

# 소개

html, css, java script

- **HTML , CSS, JavaScript** 기본적인 문법 학습.
- **HTML**로 **WEB의 기본 구조**를 학습하고, **화면 설계**를 함.
- **CSS**로 **HTML로 만들어진 문서를 화면 구현**함. ( , , , )
- **JavaScript**로 **웹 사이트를 interactive**하게 제작하고, 사용자 참여를 유도함. ( , , 가 . )
- **프런트 엔드 개발자의 기초** 과목.
- 앞으로 진행할 프로젝트의 가장 기본적인 과목.

# 교육 개요

- 1단원: HTML 기본 문법
- 2단원: CSS 기본 문법
- 3단원: JavaScript 기본 문법 JQuery 기본 활용
- 4단원: ~~카카오 오픈~~ 활용한 화면 설계

# 1단원 HTML 기본 문법 공부 : 목표

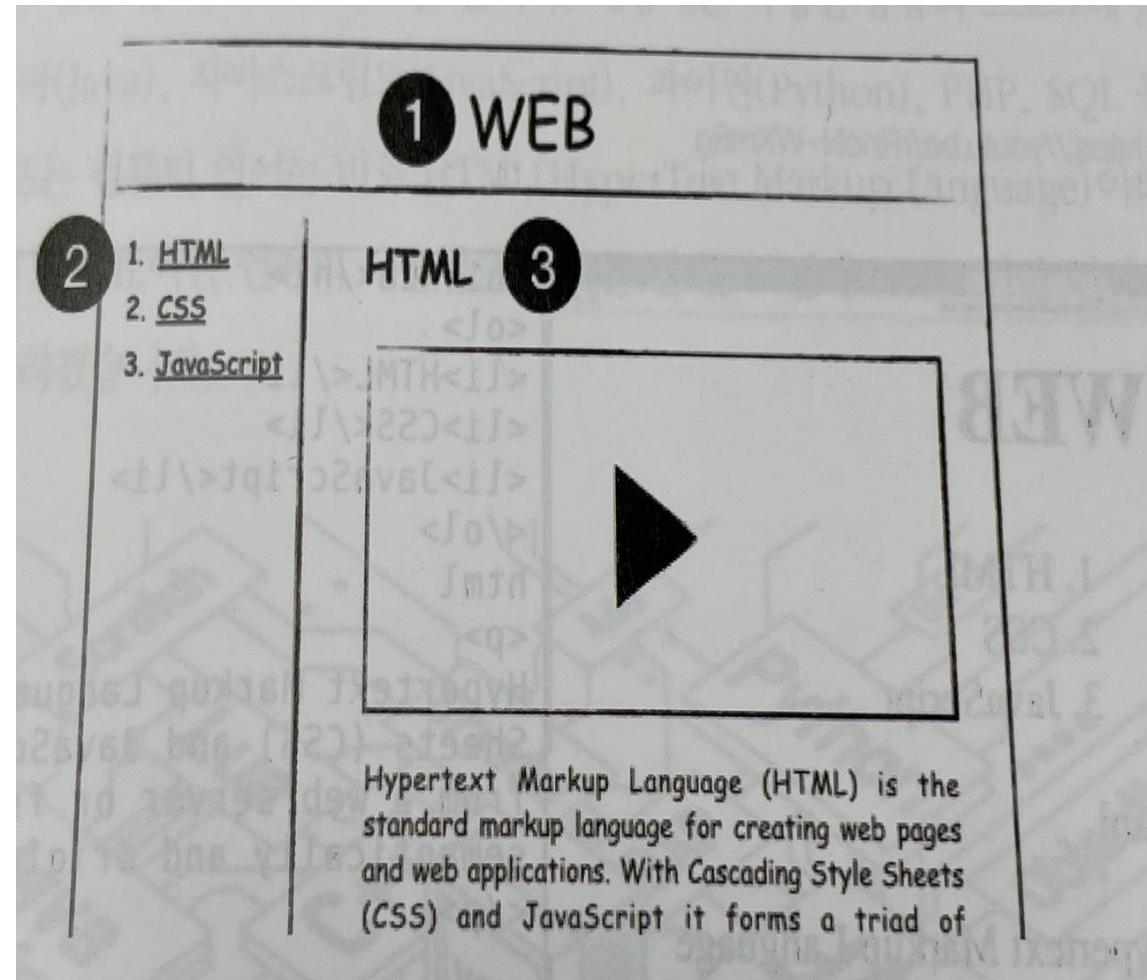
- HTML 수업에서 만들게 될 예제를 간략하게 기획하고 밑그림 그리기
- HTML 코딩 실습을 위한 환경 구축하기 vs code
- 웹 페이지를 구성하는 기본적인 요소와 문법을 이해하기
- 통계에 기반한 주요 태그 공부하기
- 간단한 예제로 웹사이트 완성해 보기
- 서버와 클라이언트 구조 이해하기
- 웹 호스팅 및 웹 서버 간단 구축해 보기
- 무료 도메인을 이용한 웹 사이트 접속하기

(NextJS, Vercel)

)

# 1단원-1: 기획

- 뭔가를 만들기 전에 무엇을 만들 것인가를 설계하거나 그려보는 과정을 '기획'이라 합니다.
- 이런 일을 하는 분들은 현업에서 '기획자'라고 합니다.
- 코딩 수업과 내용을 잘 정리 정돈할 수 있는 웹사이트 그림을 한번 그려봅니다.



# 1단원-2: 코딩과 HTML (1)

기획을 마치고 구현할 차례.

실행 결과 화면 css가

예제 코드

```
<h1>WEB</h1>
<ol>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ol>
html
```

```
<p>
HTML -> Hypertext Markup Language
CSS -> Cascading Style Sheets
JavaScript
</p>
```

WEB

1. HTML

2. CSS

3. JavaScript

html

HTML -> Hypertext Markup Language CSS -> Cascading Style Sheets JavaScript

# 1단원-2: 코딩과 HTML (2)

사람이 하는 일, 원인을 가리키는 여러 가지 표현이 있습니다.

부호 또는 신호라는 의미에서 **코드(code)**, 원천이라는 뜻의 **소스(source)**

기계도 이해할 수 있고 사람도 이해할 수 있는 공통의 약속이라는 의미에서 언어 (language), 그 중에서 프로그래밍 언어라는 표현을 사용합니다.

기계가 하는 일, 결과를 가리키는 여러 표현이 있는데

**애플리케이션(application)**, 응용 프로그램, 줄여서 **앱(app)** 또는 **프로그램(program)**

좀 더 구체적으로 **웹 페이지(web page)**, 웹 페이지가 모여 있다면 **웹사이트(website)**라는 여러 가지 표현이 있습니다.

# 1단원-2: 코딩과 HTML (3)

원인인 코드를 통해 결과를 만든다는 것이 코딩을 이해하는 핵심입니다.

우리가 하고자 하는 일은 웹 페이지를 만드는 겁니다. 여러 가지 언어가 있습니다.  
그중에서 웹 페이지를 만드는 컴퓨터 언어는 바로 HTML(Hypertext Markup Language)  
이라는 이름의 언어입니다.

HTML은 두 가지 측면에서 좋습니다. 1) 쉽다. 2) 중요하다.

- 1) 문법을 완전히 다 배우는데 10분도 채 걸리지 않습니다.
- 2) 하루에 웹 페이지를 100번 이상 보지 않나요?

) <h1></h1>

인류가 생산하는 거의 모든 디지털 정보가 담기는 가장 거대한 그릇은  
바로 웹 페이지입니다. 그래서, 쉬우면서 중요하다.

퍼블릭 도메인(public domain) 저작권이 존재하지 않는다.

1990년 이후로 살아남은 것도 기업이 독점하지 않았기 때문입니다.

# 1단원-3: 코드 실습 환경 준비(1)

## 준비물

: 웹 브라우저, 코드를 작성할 프로그램 편집 프로그램 에디터(editor)  
크롬 브라우저, vscode를 설치합니다.

(크롬 설치 주소 :[https://www.google.com/intl/ko\\_kr/chrome/](https://www.google.com/intl/ko_kr/chrome/))

## vscode 설치, 확장에서 라이브 서버 설치.

- 1) <https://code.visualstudio.com/download> 접속
- 2) 윈도우즈 환경에서 해당 비트 다운로드
- 3) **한국어 언어팩** 설치를 묻는 오른쪽 하단에 물어봄 -> 설치함.

혹시, 언어 설정이 안된다면 아래 블로그 참고.

<https://www.lainyzine.com/ko/article/how-to-change-language-setting-in-visual-studio-code/>

# 1단원-3: 코드 실습 환경 준비(2)

## 라이브 서버 설치.

왼쪽 아래에서 2번째 확장 아이콘 클릭-> live server 입력 ->설치.

### **라이브 서버 기본 브라우저 크롬으로 변경 방법**

1. 파일 -> 기본 설정-> 설정 들어가기.
2. 확장 -> Live Server Config 들어가기.
3. Settings : Custom Browser에서 null을 chrome으로 변경.

각자 바탕화면에 web 폴더 만들고.

'1.html' 파일 만들기.

hello web 입력해서, 결과 화면 브라우저로 확인해 보기.

# 1단원-4: 기본 문법과 태그 <strong>, <u>

'1.html' 파일에 텍스트 입력.

Hypertext Markup Language(HTML) is the standard markup language  
for creating web pages and web applications. -> ctrl + s 저장.

세상에서 가장 쉬운 문법 태그(Tag)입니다.

creating web pages라는 세 단어를 진하게 표현하는 것.

<strong> 태그 : 진하게 표시

<strong>creating web pages</strong>

'web'이라는 텍스트 밑에다 밑줄 긋기

underline입니다. 첫 번째 글자는 'u'

<u>web</u>

열리는 태그, 뒤에 있는 닫히는 태그. 구분하기 위해 '/' 닫히는 태그에 표시.

# 1단원-5: 혁명적인 변화 <h1> ~ <h6>(1)

난이도가 높다고 해서, 꼭 중요한 것만은 아니다. 난이도가 쉽고, 중요도가 높은 것도 있다.

**->기초가 중요.**

국제 민간표준화 기구의 홈페이지 -> <https://www.w3.org/> : 웹을 만드는 곳.

여기 우 클릭 -> **페이지 소스 보기** 하면 해당 페이지 소스를 볼 수 있습니다.  
우리는 이해할 수 없는 패턴이 나오면 어려워함.

여기에서 '<h1>'이라는 태그 검색해 보기. -> ctrl + f 단축키 -> h1 검색하면  
2개 나옵니다. 그 중에서 일부분. -> <h1>W3C</h1> 만 볼게요.

```
<h1 class="logo">
<a tabindex="2" accesskey="1" href="/">

</a> <span class="alt-logo">W3C</span>
</h1>
```

# 1단원-5: 혁명적인 변화 <h1> ~ <h6>(2)

여기서 <**h1**>W3C</**h1**>-> 이제 태그라는 것을 알았고, 그 태그가 무엇인지 알면 됩니다.  
h1이 어떤 태그 인지 검색해 보기. -> 검색어 : HTML h1 tag

[https://www.w3schools.com/tags/tag\\_hn.asp](https://www.w3schools.com/tags/tag_hn.asp) : 이 사이트 참고.

해당 예제 에디터에서 가져와서 보기. -> 현재, '1.html'파일에서 작업 중.

예제)

```
<h1>W3C</h1>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>
```

# 1단원-5: 혁명적인 변화 <h1> ~ <h6>(3)

**h 뒤에는 숫자가 붙고, 숫자가 올라갈수록 글씨가 작아지는구나,**

h와 숫자가 나오는 태그를 쓴다면 일반적인 텍스트보다는 글씨가 더 두껍구나,  
줄바꿈이 되는구나.

정의를 보면, 'HTML headings' , **heading** 은 제목이라는 뜻.-> 참고로, 6까지만 있습니다.

HTML 제목 태그 사용 해보기.

예제) 추가하기

<h1>Hypertext</h1>

Hypertext Markup Language(HTML) is the standard markup language for **creating  
<u>web</u> pages</strong>** and web applications.

# 1단원-6: 통계에 기반한 학습 (1)

현재 HTML에는 150개 이상의 태그가 있습니다. 그럼 태그를 다 외워야 할까?

태그가 무엇인지 안다면 모든 태그를 알고 있는 것과 다름없습니다.

왜냐하면, 1분이면 찾을 수 있기 때문입니다.

기본적으로 밑천으로 가지고 있어야 할 태그가 무엇인가를 스스로 판단할 수 있도록 만들어 주는 수업.

해당 통계에서 자주 사용하는 태그 목록 확인해 보기.

검색어 : advancedwebranking html tag

<https://www.advancedwebranking.com/seo/html-study/>

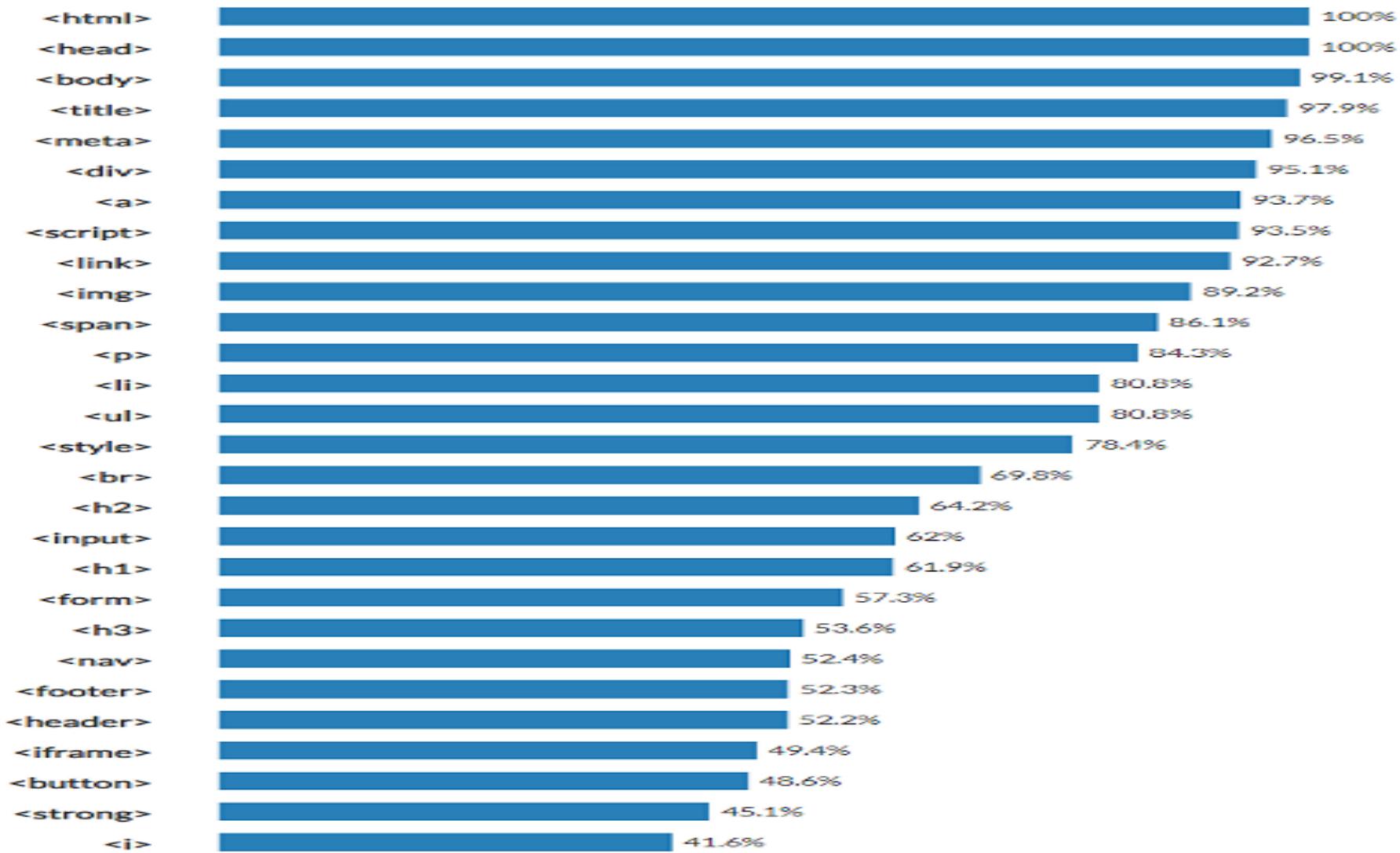
<head> 태그는 뒤에서 배우기. <h1> 태그 점유 빈도 :61.9% (22/4/4 기준)

참고 검색어 : frequency list of english words

해당 영어 단어 빈도수 결과가 나옴.

# 1단원-6: 통계에 기반한 학습 (2)

The thirty-two elements used on most pages, ordered by appearance frequency:



# 1단원-7: 줄 바꿈 : <br> vs <p> (1)

css 기술 깜짝 출연. 뒤에서 배움.

예제)

```
<h1>HTML</h1>
```

Hypertext Markup Language(HTML) is the standard markup language for **creating web** pages and web applications.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

브라우저로 결과 확인해 보기.

# 1단원-7: 줄 바꿈 : <br> vs <p> (2)

검색어 : HTML new line tag

**br 뜻은 break -> <br> 새로운 줄을 표현하는 태그**

<br><br> 2번 사용하면 : 단락처럼 보임. -> 상당히 자주 쓰는 태그 69.8% 정도.

열리는 태그만 있고, 닫히는 태그는 없음.

줄 바꿈이라는 시각적인 의미만 가지고 있기 때문에 뭔가를 감쌀 필요가 없음.

그래서, 닫히는 태그는 없음.

**<p> 태그 -> 단락을 표현할 때 쓰기 위한 태그를 만들어 놓았음.**

검색어 : html paragraph tag

paragraph는 단락이라는 의미. -> 앞 글자 p를 띤 것임.

# 1단원-7: 줄 바꿈 : <br> vs <p> (3)

<br> 태그 대신 <p> 태그로 변경해 보기.

여기서 <p> 태그가 더 적절한 태그입니다.

왜냐하면, <p> 태그는 이 웹 페이지를 좀 더 정보로써 가치 있게 만들기 때문입니다.

그 단락을 의미론적으로 표현할 수 있기 때문입니다.

그에 반해 <br> 태그는 단지 줄 바꿈일 뿐입니다.

단점, 정해진 여백만큼만 벌어짐. 자유도가 떨어짐. 하지만, css 기술이 있음.

css 에서 태그와 태그 사이의 여백을 나타내는 margin 이라는 속성이 있음.

두 번째, <p> 태그에 위쪽 여백을 주고 싶은 상태이므로

예제)

<p style="margin-top: 40px;"> css 가 ( style="margin-top: 40px;" )

시각적인 효과는 css를 통해 훨씬 더

의미 있는 태그를 사용하는 것이 더 좋은 방법임. -><p> 태그 빈도 : 84.3%

# 1단원-8: HTML이 중요한 이유 (1)

span 가 , h3 가

```
<span style="font-size: 24px;">coding</span>
<h3>coding</h3> -> 비교하기.
```

검색 엔진은 웹 페이지에 담긴 HTML 코드를 분석합니다.

태그에 근거해서 정리 정돈합니다.

제목이 "coding" 인 사이트와 그냥 시각적으로 제목처럼 보이는 "coding"이라는 정보를 가진 사이트 중에서 누구의 편을 들어줄까요?

당연히 제목이 "coding" 인 사이트의 편을 들어 줄 겁니다.

제목이 "coding"인 사이트가 좀 더 위쪽에 올라오고,

시각적으로 제목처럼 보이는 사이트는 100페이지 뒤에 나타남.

## 1단원-8: HTML이 중요한 이유 (2)

웹이 가지고 있는 본래의 의미나 정보를 탄탄히 하는 게 중요.  
HTML의 의미에 맞게 정확하게 사용한다는 것은  
비즈니스 측면에서 정말 생명줄과 같은 중요한 문제임.

HTML이 중요한 또 하나의 이유. 접근성.  
웹은 모든 운영체제에서 동작하고  
웹 페이지의 소스코드는 누구나 볼 수 있고,  
웹은 저작권이 없는 순수한 공공재입니다.

웹을 다른 기술과 구별되는 특별한 기술로 만드는 것이라고 생각합니다.  
이것을 개방성 또는 접근성 (accessibility) 라고 합니다.

# 1단원-8: HTML이 중요한 이유 (3)

웹이 중요하게 생각하는 접근성 중 하는 신체적인 장애가 있는 분들도 정 보로부터 소외 당하지 않도록 노력해야 한다는 것.

시각장애가 있는 분들은 시작적인 정보를 청각화해서 얻게 되는데, 스크린리더 같은 프로그램이나 각종 보조 장치를 이용해 정보를 접하게 됩니다. 그런데, 웹 페이지를 만들 때 예쁘게 만들고 싶다는 생각에 문자까지 이미지로 만든다면 시각 장애가 있는 분들에게는 존재하지 않는 정보가 됩니다.

그래서, 가장 중요한 첫걸음은 HTML의 태그를 정확히 알고 의미에 맞게 사용하는 것입니다. 이렇게 한다면 우리도 모르게 누군가에게 큰 도움을 주고 있는 것입니다.

이처럼 HTML은 비즈니스 측면뿐만 아니라 휴머니즘 측면에서 매우 중요한 기술입니다.

# 1단원-9: 최후의 문법 속성과 <img> 태그 (1)

이번 시간에는 속성이라 하는 태그의 심화된 문법을 배울 것입니다..  
이것까지 배운다면 HTML 문법의 모든 것을 마스터했다고 할 수 있습니다.  
아주 인기 있는 태그를 함께 살펴보겠습니다.

## 사진을 넣는 방법.

이미지를 웹 페이지에 포함시킬 때 사용하는 태그의 이름은 <img>입니다.  
image의 줄임말입니다.

<img> 태그 이름만으로 정보가 부족.

src 속성 추가

<img src="">

source를 줄여서 src로 쓰기로 함.

# 1단원-9: 최후의 문법 속성과 <img> 태그 (2)

고품질 이미지 저작권에 구애 받지 않고 사용하는 사이트

<https://unsplash.com/> -> coding으로 검색해 보기.

예제 이미지 받음. -> p53 이미지와 동일.

 설정하면, 화면이 너무 큼.

이미지 크기를 조절해서 자동으로 웹 브라우저의 크기에 맞게 이미지의 크기를 바꿀 수 있음.

width="100%" -> 추가하기.



src="coding.jpg" 와 width="100%" 위치는 상관 없음.

<img> 태그 빈도는 : 89.2%

src="coding.jpg" 와 width="100%" 를 속성(attribute)라고 합니다.

# 1단원-10: 부모 자식과 목록 (1)

```
<parent>  
    <child></child>
```

```
</parent>
```

태그가 서로 포함 관계에 있을 때 포함하고 있는 태그를 부모 태그,  
포함된 태그를 자식 태그라 합니다. -> 부모, 자식 관계에 해당하는 태그를 살펴보기.

예제) 목차 만들기.

HTML

CSS

JavaScript

이렇게 하면 줄바꿈이 안되니, br 태그로 줄바꿈 해보기.

HTML<br>

CSS<br>

JavaScript<br>

# 1단원-10: 부모 자식과 목록 (2)

이번에는, **목차 또는 목록**인데, 영어로 **List** 라함.

앞에 두 글자를 따서 **<li>** 태그를 사용하면 됨.

br 지우고 li 로 변경하기.

<li>HTML</li>

<li>CSS</li>

<li>JavaScript</li>

**팁) 다중커서. -> alt 키 잘 활용하기.**

**팁) 들여쓰기 -> Tab 키를 누르면 들여쓰기.**

구분하기 위해 띄어쓰기를 하고 싶습니다. 목록 부분에 br 을 써도 구분되겠지만,  
**<li> 태그의 부모인 <ul>이란 태그입니다.**

# 1단원-10: 부모 자식과 목록 (3)

예제)

```
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>
<ul>
  <li>JAVA</li>
  <li>JSP</li>
  <li>Spring</li>
</ul>
```

## 극단적인 예.

목록이 3개가 아닌 1억개라 생각해보기.

1,2,3 숫자를 1억번 써야 함. 앗.

그런데, 누가와서 첫번째 목록을 지워달라고 하면,...

나머지 2번 부터 숫자를 1억 -1 번이나 고쳐야 함...

그래서, 이 태그가 필요함.

**ul : Unordered List**

**ol : Ordered List**

**ul -> ol 변경.**

**항목 앞에 자동으로 숫자가 매겨지는 것을 확인됨.**

ol -> 빈도는 아쉽게 없고.

ul -> 80.8%

# 1단원-11: 문서의 구조와 슈퍼스타들 (1)

문장이 모이면 -> 페이지  
페이지 모이면 -> 책  
책에는 표지가 있고 표지에는  
제목과 저자가 표기됨.

정보가 많아짐에 따라 정보를 잘 정리 정돈  
하기 위한 체계, 구조라는 것이 필요함.

상위 랭크 슈퍼스타들.  
Html, head, body, title, meta

페이지 제목 설정하기.

<title>WEB1 - html </title>

브라우저로 표시 확인하기.

title 태그를 사용하면 사용자에게 명시적  
으로 알려주고, 검색 엔진과 같은 기계  
들이 정보로써 사용합니다.  
그래서, 안 쓰면 굉장히 손해.

제목 변경해 보기.

<h1>HTML</h1> ->

<h1>HTML 이란 무엇인가?</h1>

한글이 잘 나오지만.

웹 브라우저에게 열 때, UTF-8로 열어줘.  
코드에

<meta charset="utf-8"> 설정하기.

# 1단원-11: 문서의 구조와 슈퍼스타들 (2)

예제)

```
<title>WEB1 - html </title>  
<meta charset="utf-8">
```

char(character), 문자라는 뜻.

set은 집합이란 뜻.

UTF-8로 문서를 읽으라고 브라우저  
에게 이야기하고 있는 것.

<title> 본문의 제목이 무엇인지 설명.

<meta charset="utf-8"> 본문이 UTF-8 방  
식으로 저장돼 있다는 것을 설명.

본문은 <body>로 묶고.

본문을 설명하는 <body>를 설명하는 태그는  
<head>로 묶기로 합니다.

즉, HTML에 있는 모든 태그는 <head> 태그  
또는 <body> 태그 중 하나 아래에  
놓이게 된다는 것입니다.

이 태그들은 상당히 고위직 태그라 할 수 있습  
니다.

이 고위직 태그를 감싸는 단 하나의 최고위층  
태그가 있는데, 바로 <html>이라는 태그입니  
다.

이 <html> 태그 위에 관용적으로 이 문서에는  
HTML이 담겨 있다는 뜻에서

<!DOCTYPE html>이라고 쓰면 됩니다.

# 1단원-11: 문서의 구조와 슈퍼스타들 (3)

우리가 매일 같이 방문하는 웹 사이트를 아무 곳이나 가보세요.

웹 페이지에서 우 클릭해서 페이지 소스 보기 누르면 페이지 소스를 볼 수 있습니다.

예제)

위키피디아

슈퍼스타에 해당하는 태그들을 배움으로써 전 세계의 모든 웹 페이지가 가지고 있는 구조를 파악하게 된 겁니다.

예전에 보이지 않던 것이 보이기 시작하는 것입니다.

들리지 않았던 것이 들리기 시작하는 것입니다.

# 1단원-12: HTML 태그의 제왕 (1)

## anchor의 첫 글자를 띤 'a'

앵커는 배가 정박할 때 사용하는 닻을 의미함.

정보의 바다에 정박한다는 의미의 시적인 표현인 a 태그는 어떤 기능을 표현하는 것일까요?

바로 링크입니다.

본문의 HTML(Hypertext Markup Language)에 링크를 달 텐데, HTML 의 기술 공식 사용 설명서를 링크 걸면 좋을 것 같습니다.

HTML의 사용 설명서가 궁금하다면

검색어 : html specification

specification(설명서라는 뜻)

HTML을 만든 W3C라는 국제기구에서 만든 공식 설명서로 생각하면 됩니다.

# 1단원-12: HTML 태그의 제왕 (2)

## W3C Recommendation 권고안.

왜 권고안이냐면 W3C는 직접 브라우저를 만들지 않기 때문입니다.

멤버들이 모여 미래 웹이 어때야 하는가에 대해 의논해서 미래 웹의 기술을 결정하면 권고안 문서를 발표하고 이 문서를 웹 브라우저 제작 업체에서 보고 그에 따라 웹 브라우저를 만들어 가게 되는 것입니다.

## 링크를 걸고 싶은 곳을 <a> 태그로 감싸면 됩니다.

아직 정보가 부족.

<https://www.w3.org/TR/2011/WD-html5-20110405/>

그래서, 이동할 주소를 복사해서, 태그 속성에 적기.

하이퍼텍스트의 첫 글자인 'h'와 웹 브라우저에게 이 값을 참조하라는 의미의 reference에서 앞의 세 글자인 ref를 따서 href 라고 하는 속성에 지정합니다.

# 1단원-12: HTML 태그의 제왕 (3)

예제)

<u href="https://www.w3.org/TR/2011/WD-html5-20110405/">Hypertext Markup Language(HTML)</u> -> 해당 페이지에서 이동함.

새 탭에서 열리게 만들고 싶다면. ->target=" blank" 속성 추가.

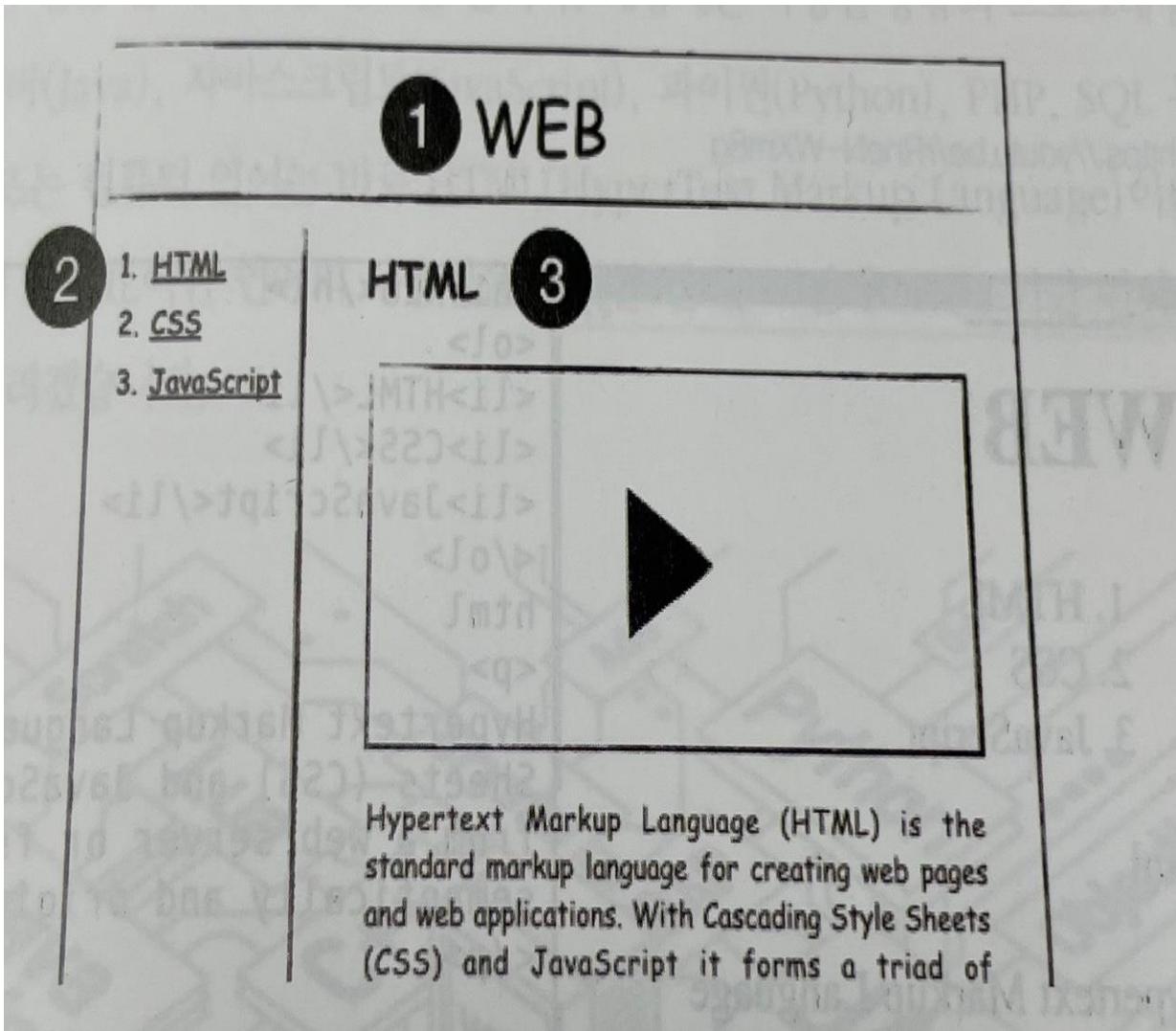
클릭하기 전에 툴팁처럼 무엇인가를 알려주고 싶다면  
마우스 커서를 가져다 놓으면 커서 아래에 설명이 나옴.

title="HTML5 specification" 속성 추가.

예제)

<a href="https://www.w3.org/TR/2011/WD-html5-20110405/" title="HTML5 specification">Hypertext Markup Language(HTML)</a>

# 1단원-13: 웹 사이트 완성 (1)



상단의 'WEB' 부분 구성하기.

책처럼 엮으려면 각각의 링크를 생성해야 함.

'WEB' -> index.html 라는 파일로 링크 걸기.

1. HTML -> 1.html

2. CSS -> 2.html

3. JavaScript -> 3.html

'WEB'을 클릭하면 index.html 파일로 갈 것이고,  
각 번호로 클릭하면 해당 페이지로 이동합니다.

<h2>HTML이란 무엇인가?</h2> 으로 변경.

# 1단원-13: 웹 사이트 완성 (2)

예제)

index.html

소스)

```
<!DOCTYPE html>
<html>
  <head>
    <title>WEB1 - HTML </title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1><a href="index.html">WEB</a></h1>
    <ol>
      <li><a href="1.html">HTML</a></li>
      <li><a href="2.html">CSS</a></li>
      <li><a href="3.html">JavaScript</a></li>
    </ol>
```

<h2>HTML이란 무엇인가?</h2>

<p><a href="https://www.w3.org/TR/2011/WD-html5-20110405/" title="HTML5 specification">Hypertext Markup Language(HTML)</a> is the standard markup language for <strong>creating <u>web</u> pages</strong> and web applications.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.</p>



<p style="margin-top: 40px;">

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

</p>

</body>

</html>

1.html 을 복사해서, 2.html, 3.html, index.html 만들기.

내용을 조금씩 수정하기.

CSS 를 클릭하면 2.html로 갈 텐데 2.html 은 css에 관한 내용으로

<h2>HTML이란 무엇인가?</h2> -> <h2>CSS란 무엇인가?</h2> 변경.

각각 파일을 변경해보기.

# 1단원-13: 웹 사이트 완성 (3)

완성 예제 코드.

```
index.html
<!DOCTYPE html>
<html>
  <head>
    <title>WEB1 - WEB </title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1><a href="index.html">WEB</a></h1>
    <ol>
      <li><a href="1.html">HTML</a></li>
      <li><a href="2.html">CSS</a></li>
      <li><a href="3.html">JavaScript</a></li>
    </ol>

    <h2>WEB이란 무엇인가?</h2>
    <p style="margin-top: 40px;">
      The World Wide Web (WWW), commonly known as the Web, is the world's dominant software platform.[1] It was originally a hypertext document management system[2] accessed over the Internet.[3]
    </p>
  </body>
</html>
```

The resources of the Web are transferred via the Hypertext Transfer Protocol (HTTP), may be accessed by users by a software application called a web browser, and are published by a software application called a web server.

Web resources may be any type of downloaded media, but web pages are hypertext documents formatted in Hypertext Markup Language (HTML). Special HTML syntax displays embedded hyperlinks with URLs, which permits users to navigate to other web resources. In addition to text, web pages may contain references to images, video, audio, and software components, which are either displayed or internally executed in the user's web browser to render pages or streams of multimedia content.

```
</p>
</body>
</html>
```

파일명: 1.html

소스코드

```
<!DOCTYPE html>
<html>
  <head>
    <title>WEB1 - HTML </title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1><a href="index.html">WEB</a></h1>
    <ol>
      <li><a href="1.html">HTML</a></li>
      <li><a href="2.html">CSS</a></li>
      <li><a href="3.html">JavaScript</a></li>
    </ol>

    <h2>HTML이란 무엇인가?</h2>
    <p style="margin-top: 40px;">
      Hypertext Markup Language(HTML) is the standard markup language for creating web pages and web applications.
    </p>
  </body>
</html>
```

<h2>HTML이란 무엇인가?</h2>

<p><a href="https://www.w3.org/TR/2011/WD-html5-20110405/" title="HTML5 specification">Hypertext Markup Language(HTML)</a> is the standard markup language for <strong>creating web pages</strong> and web applications.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.</p>

```

<p style="margin-top: 40px;">
```

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. HTML elements are delineated by tags, written using angle brackets.

```
</p>
</body>
</html>
```

# 1단원-13: 웹 사이트 완성 (4)

파일명: 2.html

소스코드

```
<!DOCTYPE html>
<html>
  <head>
    <title>WEB1 - CSS </title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1><a href="index.html">WEB</a></h1>
    <ol>
      <li><a href="1.html">HTML</a></li>
      <li><a href="2.html">CSS</a></li>
      <li><a href="3.html">JavaScript</a></li>
    </ol>
```

<h2>CSS란 무엇인가?</h2>

<p style="margin-top: 40px;">

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.[1] CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.[2]

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.[3] This separation can improve content accessibility; provide more flexibility and control in the specification of presentation characteristics; enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content

</p>

</body>

</html>

파일명: 3.html

소스코드

```
<!DOCTYPE html>
<html>
  <head>
    <title>WEB1 - JavaScript </title>
    <meta charset="utf-8">
  </head>
  <body>
    <h1><a href="index.html">WEB</a></h1>
    <ol>
      <li><a href="1.html">HTML</a></li>
      <li><a href="2.html">CSS</a></li>
      <li><a href="3.html">JavaScript</a></li>
    </ol>
```

<h2>JavaScript 무엇인가?</h2>

<p style="margin-top: 40px;">

JavaScript (/dʒɑːvəskrɪpt/),[10] often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS.[11] Over 97% of websites use JavaScript on the client side for web page behavior,[12] often incorporating third-party libraries.[13] All major web browsers have a dedicated JavaScript engine to execute

</p>

</body>

</html>

# 1단원-14: 원시 웹

이 수업은 두 개의 산으로 이루어져 있습니다.

## 하나. 웹 페이지 만들기.

-> 코딩이 무엇인가를 파악하는 것이 목표.

## 둘. 내가 만든 웹 페이지를 인터넷을 통해 누구나 가져갈 수 있게 하는 것.

인터넷 안에 -> web

핵 공격에도 견딜 수 있는 강인한 통신 시스템이 필요하게 되는데 바로 이것이 인터넷.

웹의 고향? 스위스

팀 버너스 1990년 11월에 세계 최초의 웹 브라우저인 World Wide Web 이라는 프로그램 만들. 주소 창에 -> info.cern.ch

<http://info.cern.ch/>

이곳이 웹의 메소포타미아입니다.

# 1단원-15: 인터넷을 여는 열쇠, 서버와 클라이언트

웹 브라우저가 웹 서버에 전기적 신호를 보내서, index.html 파일을 불러옴. 웹 브라우저가 요청(request)하고 웹 서버가 응답(response)합니다.

각 컴퓨터의 역할에 따라 어떻게 부를까를 고민했을 겁니다.

고객과 사업자입니다.

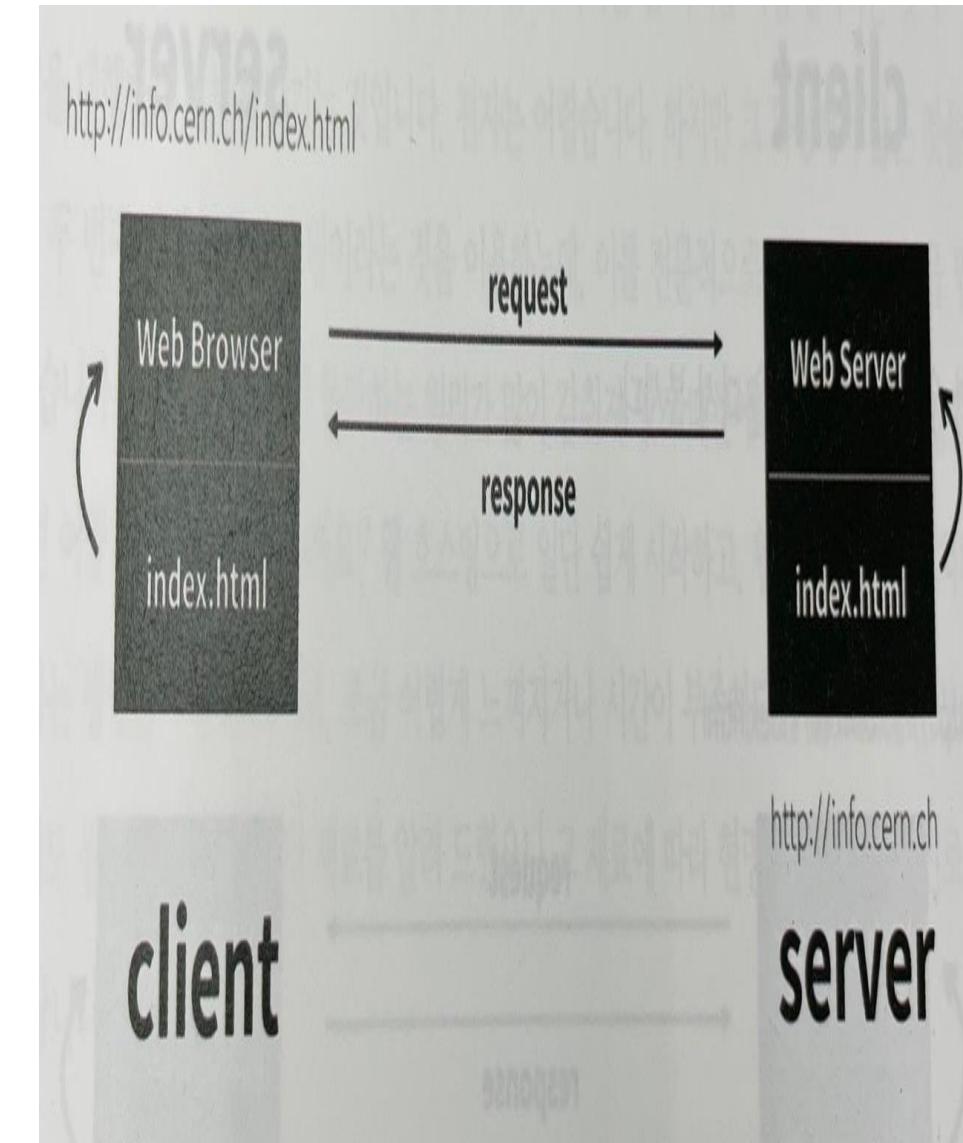
요청하는 컴퓨터 클라이언트, 응답하는 컴퓨터를 서버 컴퓨터라고 부르기도 합니다.

웹 브라우저 = 웹 클라이언트, 웹 서버 = 웹 사업자

웹 서버를 사용할 수 있다는 것은 내 컴퓨터에 있는 문서를 전 세계의 누구나 인터넷이 연결 되어있는 컴퓨터에 웹 브라우저만 깔면 가져다 볼 수 있다는 아주 혁명적인 의미입니다.

사용 방법 2가지

1. 직접 웹 서버 설치.
2. 업체 맡기기. 웹 호스팅



# 1단원-16: 인터넷을 여는 열쇠, 서버와 클라이언트 (1)

**웹 호스팅**: 깃허브 페이지

인터넷이 연결된 컴퓨터 하나하나를 호스트(host)라고 합니다.

**웹 서버를 운영하기 위한 컴퓨터, 다시 말해 호스트를 빌려주는 회사를 웹 호스팅 업체라고 합니다.**

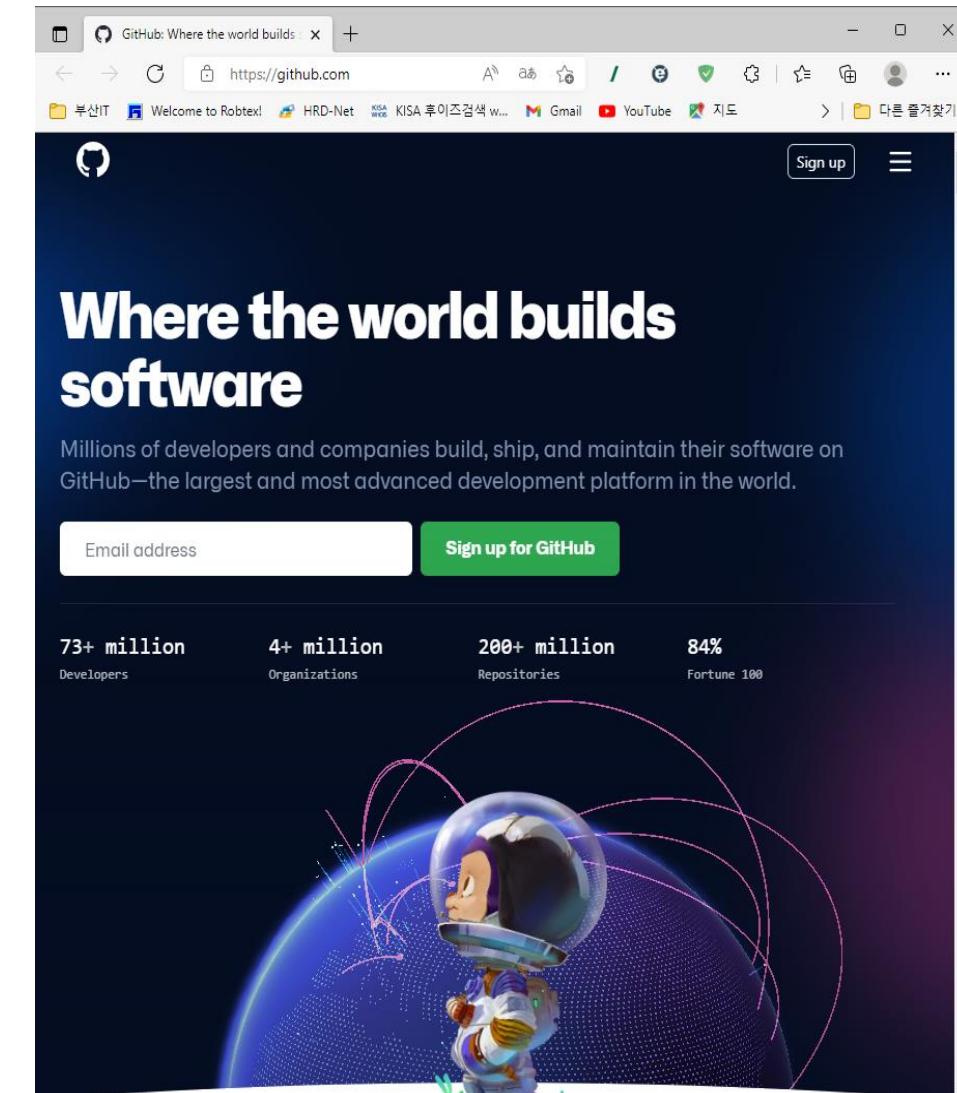
**무료**이면서, 굉장히 **유명한 사이트** 이용.

**깃허브** (GitHub, <https://github.com/>)

깃허브는 프로그래머에게는 성지와 같은 아주 중요한 서비스입니다. 왜냐하면 전 세계의 수많은 오픈 소스 프로젝트가 이곳에서 만들어 지기 때문입니다.

이 서비스가 해주는 여러 일 중 하나가 우리에게 홈페이지를 운영할 수 있는 웹 서버를 제공하는 것. -> Netlify로 운영이 변경됨.

회원가입 Sign up  
로그인하기.

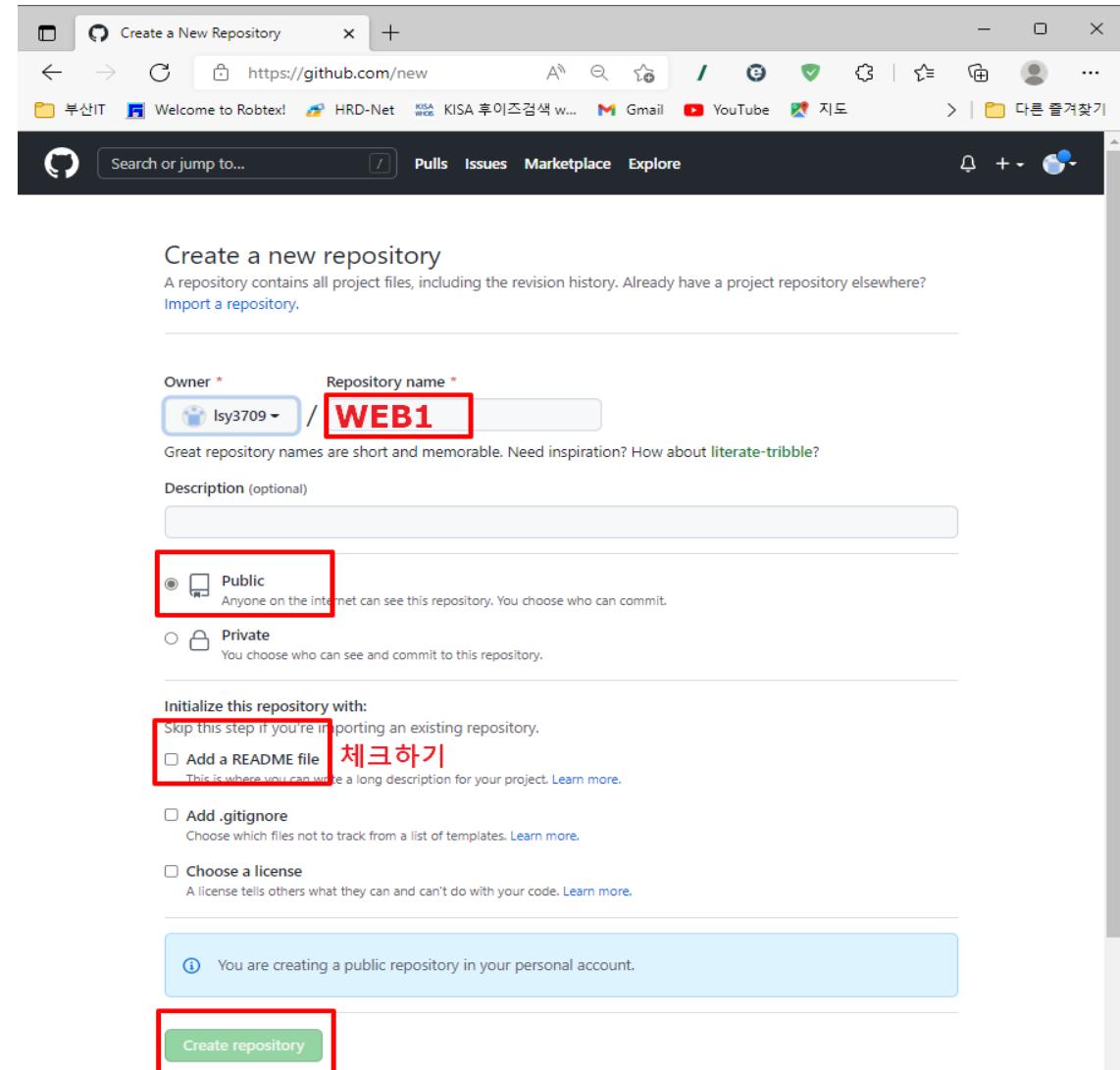


# 1단원-16: 인터넷을 여는 열쇠, 서버와 클라이언트 (2)

## 깃 허브 폴더 업로드

- 1) 왼쪽 상단에 New(Repository) 클릭.
- 2) Repository name : WEB1 설정.
- 3) public 클릭.
- 4) Add a README file : 체크하기.
- 5) Create repository
- 6) Add file 클릭.
- 7) Upload files 클릭.
- 8) 실습하고 있는 WEB1 폴더 안에 파일들을 드래그해서 이동하기.
- 9) Commit changes 항목에 : first version 입력.
- 10) Commit changes 클릭.

변경되면 다시 파일을 드래그해서 업로드하고, 변경 내용을 적고 Commit changes 클릭합니다



# 1단원-16: 인터넷을 여는 열쇠, 서버와 클라이언트 (3)

깃허브에서는 무료로 컴퓨터와 웹 서버를 제공합니다. -> 조금 변경

1) Settings 들어가기. (안보이면 오른쪽 점3개 클릭.)

2) 왼쪽 하단에 GitHub apps 클릭.

-> 안보이면 아래 사이트 접속.

<https://github.com/settings/installations>

3) Netlify -> configure 클릭.

4) Repository access -> Only select repositories에서 WEB1 클릭.

5) Save

6) netlify에서 -> Sites -> Add new site

7) Import an existing project 클릭.

8) GitHub 클릭.

9) lsy3709/WEB1 클릭. (본인 계정의 WEB 입니다.)

10) Deploy site 클릭.

시간이 조금 흐르면 1번 Your site is deployed 배포됨.

11) 2번 Set up a custom domain 클릭.

도메인 확인. 무료 도메인 설정해서. Bscoding.ga 확인 Verify 하기.

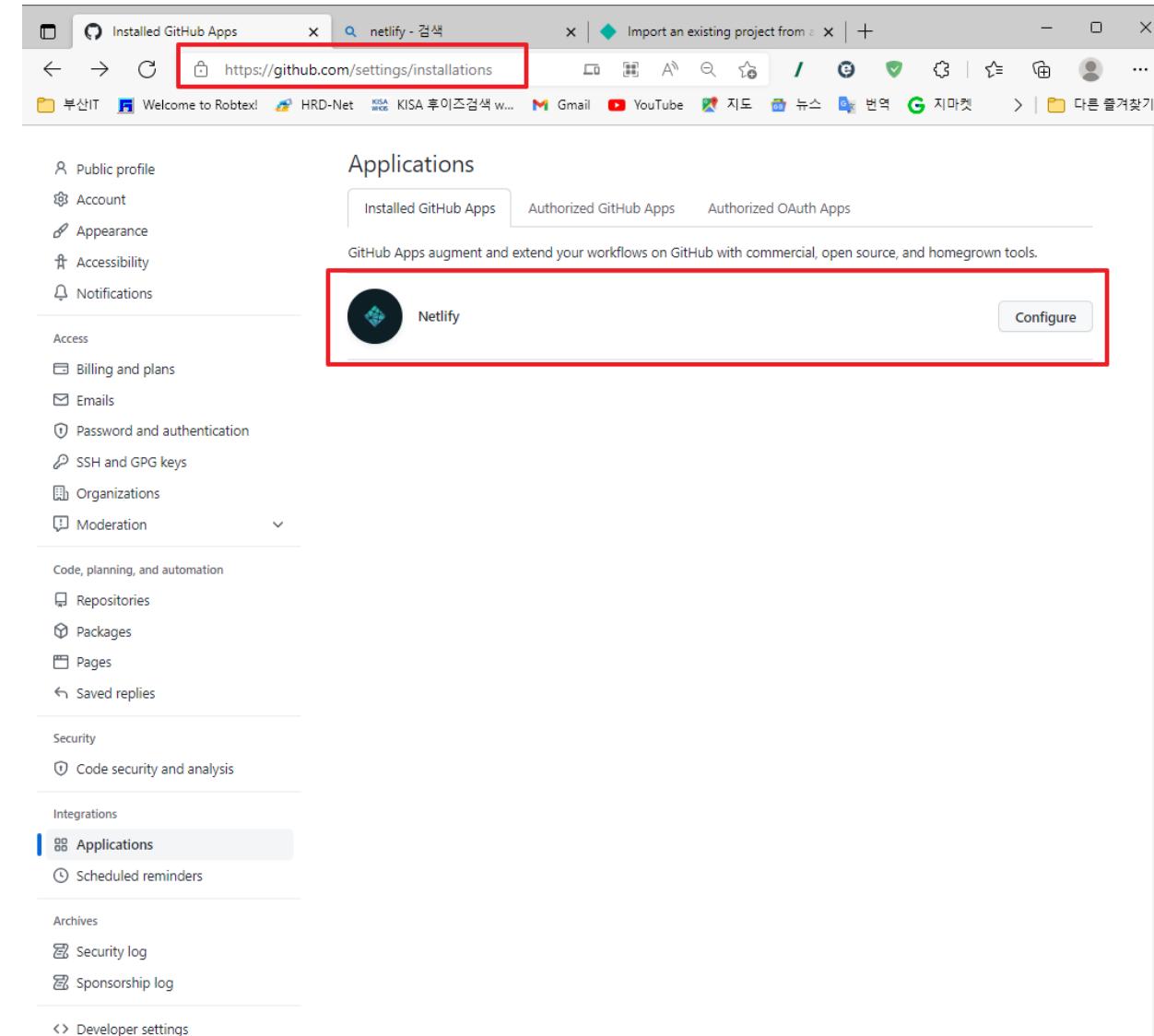
12) Add domain 클릭. -> 다음 페이지 무료 도메인 설정 후, 다시 오기.

13) HTTPS 부분에 SNI/TLS는 자동으로 설정해 줌. 시간 지나면 됨.

14) Domain management -> Custom domains

이름이 길어서 Options -> Edit site name 클릭.

Bscoding 으로 변경. -> Save



# 1단원-16: 인터넷을 여는 열쇠, 서버와 클라이언트 (4)

**무료 도메인 가입 하기.**

<https://www.freenom.com/en/index.html?lang=en>

- 1) 로그인
- 2) 검색어에서 도메인이 하나 지정 해서 검사.  
저는 busanitcoding -> bscoding.ga 가능해서 이것으로 사용하겠습니다.
- 3) 사용 기간은 12개월 무료로 선택.
- 4) 이용약관 확인하고 동의하고, 주문 완료.
- 5) 서비스 -> 내 도메인
- 6) **설정 이름의 도메인 관리 클릭.**
- 7) Freenom DNS 관리 클릭.
- 8) 설정하기.

**이름 : www, 유형: A , TTL : 3600 , 표적 75.2.60.5**

**이름 : 없이, 유형: A , TTL : 3600 , 표적 75.2.60.5**

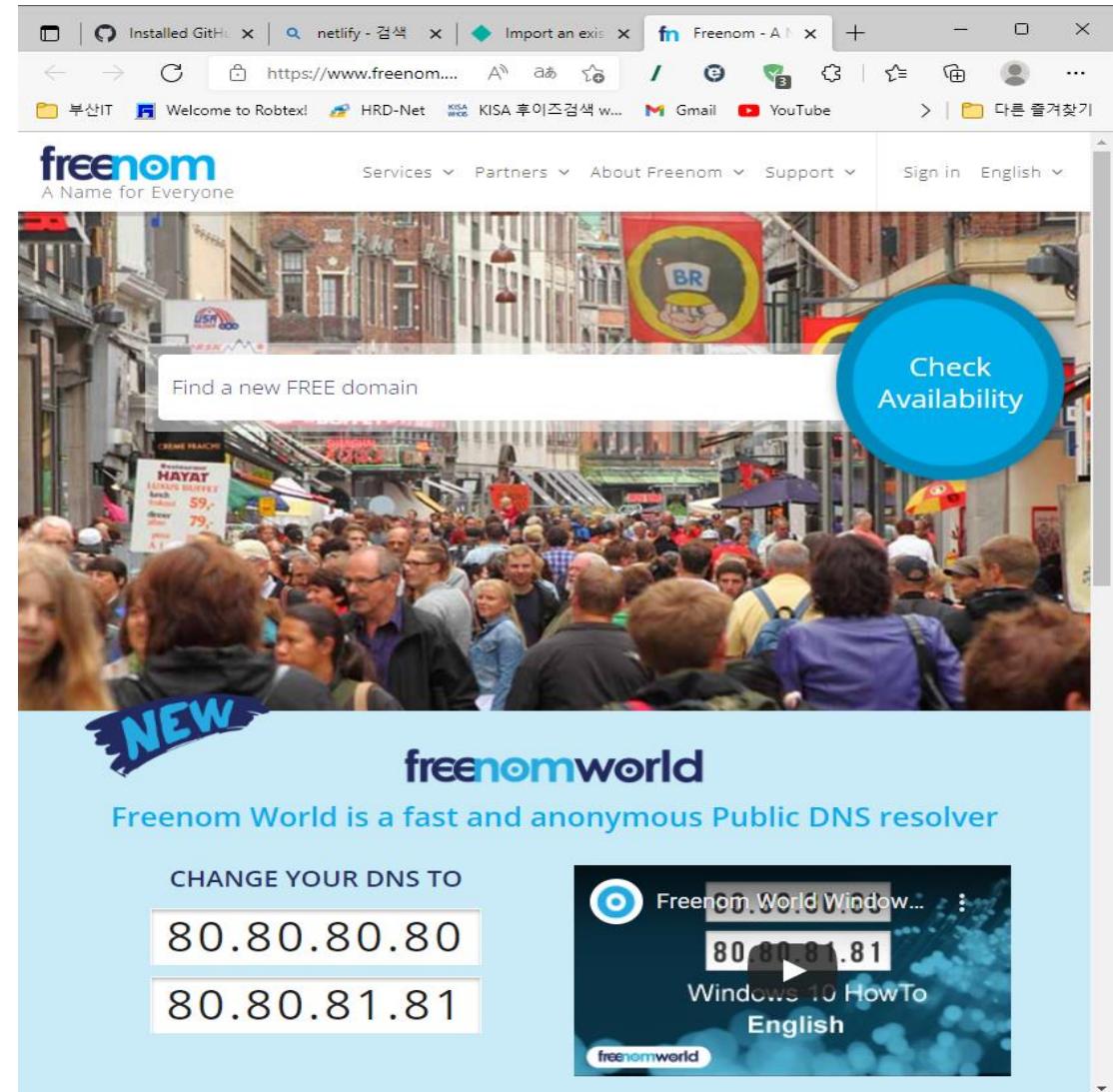
**2개 등록 하기.**

- 9) 변경사항 저장하기.

확인.

**PC, 모바일의 운영체제에서 확인 가능함.**

**https 는 시간이 지나야 활성화 됨.**



# 1단원-17: 웹 서버 운영하기 (1)

이 과정은 어렵지만, 간단히 확인만 하고 넘어 가겠습니다.

제대로 작동하는 이유는 한가지이지만, 작동하지 않는 이유는 수십 가지가 넘습니다.

혹시 작동하지 않아도 걱정할 필요는 없습니다.

실제로 웹 호스팅 업체에 운영을 맡기기 때문입니다

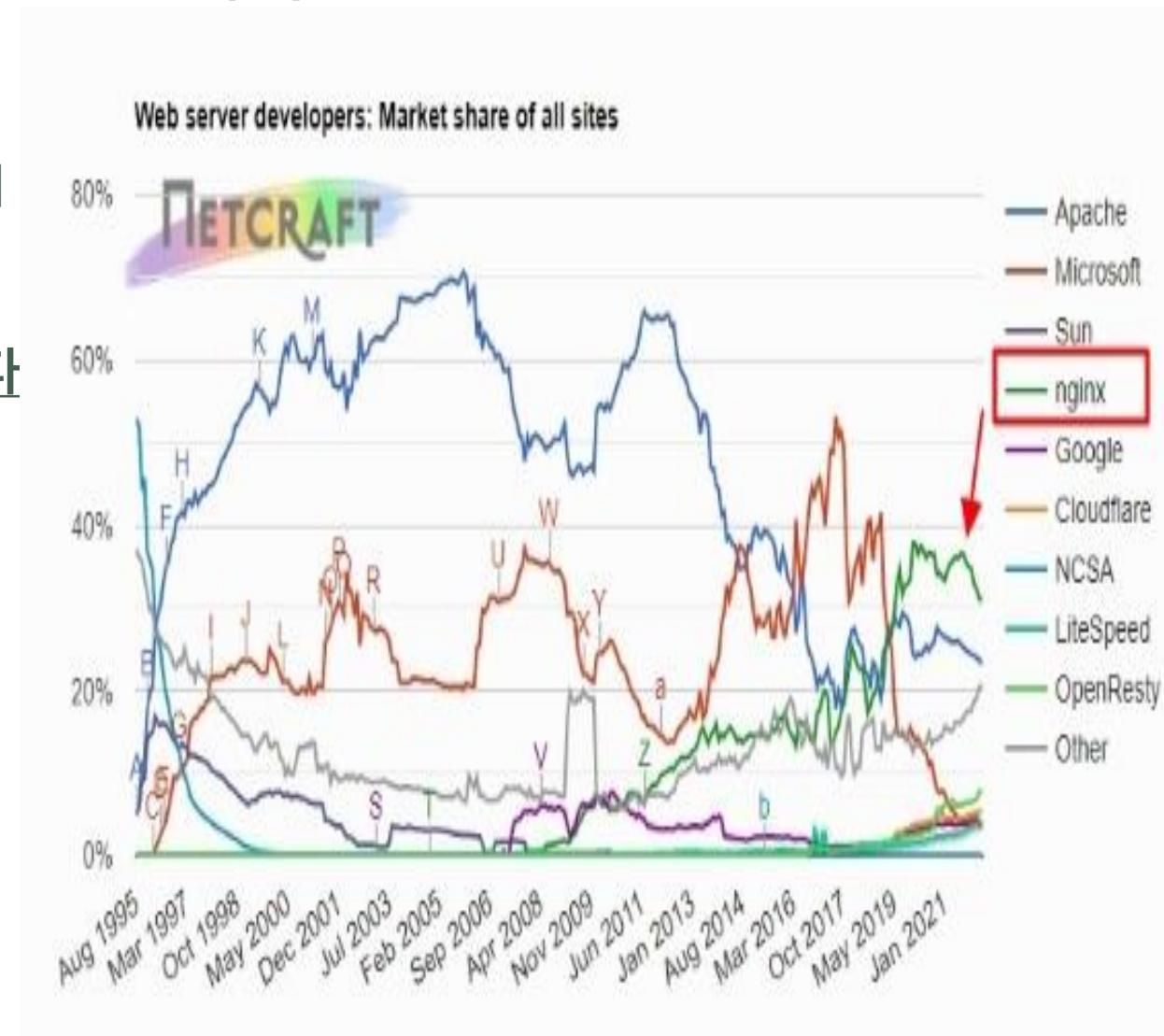
컴퓨터에 웹 서버라는 프로그램 설치하기.

웹 서버 제품군에도 여러 가지 있습니다.

아파치(Apache), IIS, 엔진엑스(nginx) 등.

오픈소스 nginx를 사용하겠습니다.

책은 아피치를 사용하지만, 순위가 변경되었습니다.



# 1단원-17: 웹 서버 운영하기 (2)

- 1) <http://nginx.org/en/download.html> 접속
- 2) Stable version : nginx/Windows-1.20.2 pgp 클릭.
- 3) 압축 해제한 폴더에 접속 후 nginx.exe 실행.
- 4) 브라우저에서 <http://localhost/> 접속 확인하기.
- 5) 종료 방법: 설치한 폴더에서 (탐색기에서 ) 폴더 경로를 알려 주는 입력란에 : 'cmd' 입력 하기.  
nginx.exe -s stop
- 6) 두번째 방법, 작업 관리자에서 작업 끝내기.
- 7) 만든 WEB1 서버에 올려 보기.
- 8) nginx 폴더 안에 conf 폴더 접속 하기.
- 9) nginx.conf 파일 편집 하기.

## 10) 44번째 줄에 root 옆에 경로 쓰기.

ex) root C:/Workspace HTML CSS JS;

저장 하고,

서버 종료 후 , 재시작.

nginx.exe -s stop

시작. -> nginx 또는 nginx -s reload

크롬 브라우저에서 <http://localhost/> 접속 확인하기

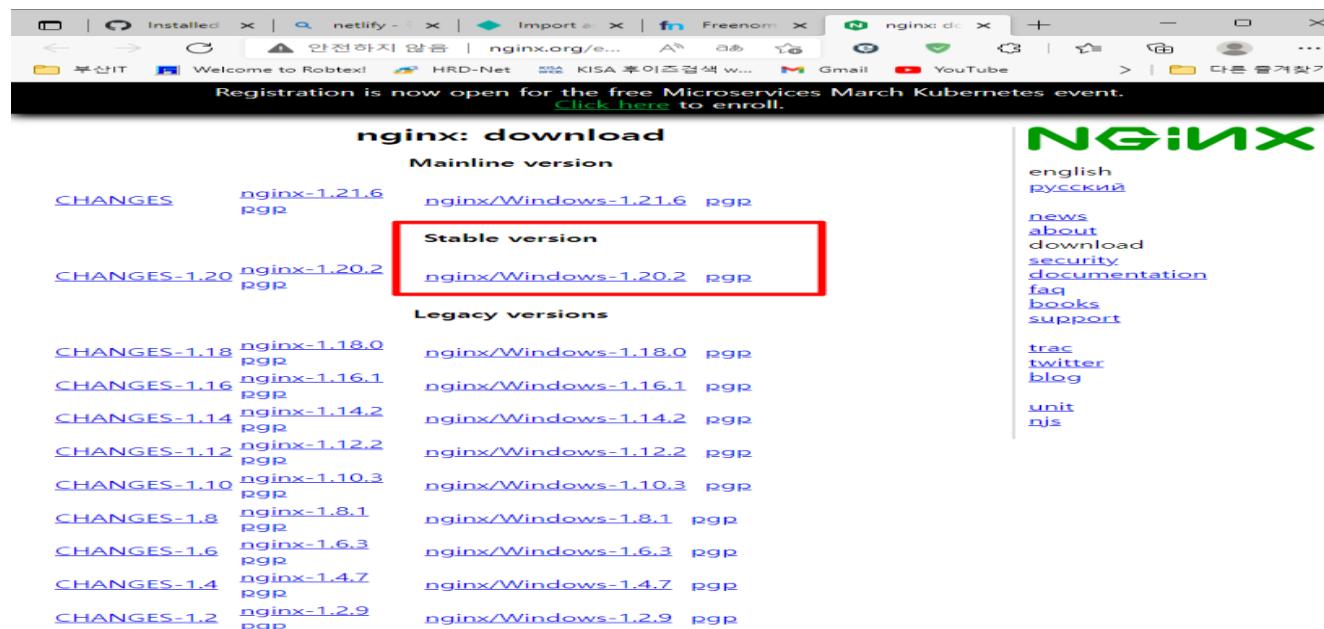
### 내 피씨에 설치한 웹 서버에 모바일로 접속 하는 방법

모바일, 웹 서버 설치한 피씨가 같은 네트워크를 사용하면 접속 확인 가능.

### 집 또는 강의장 네트워크에서 접속 확인 해보기.

#### 자기 자신 아이피 주소 확인하는 방법

- 1) win 키 + r : 실행 명령어창. -> 2) cmd 입력.
- 3) ipconfig 입력. ->
- 4) 주소 확인. 192.168.x.x 이런식의 주소가 보이면 맞음.
- 5) 모바일 브라우저에서  
192.168.x.x 로 접속 확인 하기.



# 1단원-18: HTML 수업을 마치며 (1)

웹 사이트를 아름답게 만들고 싶다면 CSS 공부하기.

CSS는 HTML이 태어 난지 약 4년 후에 만들어지기 시작함.  
웹 퍼블리셔, 웹 디자이너

사용자와 상호작용하는 웹 페이지를 만들고 싶다면 자바스크립트를 공부하기.

자바스크립트는 HTML이 태어난지 약 5년 후에 만들어지기 시작함.  
프런트 엔드 엔지니어, 웹 인터랙티브 디자이너

백엔드 기술의 필요성 예시

파일 하나를 변경 했을 때 1억 개의 웹 페이지가 동시에 변경되는 폭발적인 경험을 할 수 있음.

언어 : PHP, JSP Node.js의 익스프레스, 파이썬의 장고, 루비온레일스,  
ASP.NET 기술들. ->우리는 JSP 공부.

# Q&A