



Style Guide & Coding Convention

1. 폴더 네이밍

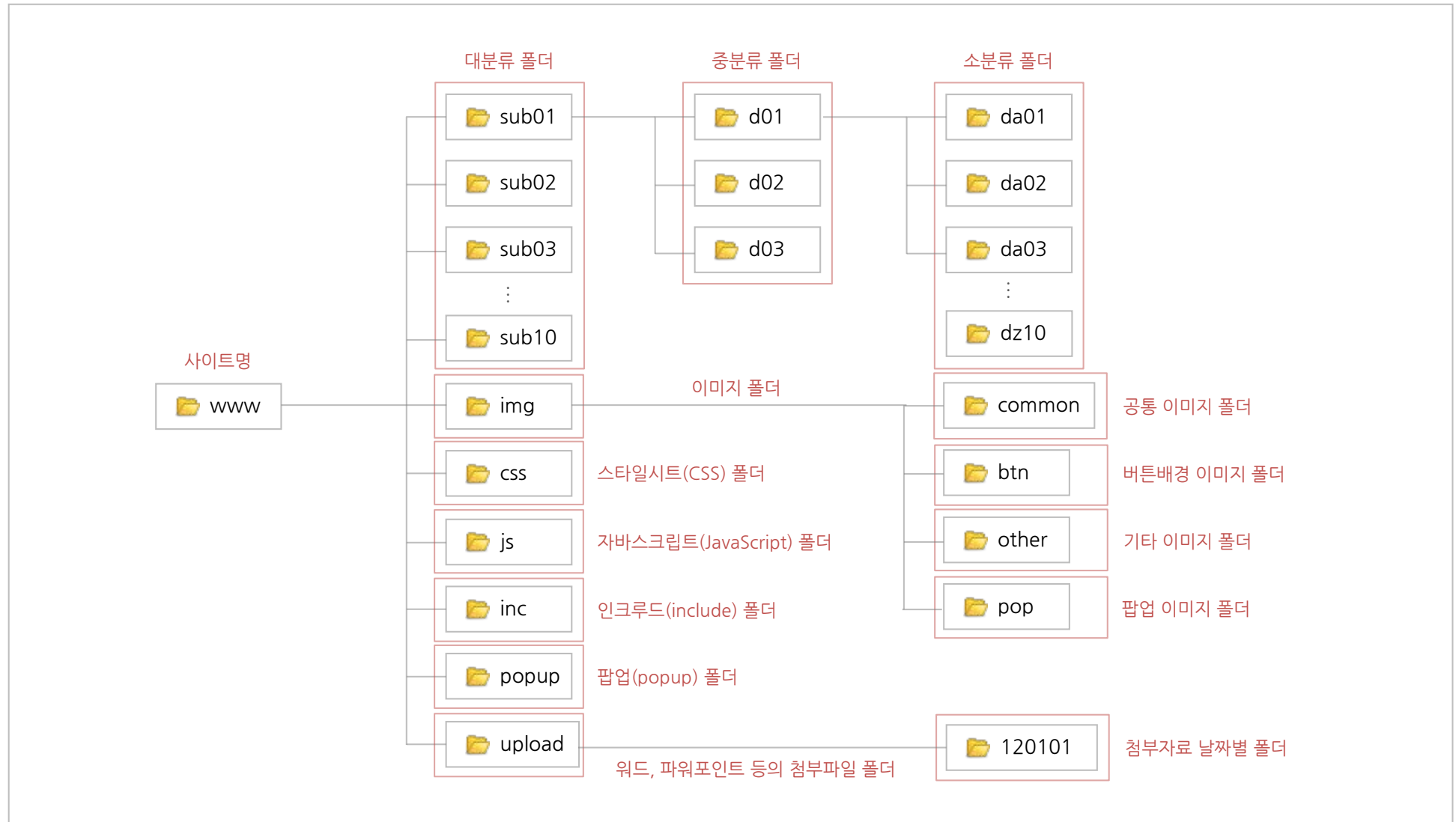
- 1) 모든 폴더명은 **소문자**로 작성하고 기호를 사용하지 않는다.
- 2) 폴더명이 영문과 숫자의 조합일 경우 일련번호를 붙여서 작성하고, 일련번호는 **2자리의 숫자**로서 순서에 따른 일련번호를 제공한다.
예) 📁 sub01, 📁 sub02, 📁 sub03...
- 3) 최상위 디렉터리는 **[사이트명]**으로 폴더를 생성하고, 임시 사이트 제작 시 **[www]**로 작성하여 폴더를 생성한다.
예) 📁 ismart (i-smart 사이트)
- 4) 최상위 디렉터리 **[사이트명]** 안의 기본 디렉터리는 **[대분류(메뉴)]**, **[img]**, **[css]**, **[js]**, **[inc]**, **[popup]**, **[upload]** 로 폴더 네이밍을 정의한다.
- 5) **[대분류]** 디렉터리 안으로 추가되는 **[중분류]**는 directory의 첫단어 'd'를 붙여 **[d(directory)+일련번호]** 순서로 작성하여 폴더를 생성한다.
예) 📁 d01, 📁 d02, 📁 d03... (중분류)
- 6) **[중분류]** 디렉터리 안으로 추가되는 폴더명은 **[d(directory)+알파벳(순서)+일련번호]** 순서로 작성하여 폴더를 생성한다.
예) 📁 da01, 📁 da02, 📁 da03... ~ 📁 dz01, 📁 dz02, 📁 dz03... (소분류)
- 7) **[대분류]** 폴더명은 사이트맵의 메뉴를 기준으로 **[sub01~10]** 단계 순으로 네이밍을 정의하고, **디렉터리는 10개 이하로 제한**한다.
(사이트맵 기획 시 대분류 메뉴는 6~7개의 구성을 기본으로 하고 분류가 복잡하여 구성이 늘어날 경우 10개 이하로 제한한다.)

- 8) **[img]** 안의 기본 디렉터리는 **[common]**, **[btn]**, **[other]**, **[pop]** 으로 폴더 네이밍을 정의한다.
- 9) **[img]** 안에는 정의된 폴더 네이밍의 디렉터리만으로 구성이 되며, 이미지 파일(gif, jpg, png)은 업로드 하지 않는다.
- 10) **[upload]** 안의 디렉터리 생성 시 **[120101(yymmdd)]**의 날짜형식을 사용하여 폴더를 생성한다.
예) 📁 120101, 📁 120102, 📁 120103...
- 11) 폴더 네이밍 규칙 **[표 1-1]**과 폴더 기본 구조 **[표 1-2]**를 참고하여 디렉터리를 생성한다. (P.5~6 참고)







[표1-1] 폴더 네이밍 규칙

폴더 분류		설명
www (임시)		프로젝트 (사이트명)
sub+일련번호 (sub01~sub10)		최상위에 종속된 대분류 (파일 업로드 x, 디렉터리 생성은 10개 이하로 규정함)
d+일련번호		대분류에 종속된 중분류
d+알파벳+일련번호		중분류에 종속된 소분류 (분류의 단계가 깊어질 경우 '알파벳+일련번호'를 추가하여 생성)
img	common	공통이미지 (로고, 배경)
	btn	버튼이미지 (버튼배경 : CSS 스타일 지정)
	other	기타이미지 (서브 이미지)
	pop	팝업이미지
css		CSS 문서
js		JavaScript 문서
inc		include 문서
popup		popup 문서
upload	120101(년월일)	워드, 파워포인트 등의 파일 (첨부파일 용도)
		년월일로 첨부파일 업로드 (yyymmdd로 날짜형식 사용)

[표1-2] 폴더 기본 구조



1. 파일 네이밍

- 1) 기본 파일명은 **디렉터리명 + 구분자 + 역할(일련번호)** 순으로 조합하여 구성한다.
- 2) 구분자는 **언더바(_)**로서 디렉터리명과 일련번호 또는 파일설명(역할)을 구분한다. 언더바는 **3단계를 초과하여 조합할 수 없다**.
- 3) 일련번호는 **2자리의 숫자**로서 순서에 따른 일련번호를 제공한다.
예)  sub01_d01_01.html,  sub01_d01_02.html,  sub01_d01_03.html...
- 4) **역할**은 파일의 역할과 관련된 용어를 사용하여 작성하도록 하고 **영문, 숫자포함 6자**를 넘기지 않도록 규정한다. 역할의 용어가 6자를 넘길 경우 약어(or 단축어)로 작성한다.
- 5) 서브 디렉터리 [**중분류**] 안에 추가되는 파일은 디렉터리의 단계 순으로 **대분류 + 중분류 + 일련번호**를 언더바로 구분하여 작성한다.
예)  sub01_d01_01.html → sub01(**대분류**)_d01(**중분류**)_01(**일련번호**).html(**확장자**)
- 6) 서브 디렉터리 [**소분류**] 안에 추가되는 파일은 중분류 디렉터리명을 배제하고 디렉터리의 단계 순으로 **대분류 + 소분류 + 일련번호**를 언더바로 구분하여 작성한다. (중분류 배제 이유 : 중분류의 디렉터리명을 포함할 경우 언더바가 3단계를 초과하기 때문)
예)  sub01_da01_01.html → sub01(**대분류**)_da01(**소분류**)_01(**일련번호**).html(**확장자**)
 sub01_dz10_01.html → sub01(**대분류**)_dz10(**소분류**)_01(**일련번호**).html(**확장자**)

7) 서브 디렉터리 안에 **tab 페이지**가 추가되는 경우 일련번호 앞에 tab의 첫단어 't'를 붙여 **디렉터리명 + 't'+일련번호'** 로 작성한다.

예)  sub01_da01_t01.html → sub01(대분류)_da01(소분류)_t01(t+일련번호).html(확장자)

8) 최상위 디렉터리 [**사이트명**] 안에 들어가는 파일은 **default.html** 기본 메인 페이지 파일만 생성 가능하며, 이 외의 파일 생성은 제한한다.

(단, 예외의 상황으로 추가 파일 생성 시 개발파트와 상의하여 파일명을 정의하고 사용하도록 한다. 예) intro.html(**인트로**))

9) css 문서는 **정의된 네이밍 파일**을 사용하도록 규정하고, 서브 페이지의 별도 스타일은 **id와 class로 지정**한다. (p.14 참고)

10) Javascript 문서의 파일명은 개발 담당자가 프로젝트 특성에 맞춰 작성하고 기획자와 협의 후 정의된 파일명을 문서화한다.

(공통으로 추가 되는 javascript 문서의 파일명은 '**common.js**' 으로 규정한다.)

11) popup 문서의 파일명은 '**pop_역할.html**' 로 작성한다.

12) include 문서의 파일명은 '**inc_역할.html**' 로 작성한다.

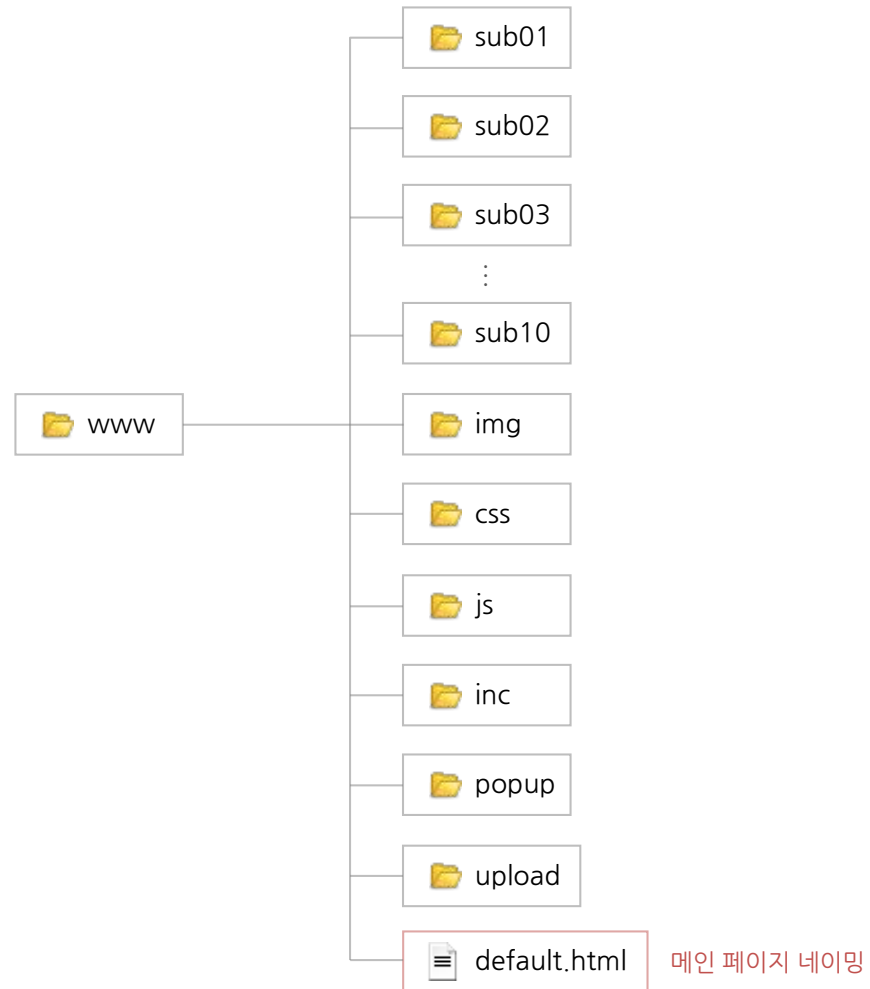
13) [**js**], [**inc**], [**popup**] 디렉터리 안의 파일명은 개발파트 네이밍 정의 문서를 확인 후 작성한다.

14) 파일 네이밍 규칙 [**표 2-1**]과 파일 네이밍 구조 [**표 2-2**] ~ [**표 2-10**]를 참고하여 파일을 생성한다. (P.9~18 참고)

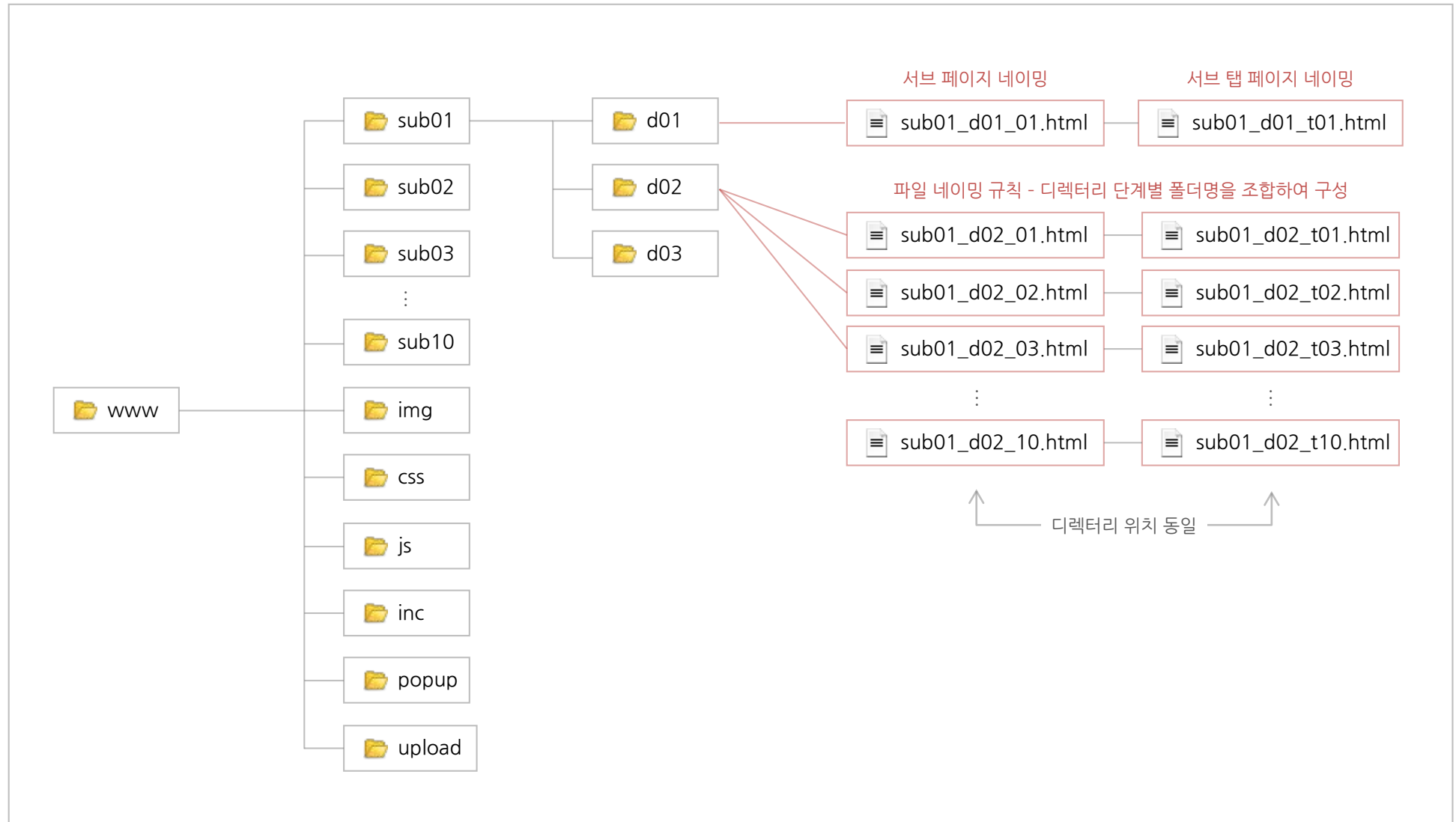
[표2-1] 파일 네이밍 규칙

분류	경로	예제	설명
메인 페이지	www	default.html	메인 페이지
서브 페이지	sub01/d01	sub01_d01_01.html	대분류_중분류_일련번호 순으로 디렉터리와 일련번호로 네이밍 작성
	sub01/d01/da01	sub01_da01_01.html	대분류_소분류_일련번호 순으로 디렉터리와 일련번호로 네이밍 작성
서브 tab 페이지	sub01/d01	sub01_d01_t01.html	대분류_중분류_+일련번호 순으로 디렉터리와 일련번호로 네이밍 작성
	sub01/d01/da01	sub01_da01_t01.html	대분류_소분류_+일련번호 순으로 디렉터리와 일련번호로 네이밍 작성
CSS 문서	css	base.css	공통 스타일 (폰트, 링크, 폼, 기타)
		layout.css	CSS 구조화 스타일 (메인, 서브 레이아웃)
		nav.css	네비게이션 스타일 (상단메뉴, 좌측메뉴)
		sub01.css~sub10.css	서브 페이지의 별도 스타일 지정 (파일생성은 10개 이하로 규정함, 서브페이지 디렉터리명)
		btn.css	버튼 스타일
JavaScript 문서	js	common.js	공통 자바스크립트 (이 외의 자바스크립트 문서 파일명은 개발파트 네이밍 정의 문서 참고)
include 파일	inc	inc_역할.html	include 파일의 역할에 맞춰 inc_역할로 네이밍 작성 (개발파트 네이밍 정의 문서 참고)
popup 파일	popup	pop_역할.html	popup 파일의 역할에 맞춰 pop_역할로 네이밍 작성 (개발파트 네이밍 정의 문서 참고)
iframe 파일	iframe	ifr_역할.html	iframe 파일은 되도록 사용하지 않도록 하고, 특이 사항으로 사용시 관련 파일의 역할에 맞춰 ifr_역할로 네이밍 작성 (특정 상황을 제외하곤 iframe 사용을 제한한다.)
첨부파일 문서	upload	파일명 규제 없음	워드, 파워포인트 등의 문서

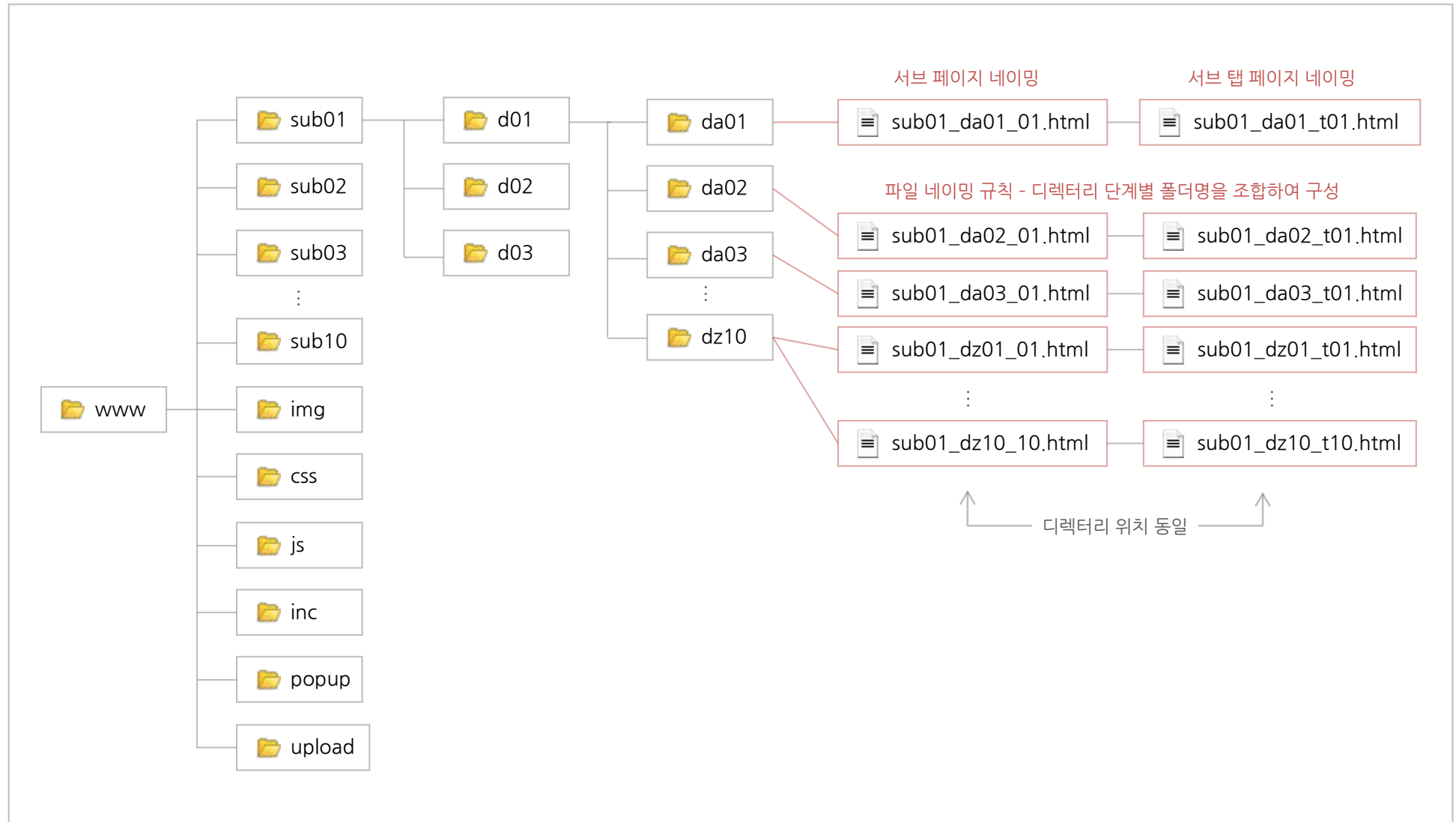
[표2-2] 파일 네이밍 구조 - 메인 페이지



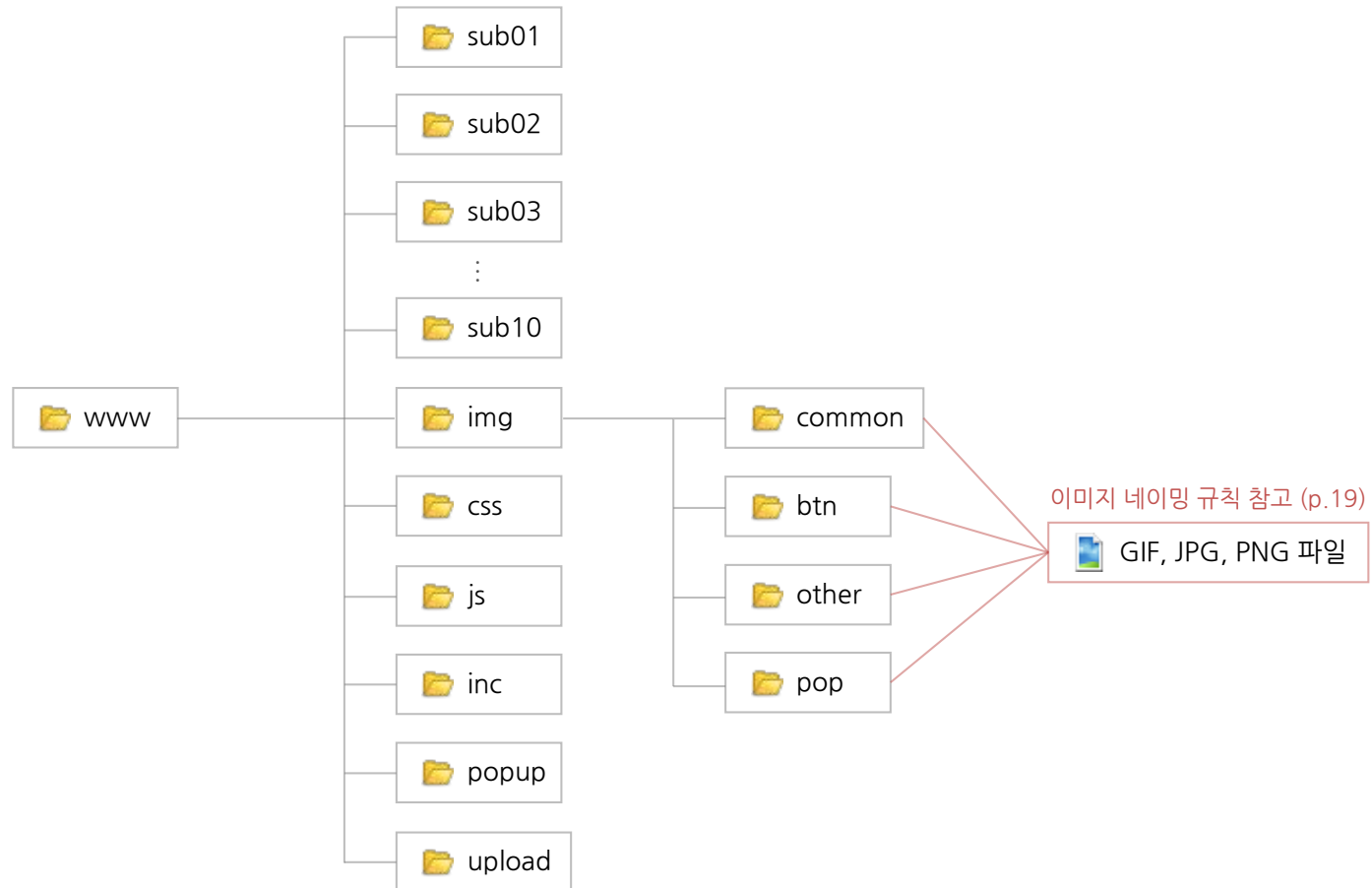
[표2-3] 파일 네이밍 구조 - 서브 페이지 (중분류)



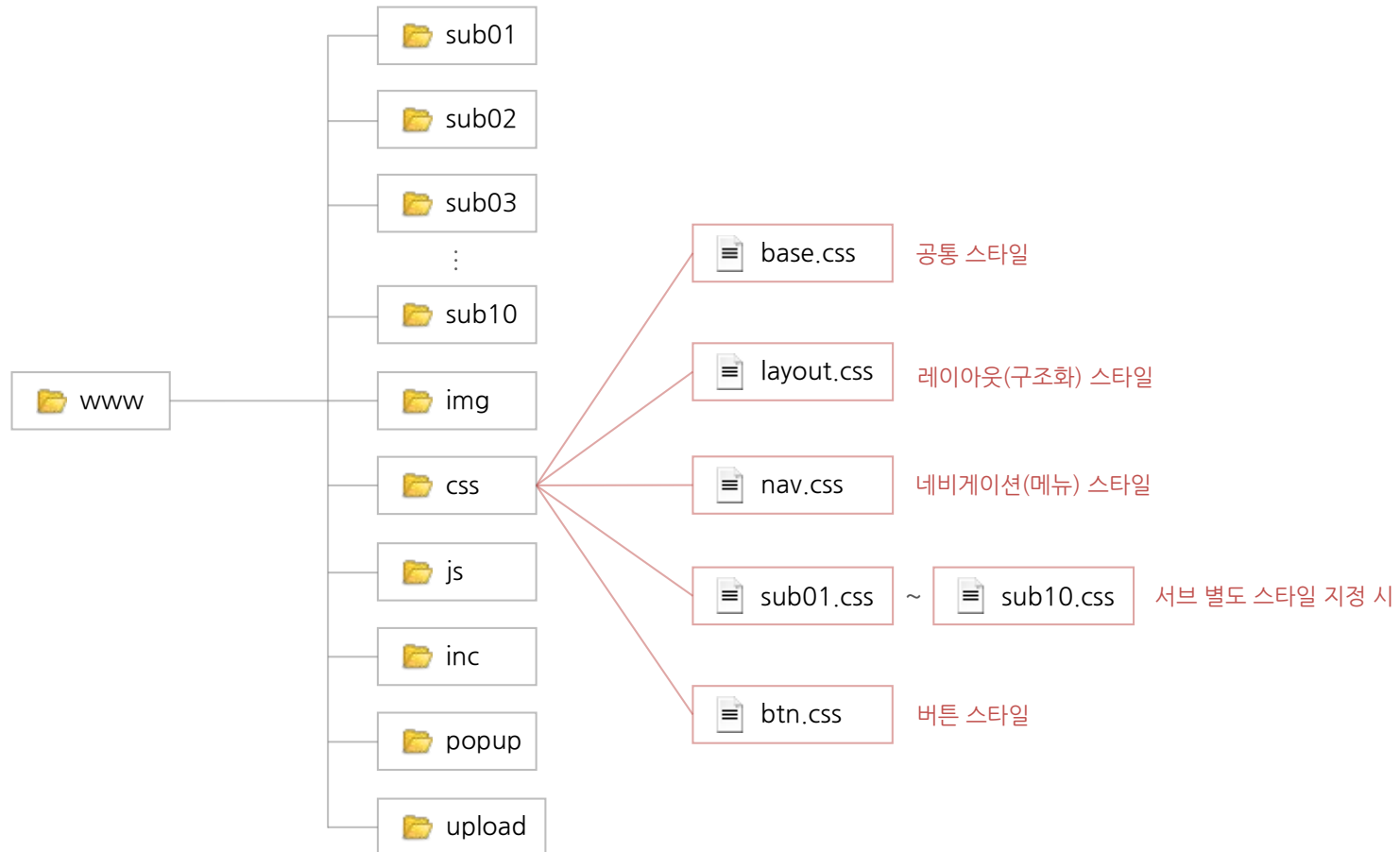
[표2-4] 파일 네이밍 구조 - 서브 페이지 (소분류)



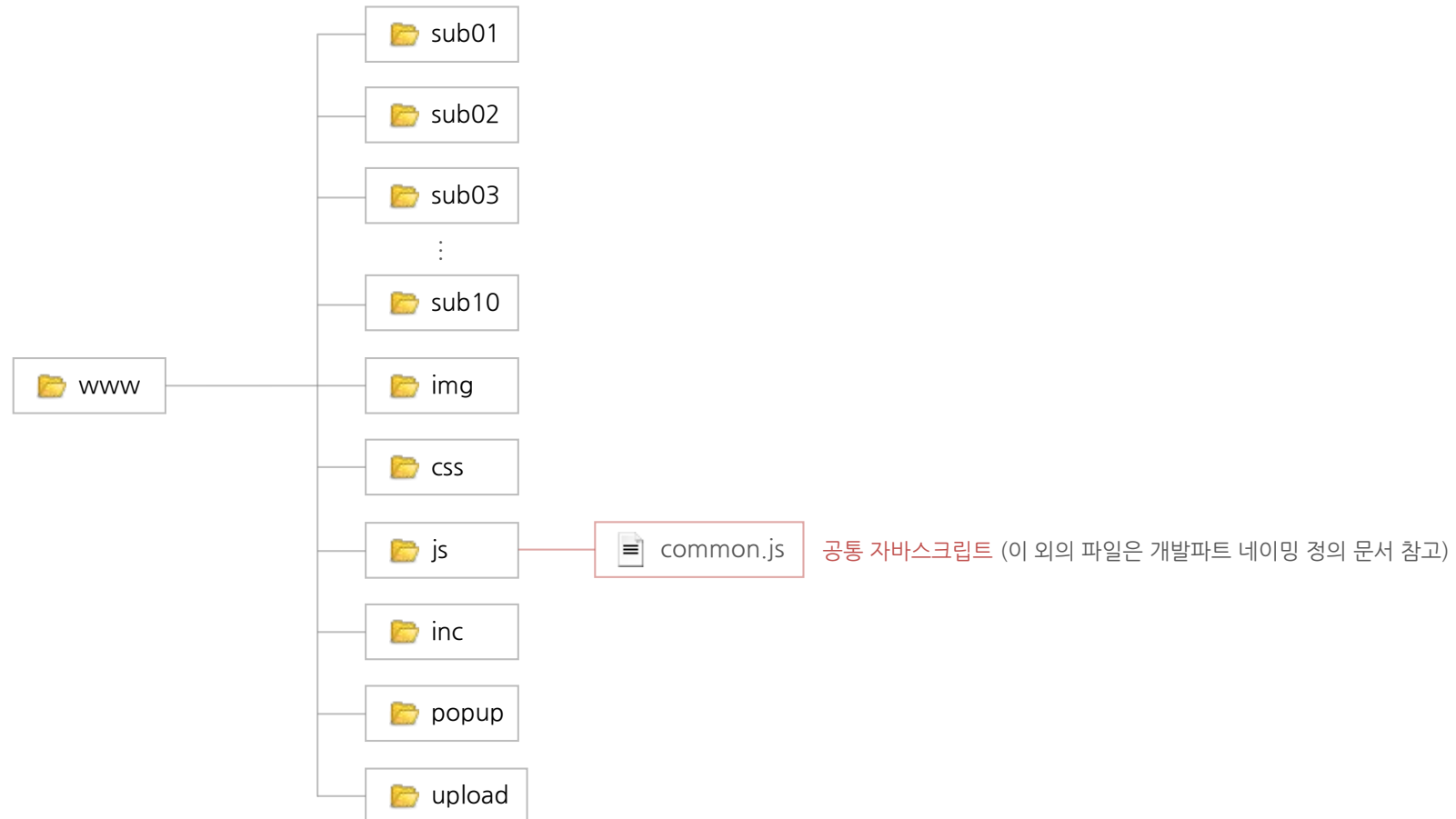
[표2-5] 파일 네이밍 구조 - 이미지 파일



[표2-6] 파일 네이밍 구조 - CSS 문서



[표2-7] 파일 네이밍 구조 - JavaScript 문서



[표2-8] 파일 네이밍 구조 - include 파일



[표2-9] 파일 네이밍 구조 - popup 파일



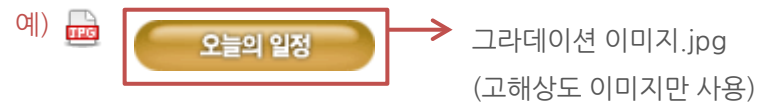
[표2-10] 파일 네이밍 구조 - 첨부파일 문서



1. 이미지 네이밍

1) 이미지 파일 기본 확장자는 **gif** 를 사용한다.

2) 두 개 이상의 이미지가 겹치는 경우 png (투명이미지)를 사용하고, 고해상도의 이미지를 적용할 경우 jpg를 사용한다.



3) 이미지 파일명은 **형태 + 위치 + 파일설명 + 상태(or 단계)**를 구분자로 구분하여 이미지 형태에 맞춰 순서대로 조합하고, 이미지 형태에 따라 위치나 상태를 배제하고 형태와 파일설명 만으로 이미지 파일명을 구성할 수 있다.

예)  img_sub01_d01_img01.gif → img(형태)_sub01(대분류 위치)_d01(중분류 위치)_img01(파일설명+단계).gif(확장자)





 nav_left_menu01_on.gif → nav(형태)_left(위치)_menu01(파일설명+단계)_img01(상태).gif(확장자)

 btn_abb_01.gif → btn(형태)_abb(파일설명)_01(단계).gif(확장자)

 ico_letter.gif → ico(형태)_letter(파일설명).gif(확장자)

→ 위와 같이 **4가지 형태의 조합으로 이미지 파일명**을 작성하도록 한다.

4) **파일설명**은 이미지와 관련된 의미를 가진 용어를 사용하여 작성하도록 하고 **영문, 숫자포함 6자**를 넘기지 않도록 규정한다. 파일설명의 용어가 6자를 넘길 경우 **약어(or 단축어)**로 작성한다.

- 5) 구분자는 **언더바(_)**로서 네이밍을 조합할 때 이미지 형태에 맞춰 구분하고, 언더바는 **4단계를 초과하여 조합할 수 없다**.
- 6) 개별 이미지 저장 시 파일 네이밍 순서는 **위쪽 좌측에서 우측 아래쪽으로 향하는 단계**로 일련번호를 작성한다. (표 3-2 참고)
- 7) 배경 이미지 저장 시 파일 네이밍 순서는 **위쪽 좌측부터 시계방향으로 향하는 단계**로 일련번호를 작성한다. (표 3-5 참고)
- 8) 이미지 파일명 구성할 때 네이밍이 길어져 구분자가 3단계 초과 시 **파일설명+단계**를 붙여서 파일명을 구성한다.
 예)  nav_left_menu01_on.gif → nav_left_menu01(파일설명+단계)_on.gif
- 9) 이미지 파일명은 이미지 **확장자에 관계 없이 순차적**으로 적용한다.
 예)  btn_abb_01.jpg,  btn_abb_01.gif,  btn_abb_01.jpg
- 10) 이미지 네이밍 규칙 [표 3-1]과 이미지 네이밍 적용 예 [표 3-2~3-11]을 참고하여 이미지 파일명을 작성한다. (P.21~33 참고)

[표3-1] 이미지 네이밍 규칙

예약어 (형태)	범위
tit_	제목
nav_	네비게이션 (메뉴)
tab_	탭
bg_	배경, 박스
_left, _right, _top, _bot	이미지 위치 (왼쪽, 오른쪽, 위, 아래)
btn_	버튼
_in, _out, _go, _more	버튼 상태 (테이블 안, 테이블 밖, 바로가기(이동), 더보기)
_on, _off	상태 변화 (rollover)
bl_	블릿
_h1~h10	블릿 제목 사이즈 (단계)
img_	이미지 (메인, 서브 기타 이미지)
line_	선
_000000, _dot	선 스타일 (색상코드 6자리 (코드 중복 시3 자리사용), 도트)
ico_	아이콘
pop_	팝업 이미지

[표3-2] 이미지 네이밍 적용 예 - 제목

예약어 (형태)	범위
tit_	제목

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

메인페이지



1. **오늘의 할일** (2)

↓	순서	업무명	기간	상태
A	1	Web Style Guide 작성	~2012-06-01	진행
A	2	i-Smart 디자인 개선	~2012-05-29	진행

2. **Today 스케줄** (0)

등록된 일정이 없습니다.

3. **요청처리**

▲ 요청 (0) ▲ 미처리 (0)

▲ 요청 (0)

등록된 요청처리가 없습니다.

▲ 미처리 (0)

- 1 tit_mainbox_01.gif
(형태_파일설명_단계)
- 2 tit_mainbox_02.gif
- 3 tit_mainbox_03.gif

※ 이미지 파일 단계
: 일련번호 작성 순서 (Z방향)

[표3-3] 이미지 네이밍 적용 예 - 네비게이션

예약어 (형태)	범위
nav_	네비게이션 (메뉴)
_left, _right, _top, _bot	이미지 위치 (왼쪽, 오른쪽, 위, 아래)
_on, _off	상태 변화 (rollover)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)



1 nav_top_menu01_on.gif
(형태_위치_파일설명+단계_상태)

2 nav_top_menu02_off.gif

3 nav_top_menu03_off.gif

4 nav_top_menu04_off.gif

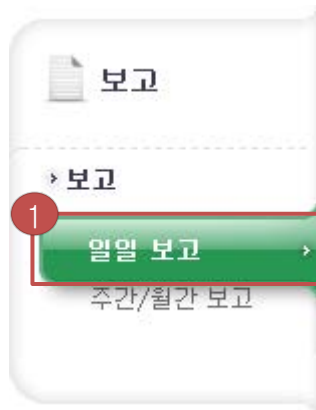
5 nav_top_menu05_off.gif

[표3-3] 이미지 네이밍 적용 예 - 네비게이션

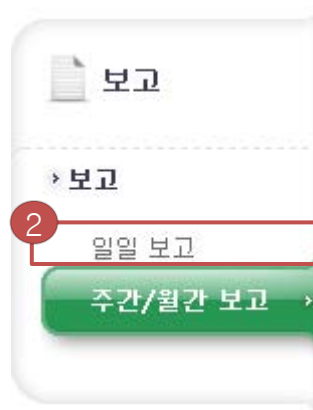
예약어 (형태)	범위
nav_	네비게이션
_left, _right, _top, _bot	이미지 위치 (왼쪽, 오른쪽, 위, 아래)
_on, _off	상태 변화 (rollover)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

좌측메뉴 - 롤오버 (상태)



마우스 클릭 후



마우스 클릭 전

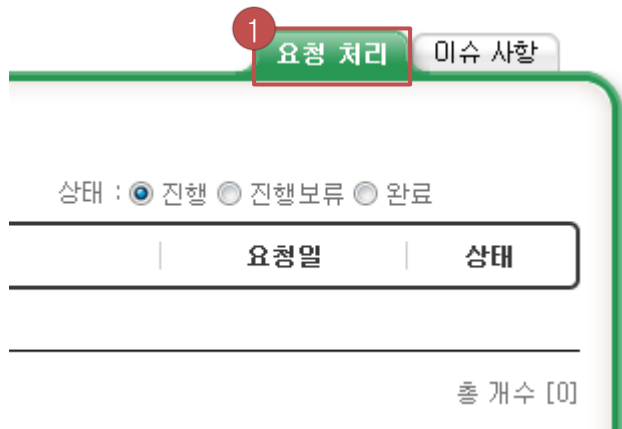
- 1 nav_left_menu01_on.gif
(형태_위치_파일설명+단계_상태)
- 2 nav_left_menu01_off.gif

[표3-4] 이미지 네이밍 적용 예 - 탭

예약어 (형태)	범위
tab_	탭
_on, _off	상태 변화 (rollover)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

sub03 페이지 탭 메뉴



1 tab_sub03_request_on.gif
(형태_위치_파일설명_상태)

2 tab_sub03_request_off.gif

[표3-5] 이미지 네이밍 적용 예 - 배경, 박스

예약어 (형태)	범위
bg_	배경, 박스
_left, _right, _top, _bot	이미지 위치 (왼쪽, 오른쪽, 위, 아래)

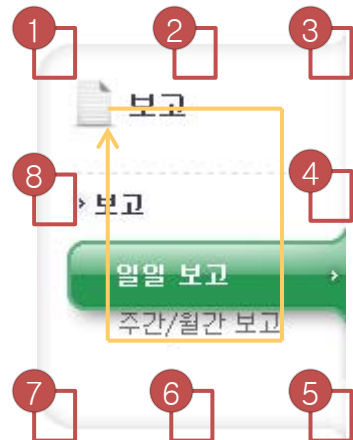
파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

상단메뉴



1 bg_top_nav.gif
(형태_위치_파일설명)

좌측메뉴



- | | |
|-------------------------------------|---------------------|
| 1 bg_menubox_01.gif
(형태_파일설명_단계) | 5 bg_menubox_05.gif |
| 2 bg_menubox_02.gif | 6 bg_menubox_06.gif |
| 3 bg_menubox_03.gif | 7 bg_menubox_07.gif |
| 4 bg_menubox_04.gif | 8 bg_menubox_08.gif |

※ 이미지 파일 단계
: 일련번호 작성 순서 (시계방향)

[표3-6] 이미지 네이밍 적용 예 - 버튼

예약어 (형태)	범위
btn_	버튼
_in, _out, _go, _more	버튼 상태 (테이블 안, 테이블 밖, 바로가기(이동), 더보기)
_on, _off	상태 변화 (rollover)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

이미지 폰트체 버튼 => 이미지화 된 폰트체



마우스 클릭 전



마우스 클릭 후

1 btn_in_login_off.gif
(형태_위치_파일설명_상태)

2 btn_in_login_on.gif

[표3-6] 이미지 네이밍 적용 예 - 버튼

예약어 (형태)	범위
btn_	버튼
_in, _out, _go, _more	버튼 상태 (테이블 안, 테이블 밖, 바로가기(이동), 더보기)
_on, _off	상태 변화 (rollover)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

시스템 폰트체 버튼

=> 문서에서 텍스트를 불러 온 폰트체

2012년 06월 01일(금) 오전 10:52:50



2012년 06월 01일(금) 오전 11:19:23



1 btn_in_loginbg_off.gif
(형태_위치_파일설명_상태)

2 btn_in_loginbg_on.gif

특정 버튼 배경 사용 시

또는

1 btn_in_bg01_off.gif
(형태_위치_파일설명+단계_상태)

2 btn_in_bg01_on.gif

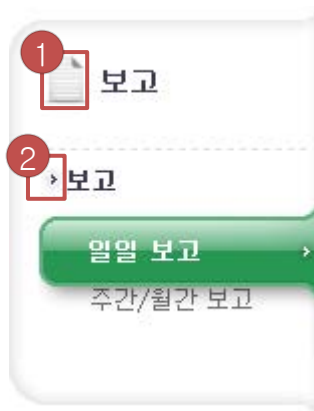
공통 버튼 배경 사용 시

[표3-7] 이미지 네이밍 적용 예 - 불릿

예약어 (형태)	범위
bl_	불릿
_h1~h10	불릿 제목 사이즈 (단계)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

좌측메뉴 제목



1 bl_report_h1.gif
(형태_파일설명_단계)

2 bl_report_h2.gif

서브 페이지 제목



1 bl_tit_h1.gif
(형태_파일설명_단계)

2 bl_list_h2.gif

2 bl_list_h3.gif

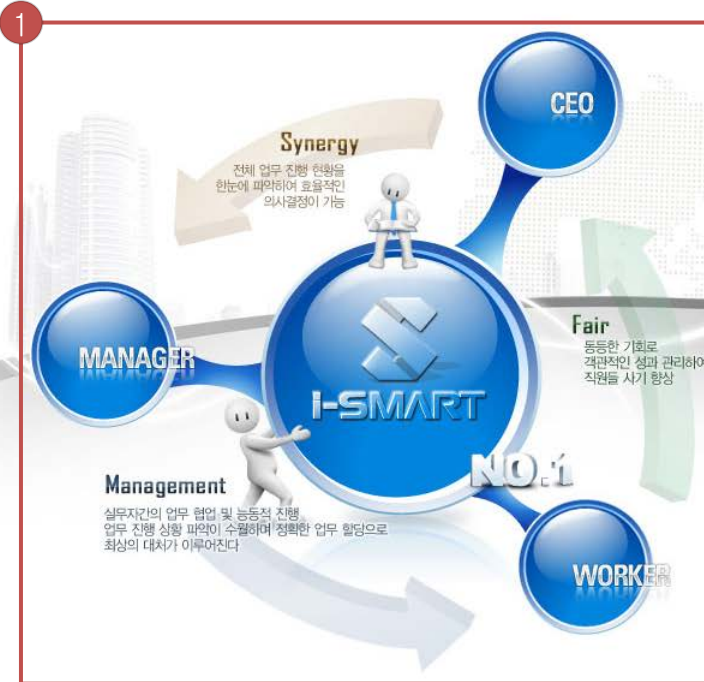
[표3-8] 이미지 네이밍 적용 예 - 기타 이미지

예약어 (형태)	범위
img_	이미지 (메인, 서브 기타 이미지)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

메인페이지 (차트)

서비스 사용법 서비스 소개 서비스 신청



2

WPM
Workflow Performance Management

i - Smart Login

Site ID: ubasic

ID: inyoung

Password: ●●●●●●

Login

☒ ID 저장하기 ☒ PW 저장하기

1 img_visual_01.gif
(형태_파일설명_단계)

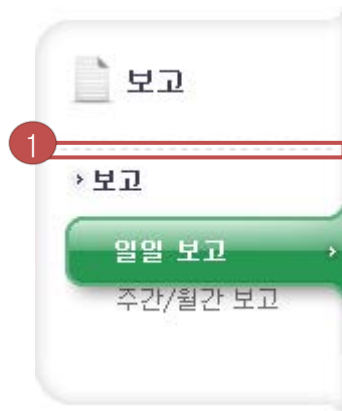
2 img_visual_02.gif

[표3-9] 이미지 네이밍 적용 예 - 선

예약어 (형태)	범위
line_	선
_000000, _dot	선 스타일 (색상코드 6자리 (코드 중복 시 2자리사용), 도트)

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

좌측메뉴 - 롤오버 (상태)



- 1 line_dot_e0.gif
(형태_형태_파일설명(색상코드 e0e0e0))

상단메뉴



- 2 line_fdd65a_4px.gif
(형태_파일설명(색상코드)_단계(height 4px))

메인 상단 탭 메뉴



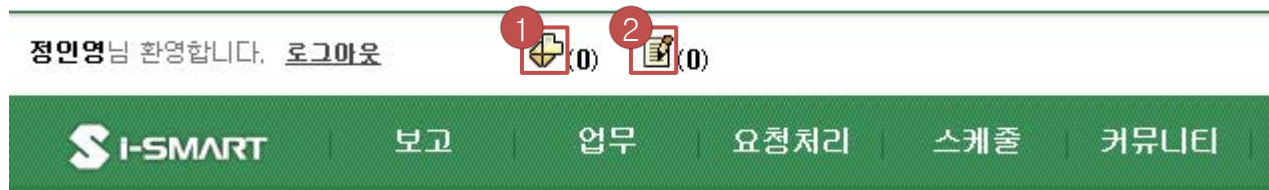
- 3 line_2c_3px.gif
(형태_형태_파일설명(색상코드 2c2c2c))

[표3-10] 이미지 네이밍 적용 예 - 아이콘

예약어 (형태)	범위
ico_	아이콘

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

메인페이지 (상단)



1 ico_letter.gif
(형태_파일설명)

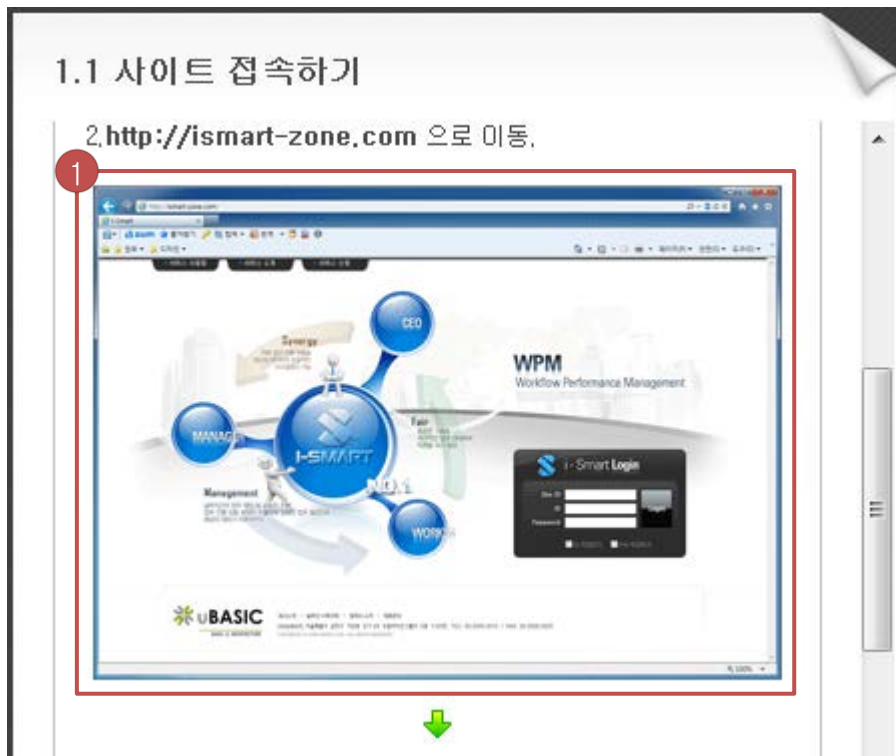
2 ico_meno.gif

[표3-11] 이미지 네이밍 적용 예

예약어 (형태)	범위
pop_	팝업 이미지

파일명 규칙 - 형태_위치_파일설명_상태(or 단계)

메인페이지 (상단)



- 1 pop_help_01.gif
(형태_파일설명_단계)

1. 주석

- 시작은 **[START]**로 작성하고, 끝은 **[END]**로 작성하여 마무리한다. 영역의 명칭은 '**[START]**▼**영역명**' 과 같이 한 칸의 공백을 두고 작성한다.

예) <!-- [START] DTD 정의 --> : 영역이 시작되는 행 위로 삽입되는 주석

<!-- [END] DTD 정의 --> : 영역이 끝나는 행 아래로 삽입되는 주석

2. 공백

- 코딩 시 모든 글자(tag)와 글자(tag) 사이의 공백은 한 칸 만을 허용한다. (키보드 spacebar로 한 칸)

예) 주석 : <!--▼[START]▼DTD 정의▼-->, 태그 : <span▼class="abc">내용

3. 들여쓰기

- 들여쓰기를 하면 코드의 가독성이 높아지고 전체 HTML 구조를 쉽게 파악할 수 있다.
- 마크업의 중첩이 깊어질 때마다 자식 엘리먼트는 **1탭 들여 쓰고**, 1탭의 크기는 **공백 4칸으로 설정**한다.

예-1) div 구조

```
<div id="wrap">
▼▼▼▼<div id="header"></div>
▼▼▼▼<div id="content">내용</div>
▼▼▼▼<div id="footer"></div>
</div>
```

예-2) table 구조

```
<table>
▼▼▼▼<tr>
▼▼▼▼▼▼▼▼<td>내용</td>
▼▼▼▼</tr>
</table>
```

예-3) 불규칙한 목록 사용

```
<ul>
<li>• 내용</li>
<li>• 내용</li>
</ul>
```

예-4) 규칙적인 목록 사용

```
<ol>
<li>1. 내용</li>
<li>2. 내용</li>
</ol>
```

예-5) 정의할 목록의 용어와 서술에 사용

```
<dl>
<dt>이름 :</dt>
<dd>홍길동</dd>
</ul>
```

- 위와 같은 목록 태그 [예-3], [예-4], [예-5] 는 들여쓰기를 하지 않는다.

4. 영문 소문자 사용

- DTD를 제외한 모든 엘리먼트와 애트리뷰트는 **소문자로 작성**한다.

예) `내용` → 엘리먼트 : `span`, 애트리뷰트 : `abc`

5. 애트리뷰트 값 표시

- 애트리뷰트 값은 큰따옴표 (" ")로 묶는다.

예) `내용`

6. 빈 줄

- 빈 줄은 코드 그룹간 1줄씩 빈 줄을 만드는 것을 허용하고, 1줄을 초과하지 않는다.

예) `<div>내용</div>`

`<div>내용</div>` → 빈 줄 (enter)

1. DOCTYPE

- 정확한 표준모드로 사용하기 위해서 W3C에서 사용할 것을 권장한 DOCTYPE 으로 선언한다.

```
<!-- [START] DTD 선언 -->  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">  
<!-- [END] DTD 선언 -->
```

- HTML5** DOCTYPE 선언 방법 - html5에서는 공식 DTD(Document Type Definition)가 없다.

```
<!-- [START] DTD 선언 -->  
<!DOCTYPE html>  
<!-- [END] DTD 선언 -->
```

2. META

- 웹 문서의 인코딩은 다국어 지원이 가능하도록 UTF-8로 지정한다.

```
<!-- [START] 문자 인코딩 지정 -->  
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">  
<!-- [END] 문자 인코딩 지정 -->
```

- **HTML5** 문자 인코딩 지정 방법 - UTF-8로 지정한다.

```
<!-- [START] 문자 인코딩 지정 -->  
<meta charset="utf-8">  
<!-- [END] 문자 인코딩 지정 -->
```

3. TITLE

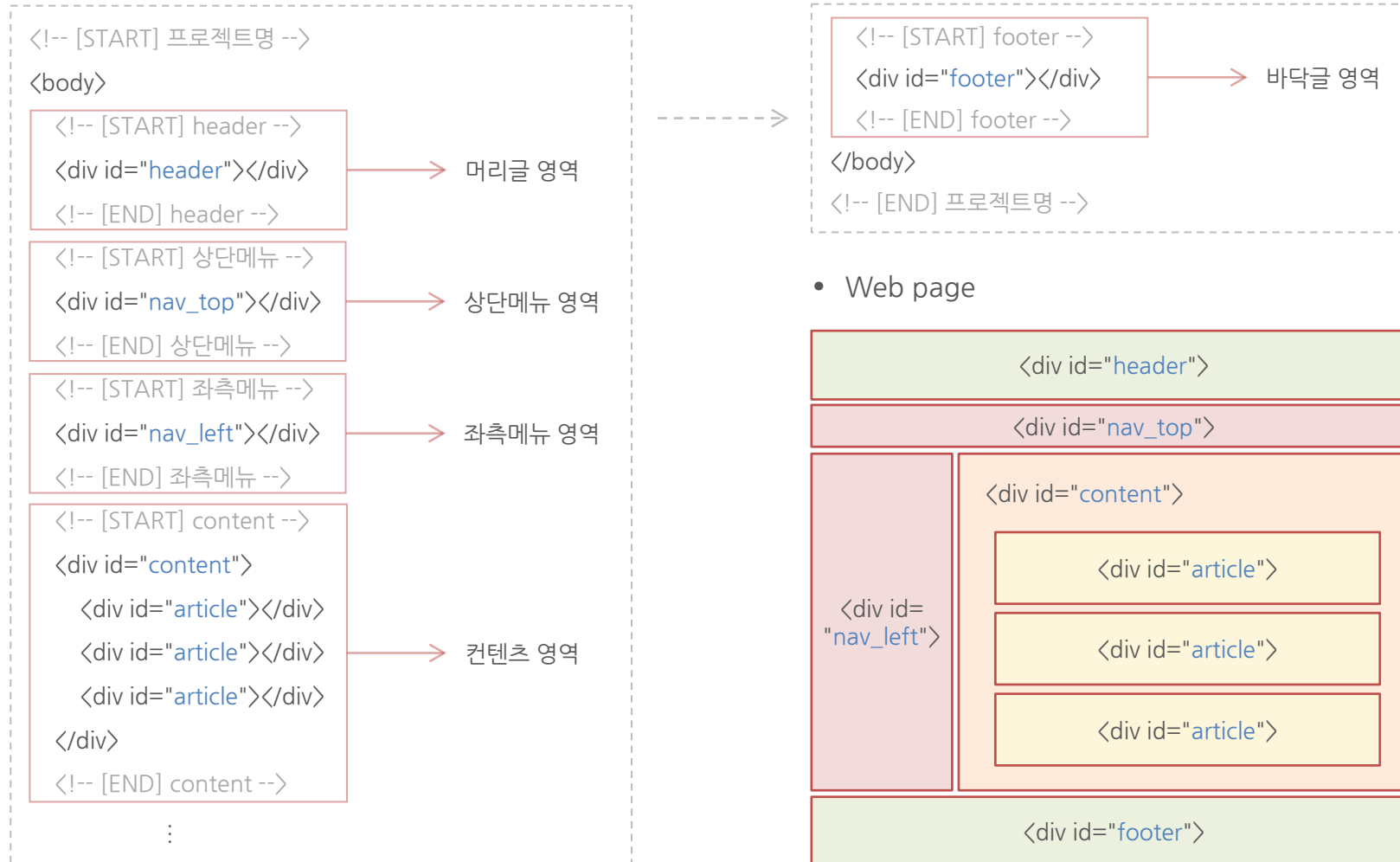
- 웹 페이지의 사이트명(프로젝트명)을 지정한다.

```
<!-- [START] title -->  
<title>사이트명(프로젝트명)</title>  
<!-- [END] title -->
```

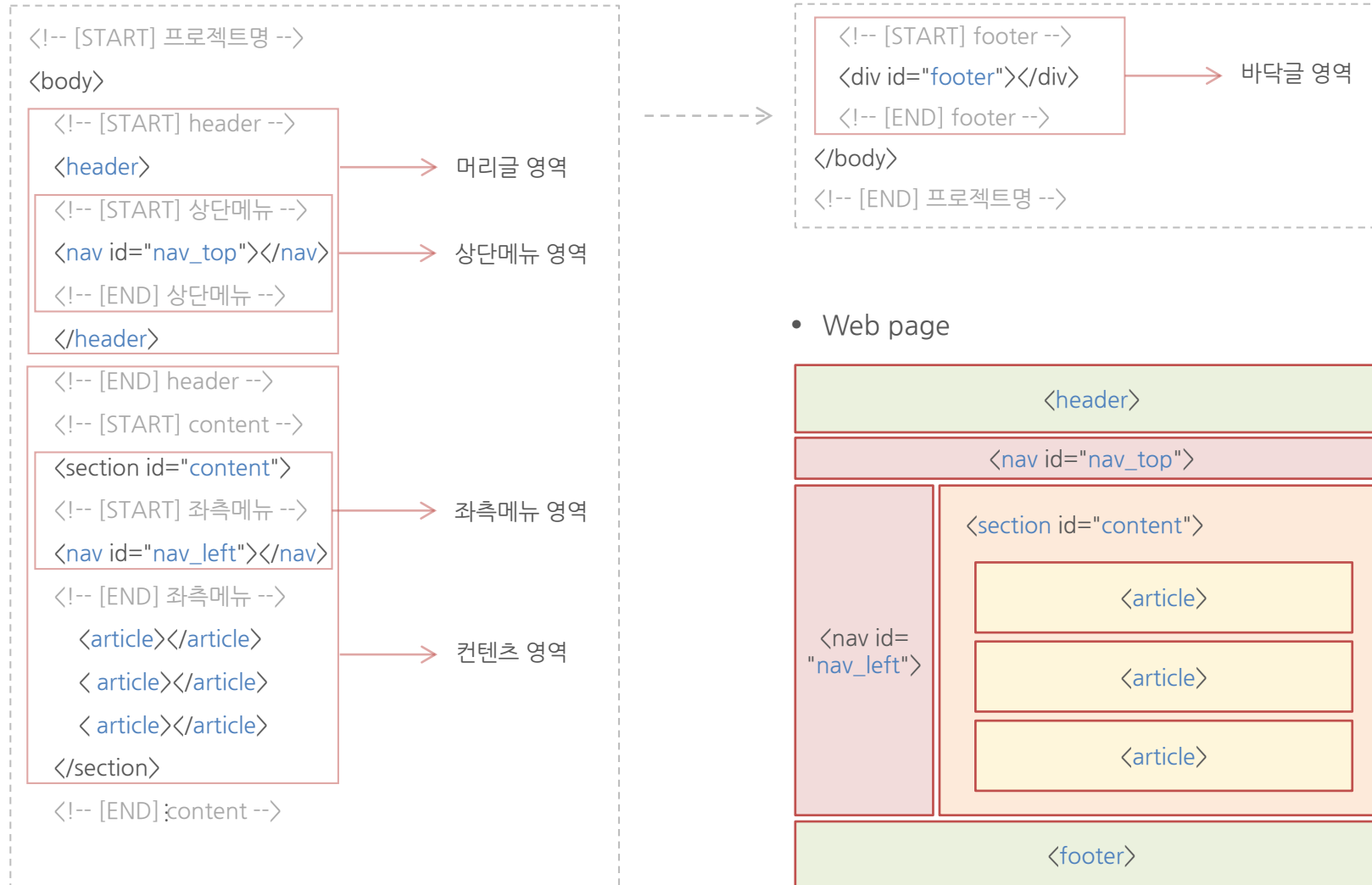
4. HTML 구조

- 1) 기본 html 구조를 참고하여 파일을 제작한다.
- 2) 브라우저의 호환이 되면 html5로 쉽게 변경 할 수 있도록 div의 id값을 html5의 태그로 의미를 두었다.
- 3) 기본 HTML 구조[표 5-1]을 참고하여 제작하고, 향 후 브라우저가 호환이 되면 [표 5-2]를 참고하여 HTML5로 변경한다.

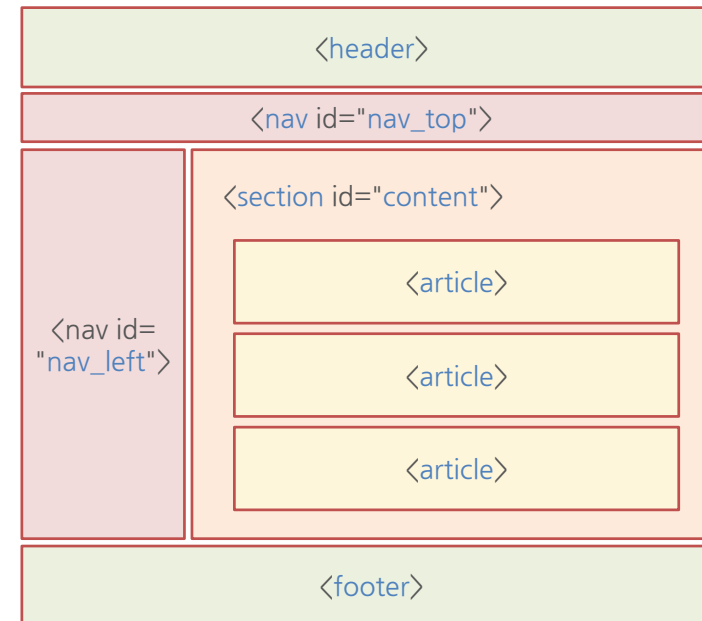
[표5-1] html 기본 문서 예



[표5-2] html5 기본 문서 예



• Web page



1. CSS 기본 문서

문서	설명
base.css	공통 스타일 (폰트, 링크, 폼, 기타)
layout.css	CSS 구조화 스타일 (메인, 서브 레이아웃)
nav.css	네비게이션 스타일 (상단메뉴, 좌측메뉴)
sub01.css~sub10.css	서브 페이지의 별도 스타일 지정 (파일생성은 10개 이하로 규정함, 서브페이지 디렉터리명)
btn.css	버튼 스타일

2. id 및 class 네이밍

- 1) id는 문서 전체의 고유 식별자 이므로 한 문서에서 동일한 id를 여러 번 사용할 수 없다.
- 2) id 지정은 레이아웃과 콘텐츠 영역 설정만 하고, 이 외의 스타일은 class로 지정한다.
- 3) 레이아웃과 콘텐츠 영역 안에 디자인 스타일을 추가하려면 class를 사용한다.
- 4) 팝업 스타일 지정 시 예약어 앞에 “pop_”를 조합하여 class로 지정한다.
- 5) 문서의 스타일에 맞게 class는 문서 안에서 여러 번 사용할 수 있다. (중복되지 스타일이 없도록 class 지정)
- 6) 레이아웃 이외 의 id와 class 네이밍 지정은 프로젝트(사이트)에 따라 스타일가이드를 제작한다.

3. 레이아웃 예약어

1)

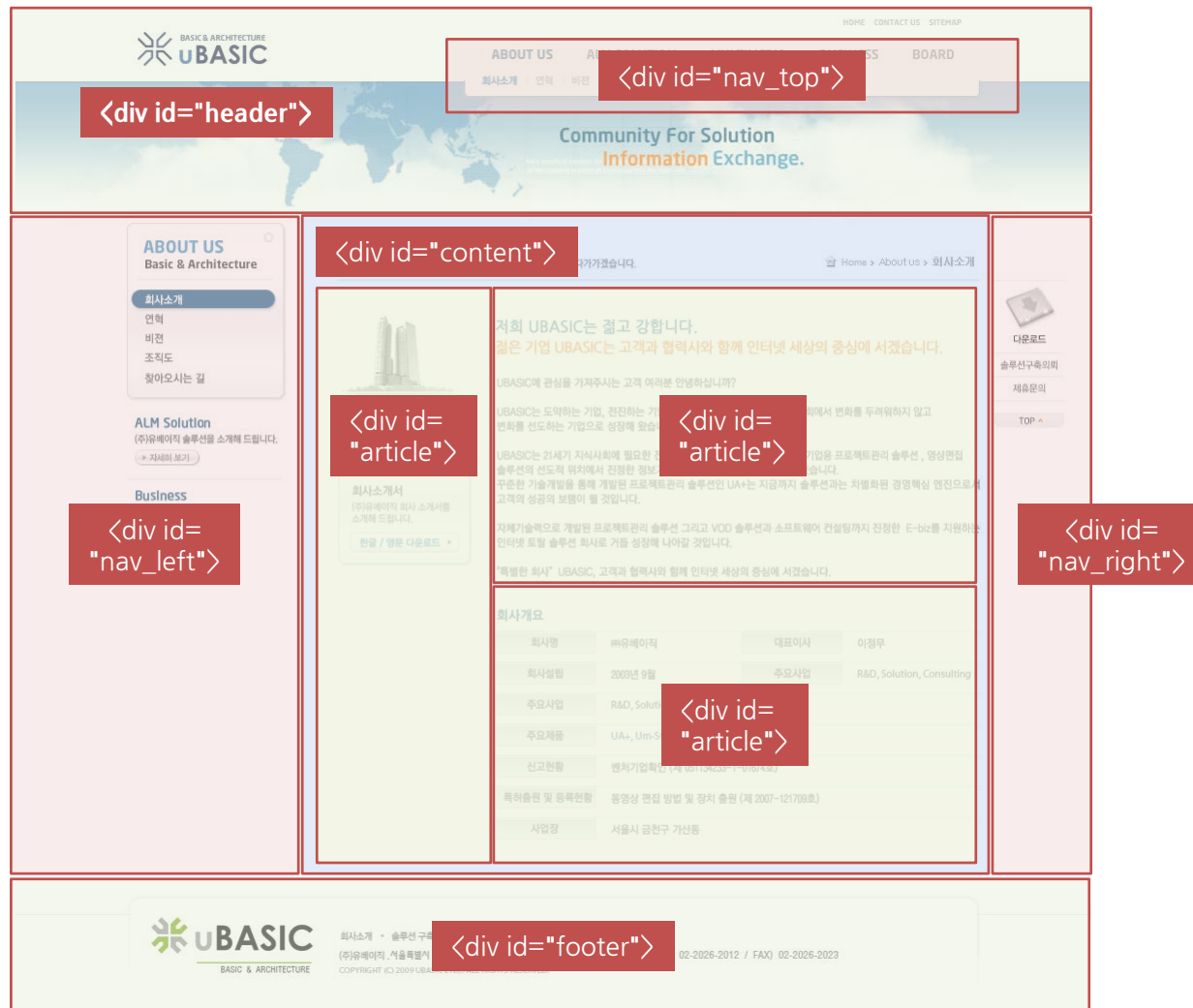
1) CSS 구조 스타일 지정 시 레이아웃 예약어 [표 6-1]을 참고하여 CSS 문서를 제작한다.

2) CSS 구조 분석 예 [표 6-2]을 참고하여 CSS 문서 제작 시 문서의 구조를 파악하고 제작을 한다.

[표6-1] 예약어

문서		설명
header		머리글 영역
nav	_top	상단메뉴
	_left	좌측메뉴
	_right	우측메뉴
content		컨텐츠 영역
article	_일련번호	컨텐츠 상세 영역
footer		바닥글 영역

[표6-2] CSS 구조 분석 (uBASIC 사이트)



Style Guide & Coding Convention

2012. 6. 5

작성자 정인영