



Publishing Guide

Date : 2012.12.07

Version : 3.00



History

[illegible]



table of contents

1. HTML 코드 작성 규칙
2. HTML 엘리먼트 작성 규칙
3. CSS 코드 작성 규칙
4. 네이밍 규칙
5. 제이쿼리 달력, 에디터
6. Template

1. HTML 코드 작성 규칙



파일의 경로

유지보수의 편의를 및 HTML 작업자의 퍼포먼스를 위하여 **상대 경로**로 작업한다.

작업이 완료된 HTML 파일은 로컬 브라우저와 다양한 OS에서 확인이 가능하여야 한다.

```
1 <!-- 스타일시트 파일 경로 -->
2 <link rel="stylesheet" type="text/css" href="../../common/css/common.css" />
3
4 <!-- 스크립트 파일 경로 -->
5 <script type="text/javascript" src="../../common/js/jquery-1.4.2.min.js"></script>
6
7 <!-- 이미지 파일 경로 -->
8 
```



head 영역의 파일 링크

웹 문서 전송 속도를 높이기 위하여 최소한의 파일을 사용한다. 아래의 4개의 파일이 기본 파일이다.

```
1 <!-- 기본 전역 설정 스타일 시트 -->
2 <link rel="stylesheet" type="text/css" href="../common/css/basic.css" />
3
4 <!-- 사이트 공통 레이아웃 설정 -->
5 <link rel="stylesheet" type="text/css" href="../common/css/layout.css" />
6
7 <!-- jQuery 라이브러리 사용(최대로 압축된 파일 사용) -->
8 <script type="text/javascript" src="../common/js/jquery-1.4.2.min.js"></script>
9
10 <!-- 프론트에 사용되는 스크립트 -->
11 <script type="text/javascript" src="../common/js/basic.js"></script>
```

1. 기본 레이아웃과 구조를 달리하는 페이지는 별도의 layout.css를 링크한다. 예를 들어 팝업 페이지의 경우 popup_layout.css
2. 반응형웹의 경우 layout.css를 basic.css에 포함하여 퍼블리싱 한다.
3. 기본 지정된 js 파일 이외의 스크립트 파일의 사용이 있을 경우 body 영역안에서의 javascript 파일의 링크를 허용한다.



표준 문법의 사용

HTML은 해당 DTD의 명세에 맞게 작성하며, W3C validation을 통과해야 한다.

1. DTD를 제외한 모든 엘리먼트와 애트리뷰트는 소문자로 작성한다.
2. 애트리뷰트의 값은 큰따옴표("")로 묶는다.
3. 특수기호는 Entity name을 사용하여 entity 코드로 변환한다. Entity 코드는 ISO-8859-1을 기준으로 한다.

* 참고 : [HTML ISO-8859-1 Reference](https://www.w3.org/TR/html401/iso-8859-1.html)

주의 해야 할 특수기호

Character	"	&	<	>		.
Entity Name	"	&	<	>	 	ċ



파일의 인코딩

*.html, *.css, *.js의 파일의 저장 방식을 설정할때
반드시 개발과 협의가 필요하다.

문자셋(charset)별 인코딩

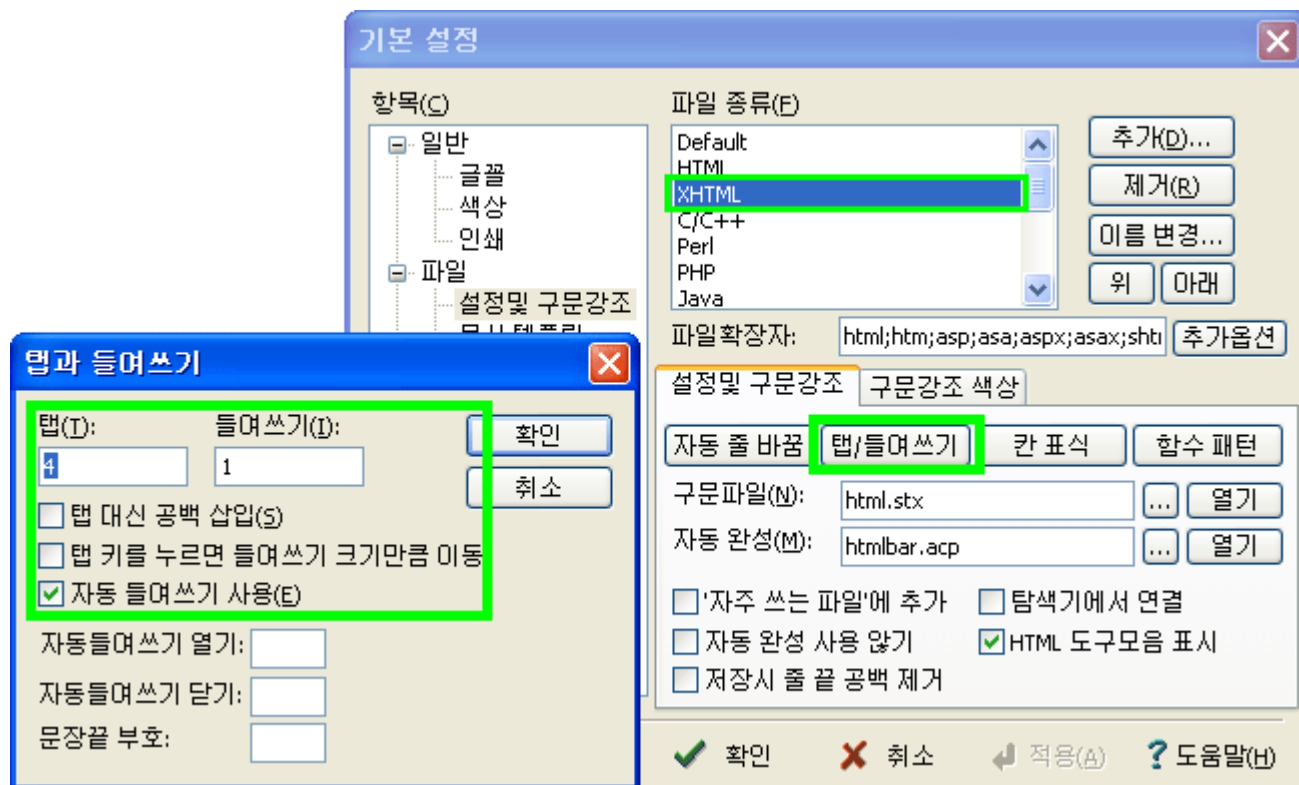
문자셋(charset)	기본 인코딩
charset=utf-8	utf-8
charset=euc-kr	ANSI



코드 들여쓰기

코드의 가독성을 높이고 전체 HTML 구조를 쉽게 파악하기 위하여 들여쓰기 규칙을 준수한다.

마크업의 중첩이 깊어짐에 따라 자식 엘리먼트를 들여 쓰고 탭의 공백을 설정한다.





코드 들여쓰기를 하지 않는 경우

1. html의 자식 엘리먼트인 head, body
2. head의 자식 엘리먼트인 meta, link, script
3. body의 자식 첫 번째 엘리먼트 div="wrapper"

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
5 <title>퍼블리싱 가이드</title>
6 <link rel="stylesheet" type="text/css" href="../common/css/basic.css" />
7 <script type="text/javascript" src="../common/js/jquery-1.4.2.min.js"></script>
8 <script type="text/javascript" src="../common/js/basic.js"></script>
9 </head>
10 <body>
11 <div id="wrapper">
```



빈줄 및 주석

1. 그룹을 감싸고 있는 객체를 구분하기 위하여 코드 그룹 간 1줄씩 빈 줄을 만드는 것은 허용하며, 빈 줄의 간격이 1줄을 초과할 수 없다.
2. 주석 기호와 주석 내용 사이에는 반드시 공백 한 칸이 있어야 한다.
3. 주석 기호와 주석 내용 사이에는 다른 기호가 올 수 없다.
4. 시작과 종료 주석의 내용은 동일해야 하며, 종료 주석 내용 앞에 //문자를 추가한다.
5. 레이아웃 엘리먼트는 id이름과 동일한 주석내용을 작성한다.
6. 레이아웃 엘리먼트를 제외한 독립된 콘텐츠 영역의 주석 표기는 선택 사항이다.
7. 개발참고 주석은 내용 앞에 [D]문자를 추가하며, 종료 주석은 사용하지 않는다.

```
1  <!-- header -->
2  <div id="header">...</div>
3  <!-- //header -->
4
5  <!-- container -->
6  <div id="container">
7      <!-- content -->
8      <div id="content">
9
10         <!-- [D]개발참고 주석문 -->
11         <div class="paging">...</div>
12
13     </div>
14     <!-- //content -->
15 </div>
16 <!-- //container -->
17
18 <!-- footer -->
19 <div id="footer">...</div>
20 <!-- //footer -->
```

2. HTML 엘리먼트 작성 규칙



문서형(Document Type Definition) 선언

문서형을 선언하지 않으면 브라우저 호환성을 확보하지 못함. 문서형을 선언하는 것은 상호 운용성을 보장하기 위함이다.

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

1 <!DOCTYPE html>
```

※ DTD 설정별 표준 문법으로 마크업 하더라도 Quirks Mode로 인식하여 바르게 해석되지 않은 경우

DTD가 선언되지 않은 경우(html 태그로 문서시작)

```
1 <html>
```

선언한 DTD 앞에 다른 문자가 오는 경우

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



언어 및 문자셋(charset) 지정

lang 애트리뷰트는 User Agent가 언어를 올바르게 해석할 수 있게 도와주며, 검색과 음성 장치에 활용된다.

```
1 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">  
  
1 <html lang="ko">
```

utf-8 인코딩 기본. DB의 인코딩 방식과도 관련이 있으므로 개발자와 협의하여 결정.

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

1. utf-8은 다국어 지원이 가능하며, euc-kr 보다 표현 가능한 한글(고어, 음절 등)이 많음.
2. 한글은 utf-8에서 3바이트, euc-kr에서 2바이트를 차지하기 때문에 utf-8이 euc-kr에 비하여 DB 저장 용량이나 트래픽이 많을수 있음.



구조와 표현의 분리

문서의 마크업 언어(HTML)와 화면 표시(CSS)언어를 본래의 목적에 맞게 최대한 분리한다.

```
1 <input type="text" id="myId" class="my_class" style="width:200px" />
2
3 <colgroup>
4   <col style="width:200px" />
5 </colgroup>
```

1. 특정 엘리먼트에 id, class, style을 선언할 때에 선언 순서는 엘리먼트의 기본 애트리뷰트 지정후 마지막에 id, class와 style을 선언한다.
2. input[text], select, 테이블 셀 엘리먼트의 경우 너비의 고정폭 지정 허용함



Img 엘리먼트

1. alt 애트리뷰트는 반드시 제공되어야 하며 빈 값을 사용하지 않는다.
2. 의미가 없는 비주얼 이미지의 경우 alt 애트리뷰트 빈 값을 허용한다.
3. 유지보수의 편의성을 위하여 width, height 애트리뷰트는 사용하지 않는다.
4. 대체 텍스트 사용으로 일부 브라우저에서 말 풍선이 나타는 것을 방지해야 할 경우 title 애트리뷰트의 빈 값을 제공한다.
5. 이미지 맵 사용을 지양하며 포지셔닝 방법을 사용한다.

```
1 
```


3. CSS 코드 작성 규칙



CSS 코드 작성 기본규칙

모든 속성은 영문 소문자로만 작성

```
1 /* 잘못된 예 */
2 .class_name{color:#FFF}
3
4 /* 올바른 예 */
5 .class_name{color:#fff}
```

마지막 선언된 속성은 세미콜론(;)을 사용하지 않는다.

```
1 /* 잘못된 예 */
2 .class_name{margin:0;padding:10px}
3
4 /* 올바른 예 */
5 .class_name{margin:0;padding:10px}
```

선택자 간, 중괄호간 공백을 사용하지 않는다.

```
1 /* 잘못된 예 */
2 .class_name, .class_name .child_class {margin:0}
3 .class_name {padding:0; margin:0}
4
5 /* 올바른 예 */
6 .class_name, .class_name .child_class {margin:0}
7 .class_name {padding:0; margin:0}
```



CSS 코드 작성 기본규칙

들여쓰기는 허용하지 않는다.

```
1 /* 잘못된 예 */
2 .class_name{
3     padding:0;
4     margin:0
5 }
6
7 /* 올바른 예 */
8 .class_name{padding:0;margin:0}
```

CSS 주석 처리는 시작 주석을 작성하며 종료 주석은 작성하지 않는다. 종료코드 이후에 오는 요소에 한 칸 띄어 쓰기를 허용한다.

```
1 /* 주석1 */
2 .class_name1{padding:0;margin:0}
3
4 /* 주석2 */
5 .class_name2{padding:0;margin:0}
```

'*' 공통 선택자는 웹 페이지의 성능을 저하를 가져오기 때문에 사용하지 않는다.

```
1 /* 잘못된 예 */
2 *{padding:0;margin:0}
```



CSS 코드 작성 기본규칙

객체를 구분하기 위하여 코드 그룹 간 1줄의 빈 줄을 허용한다. 빈 줄의 1줄을 초과하지 않는다.

```
1  /* 잘못된 예 */
2  .class_name1{padding:0;margin:0}
3
4
5  .class_name2{padding:0;margin:0}
6
7  /* 올바른 예 */
8  .class_name1{padding:0;margin:0}
9
10 .class_name2{padding:0;margin:0}
```

id와 class를 조합하여 사용하지 않는다. id를 css로 표현하기 위하여 선택자로 사용하지 않는다.

레이아웃 요소로 예약되어 있는 id에 대해서는 예외 사항을 적용한다.

```
1  /* 잘못된 예 */
2  #header .lnb{padding:0;margin:0}
3
4  #header{padding:0;margin:0}
5
6  /* 올바른 예 */
7  .header .lnb{padding:0;margin:0}
8
9  .header{padding:0;margin:0}
```



CSS 코드 작성 기본규칙

속성의 축약/약식 속성 사용

```
1  /* 축약 전 */
2  .class_name{background-color:#ffffff}
3  .class_name{background-position:left center}
4  .class_name{margin:0px}
5  .class_name{margin:15px 15px 15px 15px}
6  .class_name{margin:0 auto 0 auto}
7  .class_name{margin:25px 15px 30px 15px}
8
9  /* 축약 후 */
10 .class_name{background-color:#fff}
11 .class_name{background-position:0 50%}
12 .class_name{margin:0}
13 .class_name{margin:15px}
14 .class_name{margin:0 auto}
15 .class_name{margin:25px 15px 30px}
```

4. 네이밍 규칙



공통 규칙

잘못된 예	올바른 예	설명
Tit_section	tit_section	시작이름은 영문 대문자, 숫자, 특수문자로 시작할 수 없다.(파일 및 폴더 제외)
candle_btn, 1_tab_on_type.gif	btn_candle, tab_type_1_on.gif	네이밍의 조합은 '형태_의미_순서_상태'을 기본 순서로 사용한다.
customerService	customer_service	언더스코어(_)의 조합은 파일, 폴더, 이미지, CSS 네이밍에 사용한다.
	cyber_center, cyber_center_2	동일한 이름의 네이밍의 경우 가장 뒤에 숫자를 사용하여 분류지어 사용할 수 있으며 첫번째 파일은 숫자를 생략하여 사용할 수 있다.



파일 및 폴더

개발과 기획에서 정의된 디렉토리 구조도 및 화면 아이디어가 정의가 되어있다면 그에 준한 규칙에 따라 파일 및 폴더의 이름을 사용한다.

분류	예제	설명
Folder	html	메뉴별 의미에 맞는 영문명으로 생성
	common	css, img, js 폴더 모음
	css	css파일 폴더
	img	img파일 폴더
	js	Javascript파일 폴더
CSS	import.css	reset.css, contents.css, layout.css 를 @import 한다.
	reset.css	브라우저 기본 css 속성을 초기화한 파일
	contents.css	콘텐츠 내용의 표현을 정의한 파일
	layout.css	레이아웃 구조의 표현을 정의한 파일
	basic.css	reset.css, contents.css, layout.css 등 @import 하지 않고 하나의 파일에 작성
Javascript	basic.js	전역에 사용하는 스크립트를 작성한다.



레이아웃 예약어

예약어	설명
wrap	페이지 전체 영역 그룹핑
header	머리글 영역
container	본문영역 그룹핑
content	주용컨텐츠 영역
footer	바닥글
pop_wrap	팝업 페이지 전체 영역 그룹핑
pop_header	팝업 머리글 영역
pop_container	팝업 본문영역 그룹핑
pop_content	팝업 주용컨텐츠 영역
pop_footer	팝업 바닥글



이미지 예약어

분류	예약어	설명
텍스트	tit_*	heading 요소에 사용
	txt_*	문단의 이미지로된 텍스트에 사용
링크	tab_*	탭 메뉴 형태에 사용
	btn_*	버튼 형식에 사용
박스	box_*	grouping 목적으로 사용
불릿	bul_*	리스트 내용 앞에 붙는 불릿에 사용
아이콘	ico_*	아이콘 이미지에 사용
상태	*_on, *_off, *_over	활성, 비활성, 오버 상태에 대한 표현에 사용

5. 제이쿼리 달력, 에디터



제이쿼리 달력

달력 UI는 제이쿼리에서 제공해주는 기본 스타일을 따른다. (색상 변경 가능)

<http://jqueryui.com/themeroller/>

December 2012						
Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

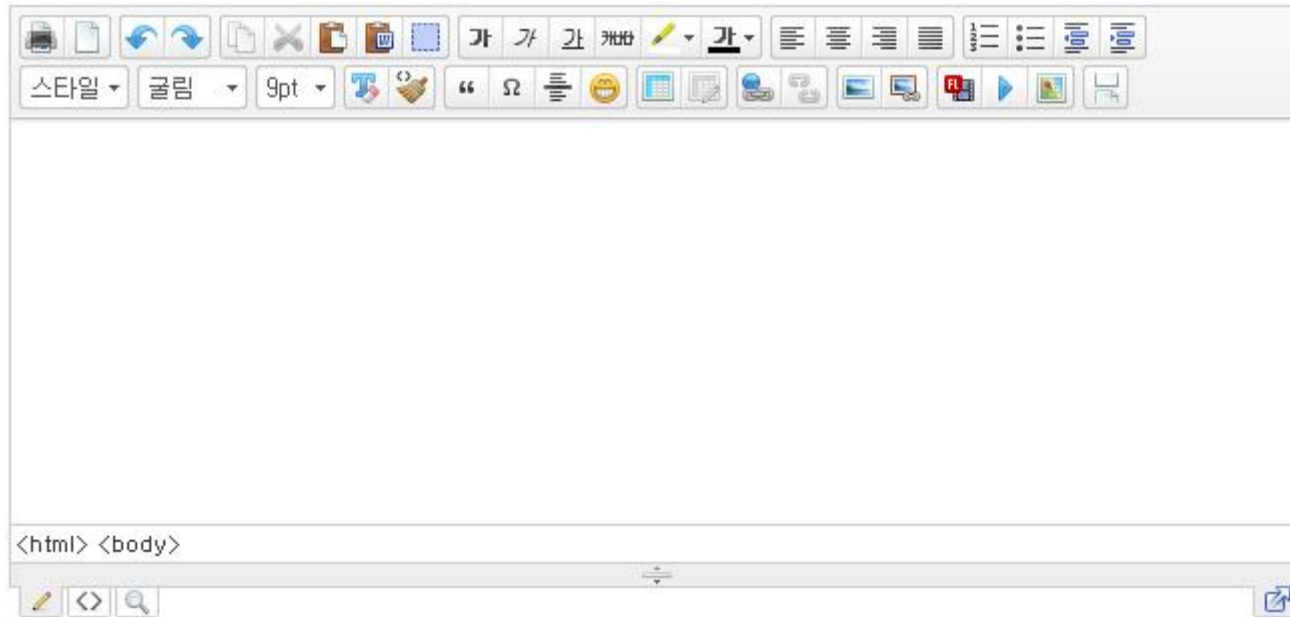
December 2012						
Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

December 2012						
Su	Mo	Tu	We	Th	Fr	Sa
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

에디터

에디터 기본 스타일

<http://www.chcode.com/cheditor/demo.shtml>



6. Template



제이쿼리 달력

1. select box 1 (스크롤 생길시 IE브라우저 문제됨)
2. select box 2 (ul, button 엘리먼트로 구성)
3. 디자인 scroll bar
4. 모바일 디바이스 scroll bar
5. 화면에 따라다니며, 클릭시 화면이 최상단으로 이동
6. 기본 터치 슬라이더(가벼움)
7. 기능이 여러가지인 터치 슬라이더
8. 콘텐츠 로테이션
9. 레이어 토글
10. FAQ 목록
11. Tree Navigation
12. Tab Navigation + List Item Navigation
13. 가로 Navigation
14. 세로 Navigation
15. input focus, label 숨기기
16. 디자인 파일찾기 input

로컬 소스 경로 : 퍼블리싱 가이드 v3.00 / source

웹 소스 경로 : <http://pub.s-prism.co.kr/source/index.html>