



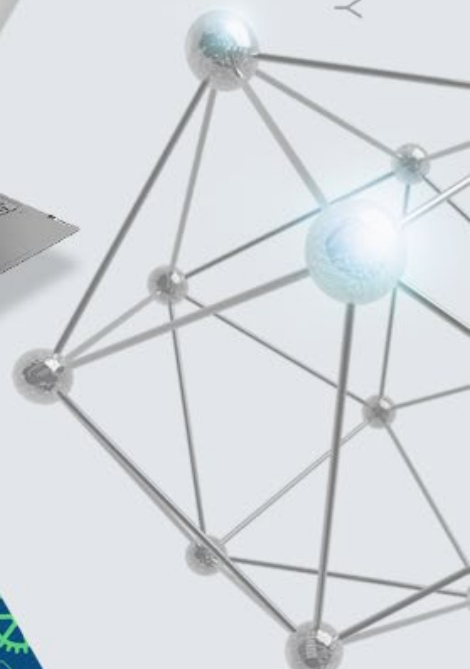
한국기술교육대학교
온라인평생교육원

The 4th Industrial Revolution is characterized by super connectivity and super intelligence, where various products and services are connected to the network, and artificial intelligence and information communication technologies are used in 3D printing, unmanned transportation, robotics. Of the world's most advanced technologies.

T
E
C
H
N
O
L
O
G
Y

웹 표준에 맞는 HTML5 프로그래밍

화면설계 확인하기



The 4th Industrial Revolution is characterized by super connectivity and super intelligence, where various products and services are connected to the network, and artificial intelligence and information communication technologies are used in 3D printing, unmanned transportation, robotics. Of the world's most advanced technologies.



화면 설계 확인하기



학/습/목/표

1. 와이어프레임의 개념과 구성요소를 설명할 수 있다.
2. HTML과 CSS를 활용하여 기본적인 웹 레이아웃을 구현할 수 있다.



학/습/내/용

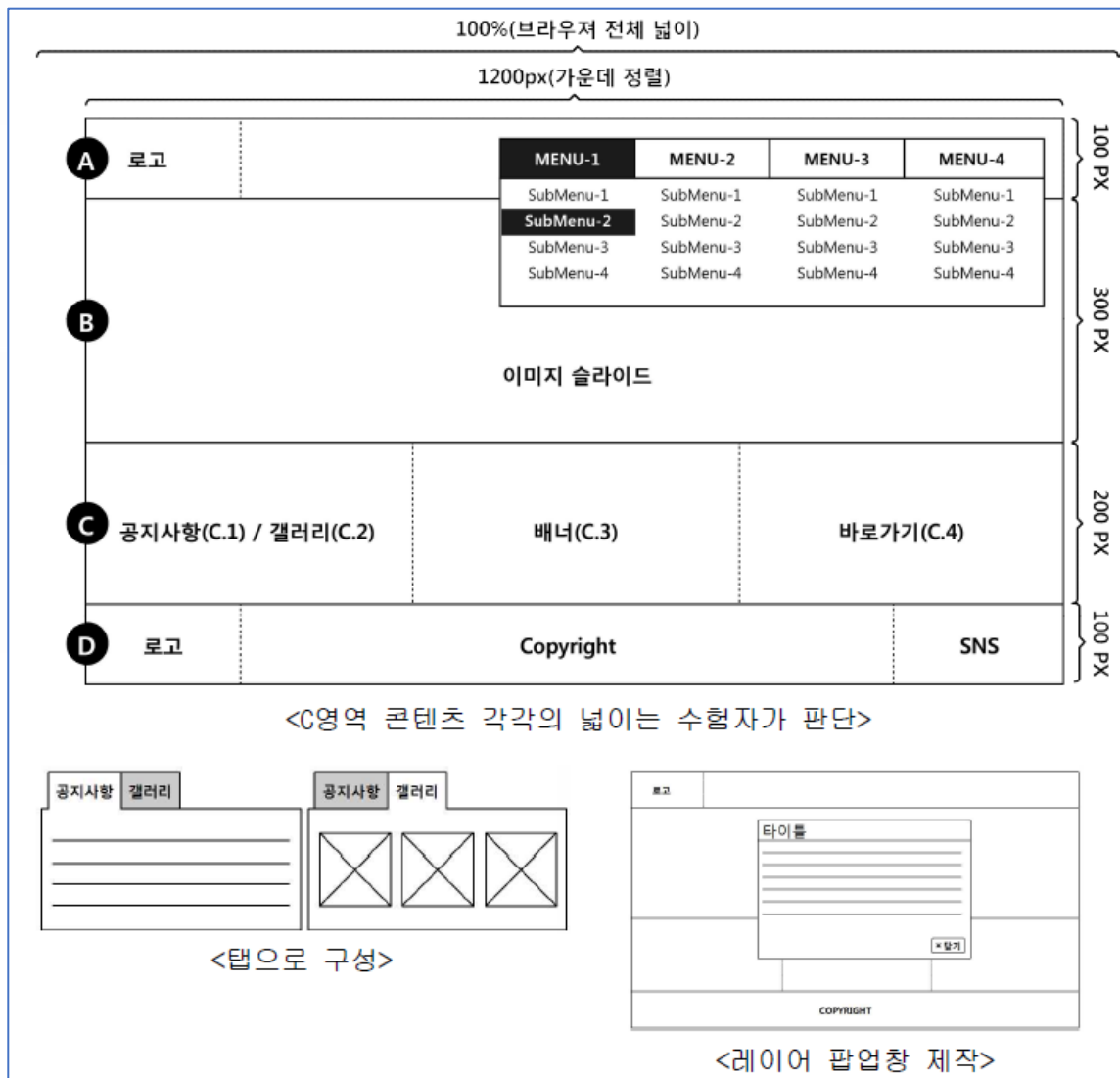
1. 와이어프레임
2. 레이아웃 구현하기

1. 와이어프레임

1) 와이어프레임의 개념

(1) 와이어프레임 (Wireframe)의 개념

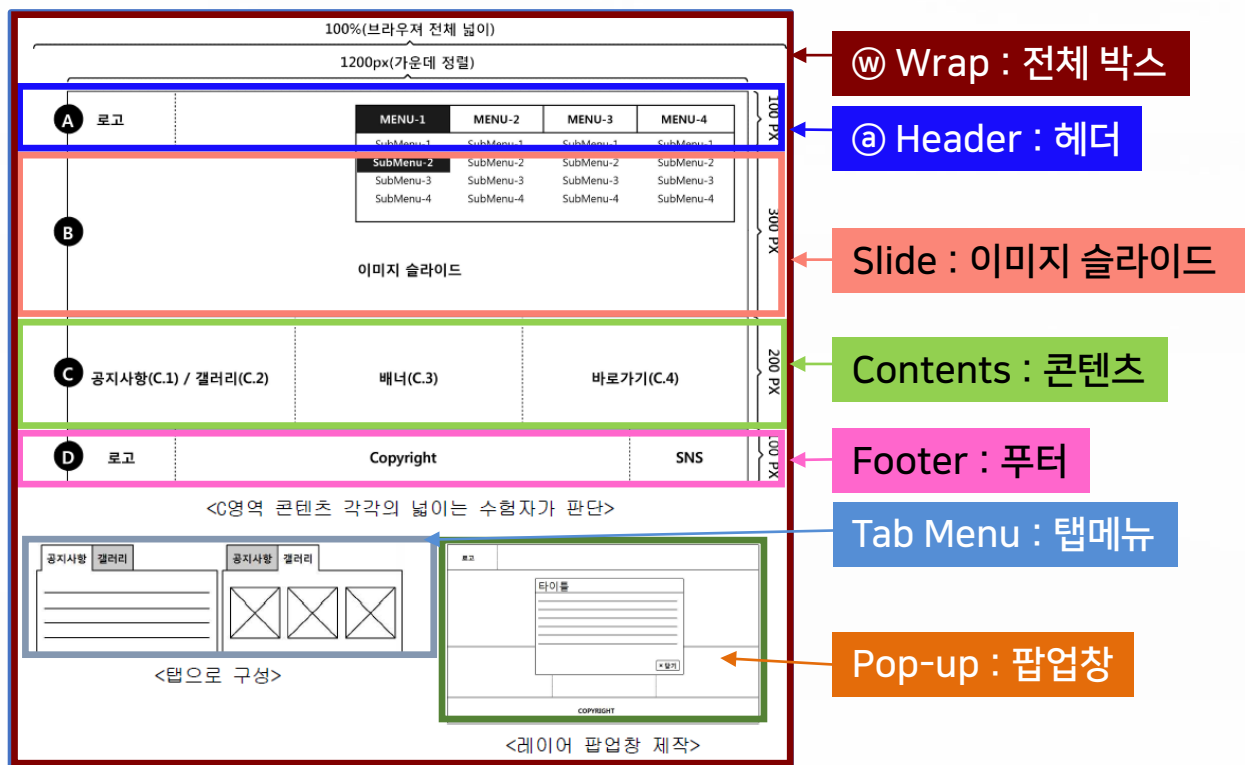
- 와이어프레임 : 웹사이트의 **회로도 (schematic)** 또는 **청사진 (blueprint)**라고도 불리우며, 웹사이트의 골격을 나타내는 시각적 가이드



1. 와이어프레임

2) 와이어프레임의 구성 요소

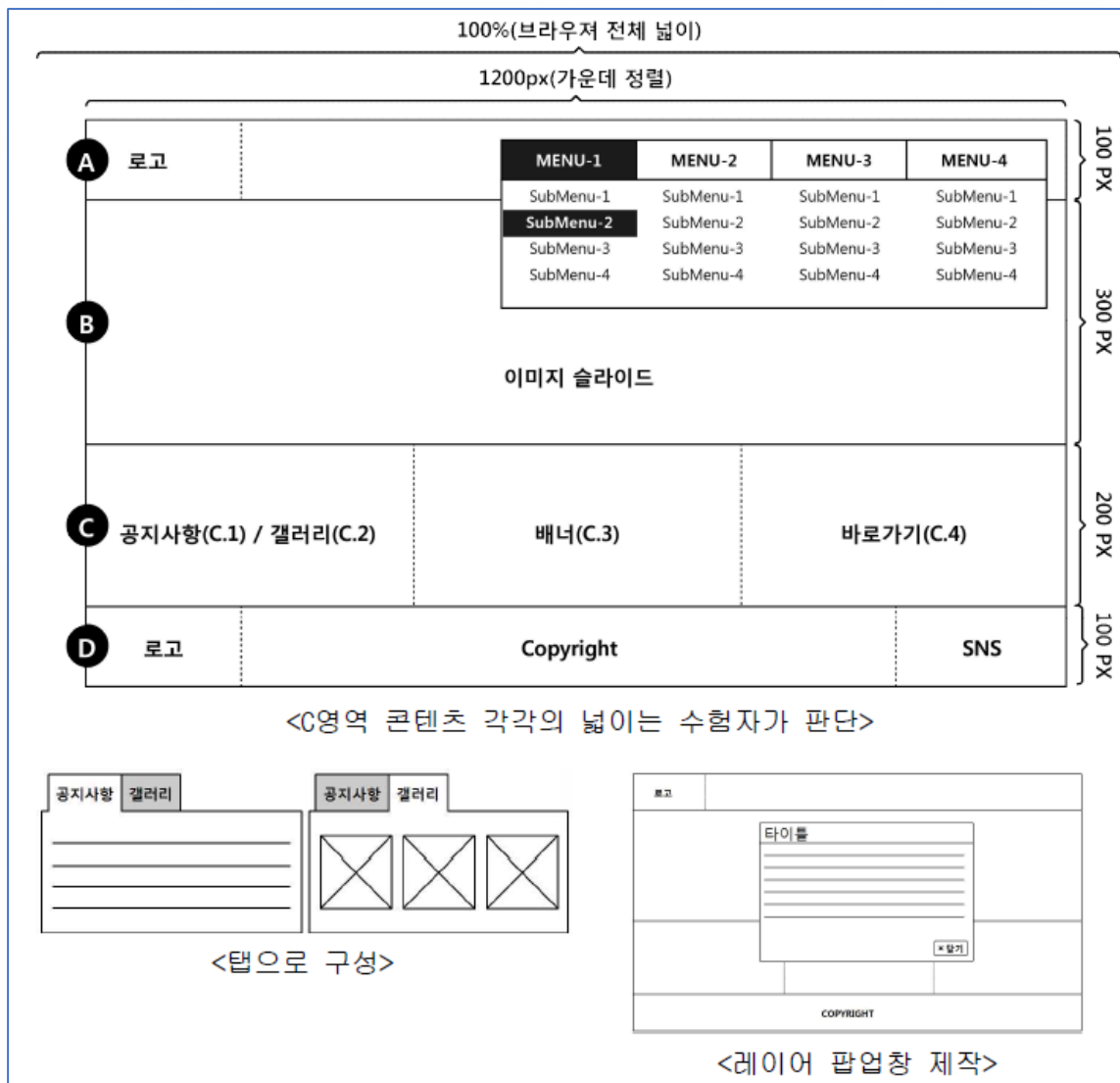
(1) 와이어프레임 (Wireframe)의 구성 요소



1. 와이어프레임

3) 와이어프레임의 종류

(1) 와이어프레임 (Wireframe)의 종류

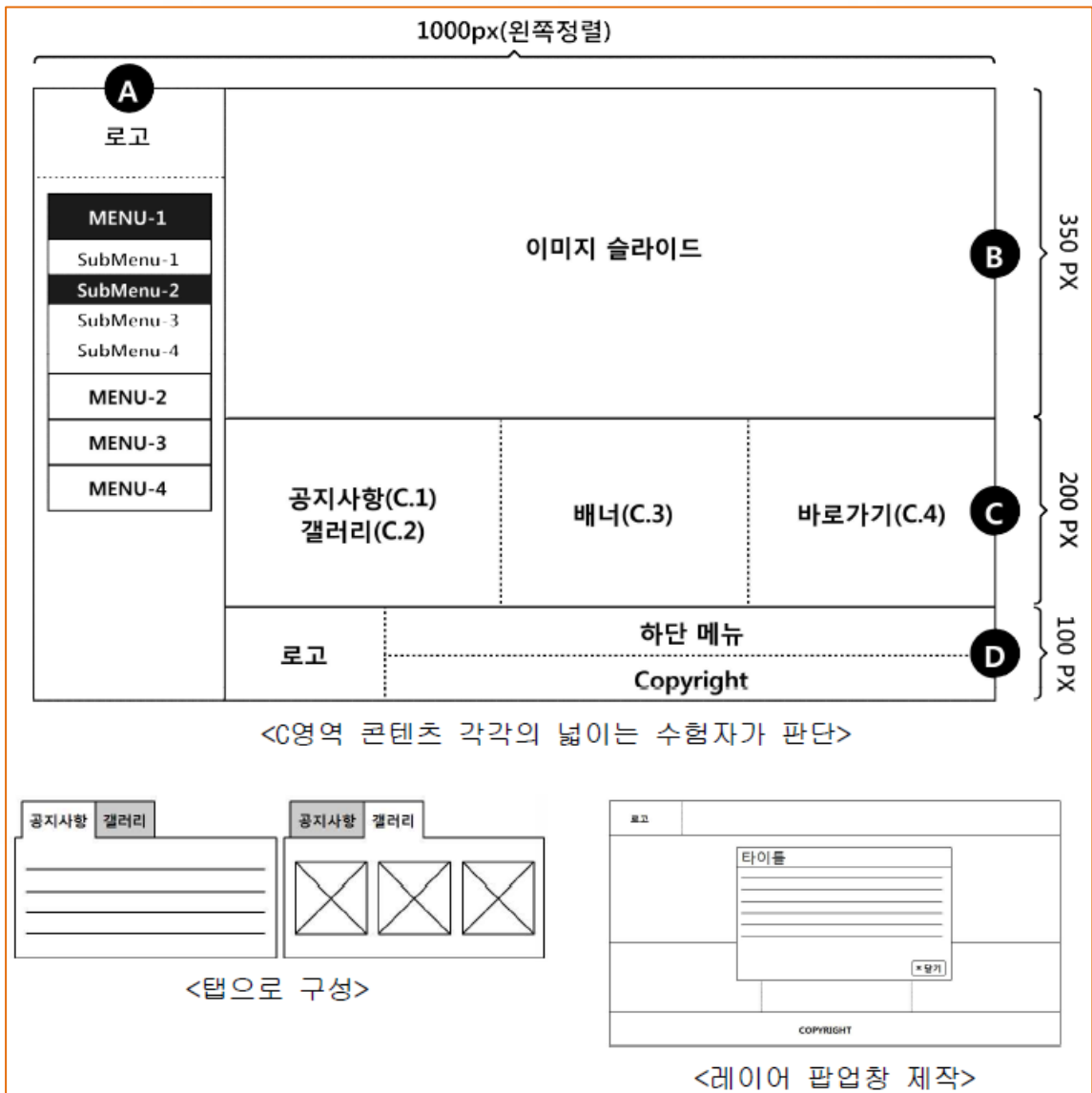


[가로형]

1. 와이어프레임

3) 와이어프레임의 종류

(1) 와이어프레임 (Wireframe)의 종류

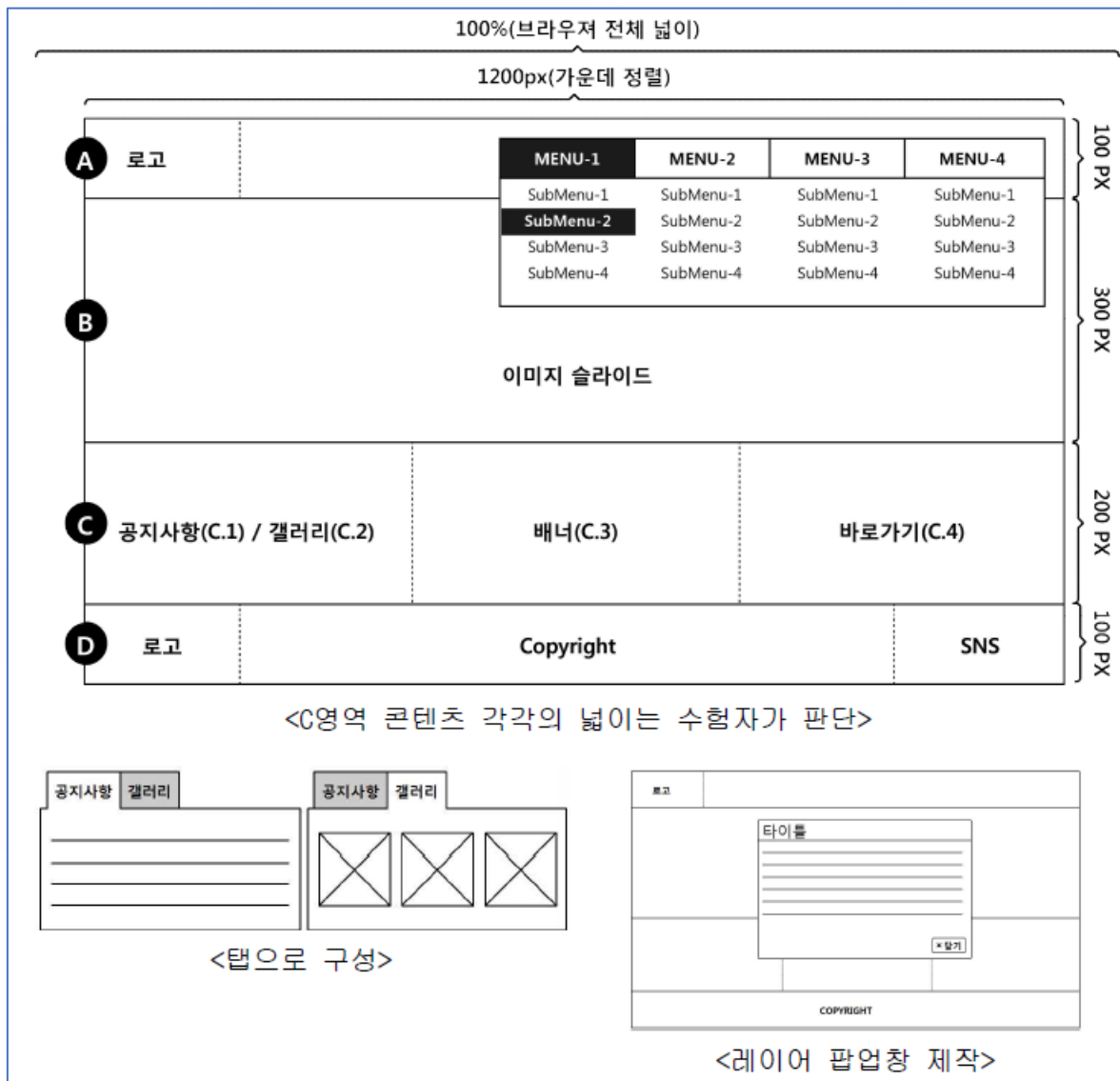


[세로형]

2. 레이아웃 구현하기

1) HTML 기본 구성 입력과 CSS Reset하기

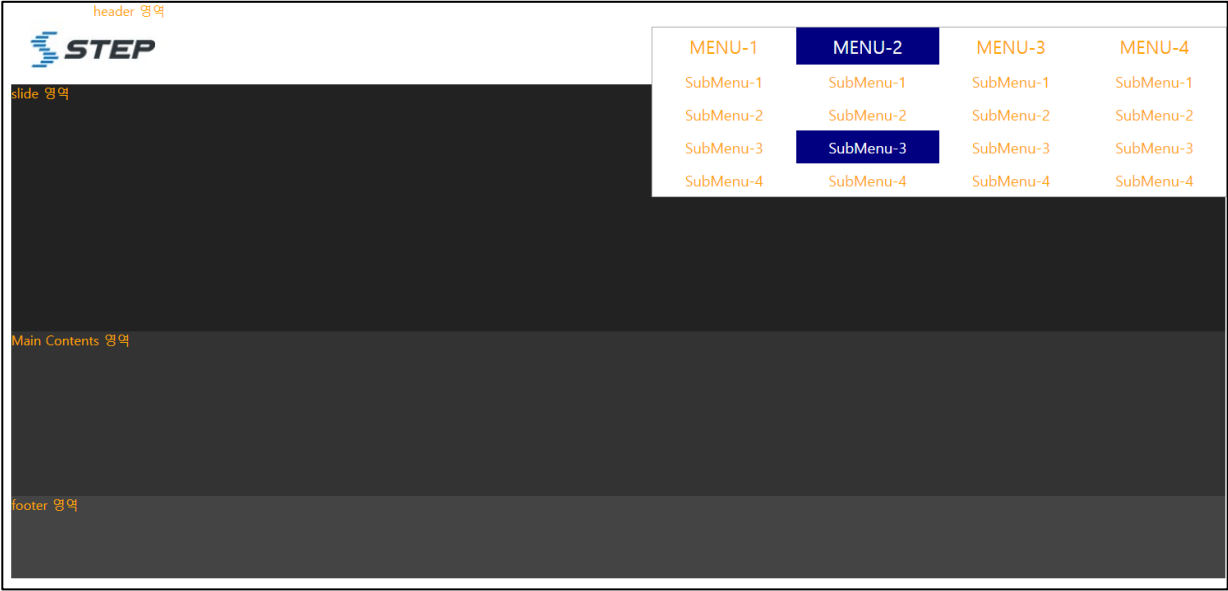
(1) 레이아웃 확인하기



2. 레이아웃 구현하기

1) HTML 기본 구성 입력과 CSS Reset하기

(2) 최종 구현



2. 레이아웃 구현하기

1) HTML 기본 구성 입력과 CSS Reset하기

(2) 웹에디터의 Emmet 기능을 활용하여 html 기본 구성 완성하기 (index.html)

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="utf-8" />
    <title>Layout</title>
    <link rel="stylesheet" href="css/style.css" type="text/css"
  />
</head>
<body></body>
</html>
```

(3) CSS 초기화 (Reset)

```
* {
  margin: 0;
  padding: 0;
}
ul { list-style: none; }
a {
  text-decoration: none;
  color: inherit;
}
.clearfix::after {
  display: table;
  content: '';
  clear: both;
}
img {
  vertical-align: top;
  border: 0;
}
body { color: #ffa500; }
```

float로 인한 버그를 해결

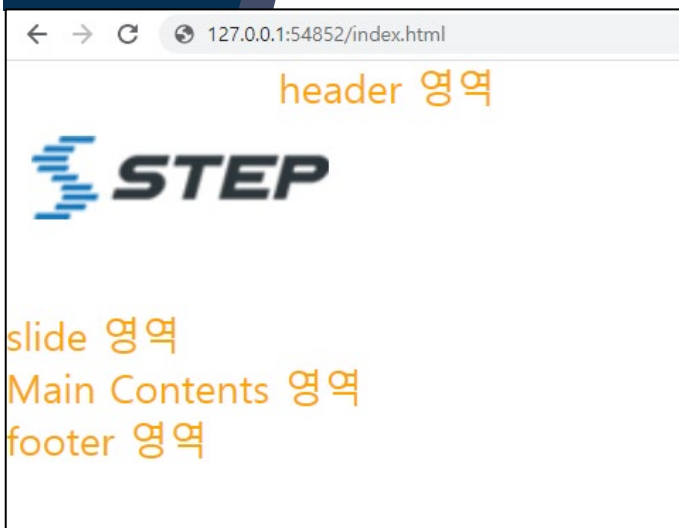
2. 레이아웃 구현하기

2) HTML 기본 레이아웃 구현하기

(1) HTML 기본 레이아웃 구현

```
<body>
  <div class="wrap">
    <header>header 영역
      <h1>
        <a href="#"> </a>
      </h1>
    </header>
    <div class="slide">slide 영역</div>
    <div class="contents"> Main Contents 영역 </div>
    <footer>footer 영역</footer>
  </div>
</body>
```

실행 결과



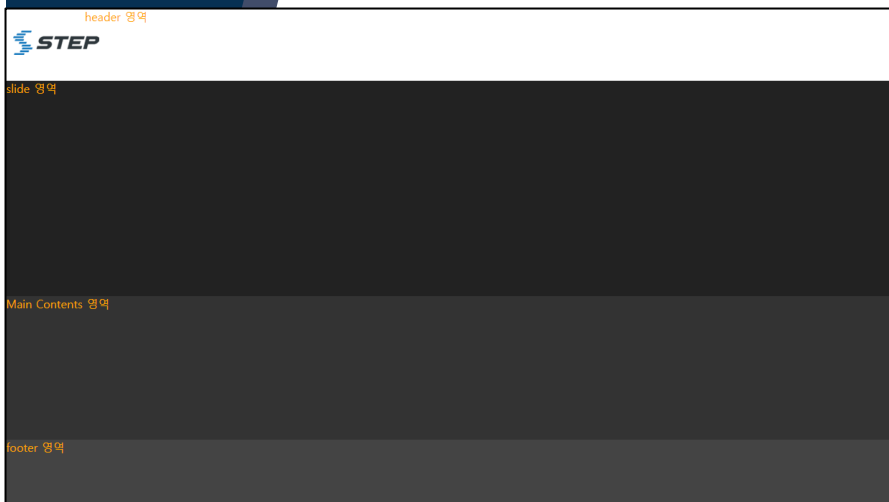
2. 레이아웃 구현하기

3) CSS로 스타일 구현하기

(1) CSS로 스타일 구현

```
.wrap {  
  width: 1200px;  
  margin: 0 auto;  
  background: #111  
}  
header {  
  width: 100%; height: 100px;  
}  
.slide {  
  width: 100%; height: 300px;  
  background: #222;  
}  
.contents {  
  width: 100%; height: 200px;  
  background: #333;  
}  
footer {  
  width: 100%; height: 100px;  
  background: #444;  
}
```

실행 결과



2. 레이아웃 구현하기

4) Header 영역 구조 꾸미기

(1) Header 영역 구조 (HTML)

```
<header class="clearfix">
  header 영역
  <h1>
    <a href="#">
      </a>
    </h1>
    <nav>
      <ul class="gnb">
        <li>
          <a href="#">MENU-1</a>
          <ul class="sub">
            <li><a href="#">SubMenu-1</a></li>
            <li><a href="#">SubMenu-2</a></li>
            <li><a href="#">SubMenu-3</a></li>
            <li><a href="#">SubMenu-4</a></li>
          </ul>
        </li>
        <li>
          <a href="#">MENU-2</a>
          <ul class="sub">
            <li><a href="#">SubMenu-1</a></li>
            <li><a href="#">SubMenu-2</a></li>
            <li><a href="#">SubMenu-3</a></li>
            <li><a href="#">SubMenu-4</a></li>
          </ul>
        </li>
      </ul>
    </nav>
  </header>
```

1/2

→ nav : 메뉴를 구성하기 위한 네비게이션 태그 사용

→ ul : 메뉴바의 전체 목록을 위한 태그

2. 레이아웃 구현하기

4) Header 영역 구조 꾸미기

(1) Header 영역 구조 (HTML)

```
</li>
<li>
  <a href="#">MENU-3</a>
  <ul class="sub">
    <li><a href="#">SubMenu-1</a></li>
    <li><a href="#">SubMenu-2</a></li>
    <li><a href="#">SubMenu-3</a></li>
    <li><a href="#">SubMenu-4</a></li>
  </ul>
</li>
<li>
  <a href="#">MENU-4</a>
  <ul class="sub">
    <li><a href="#">SubMenu-1</a></li>
    <li><a href="#">SubMenu-2</a></li>
    <li><a href="#">SubMenu-3</a></li>
    <li><a href="#">SubMenu-4</a></li>
  </ul>
</li>
</ul>
</nav>
</header>
```

2/2

→ 메뉴바의 메뉴 1, 메뉴 2, 메뉴 3, 메뉴 4를 구성

→ 각 메뉴의 소분류

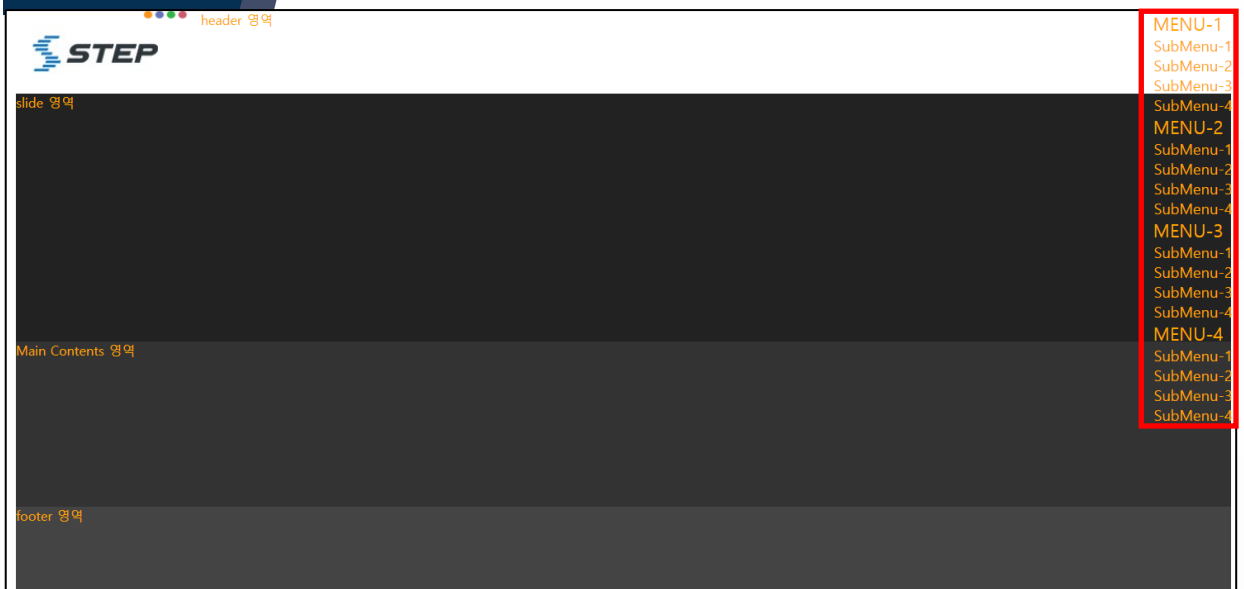


2. 레이아웃 구현하기

4) Header 영역 구조 꾸미기

(1) Header 영역 구조 (HTML)

실행 결과



2. 레이아웃 구현하기

5) 메뉴 영역 꾸미기

(1) 메뉴 영역 (CSS)

```
.wrap {  
  width: 1200px; margin: 0 auto;  
}  
header {  
  width: 100%;  
  height: 100px;  
  background: #fff;  
}  
header h1 {  
  float: left;  
  width: 100px;  
  padding-top: 30px;  
}  
header nav {  
  float: right;  
  width: 700px;  
  padding-top: 30px;  
}  
header nav ul.gnb {  
  width: 100%;  
  text-align: center;  
  font-size: 18px;  
}  
header nav ul.gnb > li {  
  width: 25%; float: left;  
}  
header nav ul.gnb > li > a {  
  font-size: 22px;  
}
```

```
<ul class="gnb" clearfix>
```

→ wrap : 전체 콘텐츠를 감싸는 영역

→ header h1 : 제목에 대한 스타일

→ clearfix : CSS 초기화

5) 메뉴 영역 꾸미기

(1) 메뉴 영역 (CSS)

STEP

header 영역

slide 영역

Main Contents 영역

footer 영역

MENU-1	MENU-2	MENU-3	MENU-4
SubMenu-1	SubMenu-1	SubMenu-1	SubMenu-1
SubMenu-2	SubMenu-2	SubMenu-2	SubMenu-2
SubMenu-3	SubMenu-3	SubMenu-3	SubMenu-3
SubMenu-4	SubMenu-4	SubMenu-4	SubMenu-4

2. 레이아웃 구현하기

5) 메뉴 영역 꾸미기

(1) 메뉴 영역 (CSS)

```
header nav ul.gnb {
  width: 100%;
  text-align: center;
  font-size: 18px;
  box-sizing: border-box;
  border: 1px solid #aaa;
  background: #fff;
}
header nav ul.gnb > li {
  width: 25%;
  float: left;
}
header nav ul.gnb > li > a {
  font-size: 22px;
  display: block;
  width: 100%;
}
header nav ul.gnb > li:hover > a {
  background: #000080;
  color: #fff;
}
header nav ul.gnb > li > ul.sub {
  width: 100%;
  line-height: 40px;
}
header nav ul.gnb > li > ul.sub > li {
  width: 100%;
  height: 40px;
}
header nav ul.gnb > li > ul.sub > li > a {
  display: block;
  width: 100%;
  height: 100%;
}
header nav ul.gnb > li > ul.sub > li:hover {
  background: #000080;
  color: #fff;
}
```

→ 마우스가 콘텐츠 위로 올라갔을 때 효과를 주는 스타일

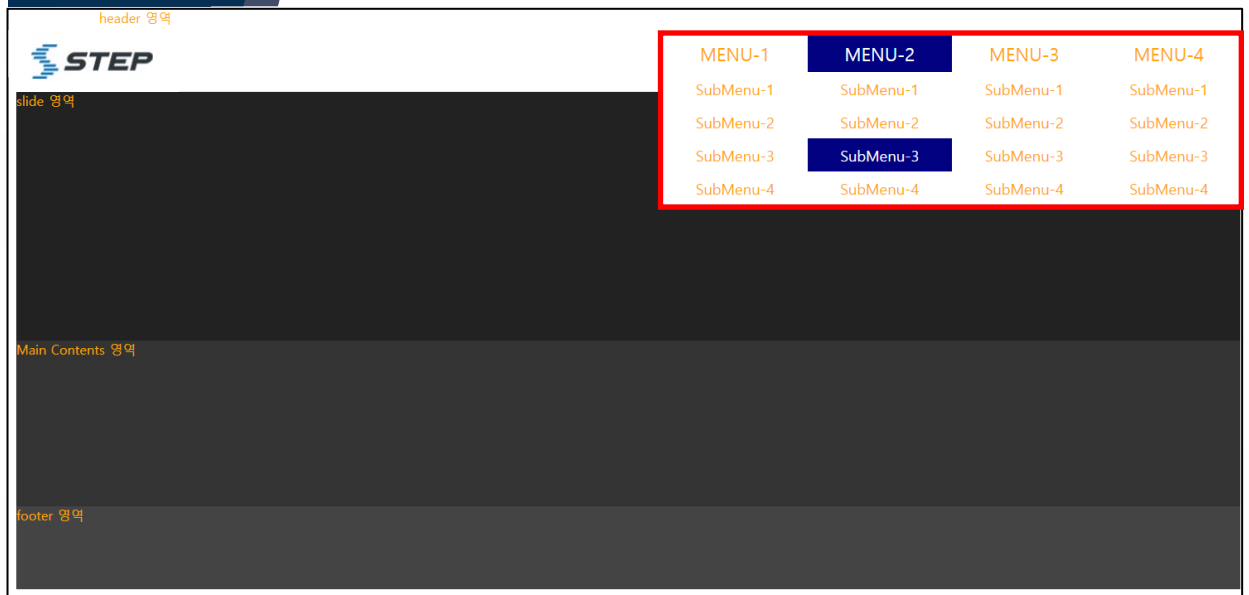
2. 레이아웃 구현하기

5) 메뉴 영역 꾸미기

(1) 메뉴 영역 (CSS)

실행 결과 - 2

header 영역



slide 영역

Main Contents 영역

footer 영역

MENU-1	MENU-2	MENU-3	MENU-4
SubMenu-1	SubMenu-1	SubMenu-1	SubMenu-1
SubMenu-2	SubMenu-2	SubMenu-2	SubMenu-2
SubMenu-3	SubMenu-3	SubMenu-3	SubMenu-3
SubMenu-4	SubMenu-4	SubMenu-4	SubMenu-4



1. 와이어프레임

- 와이어프레임(wireframe)의 개념
 - 웹사이트의 회로도 (schematic) 또는 청사진 (blueprint)라고도 불리우며, 웹사이트의 골격을 나타내는 시각적 가이드
- 와이어프레임(wireframe)의 구성요소
 - Wrap : 전체 박스
 - Header : 헤더
 - Slide : 이미지 슬라이드
 - Contents : 콘텐츠
 - Footer : 푸터
 - Tab Menu : 탭메뉴
 - Pop-up : 팝업창
- 와이어프레임(wireframe)의 종류
 - 가로형과 세로형이 있음



2. 레이아웃 구현하기

- HTML 기본구성 입력과 CSS Reset하기
 - 웹에디터의 Emmet 기능을 활용하여 html 기본 구성을 완성함
 - HTML 태그에 대한 기본 CSS요소를 초기화
- HTML 기본 레이아웃 구현하기
 - header, slide, Main Contents, footer에 대한 HTML 영역을 구현
- CSS로 스타일 구현하기
 - header, slide, Main Contents, footer에 대한 CSS 스타일을 구현
- Header 영역 구조 꾸미기
 - ul, li, a에 대한 HTML 영역 구조를 제작함
- 메뉴 영역 꾸미기
 - ul, li, a에 대한 CSS 스타일을 제작함