

IT비전학교

YEAR_2024

CONTENTS_프론트엔드 Basic

프론트엔드 Basic 42~43_day



송현진 SongHyunJin
Web Designer + Frontend

Contents

Chapter1. head태그 살펴보기

Chapter2. html 기본구조

Chapter3. 주석

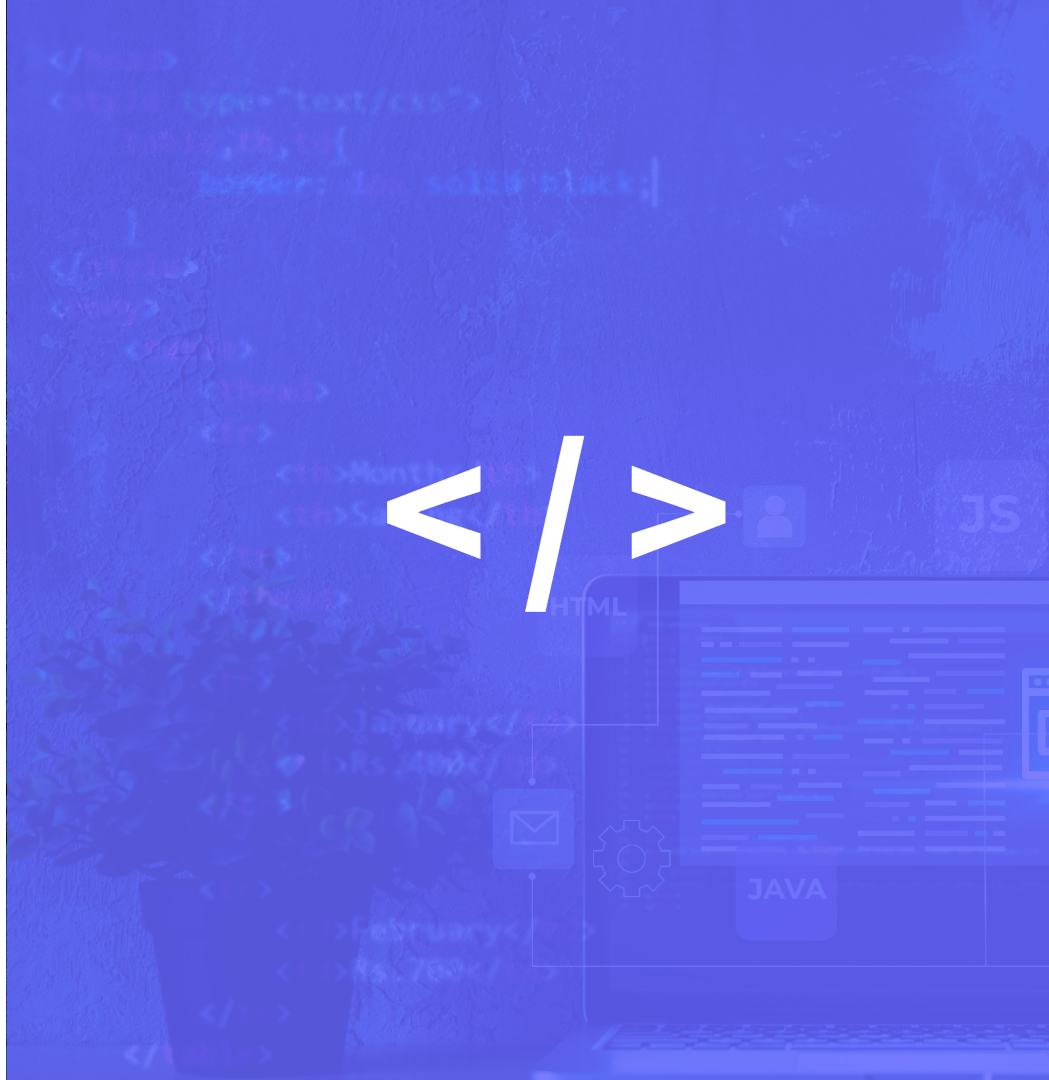
Chapter4. 부모, 자식, 형제 관계

Chapter5. div, p, h2, form, input, button, a

Chapter6. css 기본구조

Chapter7. css 선택자_기본

Chapter8. 로그인페이지 만들기



html 기본구조

- Chapter1 *html* 기본 구조

요소(Element)

<title> Login </title>

시작태그

Open tag

콘텐츠

종료태그

Close tag

태그

웹페이지를 구성하는 텍스트,
이미지,버튼 등의
요소를 정의하는 역할을 하는
html 가장 작은 단위

- Chapter1 *html* 기본 구조

시작태그 *Open tag*

<html lang="ko">

↓
속성명

↓
속성값

</html>

종료태그 *Close tag*

속성

태그에 어떤 의미나 기능을 보충
(옵션 이라고 보면 됨)

태그 없이 단독으로
사용할 수 없음

- Chapter1 *html* 기본 구조

 : 줄바꿈 태그

콘텐츠가 없는 태그

head태그 살펴보기

- Chapter2 head태그 살펴보기

```
<> index.html > ...
1  <!DOCTYPE html>
2  <html>
3      <head>
4          
5      </head>
6      <body>
7
8      </body>
9  </html>
```

< head > </head>

html문서의 **메타데이터**를 정의하는 영역입니다.

메타데이터란

HTML 문서의 정보로 직접 노출 되지는 않습니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

< meta charset = "UTF-8" >

meta 태그는 정보를 의미합니다.(웹페이지의 내용, 키워드 같은 정보들을 검색엔진이나 브라우저에게 제공합니다.)

charset(Character Set)속성은 **문자 인코딩 방식**을 지정하는 HTML의 속성입니다.

문자 인코딩 (Encoding)은 기호나 문자들을 컴퓨터가 사용할 수 있는 신호로 만드는 것이며,

대표적으로 웹에서는 **UTF-8 사용**을 권장합니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

< **meta name = "viewport" content = "width=device-width, intial-scale=1.0" >**

meta 태그는 정보를 의미합니다.

name 속성은 **정보의 종류**를 나타냅니다. /viewport는 웹페이지의 영역이고 (모바일을 만들 시 활용되는 것입니다)

content 속성은 **정보의 값**을 나타냅니다. /가로 사이즈는 기기의 가로사이즈에 맞추고, 기본 크기는 1.0배수로 출력한다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

< meta name = "author" content = "HYUNJIN" >

meta 태그는 정보를 의미합니다. 이 정보가 무엇인지를 **name 속성**에서 나타냅니다.

name= "**author**"은 **제작자** 라는 뜻으로 웹페이지가 어떤 사람에 의해 제작 되는지 나타내줍니다.

값은 content = "HYUNJIN"의 속성 값에 누구누구~라고 나타냅니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

< **title** >Document< / **title** >



title 태그는 HTML문서의 제목을 나타냅니다.

웹브라우저의 탭에 표시됩니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

<link rel = "stylesheet" href = "./style.css" >

link 태그는 HTML문서의 외부 문서(거의 css 파일)를 연결할 때 사용합니다.

rel (Relationship) 은 HTML 문서와의 어떤 관계를 나타내는지 명시하는 **속성(Attribute)**입니다.

href 는 가져올 문서의 경로를 나타냅니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <style>
    h2{
      color: blueviolet
    }
  </style>
  <script src="./main.js"></script>
</head>
```

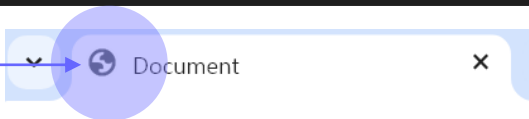
1. CSS를 파일로 만들어서 link로 가져옴

2. CSS를 HTML에다가 직접적으로 작성

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

<link rel = "icon" href = "./favicon.png" >



link 태그는 HTML문서의 외부 문서(거의 css 파일)를 연결할 때 사용합니다.

rel (Relationship) 은 HTML 문서와의 어떤 관계를 나타내는지 명시하는 **속성(Attribute)**입니다.

href 는 가져올 문서의 경로를 나타냅니다.

※HTML 문서의 대표 아이콘인 **Favicon(Favorite Icon)**을 불러올 때 사용합니다.

favicon.ico / favicon.png 파일을 가져 오는 것이 좋습니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
</head>
```

<script src = "/main.js" ></ script >

script 태그는 HTML문서의 자바스크립트 파일을 가져올 때 사용합니다.

src (Source) 은 외부의 있는 파일을 가져올 때 사용하는 **속성(Attribute)**입니다.

- Chapter2 head태그 살펴보기

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="./style.css">
  <script src="./main.js"></script>
  <script>
    console.log('hello');
  </script>
</head>
```

1. JS를 파일로 만들어서 link로 가져옴

2. JS를 HTML에다가 직접적으로 작성

주석

- Chapter3 주석

html 주석

<!-- 주식내용 -->

css주석

/* 주식내용 */

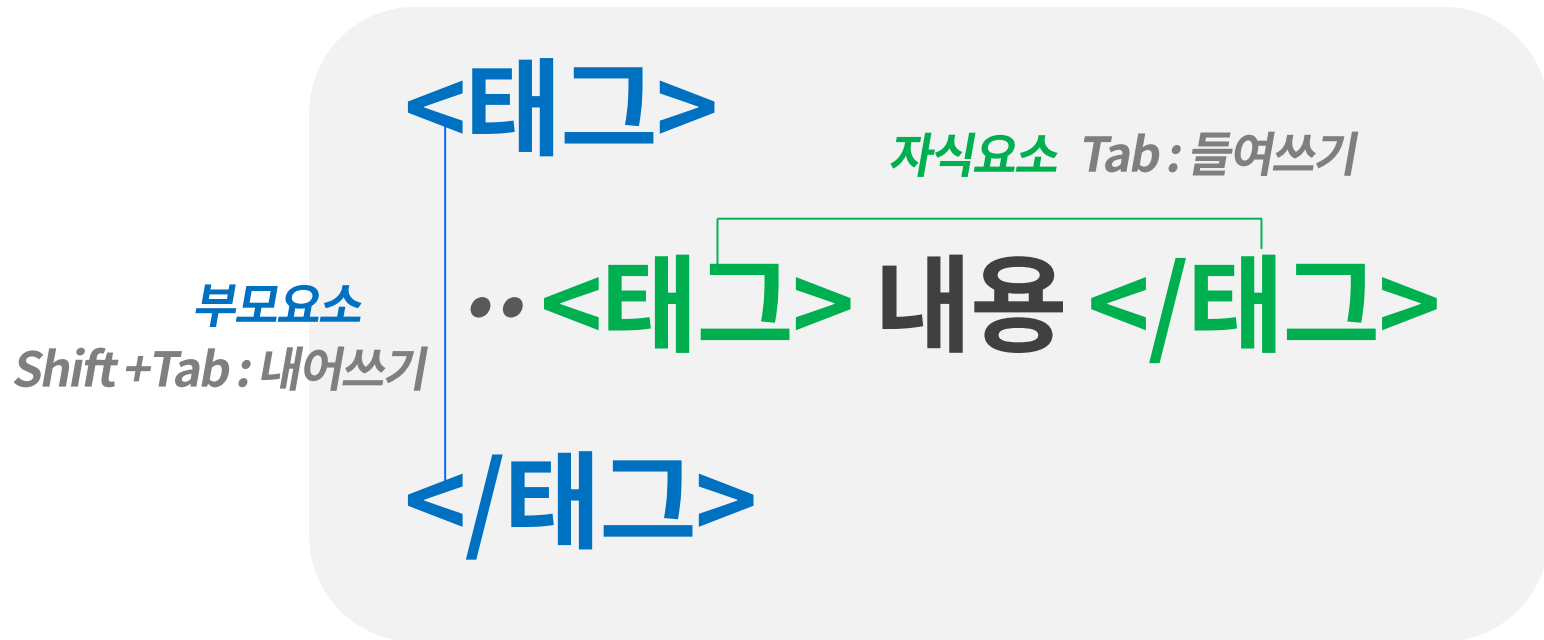
* 단축키: **Ctrl + /** (슬래쉬)

js주석

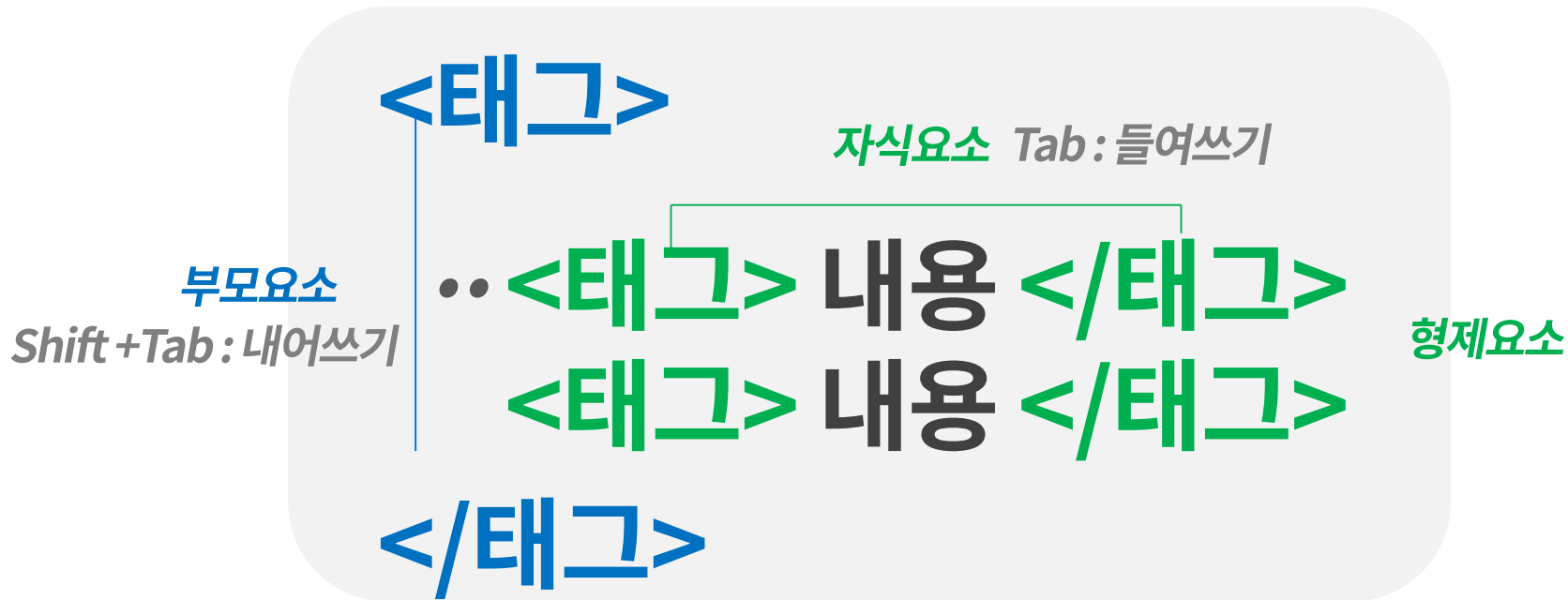
// 주식내용

부모, 자식, 형제관계

- Chapter4 부모, 자식, 형제 관계



- Chapter4 부모, 자식, 형제 관계



- Chapter4 부모, 자식, 형제 관계

<태그>

부모요소

<태그>

<태그> 내용 </태그>

</태그>

</태그>

- Chapter4 부모, 자식, 형제 관계

<태그>

<태그>

상위(조상)요소

<태그> 내용 </태그>

</태그>

</태그>

- Chapter4 부모, 자식, 형제 관계

<태그>

자식요소

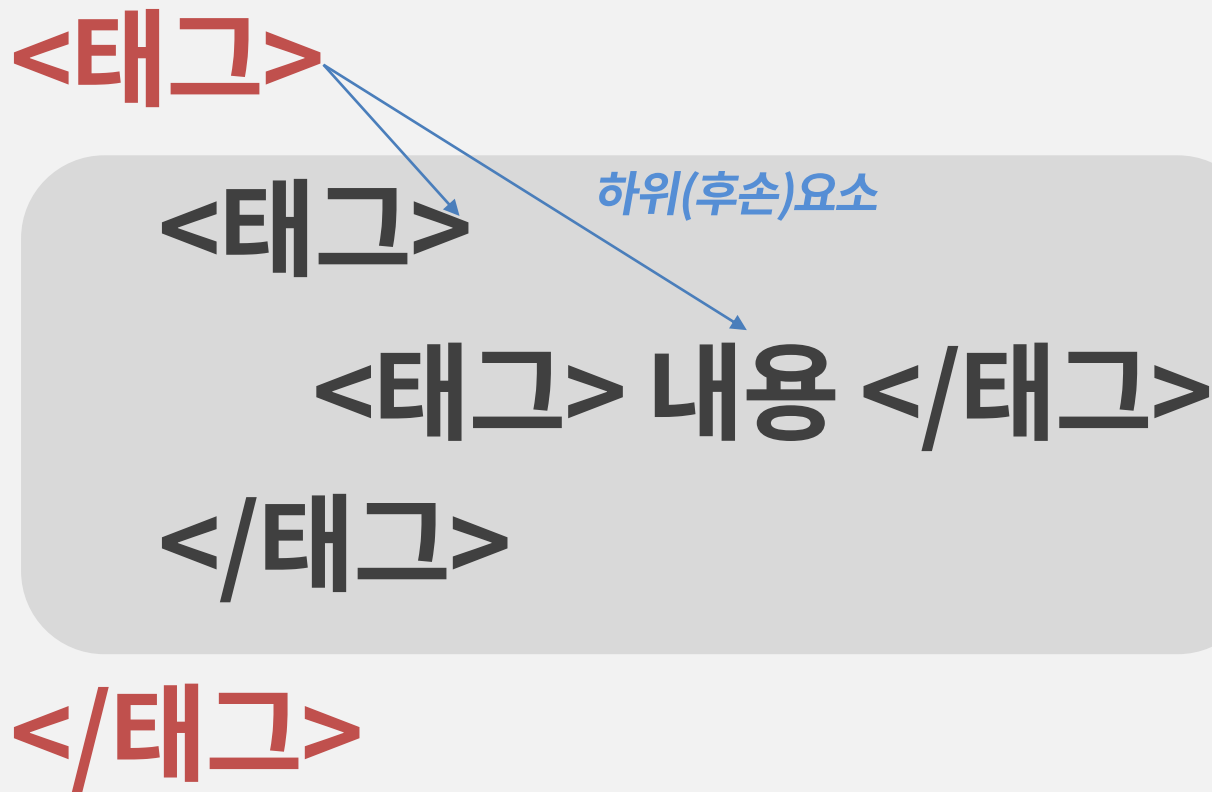
<태그>

<태그> 내용 </태그>

</태그>

</태그>

- Chapter4 부모, 자식, 형제 관계



태그: div, p, hn, form, input, button, a

- Chapter5 **블록(Block)**요소

div(Divide)

div태그는 "영역"을 나눌 때 씁니다.

<div> </div>

- Chapter5 **블록(Block)요소**

P(Paragraph)

Paragraph 태그는 "문장"을 나타내는 태그입니다.

<p> </p>

- Chapter5 **블록(Block)요소**

h1~h6 (Heading)

Heading 태그는 "제목"을 나타내는 태그입니다.

<h1> </h1>
~
<h6> </h6>

- Chapter5 **블록(Block)요소**

form

form은 "웹페이지의 입력 양식" 을 나타냅니다. 로그인 창이나 회원가입 창을 만듭니다.

form 태그는 전체 양식을 의미하며 화면에는 보이지 않습니다.

백엔드 코드와 함께 form을 사용하기 위해서 **속성**들이 필요합니다.

양식을 입력하기 위해서는 **input**태그 등을 사용합니다.

<form></form>

■ Chapter5 *form* 속성

속성명	설명
name	폼의 이름
action	폼 데이터가 전송되는 백엔드 URL
method	폼 전송 방식(GET / POST)/정보를 주고 받는 방식
...	

GET: 전송된 데이터는 URL의 파라미터로서 주소창에 표시됩니다.

POST: 전송된 데이터는 URL에 표시되지 않습니다. 긴 데이터를 보낼 때 적합합니다.

GET 방식

예시: "너 오늘 점심 뭐 먹었어?"라고 물어보는 것처럼, **웹사이트에 정보를 요청**하는 방법

특징:

- 1.편지 봉투 겉면에 쓰기:** 정보를 주소(URL)에 붙여서 보내. 마치 편지 봉투 겉면에 질문을 적어 보내는 것과 같아서 **누구나 볼 수 있음**
- 2.짧은 내용만:** 주소에 붙일 수 있는 정보가 많지 않기 때문에, **간단한 정보를 요청할 때 주로 사용함**

POST 방식

예시: "비밀 편지를 보내는 것처럼" 웹사이트에 정보를 안전하게 전달하는 방법

특징:

- 1. 편지 봉투 안에 쓰기:** 정보를 웹사이트의 주소(URL)에 보이지 않게 숨겨서 보냄.
마치 비밀 편지를 봉투 안에 넣어 보내는 것처럼 하면 **다른 사람이 내용을 쉽게 볼 수 없음**
- 2. 많은 내용 보내기:** 편지 봉투 안에 많은 정보를 담을 수 있기 때문에
많은 정보를 전달할 때 주로 사용함

- Chapter5 **인라인(inline)요소**

input

input 는 "사용자로 부터 정보를 받을 때"사용합니다.
form 태그 내부에 사용합니다.
input태그는 빈 태그라서 **속성**만을 포함하고 있습니다.

<input>

- Chapter5 **인라인(inline)요소**

button

button 태그는 "클릭할 수 있는 버튼"을 만들 수 있습니다.

button 요소는 텍스트나 콘텐츠를 삽입할 수 있으나,

input 요소의 버튼은 삽입할 수 없습니다.

<button></button>

- Chapter5 **인라인(inline)요소**

a[anchor]

a 태그는 "HTML에서 하이퍼링크를 생성하는 데" 사용됩니다.

다른 웹페이지나 같은 페이지의 특정부분으로 스크롤하도록 할 수 있습니다.

이 태그는 링크의 대상을 지정하는 속성인 **href**를 포함합니다.

<a>

CSS기본 문법

- Chapter6 CSS기본구조

선택자 { 속성:값; }

- Chapter6 CSS기본구조

CSS를 적용할 대상

선택자 { 속성:값; }

- Chapter6 CSS기본구조

선택자 { 속성:값; }

CSS의 종류

- Chapter6 CSS기본구조

선택자 { 속성:값; }

CSS의 값

- Chapter6 CSS기본구조

선택자 { 속성:값; }
은 이다

- Chapter6 CSS기본구조

선택자 { 속성:값; }

시작

끝

- Chapter6 CSS기/본구조

선택자 { 속성:값; 속성:값; }

- Chapter6 CSS기본구조

```
선택자 { 속성:값;  
        속성:값;  
}
```

- Chapter6 CSS기본구조

```
div { margin: 50px;  
      color: blue;  
}
```

CSS 선택자_기본

CSS 선택자

기본

복합

가상 클래스

가상 요소

속성

CSS선택자_기본

- Chapter7 CSS선택자_기본

기본

* (에스터리스크)전체 선택자 (Universal Selector)

```
* {  
    color: white;  
    padding: 50px 20px;  
}
```

- Chapter7 CSS선택자_기본

기본

태그 선택자 (Universal Selector)

```
div {  
  color: white;  
  padding: 50px 20px;  
}
```

- Chapter7 CSS선택자_기본

기본

클래스 선택자 (Class Selector)

```
.btn {  
    color: white;  
    padding: 50px 20px;  
}
```

- Chapter7 CSS선택자_기본

기본

아이디 선택자 (ID Selector)

```
#btn {  
    color: white;  
    padding: 50px 20px;  
}
```

로그인 페이지 만들기

End

고생하셨습니다~!