 16092019 Corrected content -->  ...  <!-- //16092019 Corrected content --> |

**1.6.4 삭제 및 숨김 처리된 부분의 주석 표기**

- 삭제되었거나 숨김 처리된 부분은 해당 영역의 시작과 끝에 주석을 표기하며, 주석 표기와 내용은 줄을 구분한다.

- 주석 작성 규칙은 1.6.3 수정된 부분의 주석 표기와 동일하다.

- 삭제된 부분의 주석 표기는 선택 사항이며 삭제 및 숨김 주석에 포함된 주석은 삭제한다.

|  |
| --- |
| <!-- 091120 Delete  ...  //091120 Delete -->  <!-- 091120 Hiding  ...  //091120 Hiding --> |

**1.6.5 설명이 필요한 부분의 주석 표기**

- 작업자 간에 설명이 필요할 경우 주석을 표기 할 수 있다.

- 설명이 필요한 부분의 주석 표기는 선택 사항이다.

|  |
| --- |
| <ul><!-- At least two lists -->  <li>…</li>  <li>…</li>  </ul> |

**2 HTML 요소 작성 규칙**

**2.1 기본 규칙**

- HTML 요소에 class, style을 선언할 때는 선언 순서를 준수한다. 다음과 같이 class와 style은 제일 뒷부분에 선언한다.

|  |
| --- |
| <input type="text" id="user\_id" title="User ID" class="input\_txt" style="width:100px;" /> |

**2.2 전역 구조화 요소**

**2.2.1 <html>**

- 다음과 같이 lang 속성을 선언한다.

- class 속성은 선언하지 않는다.

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 문서

|  |
| --- |
| <html lang="en"> |

- XHTML 문서

|  |
| --- |
| <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en"> |

- lang 속성은 User Agent가 언어를 올바로 해석할 수 있게 도와주며, 검색과 음성 장치(speech

synthesizers)에 활용된다.

**2.2.2 <head>**

- meta, link, script, title, style 순서로 meta 요소를 선언한다.

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 문서

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>  <meta charset="utf-8" />  <meta http-equiv="X-UA-Compatible" content="IE=Edge" />  <meta name="viewport" content="width=device-width, initial-scale=1.0, maximumscale=1.5, minimum-scale=1.0, user-scalable=yes" />  <meta name="title" content="title of site" />  <meta name="keywords" content="Keyword" />  <meta name="description" content="details of site" />  <link rel="stylesheet" href="css/main.css" />  <script src="http://code.jquery.com/jquery-latest.js"></script>  <script src="js/common.js"></script>  <script src="js/main.js"></script>  <title>Contents title</title>  </head>  <body>  </body>  </html> |

- XHTML 문서

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html lang="en">  <head>  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />  <meta http-equiv="X-UA-Compatible" content="IE=edge" />  <meta name="title" content="title of site" />  <meta name="keywords" content="keyword" />  <meta name="description" content="details of site” />  <link rel="stylesheet" type="text/css" href="css/main.css" />  <script type="text/javascript" src="http://code.jquery.com/jquery-latest.js"></script>  <script type="text/javascript" src="js/common.js"></script>  <script type="text/javascript" src="js/main.js"></script>  <title>Contents title</title>  </head>  <body>  </body>  </html> |

**2.2.3. <meta>**

charset, http-equiv, name 순서로 요소를 선언한다.

|  |
| --- |
| <meta charset="utf-8" />  <meta http-equiv="X-UA-Compatible" content="IE=edge" />  <meta name="title" content="Site title" /> |

**2.2.4 <link>**

- rel, href 순서로 속성을 선언한다.

|  |
| --- |
| <link rel="stylesheet" href=“css/default.css" /> |

- rel, type, href 순서로 속성을 선언한다.

|  |
| --- |
| <link rel="stylesheet" type="text/css" href=“css/default.css" /> |

**2.2.5 <script>**

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 : src 속성을 선언한다.

|  |
| --- |
| <script src=“js/default.js"></script> |

- XHTML : type, src 순서로 속성을 선언한다

|  |
| --- |
| <script type="text/javascript" src=“js/default.js"></script> |

**참고**

language 속성은 HTML4 이전 버전의 하위 호환성을 위해 사용하는 비 표준 속성으로 사용하지

않는다.

**2.2.6 <title>**

"페이지 제목 - 사이트 이름"의 형식으로 작성한다.

|  |
| --- |
| <title>Content title – Site name</title> |

**2.2.7 <style>**

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 : type 속성을 생략한다.

|  |
| --- |
| <style>  </style> |

- XHTML : type 속성을 선언한다.

|  |
| --- |
| <style type="text/css">  </style> |

**2.2.8 <body>**

- class 속성을 웹페이지의 스킨셋을 변경해야 할 때 선택적으로 사용한다.

|  |
| --- |
| <body class=”sub01”> |

**2.3 form 요소**

폼 컨트롤 요소를 마크 업 할 때 form, fieldset, legend 요소를 다음과 같이 선언한다.

|  |
| --- |
| <form action="">  <fieldset>  <legend>Form title</legend>  </fieldset>  </form> |

**2.3.1 <form>**

action 속성은 서버 사이드 폼 핸들러 이나 필수 선언 속성이기 때문에, 마크업 단계에서 다음과

같이 선언한다.

|  |
| --- |
| <form action="">  </form> |

CSS로 디자인을 컨트롤하지 않는다.

**2.3.2 <fieldset>**

form 요소의 자식 노드로 선언하여 폼 컨트롤 요소들을 그룹화 위해 선언한다.

|  |
| --- |
| <form action="">  <fieldset>  <legend>Form title</legend>  </fieldset>  </form> |

**2.3.3 <legend>**

폼 컨트롤 그룹인 fieldset의 자식 요소로서 폼 컨트롤 요소들의 그룹 이름을 표현하기 위해

선언한다.

|  |
| --- |
| <form action="">  <fieldset>  <legend>Form title</legend>  [form control element here]  </fieldset>  </form> |

**2.3.4 <input>**

다음과 같이 label 요소 또는 title 속성을 선언한다 (type="image"이면 alt 속성 사용).

|  |
| --- |
| <label for="user\_id">User ID</label>  <input type="text" id="user\_id" name="user\_id" />  <input type="image" src=“btn\_confirm.gif" alt="Confirm" />  <label for="num1">Call number</label>  <input type="text" id="num1" name="num1" title="Area number" /> |

label 요소, title 속성, alt 속성을 통해 스크린 리더 사용자는 주변 맥락에 대한 이해 없이 각

요소에 독립적으로 접근해도 폼을 이해할 수 있다.

다음과 같이 type값에 따라 속성을 선언한다.

- type이 text인 경우: type, id, class, title, placeholder 순서로 속성을 선언한다

|  |
| --- |
| .<input type="text" id="user\_id" title="User ID" placeholder="Please enter your ID." /> |

동일한 스타일의 텍스트 필드이나 width 값이 다를 경우 style 속성을 이용하여 width 값을 제어한다.

텍스트필드에 가이드를 위한 내용이 들어갈 경우 value 속성을 선택적으로 사용할 수 있다.

- type이 radio, checkbox인 경우: type, name, id, checked 순서로 속성을 선언한다.

|  |
| --- |
| <input type="radio" name="vt\_align" id="align\_lft" checked="checked" />  <label for="align\_lft">Left align</label>  <input type="radio" name="vt\_align" id="align\_cnt" />  <label for="align\_cnt">Center align</label>  <input type="radio" name="vt\_align" id="align\_rgt" />  <label for="align\_rgt">Right align</label>  <input type="checkbox" name="sports" id="soccer" checked="checked" />  <label for="soccer">Soccer</label>  <input type="checkbox" name="sports" id="basketball" />  <label for="basketball">Basketball</label>  <input type="checkbox" name="sports" id="tennis" />  <label for="tennis">Tennis</label> |

그룹화가 필요하면 선택적으로 name 속성을 이용하여 항목들을 그룹화 한다.

기본 선택에 대한 표현이 필요할 경우 checked="checked"라고 표기한다.

title 속성은 선언하지 않는다.

- type이 image인 경우: type, src, alt 순서로 속성을 선언한다. 폼 전송 역할을 하는 디자인

버튼은 image 타입을 사용한다.

- 디자인이 적용되지 않은 버튼은 <button type="submit">로 사용 한다.

|  |
| --- |
| <input type="image" src="img/btn\_confirm.gif" alt="send" /> |

**Input 요소 HTML 5 추가 속성**

**(a) 공통속성**

- autofocus : 페이지 로딩 완료 시 자동으로 해당 컨트롤에 포커싱이 됨

**(b) input 요소**

- type 속성

\* search : 검색 필드 기능을 한다.

\* tel : 전화번호 입력을 위한 타입이다.

\* url : 절대경로 URL 입력을 위한 타입이다.

\* email : E-Mail 주소를 입력 하기 위한 타입이다.

\* date : 년/월/일 입력을 위한 타입으로, 캘린더가 제공된다.

\* month : 년/월 입력을 위한 타입으로, 캘린더가 제공된다.

\* time : 시간 입력을 위한 타입으로 1/1000초 단위까지 입력이 가능하다.

\* datetime : UTC 일자, 시간 입력을 위한 타입이다.

\* datetime-local : 지역화 된 일자/시간입력을 위한 타입이다.

\* week : 년도와 주차를 입력하기 위한 타입이다.

\* number : 한 줄의 숫자 데이터를 입력 하기 위한 타입이다.

\* range : 슬라이더 형태로 숫자 범위 데이터를 입력할 수 있다.

\* color : 컬러 피커에서 컬러를 선택하여 데이터를 입력할 수 있다.

- autocomplete : 입력 필드 값의 자동 완성 여부를 지정. boolean 타입이다.

- list : 서제스트를 제공할 요소의 id를 지정한다.

- required : boolean 속성으로, 필수 입력 여부를 지정한다.

- multiple : 다수 값 입력 여부를 지정 콤마(,)로 구분한다.

- pattern : 컨트롤 값을 체크하는 정규 표현식 패턴을 지정한다.

- min : 컨트롤의 최소값을 지정한다. (숫자만 가능)

- max : 컨트롤의 최대값을 지정한다. (숫자만 가능)

- step : 컨트롤의 증가 단위 값을 지정한다.

- placeholder : 입력값에 대한 힌트를 지정한다.

상위 추가 속성은 하위 브라우저 미 지원 및 브라우저별, 버전별 지원 및 표현이 상이 하므로

사용시 주의 하며, 점진적 향상으로 사용 한다.

**2.3.5 <select>**

동일한 스타일의 셀렉트 박스이나 width 값이 다르면 선택적으로 style 속성을 이용하여 width값을 제어한다.

|  |
| --- |
| <select style="width:100px;">  <option>option1</option>  </select> |

**2.3.6 <label>**

라디오 버튼, 체크 박스는 for 속성을 부여하여 해당 요소의 id값과 동일한 이름으로 연결(coupling)한다.

|  |
| --- |
| <input type="radio" name="alignment" id="align\_lft" />  <label for="align\_lft">Left align</label>  <input type="checkbox" name="sports" id="soccer" />  <label for="soccer">Soccer</label> |

**2.3.7 <textarea>**

cols, rows 순서로 속성을 선언한다.

CSS를 정상적으로 불러오지 못하는 상황에서도 사용에 문제가 없도록 col, rows의 속성값은 각각

최소 30, 5 이상이 되도록 선언한다.

|  |
| --- |
| <textarea cols="30" rows="5"></textarea> |

**2.3.8 <button>**

type 속성을 선언한다.

- type 속성을 button으로 선언하여 열기/닫기, 접기/펼치기 등의 UI를 제어한다.

- 폼 전송 역할을 하는 디자인 적용 버튼은 submit 타입을 사용한다

|  |
| --- |
| <button type="button">Open</button>  <button type="submit">Send</button> |

**2.4 표 요소**

- 표 요소를 아래와 같이 배치한다.

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 에서는 summary 속성을 사용하지 않는다.

- XHTML 에서는 summary 속성을 반드시 사용한다.

|  |
| --- |
| <table>  <caption>Compare prices and calories for apples and oranges</caption>  <colgroup>  <col style=”width:100px;” />  <col />  <col />  </colgroup>  <thead>  <tr>  <th>division</th>  <th scope="col">price($)</th>  <th scope="col">calories(kcal)</th>  </tr>  </thead>  <tbody>  <tr>  <th scope="row">apple</th>  <td>2</td>  <td>300</td>  </tr>  <tr>  <th scope="row">oranges</th>  <td>3</td>  <td>350</td>  </tr>  </tbody>  <tfoot>  <tr>  <th scope="row">total</th>  <td>5</td>  <td>650</td>  </tr>  </tfoot>  </table> |

**2.4.1 <table>**

일반적으로 레이아웃을 표현하기 위해 사용하지 않고, 2차원의 격자형 데이터를 표현하기 위해

사용한다.

cellspacing="0", cellpadding="0"은 선언하지 않고, CSS에서 셀 여백 및 테두리 선을 초기화한다.

- XHTML DTD 에서는 summary 속성을 선언한다.

|  |
| --- |
| <table summary=”descriptions of table”> |

- HTML5 Summary 속성을 제외한다.

<caption> 요소를 사용하거나, 테이블 앞/뒤에서 요약 설명 글을 포함한다.

\* Html5 Summary 예제 1 (위쪽, 왼쪽에 제목 셀 제공 예시 )

|  |
| --- |
| <table>  <caption>Title of table – descriptions of table</caption>  [ table contents .... .]  </table> |

\* Html5 Summary 예제 2 (설명이 필요한 복잡한 구조 예시. Ex) id, header 사용 등)

|  |
| --- |
| <table>  <caption>  <strong>Title of table</strong>  <p>descriptions of table</p>  </caption>  <thead> |

\* Html5 Summary 예제 3 (설명이 필요한 복잡한 구조 예시. Ex) id, header 사용 등)

|  |
| --- |
| <figure>  <figcaption>  <strong>Title of table</strong>  <p>descriptions of table</p>  </figcaption>  <table>  [ table contents....... ]  </table>  </figure> |

**2.4.2 <caption>**

표의 제목을 표현하기 위해 사용한다.

|  |
| --- |
| <caption>Compare prices and calories for apples and oranges</caption> |

**2.4.3 <colgroup>**

col 요소를 그루핑 하여 디자인을 제어할 때 선언한다. 이 요소는 선택적으로 사용한다.

|  |
| --- |
| <colgroup>  <col style=”width:100px;" />  <col />  <col />  </colgroup> |

**2.4.4 <col>**

표 각 열의 너비 지정을 위해 선언한다. Style=”width:100px;”, span 속성을 필요에 따라 선택적으로 사용하며 아래와 같은 순서로 선언한다.

|  |
| --- |
| <colgroup>  <col style=”width:100px;” span="2" />  <col style=”width:100px;” />  <col style=”width:100px;” />  <col style=”width:100px;” />  <col />  <col style=”width:10%;” />  <col style=”width:10%;” />  </colgroup> |

**2.4.5 <thead>**

표 머리글을 그룹화 할 때 선언한다. 이 요소는 th 요소만으로 그룹화 되지 않으면 선언하지

않는다.

|  |
| --- |
| <thead>  <tr>  <th>division</th>  <th scope="col">price($)</th>  <th scope="col">calories(kcal)</th>  </tr>  </thead> |

**2.4.6 <tfoot>**

표 바닥글을 그루핑할 때 선언한다. 이 속성은 선택적으로 사용한다.

- 본 가이드에서는 **HTML5 문서를 원칙**으로 하는 것으로 정의 한다.

- HTML5 : tbody 아래에 위치해야 한다.

|  |
| --- |
| <thead>  …  </thead>  <tbody>  …  </tbody>  <tfoot>  <tr>  <th scope=”row”>total</th>  <td>5</td>  <td>650</td>  </tr>  </tfoot> |

- XHTML : thead와 tbody 요소 사이에 위치해야 한다.

|  |
| --- |
| <thead>  …  </thead>  <tfoot>  <tr>  <th scope=”row”>total</th>  <td>5</td>  <td>650</td>  </tr>  </tfoot>  <tbody>  …  </tbody> |

**2.4.7 <th>**

- 표에 셀 제목이 명시되지 않은 경우에도 th 요소를 필수로 선언하여 의미에 맞는 제목을

명시한다.

표에 셀 제목이 없는 문서는 다른 요소를 사용하여 마크업 한다.

- scope 속성은 반드시 선언해야 한다.

|  |
| --- |
| <th scope="col">price($)</th> |

**참고**

scope 속성: th 요소에 동반되는 속성으로서 현재의 셀이 어느 셀의 제목인지 범위를 명시한다.

scope 속성의 값으로는 col, row가 있다.

colgroup, rowgroup은 tbody 전체에 영향을 주는 속성으로 사용 시 주의한다.

**2.4.8 <tbody>**

표 본문을 그룹화 하기 위해 선언한다. 테이블의 본문(body)이 하나이고 thead나 tfoot이 없을

경우 생략할 수 있다.

|  |
| --- |
| <tbody>  <tr>  <th scope="row">apple</th>  <td>2</td>  <td>300</td>  </tr>  </tbody> |

**2.5 기타 요소**

**2.5.1 <a>**

href, target, title 순서로 속성을 선언한다.

- 새 창으로 페이지를 표시해야 할 때 target 속성을 선택적으로 사용한다.

- class로 디자인을 제어할 경우 class 속성을 선택적으로 사용한다.

- title 속성은 예고 없이 새 창을 표시해야 하거나 이동 경로를 정확히 알 수 없을 때, 또는

브라우저에 독립적으로 툴팁을 표현하기 위해 사용한다.

|  |
| --- |
| <a href="print.htnl" target="\_blank" title="opens in a new window">print</a> |

**2.5.2 <iframe>**

- iframe은 페이지 성능에 영향을 주기 때문에 가급적 사용하지 않는다. 부득이하게 사용해야 할

경우 아래 설명을 참고하여 선언한다.

- XHTML의 DTD

\* src, width, height, title, frameborder, marginwidth, marginheight, scrolling 순서로 속성을

선언한다. .

\* HTML5 속성 추가, 제외

\* 추가 : srcdoc, sandbox

\* 제외 : Align, frameborder, longdesc, marginheight, marginwidth, scrolling

\*- 스크린 리더 사용자를 위해 title 속성에 제목을 표기한다.

|  |
| --- |
| <iframe src="nbox.html" style=”width:410px;height:800px;” title="notice posts" ></iframe> |

|  |
| --- |
| <iframe sandbox src="nbox.html" style=”width:410px;height:800px;” title="notice posts" ></iframe> |

**2.5.3 <img>**

src, width, height, title, alt, usemap 순서로 속성을 선언한다.

- 이미지 내용과 동일한 값을 alt 속성에 표기하여, 이미지를 볼 수 없는 환경(스크린 리더,

이미지 서버 장애, 이미지 표시 하지 않음 설정)에서도 내용을 확인할 수 있게 한다.

- title 속성을 선언한 경우에도 alt 속성을 함께 선언한다.

- title 속성은 브라우저에 독립적으로 tooltip을 표현하기 위해 사용한다.

- width, height 속성은 선택적으로 사용한다

- html5 에서 longdesc 제외되었지만 html5.1 속성 유지로 안내하고 있다.

|  |
| --- |
| <img src="service\_center.png" style=”width:30px;height:10px;” title="service center" alt="service center -asking questions" usemap="#help"> |

**2.5.4 <map>**

map 요소의 name 속성을 선언하여 img 요소의 usemap 속성과 같은 이름으로 연결한다.

|  |
| --- |
| <img src=“img\_main.gif title="event" alt="New year events" usemap="#event">  <map name="event" id="event">  <area shape="rect" coords="506,48,608,139" href=“" alt="place of new year events" />  </map> |

**2.5.5 <area>**

shape, coords, href, target, title, alt 순서로 속성을 선언한다.

- title 속성을 선언한 경우에도 alt 속성을 함께 선언한다.

- target 속성은 새 창으로 페이지를 표시해야 할 때 사용한다.

- title 속성은 예고 없이 새 창을 표시해야 하거나 이동 경로를 정확히 알 수 없을 때 또는

브라우저에 독립적으로 tooltip을 표현하기 위해 사용한다.

|  |
| --- |
| <area shape="rect" coords="506,48,608,139" href="" target="\_blank" title="opens in a new window" alt="place of new year events " /> |

**3 CSS 코드 작성 규칙**

**3.1 기본 규칙**

**3.1.1 영문 소문자 사용**

모든 속성은 영문 소문자로만 작성한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| .class{Font-Family:AppleGothic;} | .class{font-family:AppleGothic;} |

**3.1.2 따옴표 사용 범위**

한글 폰트 선언, <string> 데이터 타입은 작은따옴표(' ')로 감싸며, 그 외의 경우에는 따옴표를

사용하지 않는다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| font-family:돋움 | font-family:'돋움' |
| background:url('bg.gif') 0 0 no-repeat | background:url(bg.gif) 0 0 no-repeat |
| content:"Chapter:" | content:'Chapter:' |

**3.2 들여쓰기**

CSS 코드를 작성할 때는 들여쓰기를 하지 않는다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| #content{border:1px solid #000;}  #content .class{color:#000;} | #content{border:1px solid #000;}  #content .class{color:#000;} |

**3.3 공백**

**3.3.1 선택자 간 공백 제거**

쉼표로 구분되는 선택자 간 공백은 없음을 원칙으로 하되 1칸 공백을 허용한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| a:hover,^^a:active,^^a:focus{textdecoration:  underline;} | a:hover,a:active,a:focus{text-decoration:underline;} |
| a:hover,^a:active,^a:focus{text-decoration:underline;} |

**3.3.2 속성 간 공백 제거**

속성 간 공백은 없음을 원칙으로 하되 1칸 공백을 허용한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| .class p{color:#000;^^line-height:18px;} | .class p{color:#000;line-height:18px;} |
| .class p{color:#000;^line-height:18px;} |

**3.3.3 중괄호 좌우 공백 제거**

중괄호 좌, 우의 공백은 제거한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| .class p^{color:#000;} | .class p{color:#000;} |
| .class p{^color:#000; line-height:18px;^} | .class p{color:#000;line-height:18px;} |

**3.4 빈 줄**

의미 있는 객체를 구분하기 위하여 코드 그룹 간 1줄씩 빈 줄을 넣을 수 있다. 단, 빈 줄의 간격은 1줄을 초과하지 않게 한다. 빈 줄의 사용은 선택 사항이다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| /\* Meaningful group 1 \*/  .class1{border:1px solid #000;}  .class1 img{border:1px solid #000;}  Blank line  Blank line  /\* Meaningful group 2 \*/  .class2{border:1px solid #000;}  .class2 img{border:1px solid #000;} | /\* Meaningful group 1 \*/  .class1{border:1px solid #000;}  .class1 img{border:1px solid #000;}  Blank line  /\* Meaningful group 2 \*/  .class2{border:1px solid #000;}  .class2 img{border:1px solid #000;} |

**3.5 인코딩**

폰트 이름이 영문이 아닐 때 이를 브라우저에서 바르게 표현하고, HTML에서 불러온 스타일을

제대로 렌더링 하려면 반드시 CSS 인코딩을 선언해야 한다. HTML과 동일한 인코딩을 문서 첫

줄에 공백 없이 선언한다.

|  |
| --- |
| @charset "utf-8"; |

파일을 저장할 때는 반드시 선언한 인코딩과 동일한 인코딩을 선택한다.

**3.6 선택자**

**3.6.1 공통 선택자 사용 지양**

공통 선택자 '\*'는 웹 페이지의 성능을 떨어뜨리고, Internet Explorer에서는 주석까지 영향을 받을

수 있으므로 사용하지 않는다.

**3.6.2 id, class 선택자 범위**

id 선택자는 4.2 id 및 class 네이밍 규칙에서 정의한 내용을 참조한다.

**3.7 속성**

**3.7.1 속성 선언 순서**

속성을 선언할 때는 그 쓰임새가 레이아웃과 관련이 큰 것에서 시작하여 레이아웃과 무관한 것

순서로 선언한다.

대표되는 속성 중 (그룹) 표기된 것은 약식속성으로, 약식속성을 선언했을 때 속성 값의 순서와

동일하게 전체 속성을 선언하며 다음 표에 표기된 순서를 참고한다.

|  |  |  |  |
| --- | --- | --- | --- |
| 순서 | 의미 | 대표되는 속성(그룹) | 관련 항목 |
| 1 | 위치 | position | top, right, bottom, left, z-index |
| 2 | 흐름 | float | clear |
| 3 | 표시 | display | visibility |
| 4 | 크기 | width $ height | min-, max- |
| 5 | 간격 | padding $ margin(Group) |  |
| 6 | 폰트 | font(Group) | font-size, line-height, color, font-weight, letter-spacing, text-align, vertical-align, text-indent, text-shadow, text-decoration, white-spacing, word-wrap 등 |
| 7 | 테두리 | border(Group) | border-radius, box-sizing |
| 8 | 배경 | background(Group) | background-size, gradients |
| 9 | 효과 | transform, transitions | transform, transition, perspective, animateions |
| 10 | 넘침 | overflow | - |
| 11 | 기타 |  | 위에 언급되지 않은 나머지 속성들로 배경의 관련 속성 이후에 선언하며, 기타 속성 내의 선언 순서는 무관함. |

\* [속성 선언 순서 기준: 1~5: 레이아웃, 6 폰트, 7~8: 테두리/배경, 9: 효과, 10 넘침, 11: 기타]

약식속성 전체속성 선언 순서

|  |  |
| --- | --- |
| 약식 속성 | 전체 속성 선언 순서 |
| padding | padding-top, padding-right, padding-bottom, padding-left |
| margin | margin-top, margin-right, margin-bottom, margin-left |
| font | font-size, font-weight, font-family, font-style, font-variant |
| border | border-top, border-right, border-bottom, border-left, border-width, border-top-width, border-right-width, border-bottom-width,  border-left-width, border-style, border-top-style, border-right-style, border-bottom-style, border-left-style, border-color, border-top-color, border-right-color, border-bottom-color, border-left-color |
| border-radius | border-top-left-radius, border-top-right-radius, border-botom-right-radius, border-bottomleft-radius |
| background | background-image, background-position, background-color, background-repeat, background-attachment,, background-size |
| transition | transition-duration, transition-timing-function, transition-delay, transition-property |

**3.7.2 속성값 축약**

CSS 최적화를 위해 다음과 같이 속성값을 축약한다.

|  |  |  |
| --- | --- | --- |
| 축약 전 | 축약 후 | 설명 |
| #555555 | #555 | 16 진수 컬러 코드값  동일한 값으로 이루어진 16 진수 컬러 코드값은 세 자리수로 축약하여 사용한다. 단 IE 전용 속성인 CSS filter 를 사용했을 경우 축약 속성을 인식하지 못하는 오류가 있기 때문에 속성값을 축약하지 않는다. |
| #ff4400 | #f40 |
| top:0px | top:0 | 0 의 단위  속성값이 0 일 경우 단위를 표시하지 않는다. |
| padding:20px 20px 20px 20px | padding:20px | 동일한 속성값  상, 우, 하, 좌의 속성값이 동일하면 축약한다. |
| margin:0 auto 0 auto | margin:0 auto |
| padding:20px 30px 50px 30px | padding:20px 30px 50px |
| Border-color:#ccc #eee #ccc #eee | Border-color:#ccc #eee |

**참고**

문자로 표현된 위치 값을 숫자로 변환할 때 다음과 같이 한다.

|  |
| --- |
| top, left → 0  right, bottom → 100% |

**3.7.3 약식 속성 사용 범위**

background와 border는 약식 속성을 우선적으로 사용하며, font 약식 속성은 사용하지 않는다.

**3.7.3.1 background**

속성값은 background-image, background-position, background-repeat, background-color, background-origin, background-clip, background-attachment 순서로 선언한다.

- 배경 스타일 속성을 초기 선언할 때는 반드시 background 단일 속성을 사용하며, 이후 배경의

부분적인 속성이 변경될 경우 background 관련 속성을 선언한다.

|  |
| --- |
| .class{background:url(bg.gif) 0 0 no-repeat #ccc;background-size:100% 50%;}  .class\_v1{background-position:0 -50px;}  .class\_v2{background-position:0 -100px;} |

- 다중 배경 스타일

|  |
| --- |
| .class{background:url(bg.gif) 0 0 no-repeat, url(img\_flwr.gif) right bottom no-repeat;} |

**3.7.3.2 border**

속성값은 border-width, border-style, border-color 순서로 선언한다.

테두리 스타일 속성을 초기 선언할 때는 반드시 border 단일 속성을 사용하고, 이후 테두리의

부분적인 속성이 변경될 경우 border 관련 속성(border-width, border-style, border-color)을

선언한다.

|  |
| --- |
| .class{border:1px solid #ccc;}  .class\_v1{border-color:#666;}  .class\_v2{border-style:dotted;}  .class2{border-top:1px solid #ccc;}  .class2\_v1{border-top-color:#666;}  .class2\_v2{border-top-style:dotted;} |

테두리의 상, 우, 하, 좌 스타일이 2개 이상 다르면, 공통 스타일을 약식 속성으로 선언한 후 다른

부분은 관련 속성으로 선언한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| .class{border:1px solid #ddd;border-bottom  :1px solid #eee;border-left:1px solid  #eee;} | .class{border:1px solid;border-color:#ddd  #ddd #eee #eee;} |
| .class{border-top:1px solid #ddd;border-right:  1px dotted #ddd;border-bottom:1px solid  #eee;border-left:1px dotted #eee;} | .class{border:1px; border-style:solid dotted;  border-color:#ddd #ddd #eee #eee;} |

**3.7.3.3 font**

- 폰트 스타일은 약식 속성으로 선언하지 않는다.

- 폰트 스타일을 약식 속성으로 선언하면 다음과 같은 문제가 발생한다.

- 아래와 같이 선언하면, font-weight:bold는 상속되지 않고 font 속성의 기본 값인 fontweight:

normal로 변경되기 때문에 불필요한 속성값을 선언해야 하는 문제가 발생한다.

|  |
| --- |
| .class{font-size:12px; font-weight:bold;font-family:dotum;}  .class p{font:15px gulim;} |

결국, class p의 폰트 스타일은 아래와 같아진다.

|  |
| --- |
| .class p{font-family:gulim;font-style:normal;font-variant:normal;font-weight:normal; fontsize:15px;line-height:normal} |

**3.8 z-index**

z-index 속성의 속성값의 범위는 최소 10, 최고 1000이며, 10 단위로 증감한다. 단, 10 단위 사이

의 예외 변수가 발생하면 1 단위 값을 지정할 수 있다.

팝업 등 레이어의 경우는 1000부터 시작 한다.

각 서비스의 z-index값은 상황에 맞게 선택적으로 선언한다.

**3.9 핵(hack)**

핵(hack)은 가능한 사용하지 않는다.

**3.10 @font-face (웹폰트)**

Font-family, font-weight, src 순서로 선언한다.

|  |
| --- |
| @font-face{  font-family:'NotoKhmer';  font-weight: 400;  src:url("font/NotoSansKhmer-Thin.ttf") format("truetype");  } |

**3.11 @media**

다양한 디바이스 특징의 일부를 참조하여 여러가지 뷰를 갖도록 구현할 때 선언 한다.

|  |
| --- |
| @media only all and (Conditional statement) {  Execution statement  } |

조건문이 될 수 있는 특징들

|  |
| --- |
| Width / height, Device-width / device-height, Orientation, Aspetct-ratio,  Device-aspect-ratio, Color, Color-index, Monochrome, Resolution, Scan, Grid |

@media는 선언 할 선택자의 의미 있는 그룹화 된 영역 다음에 위치 한다.

**3.12 CSS3 Support in Browser**

CSS3 표준안이 아직 확정되지 않은 상태에서 각 브라우저들은 Vendor Prefix(벤더 프리픽스)를 통

하여 CSS3 속성을 실험적으로 제공하고 있으며, 각 브라우저별 Vendor Prefix는 아래와 같다.

|  |
| --- |
| webkit based browser (Safari, Chrome) : -webkit-  Mozilla (Firefox) : -moz-  Opera : -o-  Internet Explorer : -ms- |

단, 웹 호환성 점검 시 문제 될 경우 사용하지 않는다.

**3.13 CSS 선언 타입**

CSS 선언 타입은 크게 세 가지로 분류하며, 용도에 맞게 사용한다.

**3.13.1 External type**

CSS를 선언하는 가장 기본적인 방식으로, CSS 파일이 별도로 존재하는 형태이다. link 요소를 통해 HTML과 CSS 파일을 연결한다.

|  |
| --- |
| <link rel="stylesheet" href="/common/css/common.css"> |

**3.13.2 Internal(embedded) type**

HTML 파일의 head 안에 스타일을 선언하는 방식으로, 단발성 페이지의 CSS 분량이 작을 경우

사용한다. 단, 권장하지 않는다.

|  |
| --- |
| <head>  <style>  …  </style>  </head> |

**3.13.3 Inline type**

HTML의 개별 요소에 style 속성을 이용하여 스타일을 선언하는 방식으로, 속성값이 동적으로

변경되어야 하는 경우 사용한다.

속성값에 사용되는 %와 같은 특수기호는 entity 코드로 변환하지 않는다.

|  |
| --- |
| <div style=”top:0;left:50%;”>  …  </div> |

**3.14 주석**

**3.14.1 기본 형식**

CSS 주석의 시작과 종료 주석은 아래와 같이 표기하며, 기본 형식에 맞게 작성한다.

|  |
| --- |
| Start an annotation /\* content of annotation \*/  End an annotation /\* //content of annotation \*/ |

- 주석 기호와 주석 내용 사이에는 반드시 공백 한 칸이 있어야 한다.

- 주석 기호와 주석 내용 사이에는 다른 기호가 올 수 없다.

- 주석 기호와 주석 내용 사이의 줄 바꿈은 허용하지 않는다. 단, 주석 내용 간 줄 바꿈은 허용한다.

- 시작과 종료 주석의 주석 내용은 동일해야 하며, 종료 주석은 생략 할 수 있다.

**3.14.2 작성자 정보 표기**

작성자 정보에는 소속 부서, 작성자, CSS 파일 생성 월, 연도(MMYY 형식)를 작성한다.

작성자 정보 밑에는 빈 줄(한 줄)을 두어 스타일을 선언하는 문장과 구분되도록 한다.

|  |
| --- |
| /\* Department Writer 0919 \*/  Blank line |

**3.14.3 의미 있는 그룹 영역의 주석 표기**

의미 있는 객체를 구분하기 위한 주석은 영역의 윗부분에 시작 주석만 표기한다. 초기화와

레이아웃 스타일 그룹을 제외한 의미 있는 그룹 영역의 주석 표기는 선택 사항이다.

|  |
| --- |
| /\* news \*/  #news{position:relative;top:25px;height:125px;}  #news ul{float:left;display:inline;width:365px;margin:0;}  #news ul li{float:left;display:inline;padding:0 1px;}  #news ul li.first{padding-left:0;background:none;}  #news ul li img{vertical-align:top;} |

**3.14.3.1 초기화 스타일 그룹**

레이아웃 템플릿에서 제공하는 default.css 사용한다.

**3.14.3.2 레이아웃 스타일 그룹**

레이아웃을 위한 스타일 선언 시 /\* Layout \*/ 또는 /\* 각 요소 명칭 \*/으로 그루핑한다

|  |
| --- |
| /\* Layout \*/  #wrap{…}  #header{…}  #container{…}  #footer{…} |

**4 네이밍 규칙**

**4.1 기본 규칙**

**4.1.1 일반 규칙**

이름은 영문 대, 소문자, 숫자, 언더스코어(\_), 하이픈(-)만을 사용하여 작성한다.

**4.1.2 시작이름**

이름은 영문 대문자, 숫자로 시작할 수 없다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| Btn | btn |
| 2btn | Btn2 |

**4.1.3 예약어**

- 예약어가 있는 경우 예약어를 사용한다.

- 예약어가 없으면, 종류와 특성을 나타내도록 네이밍 한다.

- 영문 대, 소문자, 숫자, 언더스코어(\_), 하이픈(-)만 사용할 수 있다.

- 언더스코어(\_) 와 하이픈(-)은 2개 이상의 단어를 조합할 때만 사용한다.

- 언더스코어(\_) 와 하이픈(-)을 이용하여 3단계 이내에서 조합한다.

단, 모바일임을 표현하는 단어 추가시는 4단계까지 허용한다.

- 단어와 숫자를 조합하는 경우 언더스코어(\_) 와 하이픈(-)을 사용하지 않으며, 숫자가 없으면

‘1’이라는 숫자가 생략된 것으로 간주한다.

- 숫자 정보 경우 언더스코어(\_) 와 하이픈(-)을 사용 하며 ‘1‘ 표기 한다

|  |  |  |
| --- | --- | --- |
| 구분 | 잘못된 예 | 올바른 예 |
| 일반형 | sectionLst | section\_list |
| 숫자형 | section\_list\_style\_type  section, section\_1, section\_2 | section\_list\_type  section, section2, section3  no\_1, no\_2, no\_3 ...... no\_10 |

**4.1.4 축약어**

- 축약어는 명시적으로 의미를 이해 할 수 있도록 지정한다.

- 예약어에는 축약어를 포함 할 수 있다.

- 아래 지정한 축약어만 사용 할 수 있다.

|  |  |
| --- | --- |
| 축약어 | 단어 |
| lnb | Local Navigation Bar |
| gnb | Global Navigation Bar |
| nav | Navigation |
| btn | Button |
| m | Mobile |
| bu | Bullet point |
| bg | Background |
| img | Image |
| info | Information |

**4.2 id 및 class 네이밍 규칙**

**4.2.1 id, class**

- id는 문서에서 한 번만 사용한다.

- class는 문서에서 여러 번 사용할 수 있다.

**4.2.2 레이아웃 예약어**

- **메인 레이아웃**

메인 화면 레이아웃에는 다음 표에 예약된 id, class 를 사용한다.

|  |  |
| --- | --- |
| 예약어 | 범위 |
| #wrapper | 콘텐츠 전체 영역 |
| .warp | 콘텐츠 전체 영역 또는 부분 영역 |
| #header | 머리글 영역 |
| .visual | 비주얼 영역 |
| #container | 본문 전체 영역 |
| .colgroup, .rowgroup | 콘텐츠 구분 영역 |
| .quckmenu | 퀵 메뉴 영역 |
| #footer | 바닥글 영역 |

- **서브 레이아웃**

서브 화면 레이아웃에는 다음 표에 예약된 id, class 를 사용한다.

|  |  |
| --- | --- |
| 예약어 | 범위 |
| #wrapper | 콘텐츠 전체 영역 |
| .warp | 콘텐츠 전체 영역 또는 부분 영역 |
| #header | 머리글 영역 |
| .sub\_visual | 비주얼 영역 |
| #container | 본문 전체 영역 |
| #side | 좌측 메뉴 전체 영역 |
| .colgroup, | 콘텐츠 전체 영역 |
| #contents | 본문 콘텐츠 영역 |
| .quckmenu | 퀵 메뉴 영역 |
| #footer | 바닥글 영역 |

**4.2.3 레이아웃 네이밍 조합**

- 레이아웃 id의 네이밍은 조합하여 사용할 수 없다.

- 레이아웃에 디자인 속성을 추가/변경하려면 class를 사용한다.

|  |  |
| --- | --- |
| 잘못된 예 | 올바른 예 |
| #container2 | #container .style2 |
| #container\_type | #contaniner .type2 |

표 레이아웃 네이밍 조합 예

|  |
| --- |
| <div id="container" class="type2">  [Contents]  </div> |

**4.2.4 객체 네이밍**

- 객체 예약어는 다음 표에 근거하여 작성한다.

|  |  |  |
| --- | --- | --- |
| 구분 | 예약어 | 분류 |
| 공통 | .gnb | 최상위 전역 네비게이션 영역 |
| .lnb | 현재 사이트의 주요 네비게이션 영역 |
| .search | 검색 영역 |
| .quickmenu | 빠른 메뉴 |
| .sub\_title | 서브 1차메뉴 제목 영역 |
| .side\_menu | 서브 메뉴 목록 영역 |
| .sub\_head | 콘텐츠 제목 영역 |
| .path | 현재 페이지의 경로 |
| .footer\_menu, .footer\_info | 바닥글 영역 |
| 페이지 | .cts\_page number | 콘텐츠 페이지 요소 |

- 작업환경에 따라 ID로 사용 할 수 있다.

- 팝업, iframe에도 동일한 규칙을 적용한다.

**4.2.5 팝업/배너 레이아웃 예약어**

- 팝업 문서의 레이아웃 지정 범위는 동일하다.

- 레이아웃 예약어 앞에 'popup\_', ‘banner\_’ 를 조합하여 사용한다.

|  |  |
| --- | --- |
| 예약어 | 범위 |
| .popup | 팝업 전체 영역 |
| .popup\_list | 팝업 목록 |
| .popup\_all\_list | 팝업 전체 목록 |
| .popup\_control | 팝업 제어 버튼 영역 |
| .popup\_btn\_ | 팝업 제어 버튼 객체 요소 |
| .banner | 배너 전체 영역 |
| .banner\_list | 배너 목록 |
| .banner\_all\_list | 배너 전체 목록 |
| .banner\_control | 배너 제어 버튼 영역 |
| .banner\_btn | 배너 제어 버튼 객체 요소 |

**4.2.6 모바일 레이아웃 예약어**

모바일 레이아웃 예약어는 상위의 레이아웃 예약어의 객체영역명 다음에 mobile 또는 m을 포함한다.

|  |  |  |
| --- | --- | --- |
| 예약어 | 법위 | 모바일 |
| .lnb\_button\_open | 주메뉴 제어 버튼 | .lnb\_m\_nav |
| .search\_btn | 검색 제어 버튼 | .search\_m\_btn |
| .popup\_button\_play | 팝업 제어 버튼 객체 요소 | .popup\_m\_button\_play |

**4.3 이미지 네이밍 규칙**

**4.3.1 이미지 네이밍**

- 이미지 네이밍은 이미지 예약어에 근거하여 작성한다.

|  |  |
| --- | --- |
| 예약어 | 분류 |
| h\_ | 제목 |
| p\_ | 문장 |
| gnb\_, lnb\_, side\_menu\_ | 네비게이션 |
| tab\_menu | 탭 |
| btn\_ | 버튼 |
| bu\_ | 블릿 |
| ico\_ | 아이콘 |
| line\_ | 선 |
| list\_ | 목록 |
| title\_ | 타이틀 |
| bg\_ | 배경, 박스 |
| img\_ | 이미지 |
| \_off, \_on, \_active, \_over | 상태변화 |
| cts\_page number | 콘텐츠 페이지 요소 |

- 같은 분류의 이미지가 두 개 이상이면 파일 이름 마지막에 숫자를 추가하여 구분한다.

- 이미지 네이밍은 이미지 확장자와 관계 없이 순차적으로 적용한다.

예) bu\_dot.gif, bu\_dot2.jpg, bu\_dot3.png

- 페이지 구성 요소 이미지 네이밍은 cts\_메뉴키\_ 형식으로 구성한다.

**4.3.2 이미지 네이밍 조합**

- 이미지 이름은 ‘의미(객체영역)\_형태\_상태’ 순서로 조합한다.

- 모바일 이미지 이름은 ‘의미(객체영역)\_모바일\_형태\_상태’ 순서로 조합한다.(4단계)

|  |  |  |
| --- | --- | --- |
| 잘못된 예 | 올바른 예 | 설명 |
| on\_recommend\_tab1 | teb\_menu\_recomm\_on | ‘의미\_형태\_상태’ 순서로 조합한다. |
| bnm.png | main\_btn.png | 임의로 축약하지 않는다. |
| btn\_search\_seoul\_mail.png | search\_btn\_mail.png | 3단계 이하로 조합한다. |
| 1btn\_search.png | search\_btn.png | 숫자로 시작하지 않는다. |
| m\_serh\_btn.png | search\_m\_btn.png | 모바일 예약어 사용시 4단계까지 허용한다 |

**4.4 파일 및 폴더 네이밍 규칙**

**4.4.1 파일 및 폴더 네이밍 규칙**

- CMS 규칙에 맞은 네이밍으로 한다.

- 산출되는 모든 HTML 파일과 폴더를 분류해야 할 경우, 파일 이름, 폴더 이름 앞에 '사이트 코드' 또는 ‘그루핑 네이밍’을 사용하여 그루핑한다.

|  |  |  |
| --- | --- | --- |
| 분류 | 올바른 예 | 설명 |
| HTML | 00.thml, site code\_00.html | 프로젝트 규정에 맞게 네이밍한다. |
| pop\_html | 팝업 파일 사용한다. |
| iframe\_html | iframe 파일 사용한다. |
| longdesc\_.html  longdesc\_.txt | longdesc\_메뉴번호.thml로 사용한다.  longdesc\_메뉴번호.txt로 사용한다. |
| CSS | main.css, board.css … | ‘서비스영문이름.css’로 사용한다. |
| Folder | images, css, js | image, css, javascript 폴더 사용 |
| contents | 콘텐츠 폴더 사용 |