**웹 제작 가이드 v.2019.04.02**

**HTML 마크업**

1. Doctype 선언

<!DOCTYPE html>

<!--[if IE 8]><html lang="ko-KR" class="no-js ie8 lt-ie10"><![endif]-->

<!--[if IE 9]><html lang="ko-KR" class="no-js ie9 lt-ie10"><![endif]-->

<!--[if !IE]><!-->

<html lang="ko-KR">

<!--<![endif]-->

2. 반응형 웹

<meta name="viewport" content="width=device-width, initial-scale=1">

3. 페이지 설명

<meta charset=”utf-8”>

<meta http-equiv="X-UA-Compatible" content="IE=Edge">

<meta name="description" content="description">

<meta name="keywords" content="keywords">

<title>사이트 이름</title>

4. 소셜 미디어

<meta property="og:type" content="website">

<meta property="og:title" content="">

<meta property="og:description" content="">

<meta property="og:image" content="">

<meta property="og:url" content="">

<meta name="twitter:card" content="summary">

<meta name="twitter:title" content="">

<meta name="twitter:image" content="">

<meta name="twitter:description" content="">

<meta name="twitter:domain" content="">

5. h1, h2, h3… 태그 사용

- section, article, nav, aside… 등 안에 outline 기준에 맞혀 사용 지향

**CSS 및 JS 작성**

CSS 스타일 시트

1. CSS3 사용

- Ie8 호환정도는 화면 UI 가 깨지지 않는 선에서 사용 ex) border-radius)

- ‘:’ 이후 공백 사용 ex) border: 4px

- 크로스브라우징을 위한 밴더프리픽스 사용

2. ID 및 Class 사용

- 스타일은 Class 사용 지향

# JS (jQuery 및 Javascript)

# 1. 변수명

- 필요시 언더스코어 “\_” 사용

- var의 중복 사용 피함

**내용과 표현, 동작의 분리**

1. HTML 과 Javascript분리

- 가능한 한 html렌더링과 자바스크립트를 분리

- javascript 파일은 동작만 담당, 마크업 파일이 렌더링 담당

2. CSS와 Javascript분리

- 가능한 한 직접 자바스크립트에서 설정하지 않고 느슨하게 연결하는 방법을 사용

$(“element”).addClass(“active”);

**네이밍 규칙**

1. 공통 규칙

- 네이밍의 조합은 ‘(소속,페이지명,서비스명)-형태-의미-순서-상태’을 기본 순서로 사용한다.

ex) btn\_calcle\_01\_off.gif, msgbox-toggle-off

- 한자리 정수는 사용하지 않으며 01, 02와 같이 사용을 권장

ex) bn\_center\_01.jpg, bn\_center\_02.jpg

- 동일한 이름의 네이밍의 경우 가장 뒤에 숫자를 사용하여 분류지어 사용할 수 있으며 첫번째 파일은 숫자를 생략하여 사용할 수 있다. ex) box-type, box-type02, box-type03

2. Id 네이밍

- 시작 이름은 영문 소문자를 사용하되 두번째 단어부터 첫 번째 문자 대문자로 처리하는 카멜케이스기법 시용 ex) #gnbNav, #lnbNav, #snbNav

3. Class 네이밍

- 시작 이름은 영문 소문자를 사용하며 두 번째 단어부터 하이픈(-)으로 연결하며 대문자는 사용하지 않는다. 3단계를 넘어가지 않도록 권장. 케밥케이스기법 사용 ex) nav-\*, .aside-\*, .section-\*

4. 이미지 네이밍

- 형태\_의미\_순서\_상태 순서로 조합한다. ex) tab\_info\_01\_on.gif

- 같은 이름의 이미지가 두 개 이상 존재하면 파일 언더스코어(\_)로 구분한다.

- 확장자에 관계없이 순차적으로 적용한다. ex) btn\_ok.gif, btn\_ok\_02.jpg, btn\_ok\_03.png

- 이미지 예약어 ex) 텍스트: tit\_\*, txt\_\*, 링크요소: nav\_\*, tab\_\*, btn\_\*, 박스: box\_\*, 아이콘: icon\_\*, 상태: \*\_on, \*\_off, \*\_over

- 이미지 수정 네이밍

ex) btn\_apply (기존), btn\_apply\_110922 (수정), btn\_apply\_110922\_v2 (재수정)

5. 네이밍 확장

- 종속 확장 class

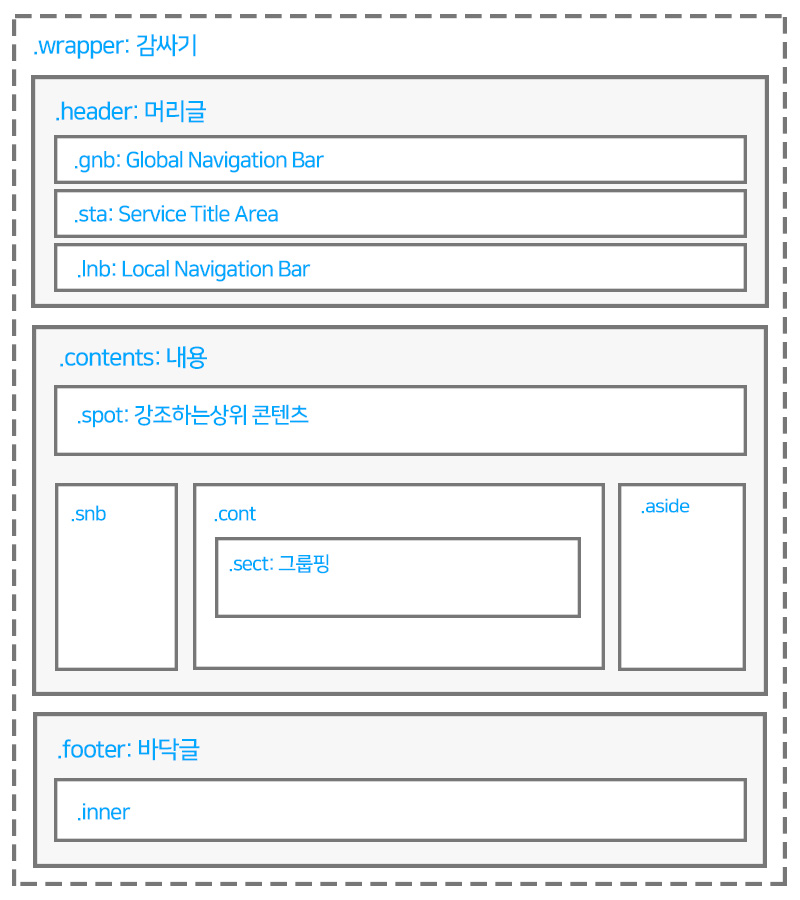
- 기본형 class에 종속되어 여백, 색깔, 행간 등의 몇 가지 속성을 부여하고자 할 대 사용

<button type=”button” class=”btn-default”><button>

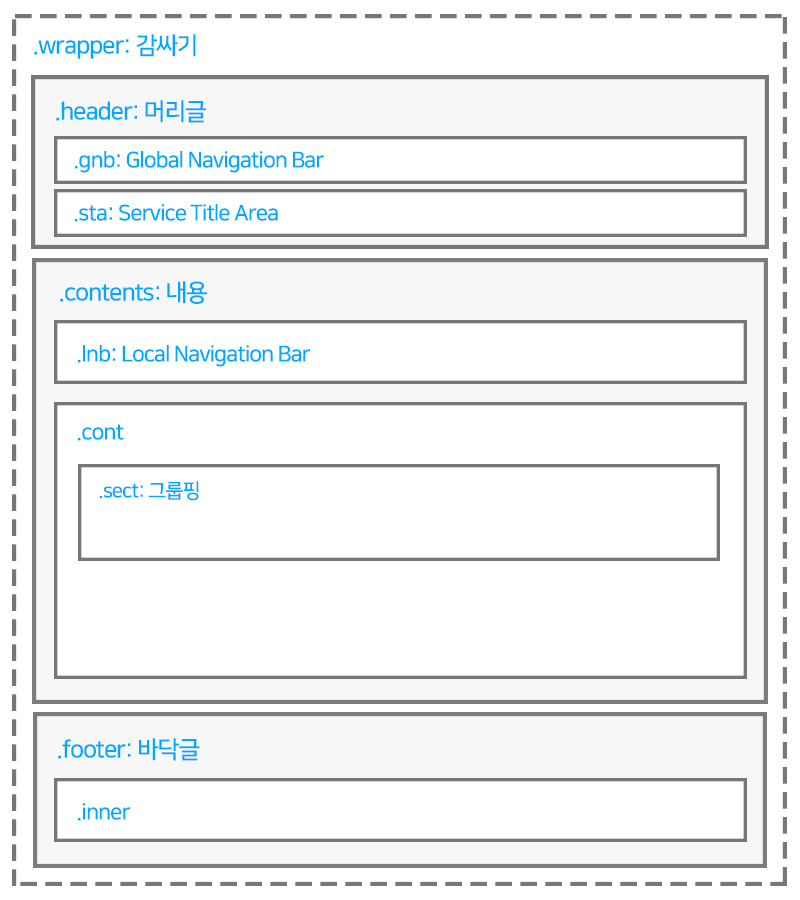
<button type=”button” class=”btn-default btn-blue”><button>

<button type=”button” class=”btn-default btn-brown”><button>

<PC HTML 기본 템플릿 구조>



<Mobile HTML 기본 템플릿 구조



<레이아웃 약속어 범위>

- 팝업 레이아웃 경우 앞에 ‘pop-’ 추가

- 약속어 범위 안에서 id사용 가능하나, 스타일은 클래스를 주어 적용

- .wrap: 페이지 전체 영역

- .header: 머리글 영역

- .contents: 내용 역역

- .cont: 주요 콘텐츠 영역

- .footer: 바닥글 영역

**prefix/subfix/suffix 정의**

1. 분류별 prefix (특명명과 함께 사용시 name-prefix로도 사용)

- 접두사를 의미하는 것으로, 주로 앞부분에 사용하며, 주로 형태를 나타내는 데 사용된다.

- 타이틀: tit

- 영역: wrapper, wrap, container, section, sect, area, inner, row

- 내비게이션: gnb, lng, snb

- 탭: tab

- 테이블: tbl

-목록: list

- 폼: tf, inp, opt, lab,

- 버튼: btn

- 박스: box

- 아이콘: ico

- 상세내용: desc

- 댓글: cmt

ex) ico-file, ico-folder, btn-prev, btn-next

2. 분류별 subfix

- 하부기호로서 subfix는 prefix와 함께 부가설명 용도로 사용한다.

- 공용: comm

- 위치변화: top, mid, bot, left, right

- 순서변화: fst, lst

- 화살표: arr

- 방향: hori, vert

- 카테고리: cate

ex) ico-arr-news

3. 분류별 suffix

- 접미사를 의미하는 것으로 prefix와 함께 부가설명 용도로 사용, 상태를 나타내는데 사용된다.

- 상태변화: on, off, over, hit, focus

- 위치변화: top, mid, bot, left, right

- 순서변화: fst, lst

- 이전/다음: prev, next

**파일 및 폴더 네이밍 규칙**

1. HTML

- pop\_.html : 팝업 파일을 사용

- winpop\_.html : 윈도우팝업 파일을 사용

2. CSS, JS

- 모바일 폴더를 분리 했을 경우(기본)

ex) init.css, m/init.css

- 모바일 폴더를 분리 안 했을 경우(원페이지스타일)

ex) init.css, init\_m.css

3. Folder

- css(html, img, js)/서비스명(or서브페이지명or기능)

ex) css/front/init.css, css/include/pop.css

**File Zilla 업로드 구조**

ex) img/common/btn\_com.gif, img/대분류명/intro\_music.gif

**폴더 구조**

dist(배포파일)

* views (html 파일)
* views\_m (html 모바일파일)
* static (css, js 파일)

- m/\* (모바일 경로)

- img (img 파일)

- m/\* (모바일 경로)

src(배포전소스파일)

* template (html 파일)
* template\_m (html 모바일파일)
* sass (scss 파일)

- m/\* (모바일 경로)

* js (js 파일)

- m/\* (모바일 경로)