

MyFood 클론 코딩

#01. 작업 개요

1) 대상 사이트

<https://www.w3schools.com/>에서 운영하는 기본 예제 중에서 하나를 선정하여 재구성

https://www.w3schools.com/w3css/tryw3css_templates_food_blog.htm

2) HTML + CSS + JS 기반 웹 페이지 작업 구조

```
├── assets          <-- 리소스(재료)가 저장될 폴더
│   ├── css        <-- 스타일시트 파일 (scss가 변환되어 자동 생성됨)
│   ├── fonts      <-- 웹폰트 파일
│   ├── helper     <-- 직접 작성한 JS 모듈
│   ├── img        <-- 이미지 파일
│   ├── js         <-- 각 html별로 개별 동작하는 js 소스코드
│   └── scss        <-- 디자인 구성 파일
└── *.html         <-- 웹 페이지 소스코드 (html파일에 대한 폴더 구성은 자율적으로...)
```

- 이 작업에서는 웹폰트를 CDN 방식으로 사용 (fonts 폴더 불필요)
- 아직 Javascript에 대한 내용을 학습하기 전이므로 `helper`, `js` 폴더 불필요
- `scss` 역시 학습 전이므로 폴더 불필요

3) 사이트 내의 이미지 가져오기

크롬 브라우저에 `Image Downloader` 설치

<https://chrome.google.com/webstore/detail/image-downloader/cnpniohnpfhjhiaiiggeabnkjhpaldj/related?hl=ko>

웹 페이지에서 익스텐션을 실행하면 해당 페이지 내의 모든 이미지 파일을 일괄 다운로드 받을 수 있다.

#02. 웹 페이지 작성 시작하기

common.html

모든 페이지에서 사용되는 공통 코드를 구현한 파일

```
<!DOCTYPE html>
<html lang="ko" translate="no">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

```

<title>MyFood</title>

<!-- SEO meta tag -->
<meta name="google" content="nottranslate" />
<meta name="description" content="검색결과에 표시될 내용" />
<meta name="robots" content="index,follow" />
<meta name="keywords" content="SEO, 검색엔진 최적화, 메타 태그" />
<meta name="author" content="leekh" />

<!-- SNS meta tag -->
<meta property="og:type" content="website" />
<meta property="og:title" content="페이지 제목" />
<meta property="og:description" content="페이지 설명" />
<meta property="og:image" content="http://www.mysite.com/myimage.jpg" />
<meta property="og:url" content="http://www.mysite.com" />

<!-- reset.css -->
<link rel="stylesheet" href="assets/css/reset.css" />

<!-- google web font -->
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
<link
    href="https://fonts.googleapis.com/css2?
family=Noto+Sans+KR:wght@100;300;400;500;700;900&display=swap"
    rel="stylesheet"
/>

<!-- font awesome -->
<link
    rel="stylesheet"
    href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css"
/>
</head>
<body></body>
</html>

```

SEO, SNS 관련 <meta>태그

검색 엔진에 노출될 정보를 설정한다.

에릭 마이어의 reset.css

모든 태그는 기본 글자 크기와 굵기, 여백 속성을 기본값으로 갖고 있다.

하지만 이 값들은 웹 브라우저마다 상이하다.

처음부터 모든 태그의 글자크기, 굵기, 여백 등을 동일하게 0으로 맞춰놓고 시작하자는 취지의 CSS 오픈 소스

이 코드를 모든 웹 페이지의 가장 첫 CSS로 참조

<https://meyerweb.com/eric/tools/css/reset/>

공통 폰트 설정

구글 웹폰트

<https://fonts.google.com/>에서 사용할 글꼴을 선정

```
<link rel="preconnect" href="https://fonts.googleapis.com" />
<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
<link
  href="https://fonts.googleapis.com/css2?
family=Noto+Sans+KR:wght@100;300;400;500;700;900&display=swap"
  rel="stylesheet"
/>
```

아이콘 폰트 (Font Awesome)

<https://cdnjs.com/libraries/font-awesome>

```
<link
  rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css"
/>
```

#03. 공통 영역(header, footer) 작업하기

`common.html`에 사이트의 상단 영역과 하단 영역을 우선적으로 작업한다.

이후 다른 페이지를 작업할 때 `common.html`을 복사하고 내용영역만 구현해 나간다

이 파일에 대한 스타일은 `common.css`로 구현

```
<!-- 공통 CSS -->
<link rel="stylesheet" href="assets/css/common.css" />
```

1) 영역 나누기

`common.html`

```
<!-- 사이트의 전체 컨테이너 -->
<div class="container">
  <!-- 상단 영역-->
  <header class="header">
```

```

    <h1>상단영역</h1>
</header>

<!-- 메인 내용 영역 -->
<section class="main">
    <h1>내용영역</h1>
</section>

<!-- 하단영역 -->
<footer class="footer">
    <h1>하단영역</h1>
</footer>
</div>

```

common.css

```

/**
 * 모든 페이지에서 사용될 공통적인 내용들을 정의하기 위한 파일
 */

/*-----
 * 공통
 *-----*/
* {
    font-family: "Noto Sans KR", sans-serif;
    box-sizing: border-box;
}

a {
    text-decoration: none;
    color: inherit;
}

/*-----
 * 상단영역
 *-----*/
.header {
    background-color: #f062;
}

/*-----
 * 내용영역
 *-----*/
.main {
    background-color: #06f2;
}

/*-----
 * 하단영역
 *-----*/
.footer {

```

```
background-color: #0002;  
}
```

2) 각 영역 안에서 넓이를 제한하기 위한 컨테이너 구성

header, main, footer 안에 `<div class="content-container"></div>` 박스를 포함하여 넓이를 제한하도록 한다.

common.html

```
<!-- 사이트의 전체 컨테이너 -->  
<div class="container">  
  <!-- 상단 영역-->  
  <header class="header">  
    <div class="content-container">  
      <h1>상단영역</h1>  
    </div>  
  </header>  
  
  <!-- 메인 내용 영역 -->  
  <section class="main">  
    <div class="content-container">  
      <h1>내용영역</h1>  
    </div>  
  </section>  
  
  <!-- 하단영역 -->  
  <footer class="footer">  
    <div class="content-container">  
      <h1>하단영역</h1>  
    </div>  
  </footer>  
</div>
```

common.css

```
/*-----  
 * 각 영역 안에서 넓이를 제한하기 위한 공통 컨테이너  
 *-----*/  
.content-container {  
  background-color: #0603;  
  max-width: 1200px;  
  margin: auto;  
}
```

3) 상단영역

```

<!-- 상단 영역-->
<header class="header">
  <div class="content-container">
    <!-- 좌측 햄버거 버튼 영역의 박스 -->
    <a href="#" class="icon-box bar-icon-box">
      <i class="fa-solid fa-bars"></i>
    </a>
    <!-- 중앙 로고 영역의 박스 -->
    <div class="title-box">
      <h1>My Food</h1>
    </div>
    <!-- 우측 메일 버튼 영역의 박스 -->
    <a href="#" class="icon-box mail-icon-box">
      <i class="fa-solid fa-envelope"></i>
    </a>
  </div>
</header>

```

```

/*-----
* 상단영역
*-----*/
/* 상단영역에 의해 가려지는 부분을 화면에 표시하기 위한 여백 */
.container {
  padding-top: 60px;
}

.header {
  /* background-color: #f062; */
  /* 상단 고정 */
  position: fixed;
  left: 0;
  top: 0;
  width: 100%;

  /* 그림자 처리 */
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2);
  background-color: #ffffff;
}

/* 상단 영역 내의 컨테이너에 속한 아이템들의 배치 구성 */
.header .content-container {
  display: flex;
  justify-content: space-between;
  padding: 0 10px;
}

/** 양쪽의 아이콘 박스 공통 속성 */
.header .icon-box {
  padding: 16px;
  font-size: 24px;
  font-weight: 900;
}

```

```

    transition: all 0.1s ease-in;
}

.header .icon-box:hover {
    background-color: #dddddd;
    color: #ffffff;
}

/** 사이트 제목 */
.header .title-box {
    display: flex;
    align-items: center;
}
.header .title-box h1 {
    font-size: 28px;
    font-weight: 600;
}

```

4) 하단영역

하단의 영역 나누기

```

<!-- 하단영역 -->
<footer class="footer">
    <div class="content-container">
        <div class="footer-item">item1</div>
        <div class="footer-item">item2</div>
        <div class="footer-item">item3</div>
    </div>
</footer>

```

```

/*-----
 * 하단영역
 *-----*/

.footer {
    background-color: #0002;
    border-top: 1px solid #dddddd;
}

.footer .content-container {
    display: flex;
}

.footer .footer-item {
    flex: 1;
}

```

하단의 각 부분 내용 채워 넣기

```
<!-- 하단영역 -->
<footer class="footer">
  <div class="content-container">
    <div class="footer-item">
      <h3>Footer</h3>
      <p>
        Praesent tincidunt sed tellus ut rutrum. Sed vitae justo
        condimentum, porta lectus vitae, ultricies congue gravida diam
        non fringilla.
      </p>
      <p>
        Powered by
        <a href="https://www.w3schools.com/w3css/default.asp">w3.css</a>
      </p>
    </div>
    <div class="footer-item">
      <h3>Blog Posts</h3>
      <ul class="bloglist">
        <li>
          <a href="#">
            
            <div class="text-box">
              <h4 class="title">Lorem</h4>
              <p>
                Sed mattis nunc Sed mattis nunc Sed mattis nunc
              </p>
            </div>
          </a>
        </li>
        <li>
          <a href="#">
            
            <div class="text-box">
              <h4 class="title">Lorem</h4>
              <p>
                Sed mattis nunc Sed mattis nunc Sed mattis nunc
              </p>
            </div>
          </a>
        </li>
      </ul>
    </div>
    <div class="footer-item">
      <h3>Popular Tags</h3>
      <ul class="tags">
        <li class="black">Travel</li>
        <li>New York</li>
        <li>Dinner</li>
        <li>Salmon</li>
        <li>France</li>
      </ul>
    </div>
  </div>
</footer>
```



```

        <li>Drinks</li>
        <li>Ideas</li>
        <li>Flavors</li>
        <li>Cuisine</li>
        <li>Chicken</li>
        <li>Dressing</li>
        <li>Fried</li>
        <li>Fish</li>
        <li>Duck</li>
    </ul>
</div>
</div>
</footer>

```

```

/*-----
 * 하단영역
 *-----*/
.footer {
    /* background-color: #0002; */
    border-top: 1px solid #dddddd;
}

.footer .content-container {
    display: flex;
}

.footer .footer-item {
    flex: 1;
    padding: 15px;
}

.footer .footer-item h3 {
    font-size: 24px;
    font-weight: 700px;
    margin: 22px 0;
    text-transform: uppercase;
}

.footer .footer-item p {
    font-size: 15px;
    line-height: 150%;
    font-weight: 300;
}

.footer .footer-item .bloglist a {
    display: flex;
    margin: 20px 0;
    padding: 0 10px;
}

.footer .footer-item .bloglist a img {

```

```

width: 65px;
height: 65px;
object-fit: cover;
margin-right: 10px;
}

.footer .footer-item .bloglist a .text-box {
display: flex;
flex-direction: column;
justify-content: center;
}

.footer .footer-item .bloglist a .text-box .title {
font-weight: bold;
margin-bottom: 5px;
font-size: 15px;
}

.footer .footer-item .bloglist a .text-box p {
line-height: 1.2;
}

.footer .footer-item .tags li {
display: inline-block;
background-color: #616161;
padding: 5px 10px;
margin-bottom: 8px;
color: #ffffff;
font-size: 13px;
}

.footer .footer-item .tags li.black {
background-color: #000000;
}

```

5) 하단 영역에 대한 반응형 기능 추가

```

@media only screen and (max-width: 768px) {
.footer .content-container {
flex-direction: column;
}

.footer .footer-item h3 {
margin-top: 0;
}
}

```

#04 메인 영역 분할

1) index.html 생성

common.html을 index.html로 복사하고 인덱스 페이지를 위한 CSS 파일(index.css)을 별도로 준비

index.html에서 CSS파일 참조

```
<!-- index style -->
<link rel="stylesheet" href="assets/css/index.css" />
```

2) 메인 영역 나누기

```
<!-- 메인 내용 영역 -->
<section class="main">
  <div class="content-container">
    <!-- 음식 갤러리 영역 -->
    <div class="food-gallery-container"></div>

    <!-- 구분선 -->
    <hr class="divider" />

    <!-- 셰프 소개 -->
    <div class="profile-container"></div>
  </div>
</section>
```

#05. 음식 갤러리 영역

1) 단위 구성

```
<!-- 음식 갤러리 영역 -->
<div class="food-gallery-container">
  <!-- 음식 목록 -->
  <ul class="food-gallery"></ul>

  <!-- 페이지 번호 -->
  <ul class="paging"></ul>
</div>

<!-- 구분선 -->
<hr class="divider" />
```

```
/*-----
* 중앙 구분선
*-----*/
.divider {
  margin: 60px 0;
  width: auto;
```

```
border-top: 1px solid #ddd;
border-bottom: 0;
}
```

2) 음식 목록

```
<!-- 음식 목록 -->
<ul class="food-gallery">
  <li class="food-item">
    <a href="#">
      <div class="img-wrapper"></div>
      <div class="food-content">
        <h2>The Perfect Sandwich, A Real NYC Classic</h2>
        <p>Just some random text, lorem ipsum text praesent tincidunt
        ipsum lipsum</p>
      </div>
    </a>
  </li>

  ... li태그 단위를 8개가 되도록 구성 ...
</ul>
```

```
/*-----
 * 음식 갤러리 영역
 *-----*/

.food-gallery {
  display: flex;
  flex-wrap: wrap;
}

.food-gallery .food-item {
  width: 25%;
  padding: 20px 10px;
}

.food-gallery .food-item a {
  display: block;
}

.food-gallery .food-item a .img-wrapper {
  overflow: hidden;
}

.food-gallery .food-item a .img-wrapper img {
  width: 100%;
  height: 360px;
  object-fit: cover;
  transition: all 0.3s ease-in-out;
}
```

```

.food-gallery .food-item a:hover .img-wrapper img {
  transform: scale(1.1, 1.1);
}

.food-gallery .food-item a .food-content {
  padding: 15px 10px;
  text-align: center;
}

.food-gallery .food-item a .food-content h2 {
  font-size: 22px;
  margin-bottom: 12px;
  line-height: 130%;
}

.food-gallery .food-item a .food-content p {
  font-size: 16px;
  font-weight: 300;
  line-height: 120%;
}

```

3) 페이지 번호

```

<!-- 페이지 번호 -->
<ul class="paging">
  <li><a href="#"><</a></li>
  <li class="active"><a href="#">1</a></li>
  <li><a href="#">2</a></li>
  <li><a href="#">3</a></li>
  <li><a href="#">4</a></li>
  <li><a href="#">5</a></li>
  <li><a href="#">></a></li>
</ul>

```

```

/*-----
* 페이지 번호 영역
*-----*/

.paging {
  display: flex;
  justify-content: center;
}

.paging li {
  margin: 0 5px;
  width: 32px;
  height: 32px;
}

```

```

.paging li a {
    display: block;
    font-size: 18px;
    text-align: center;
    line-height: 32px;
}

.paging li a:hover {
    background-color: rgba(0, 0, 0, 0.2);
}

.paging li.active {
    background-color: #000000;
}

.paging li.active a {
    color: #ffffff;
}

```

#06. 쉼표 소개

```

<!-- 쉼표 소개 -->
<div class="profile-container">
    <h2>About Me, The Food Man</h2>
    
    <h3>I am Who I Am!</h3>
    <h4>With Passion For Real, Good Food</h4>
    <p>
        Just me, myself and I, exploring the universe of unknownment. I have a
        heart of love and an interest of lorem ipsum and mauris neque quam blog. I want to
        share my world with you. Praesent tincidunt sed tellus ut rutrum. Sed vitae
        justo condimentum, porta lectus vitae, ultricies congue gravida diam non
        fringilla. Praesent tincidunt sed tellus ut rutrum. Sed vitae justo condimentum,
        porta lectus vitae, ultricies congue gravida diam non fringilla.
    </p>
</div>

```

```

/*-----
 * 쉼표 소개
 *-----*/

.profile-container {
    text-align: center;
}

.profile-container h2 {
    font-size: 32px;
    margin-bottom: 32px;
}

```

```

.profile-container img {
    max-width: 100%;
    height: auto;
    margin-bottom: 26px;
}

.profile-container h3 {
    font-size: 26px;
    margin-bottom: 16px;
}

.profile-container h4 {
    font-size: 18px;
    margin-bottom: 16px;
}

.profile-container p {
    font-size: 16px;
    font-weight: 300;
    line-height: 125%;
    margin-bottom: 30px;
    padding: 0 25px;
}

```

#07 반응형 구현

```

/*-----
 * 반응형 구현
 *-----*/

@media only screen and (min-width: 768px) {
    .food-gallery .food-item {
        width: 25%;
    }
}

@media only screen and (min-width: 600px) and (max-width: 767px) {
    .food-gallery .food-item {
        width: 50%;
    }
}

@media only screen and (max-width: 599px) {
    .food-gallery .food-item {
        width: 100%;
    }
}

```