

# team '00ZZ'



## HTML 서식 규칙

### 1) HTML 설계

- **큰영역** / 머리글, 본문, 바닥글 영역

header, main contents, footer 영역으로 크게 나누고 작은 영역으로 설계를 진행합니다.

각 영역은 주석으로 영역을 구분해 줍니다. (주석 가이드는 하단 참조)

- **main contents영역** / 본문의 자식영역 section으로 구분

main contents영역은 파트별로 section으로 나누고 상단영역에 작업자의 이니셜을 주석으로 작성하여

다른 팀원이 작업자를 확인하기 쉽도록 합니다. (소영: sy / 지희: jh / 희경: hk / 보미: bm)

- **웹 표준 준수**

설계가 완료된 html은 W3C HTML 유효성 검사기로 테스트합니다.

### 2) HTML 코드규칙

- **들여쓰기**

1탭 들여쓰기: 자식요소 작성시 1탭 들여쓰기하여 코드를 확인하기 쉽도록 정리합니다.

- **빈줄**

가독성을 고려하여 빈 줄을 추가 할 수 있습니다. 단 연속적으로 1줄을 초과하지 않습니다.

- **주석**

영역주석: 각 영역과 작성자를 팀원이 직관적으로 인식하기 위해 작성

```
<!-- 영역명 / 작성자 이니셜 -->
```

```
<div>
```

```
<p>시작과 끝에 주석을 작성하여, 마크업의 영역을 표시합니다.</p>
```

```
</div>
```

```
<!-- //영역명 -->
```

```
<!------- 큰영역명 ----->
```

```
<header>
```

```
<h1>큰영역명 구분은 -- 점선표시 10개로 작성한다.</h1>
```

```
</header>
```

```
<!------- //영역명 ----->
```

설명주석: 상단에 한번 작성, 팀원에게 설명이 필요한 부분에 작성

```
<!-- 날짜 작성자 설명내용 수정/추가/삭제 -->
```

```
<p>설명이 필요한 부분의 시작 상단에 설명할 내용을 주석으로 표기합니다. </p>
```

```
<p><strong>단, 설명 영역의 범위가 넓을 경우에만 끝나는 지점에도 주석을 작성해줍니다.</strong>
```

```
<!-- //날짜 작성자 설명내용 수정/추가/삭제 --> </p>
```

### 3) HTML 네이밍 규칙

#### • 표기법

스네이크 표기법으로 소문자로 작성합니다.

타이틀명은 구체적이고 요소의 목적을 직관적으로 반영할 수 있도록 작성합니다.

| 종류        | 구분                          | 번호                    | 예시                                   | 비고                                 |
|-----------|-----------------------------|-----------------------|--------------------------------------|------------------------------------|
| id 선택자    | 스네이크 표기법<br>(hyphen-nate)   | 한 자리수<br>(생략가능, 2, 3) | img-box,<br>slide-container          | 자바스크립트<br>영역 id 선택자 작성             |
| class 선택자 | 스네이크 표기법<br>(hyphen-nate)   | 한 자리수<br>(생략가능, 2, 3) | header-wrapper,<br>tab-page          | css영역<br>class 선택자 작성              |
| Image     | 스네이크 표기법<br>(underbar_nate) | 두 자리수<br>(01, 02, 03) | btn_cancle01.gif,<br>slider_bg05.png | 표기법을 준수하여<br>파일을 저장하고,<br>html 설계! |
| 파일 및 폴더   | 스네이크 표기법<br>(hyphen-nate)   | 한 자리수<br>(생략가능, 2, 3) | header-nav.html                      |                                    |

#### • 영역 네이밍 규칙

구체적이고 영역의 목적을 직관적으로 반영할 수 있도록 작성합니다.

| 구분   | 약속된 네이밍  | 영역                     | 비고            |
|------|----------|------------------------|---------------|
| 레이아웃 | header   | 머리글 영역                 |               |
|      | content  | 본문 영역                  |               |
|      | footer   | 바닥글 영역                 |               |
|      | section  | 본문의 자식영역               | class명과 함께 작성 |
|      | -wrapper | section 안에서 전체를 감싸는 영역 | class명으로 작성   |
| 그룹핑  | -wrap    | 전체 영역                  | class명으로 작성   |
|      | -box     | 지역 영역                  | class명으로 작성   |

#### • 이미지 네이밍 규칙

공통적으로 사용되는 이미지: 파비콘, 로고 등

| 분류  | 파일명            | 예시                | 비고                                   |
|-----|----------------|-------------------|--------------------------------------|
| 파비콘 | favicon_특징.확장자 | favicon_16x16.png | 공통적으로 사용하는<br>이미지는 common 폴더에 저장합니다. |
| 로고  | logo_특징.확장자    | logo_black.png,   |                                      |

영역별로 사용되는 이미지: 각 영역의 앞글자 한개를 넣어서 작성

| 분류  | 파일명         | 예시               | 비고                    |
|-----|-------------|------------------|-----------------------|
| 아이콘 | 영역앞글자_icon_ | f_icon_insta.png | 각 경로 image 폴더에 저장합니다. |
| 일반  | 영역앞글자_특징_   | s_bg_02.jpg      |                       |

# CSS 서식 규칙

## 1) css 설계

- 프로젝트 시작하는 단계에서 css분석 우선 실행

자바스크립트 구현 상황 최소화하기위해 CSS로 할 수 있는 요소를 분석합니다.

공통css를 만들어야 하는 부분을 분석합니다.

style.css 영역안에서 공통으로 스타일값을 줄 수 있는 부분을 분석합니다.

- 공통 css

variable.css에 웹 가이드라인에 맞는 마진, 폰트패밀리, 폰트사이즈, 포인트컬러 등을 작성합니다.

공통 css는 해당 페이지의 css 우선순위보다 낮게 연결해줍니다.

- style.css 내의 공통 스타일값

css 스타일 값이 공통인 경우도 문서의 상단에 우선 작업 진행합니다. (주석에 설명 작성 필수)

- 웹 표준 준수

설계가 완료된 html은 W3C CSS 유효성 검사기로 테스트합니다.

- 네이밍 표기

스네이크표기법을 적용하여 직관적인 네임으로 작성합니다. (예시: header-style.css)

## 2) css 기본 규칙

- 작성순서

영역.선택자 { 속성순서: 레이아웃 > 크기/간격 > 박스모양 > 폰트/정렬 > 기타 }

```
@charset "utf-8";  
/*----- 큰영역명 -----*/  
header .nav { width: 100%; height: 60px font-size: var(--fontSize-15); }
```

- 주석 / 빈줄

각 영역을 구별하기 위한 주석과 설명주석을 작성합니다. css에서 종료 주석은 작성하지 않습니다.

코드 그룹 간 1줄의 빈 줄을 만들어 가독성을 높여 줍니다.

## 3) css 문서 저장, 연결

- 저장 및 작성순서

공통 css 문서는 common 폴더에 저장해서 연결해서 사용합니다.

html 문서 연결은 공통css문서부터 작성한뒤, 직접 작성한 css문서를 아랫쪽에 연결해 줍니다.

```
/*----- 공통style.css를 위에 작성하고 후에 직접 작성한 style.css 문서를 연결합니다. -----*/  
<link rel="stylesheet" href="../common/minireset.min.css">  
<link rel="stylesheet" href="/slide3.css">
```

## JavaScript II jQuery 서식 규칙

### 1) 자바스크립트 II 제이쿼리 작성

- **간결하게 적용**

최대한 많이 찾아보고 가장 간단하게 작성할 수 있도록 합니다.

- **변수 선언**

변수 선언시 지역변수로 작성합니다

const로 우선 작성합니다.

다른 값을 할당해야할 경우에만 let으로 선언하고 주석을 작성하여 내용을 기록해 팀원이 확인할 수 있도록 합니다.

- **식별자명 네이밍 표기**

카멜표기법을 적용하여 직관적인 네임으로 작성합니다. (예시: headerScrollMagic)

- **{괄호안 작은 따옴표} 통일 작성**

괄호안에 선택자 작성시 작은 따옴표로 통일하여 작성합니다.

### 2) 자바스크립트 II 제이쿼리 작성 규칙

- **주석**

어떤 영역에 어떤 움직임을 준 것인지 팀원들이 알수있도록 상단에 한번 주석을 작성합니다.

```
// 날짜 작성자 영역 스크립트 설명내용  
/* 작성된 내용이 길어지면 이 주석을 사용합니다. */
```

- **저장 및 작성순서**

공통 js 문서는 common 폴더에 저장해서 연결해서 사용합니다.

html 문서 연결은 공통 js문서부터 작성한뒤, 직접 작성한 js문서를 아랫쪽에 연결해 줍니다.

```
/*----- 공통 js 문서를 위에 작성하고 후에 직접 작성한 js 문서를 연결합니다. -----*/  
<script src="../common/jquery-ui.min.js"></script>  
<script defer src="/slide.js"></script>
```