

웹 기초

인터넷의 역사, 웹의 동작 방식

정보가 연결되어 있는 세계 (인터넷 상상)



오래 전부터 많은 과학자들이 정보가 연결되어 있는 세계를 상상.

노버트 위너(Norbert Wiener, 1894 ~ 1964)

사람의 신경 작용에서 영감을 받아, 인간과 기계, 그리고 사회 전반에서 서로 통신하여야 한다는 사이버네틱스 이론을 펼침.

버니바 부시(Vannevar Bush, 1890 ~ 1974)

연결된 데이터들이 들어 있는 거대한 도서관을 예측.

J.C.R. 리클라이더(J.C.R. Licklider, 1915 ~ 1990)

커뮤니케이션 장치로서의 컴퓨터를 고안하여 온라인 공동체 개발 논의.

컴퓨터들의 광역 네트워크를 예측.

상상이 현실로...



패킷 교환(packet switching)

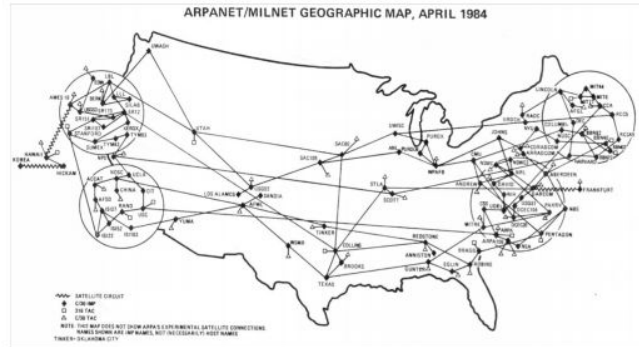
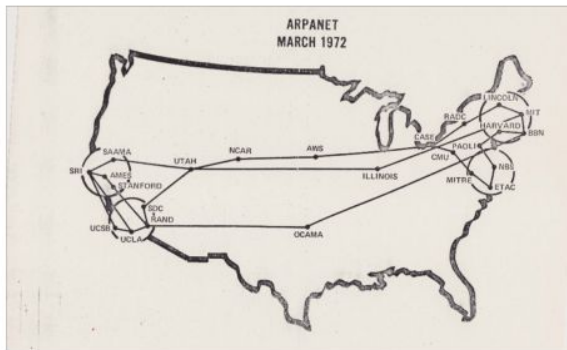
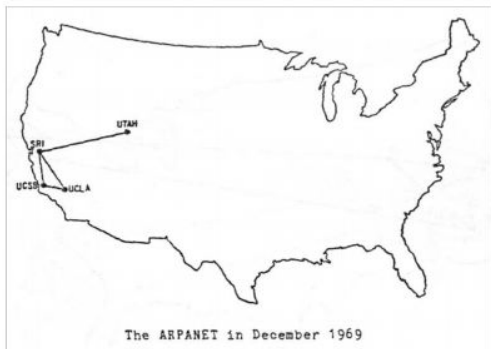
작은 블록의 패킷으로 쪼개서 데이터를 전송하는 방식. 데이터를 작게 쪼개서 먼 곳에 보냈다가 다시 합침.
1960년대 초, 영국과 미국의 연구자들이 발명.

이것이 1969년, 미 국방부에 의해 발명된 세계 최초의 인터넷의 뿌리

인터넷의 시작 ARPANET

냉전 시대인 1960년대, 미국 국방부 주도 하에 계획.

정보를 교환하여 연구 역량을 극대화하려는 목적과, 분산 네트워크로 하나가 붕괴하더라도 나머지에 영향이 없도록 하려는 목적.



ARPANET 이후

군대에서 시작되어 대학, 기업으로 퍼져나갔으나 소수 엘리트만 사용할 뿐.
초기 인터넷은 파일과 자원 공유, 원거리에 있는 컴퓨터에 로그인하려는 목적.

한편, 일반인에게도 PC(Personal Computer)가 보급되기 시작.



월드와이드웹 (WWW) 발명

유럽입자물리연구소(CERN) 소속 연구원 팀 버너스 리가 고안



한 문서에서 다른 문서로 접근한다는 개념인 하이퍼텍스트(Hypertext)를 적용한 HTML

HTML: HyperText Mark-up Language

문서에 주소를 붙이는 URL

URL: Uniform/Universal Resource Locator

컴퓨터와 컴퓨터 간에 메시지를 주고받을 수 있는 통신 규약인 HTTP

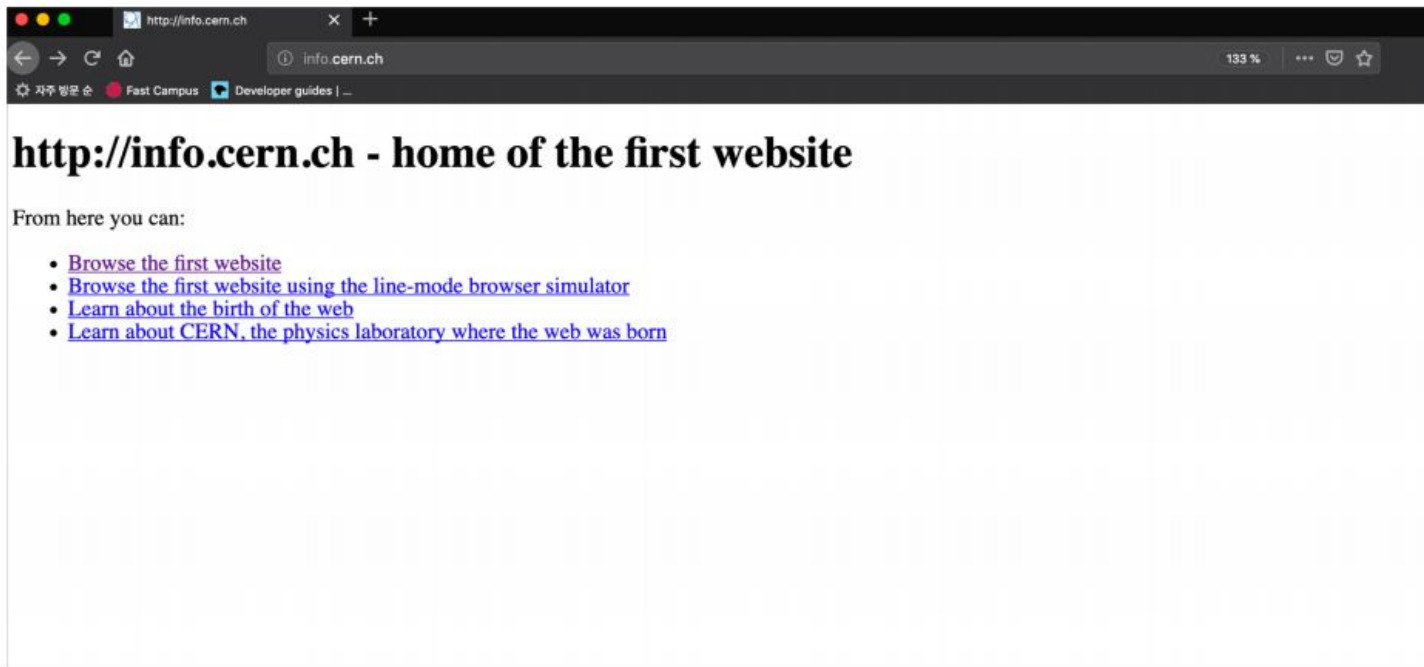
HTTP: Hypertext Transfer Protocol

전 세계 인터넷 콘텐츠를 연결해 공유할 수 있도록 하는 WWW(World Wide Web)

월드와이드웹 (WWW) 발명

최초의 사이트

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



웹의 진화



하이퍼텍스트 문서

기본적인 서식과 하이퍼링크를 내재한 텍스트 문서

웹 페이지

이미지, 오디오 등 리소스를 제공하고, 풍부하고 화려한 레이아웃 내재화

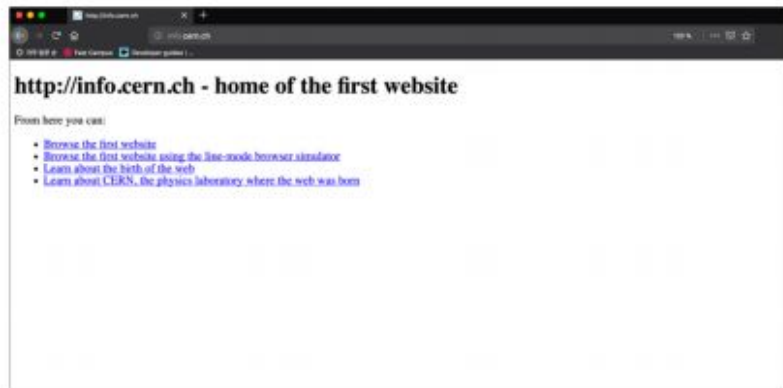
시각적으로는 아름답지만 사용자와의 상호작용이 불가능해서, 마치 종이에 인쇄된 페이지와 같음

웹 애플리케이션

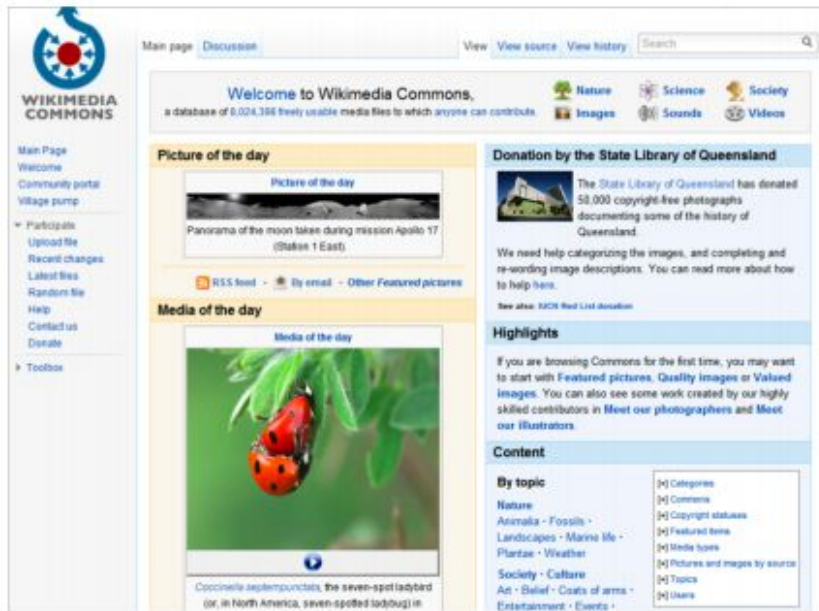
사용자와 상호작용이 가능하고 복잡한 처리 또한 가능한 웹 브라우저에서의 프로그램

웹의 진화

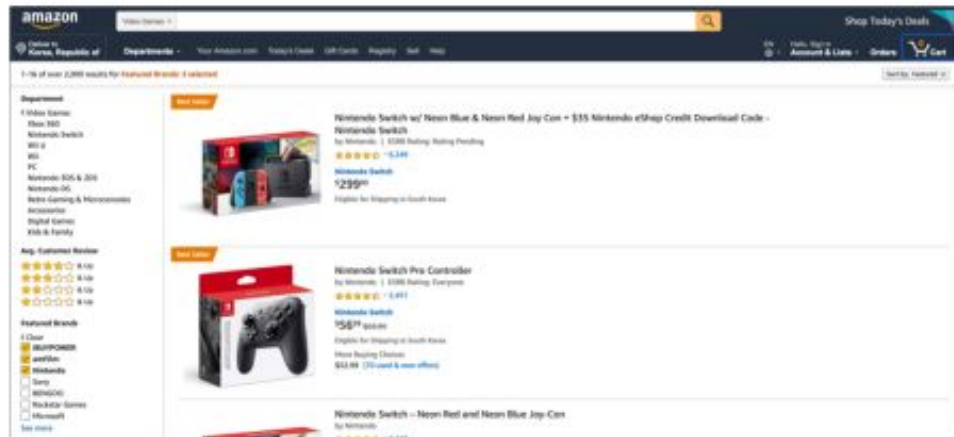
하이퍼텍스트 문서



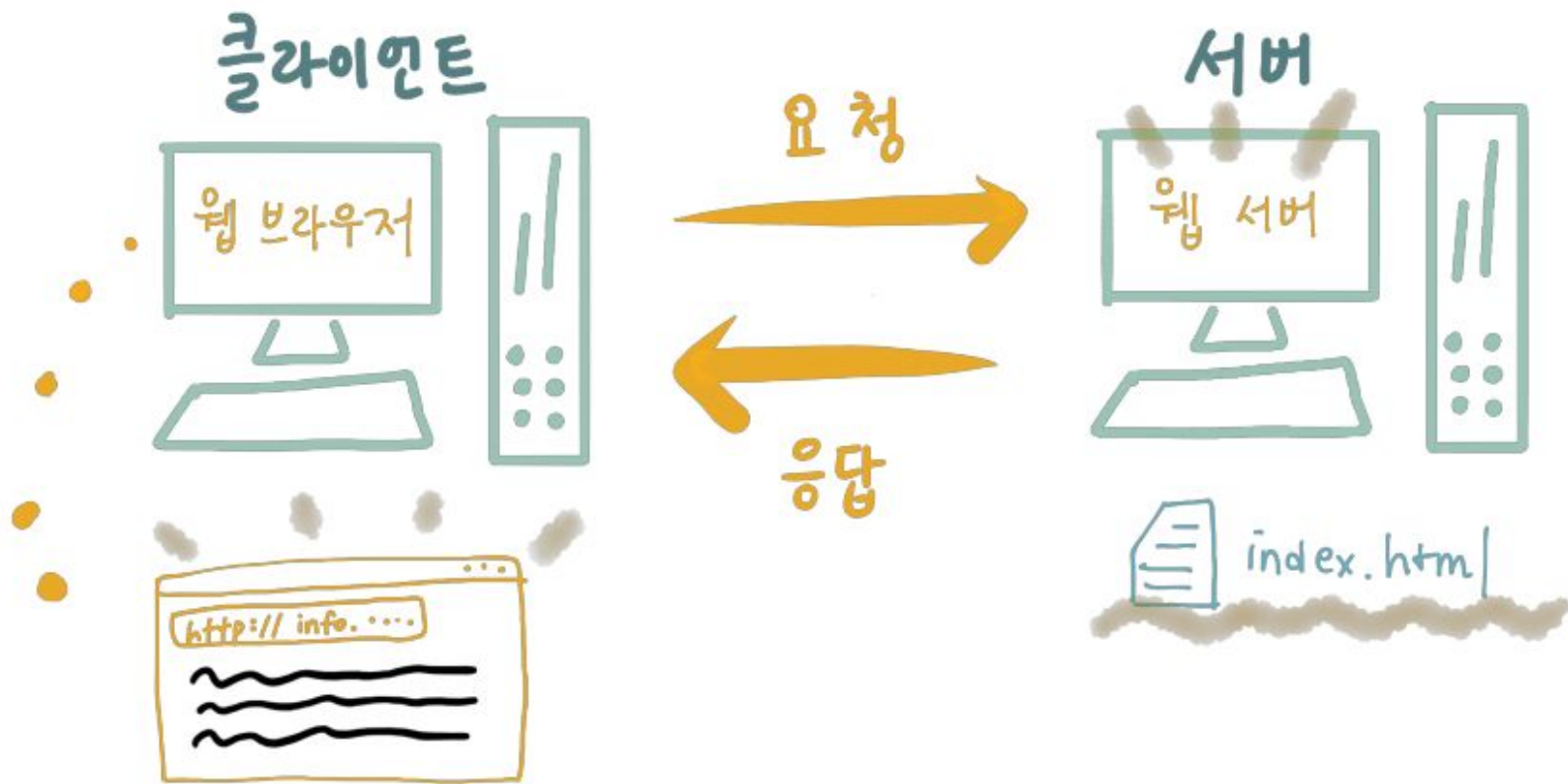
웹 페이지



웹 애플리케이션

[illegible]

서버와 클라이언트



서버와 클라이언트



서버도 요청을 보낼 수 있다. 이때, 요청을 보낸 서버가 클라이언트가 된다.

웹 브라우저만이 클라이언트가 되는 것은 아니다.

서버-클라이언트는 상대적인 개념.

서버와 클라이언트

클라이언트는 요청을 보내는 주체

- 크롬, 사파리, 엣지, 파이어폭스 등과 같은 웹 브라우저
- 데스크톱 앱 혹은 모바일 앱
- 다른 서버에 요청을 보내는 서버
- 개발자가 개발한 별도의 소프트웨어
- ...

서버는 네트워크를 통해 클라이언트에 정보 혹은 서비스를 제공하는 컴퓨터(또는 프로그램)

- 웹사이트를 저장한 컴퓨터
- 웹 서버 애플리케이션을 통해 만들어진 애플리케이션
- ...

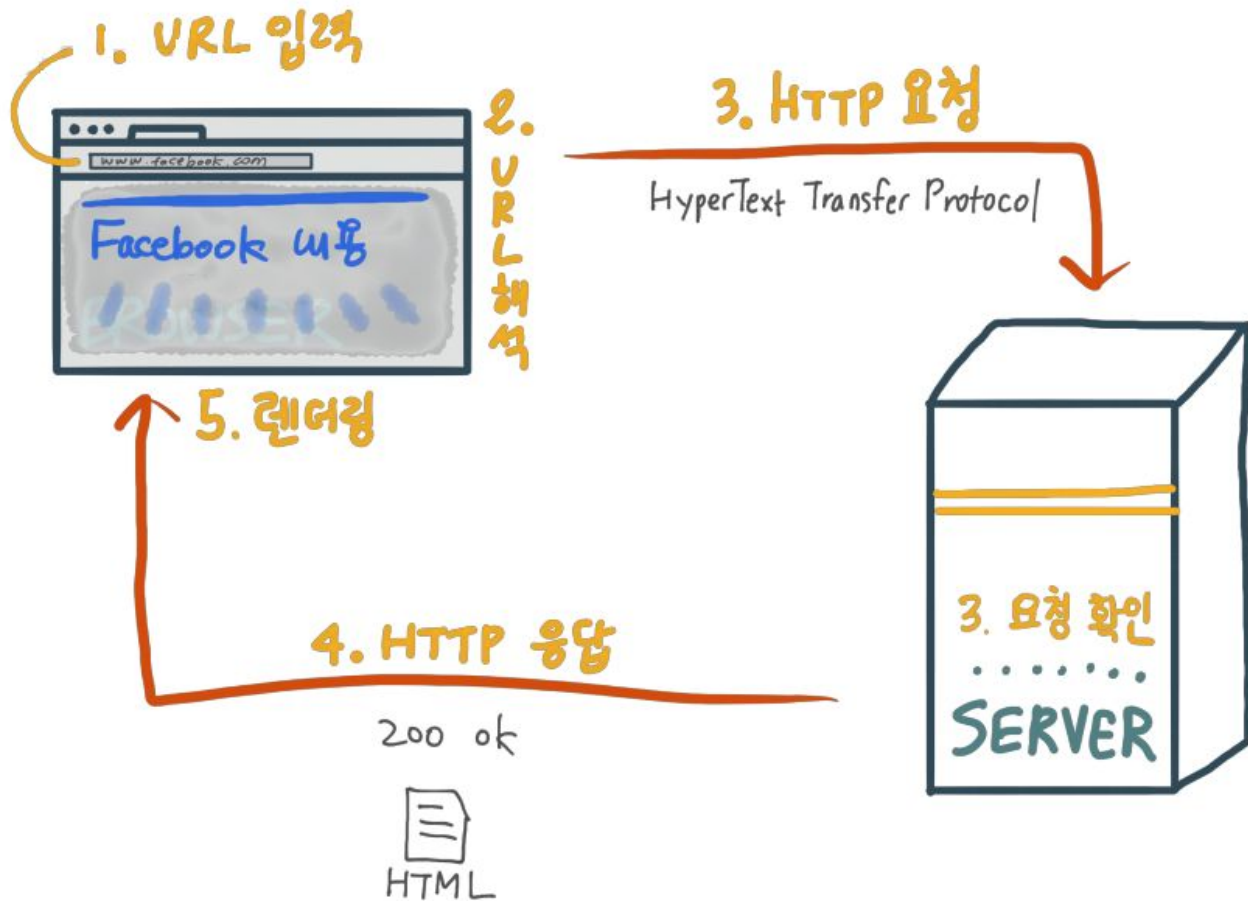
웹 프로그래밍?



웹 프로그래밍이란 ?

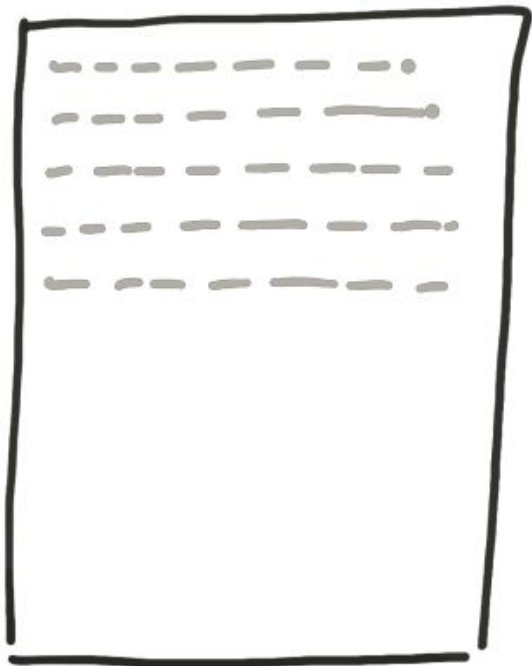
HTTP 프로토콜 (통신규약) 통신을 하는 **클라이언트** 와 **서버** 프로그램을 개발 하는 것

HTTP 통신



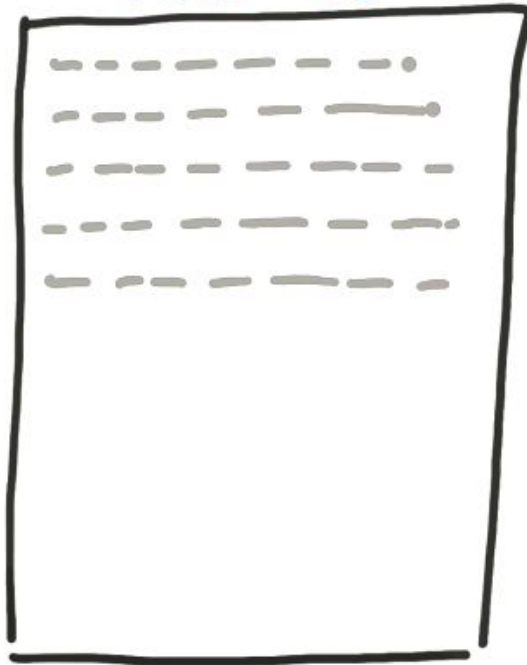
HTTP 통신

HTTP 요청



request 메시지

HTTP 응답



response 메시지

HTTP 통신

HTTP 요청

메서드 Method

서버가 수행해야 할 동작.
GET, POST, DELETE - - -

URL 주소

헤더 Header

브라우저 정보, 언어 등
여러 정보가 포함.

본문 Body

id/PW, 새 글 내용 등. 없어도 됨.

HTTP 응답

상태 코드

요청의 성공 여부.
200 OK, 404 Not Found -

헤더 Header

본문 Body

HTML 코드 등의 파일.
예러가 났다면 생략될 수도 있다.

웹의 동작

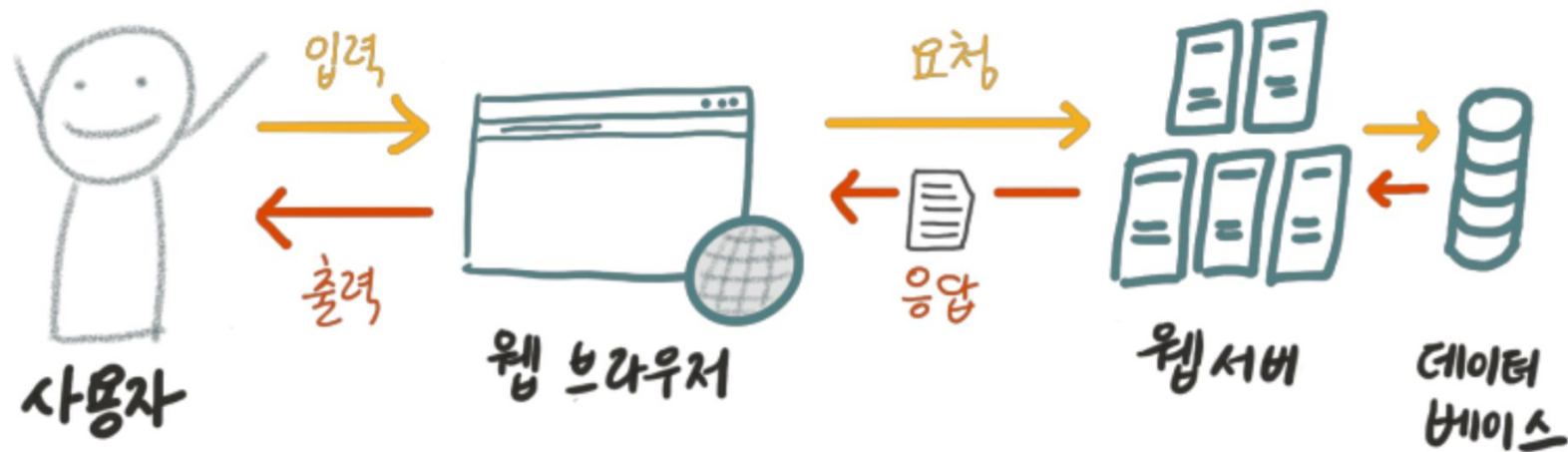


웹 브라우저 (클라이언트)와 서버는

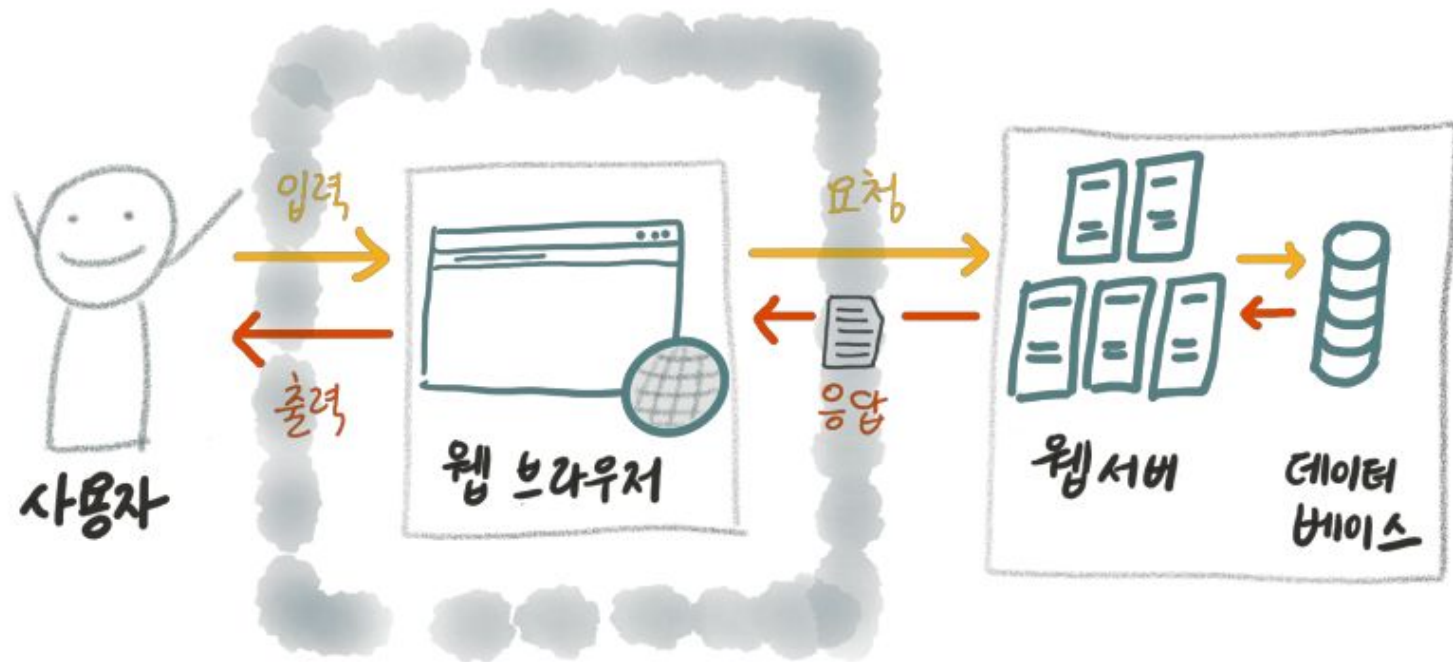
이미 정해진 약속(프로토콜: http)대로 메시지를 주고 받으며

동작한다.

FrontEnd와 BackEnd



FrontEnd



FrontEnd

BackEnd

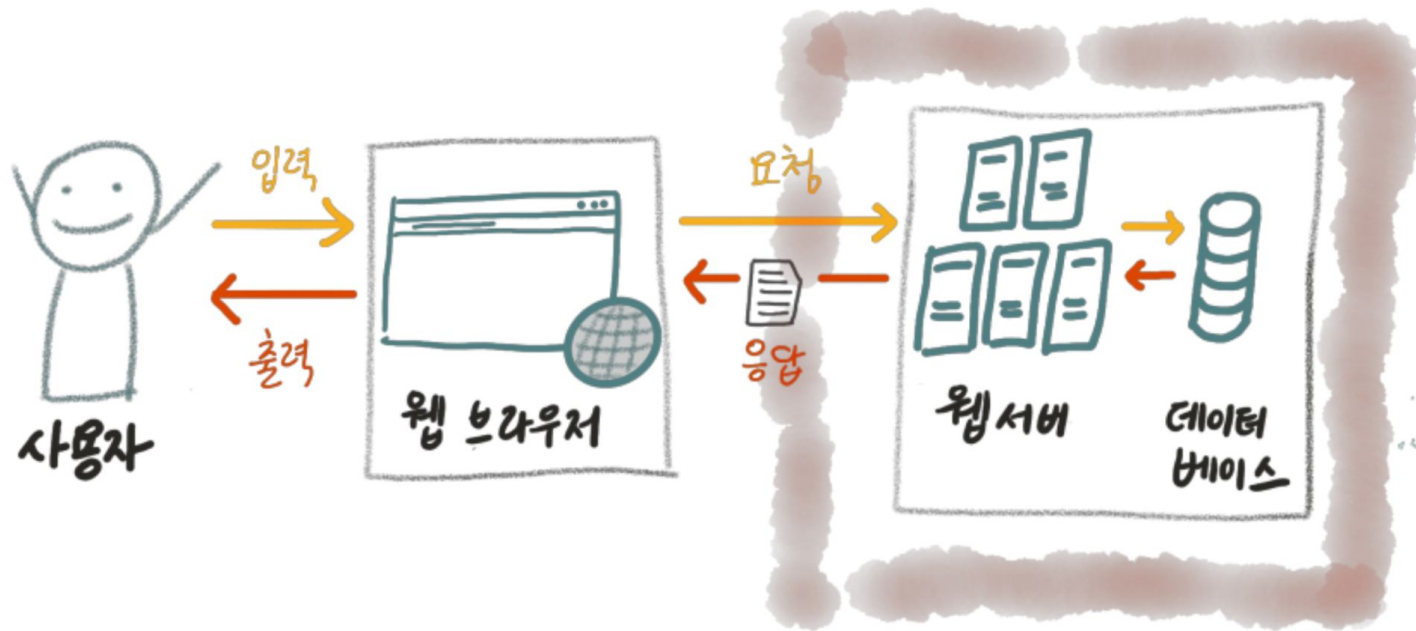
FrontEnd에서 사용되는 기술



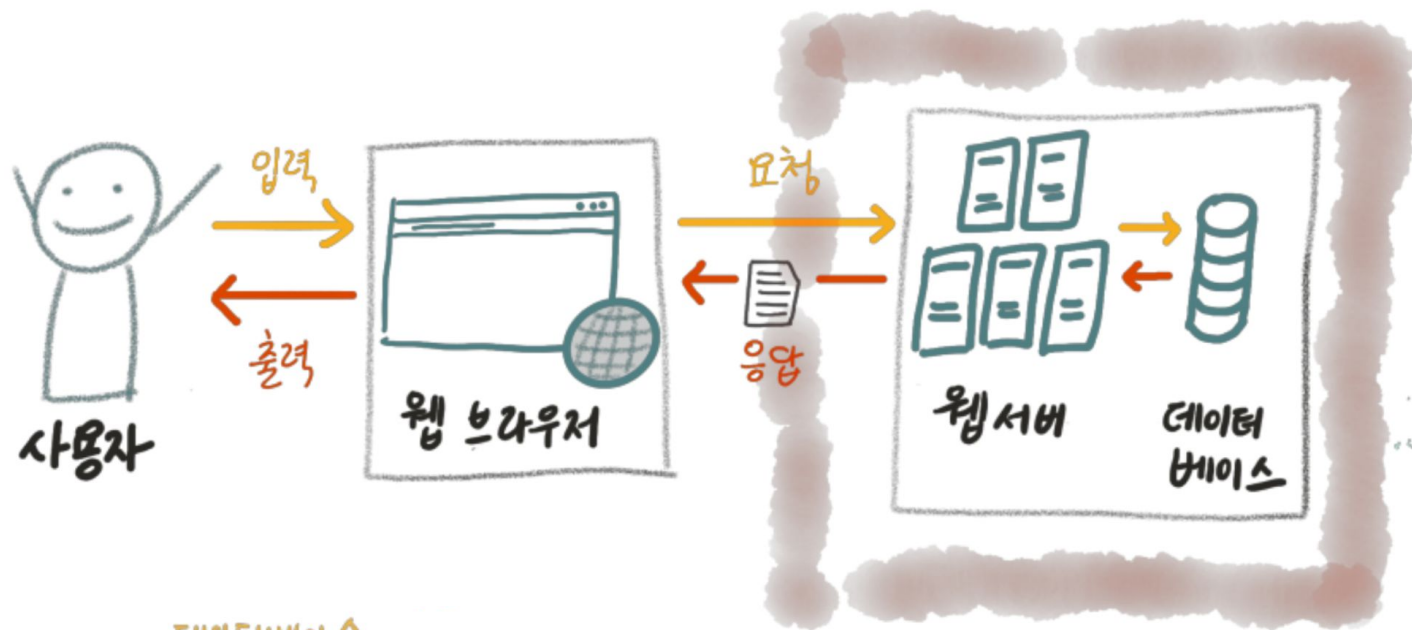
사용자의 요청을 잘 반영
적절한 레이아웃과 비주얼 디자인
웹 콘텐츠를 구조화

JavaScript
CSS
HTML

BackEnd



BackEnd 기술

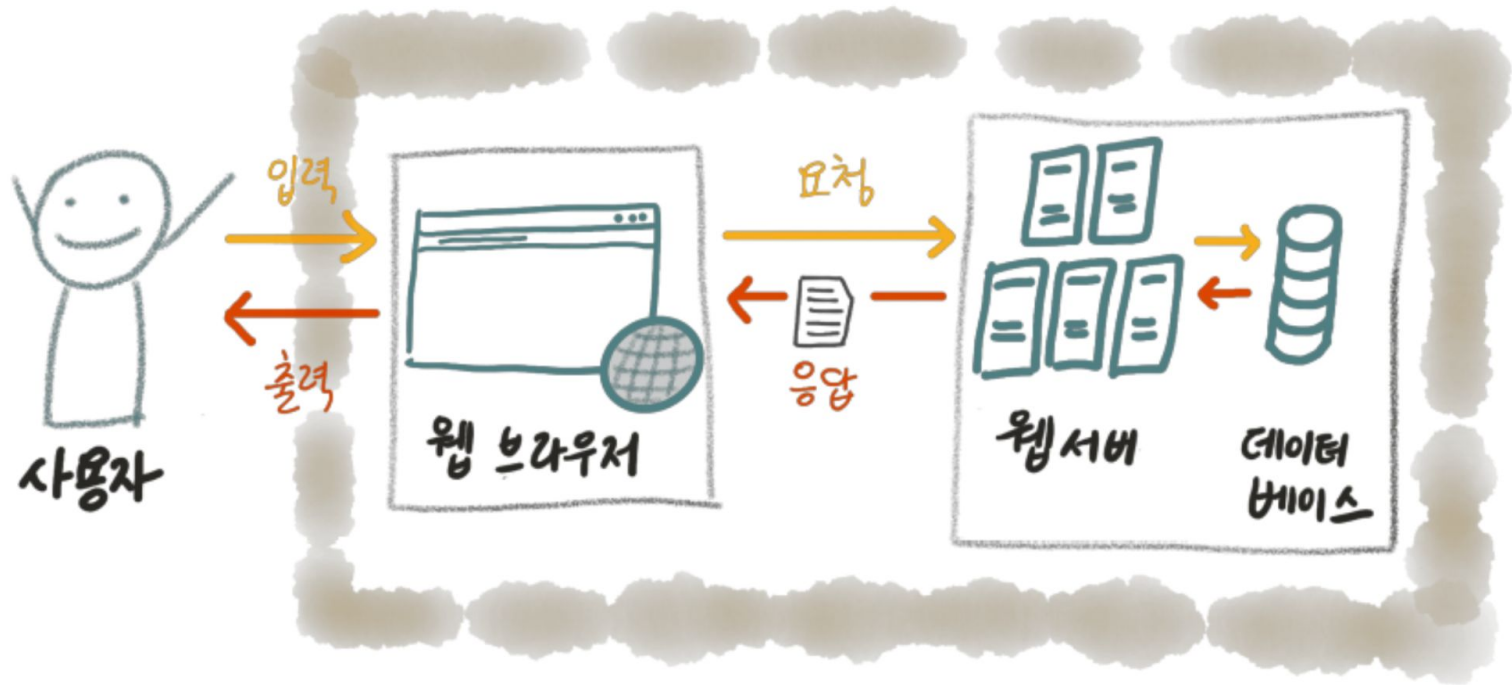


데이터베이스

Spring, Flask/Django 등 프레임워크

Java, PHP, Python 등 프로그래밍 언어

FrontEnd와 BackEnd 기술 ⇒ fullstack



HTML5, CSS3 시작

HTML

HTML(HyperText Markup Language)은
페이지에 제목, 문단, 표, 이미지, 동영상
등을 정의하고 그 구조와 의미를 부여하는
정적 언어로 **웹의 구조를 담당**

튼튼한 구조(Semantic)를 만드는 데에
HTML을 사용



CSS(Cascading Style Sheets)는 마크업 언어 (HTML, XML 등)가 실제 표시되는 방법 (색상, 크기, 폰트, 레이아웃 등)을 지정하여 콘텐츠 구조를 꾸며주는 정적 언어로 **웹의 시각적인 표현**을 담당

예쁘게 만드는 것만 집중, 적당한 크기와 아름다운 색상, 원하는 위치를 지정하는데 집중



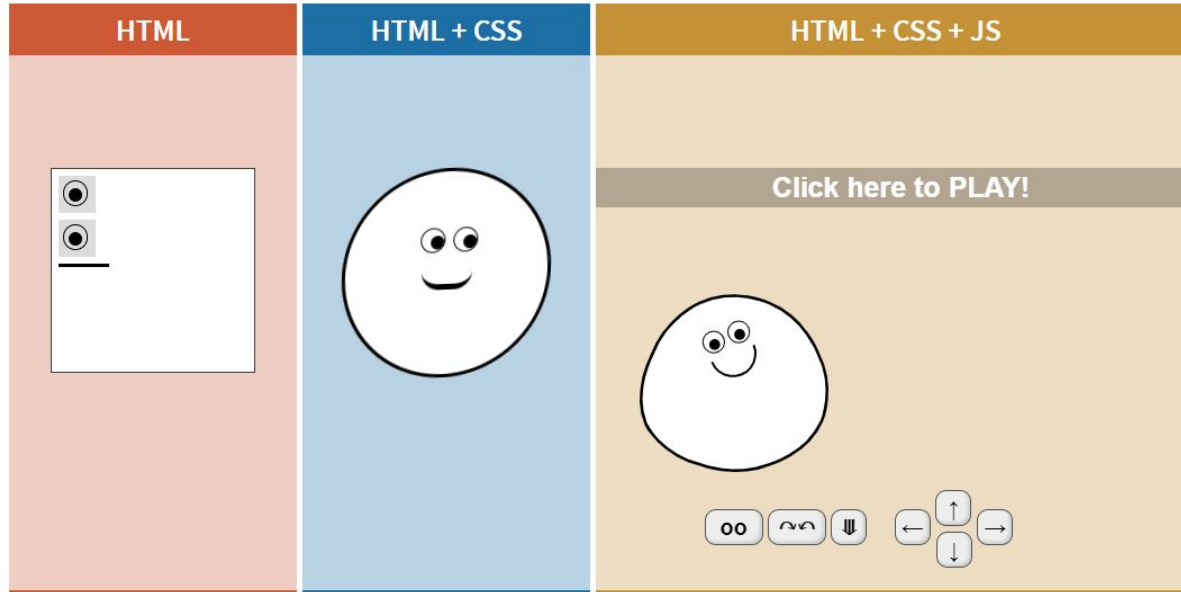
JavaScript

JS(JavaScript)는 콘텐츠를 바꾸고 움직이는 등 페이지를 동적으로 꾸며주는 역할을 하는 프로그래밍 언어로 **웹의 동적 처리**를 담당

JS는 HTML과 CSS를 동원해서 그들의 업무(구조, 시각적 표현 등)도 담당할 수 있지만, 그들만큼 잘하진 못하기 때문에(성능적으로) 되도록 동적 처리에만 집중



HTML, CSS , JavaScript



웹 표준

- ✓ 웹 표준(Web Standard)이란 ‘웹에서 사용되는 표준 기술이나 규칙’을 의미하며 **W3C의 권고안에서 나온 기술**들이 여기에 해당합니다.
- ✓ 이 표준 기술들을 기준으로 웹 브라우저들(크롬, IE, 사파리 등)이 만들어지는데, 브라우저를 만드는 업체(구글, MS, 애플 같은)에서 표준 기술을 해석하는 차이, 새로운 기술 삽입(**표준화 제정 단계**에 따른) 등으로 조금은 다르게 구동되는 브라우저가 생김

크로스 브라우징

크로스 브라우징(Cross Browsing)이란 조금은 다르게 구동되는 여러 브라우저에서, 동일한 사용자 경험(같은 화면, 같은 동작 등)을 줄 수 있도록 제작하는 기술, 방법.



크로스 브라우징

대부분의 브라우저는 최대한 웹 표준을 준수해서 제작되기 때문에 문제 되는 경우가 적지만 **MSIE**(마이크로소프트 인터넷 익스플로어) 브라우저는 웹 표준 준수 미흡함.

제멋대로 만들어져 있기 때문에 여러 의미에서 표준화하기 쉽지 않은 브라우저

그래서 **IE**에서도 동작하게 하는 것을 크로스 브라우징이라고 부르기도 합니다.(대부분의 경우 **IE**에서 문제가 없으면 다른 브라우저는 **OK!**)

웹 접근성

웹 접근성이란 정상적인 웹 콘텐츠 사용이 가능한 일반 사용자부터 고령자, 장애인 같은 신체적, 환경적 조건에 제한이 있는 사용자를 포함해 모든 사용자들이 동등하게 접근할 수 있는 웹 콘텐츠를 제작하는 방법을 말합니다.

청각 장애인을 위해 영상에 자막을 넣거나, 마우스를 사용할 수 없는 사람들을 위해 키보드를 통해서도 웹을 이용할 수 있게 하거나(혹은 그 반대), 이미지에 대체 텍스트를 제공하는 간단한 방법들까지 모두 웹 접근성에 해당

웹의 힘은 그것의 보편성에 있다. 장애에 구애 없이 모든 사람이 접근할 수 있는 것이 필수적인 요소이다. **팀 버너스 리 경 - 웹의 창시자**

웹 개발을 위한 에디터

Sublime Text



Atom



Brackets



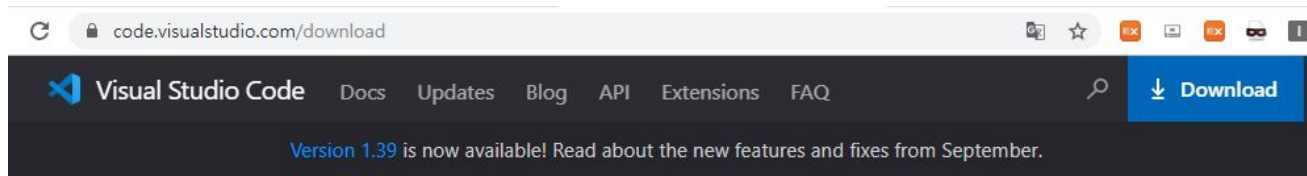
VS Code



WebStorm



VS Code 설치



Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



↓ Windows

Windows 7, 8, 10

User Installer	64 bit	32 bit
System Installer	64 bit	32 bit
.zip	64 bit	32 bit



↓ .deb

Debian, Ubuntu

↓ .rpm

Red Hat, Fedora, SUSE

.deb	64 bit
.rpm	64 bit
.tar.gz	64 bit

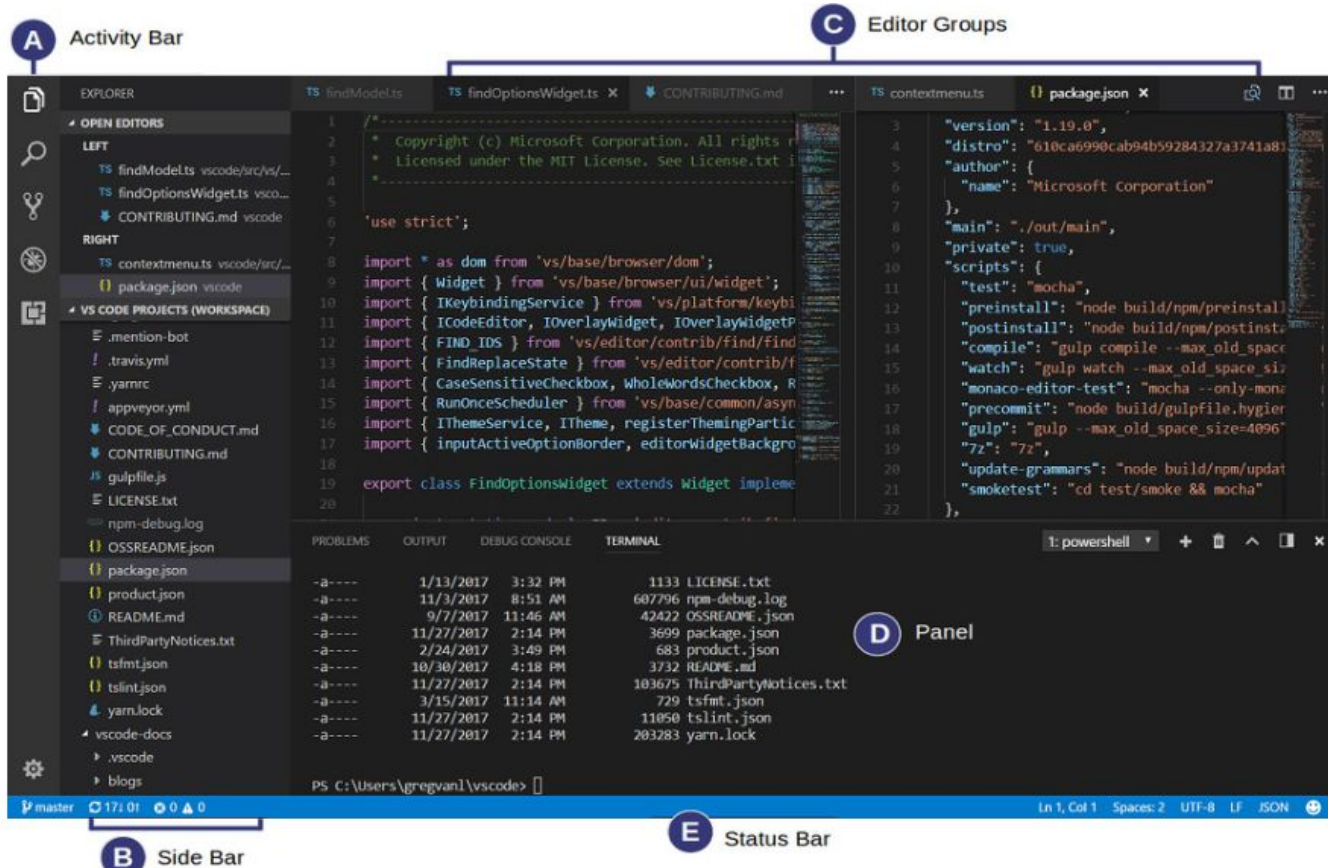
[Snap Store](#)



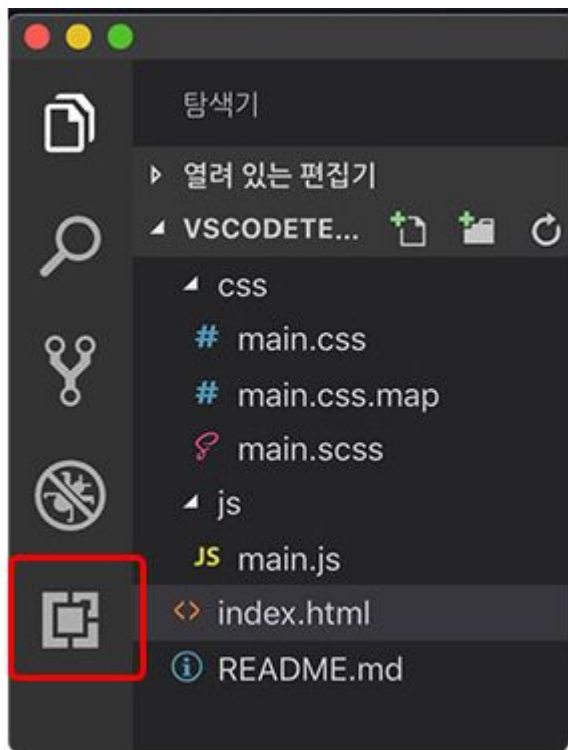
↓ Mac

macOS 10.10+

VS Code 설치



VS Code 설치



다른 에디터도 동일하지만 VS Code는 확장 기능을 제공하며 최초 설치한 버전에서 제공하지 않는 다양한 확장 기능을 외부에서 다운로드받아(설치) 연결 후 사용

에디터의 버전업 과정에서 자체적으로 흡수하는 기능도 있을 수 있으므로 몇몇 확장 기능은 에디터의 버전에 따라 이미 지원하고 있을 수 있으니 해당 기능이 제공되는지 확인

VS Code – 주요 단축키



Windows 단축키	macOS 단축키	설명
"Ctrl + B"	"Cmd + B"	사이드바 열기/닫기
"Ctrl + P"	"Cmd + P"	빠른 열기(파일이나 기호 탐색)
"Ctrl + Shift + P"	"Cmd + Shift + P"	모든 명령 표시(에디터의 모든 명령에 접근)
"Ctrl + F"	"Cmd + F"	찾기(검색)
"Ctrl + H"	"Cmd + Opt(Alt) + F"	찾기(검색)/바꾸기(대체)
"Alt + Up"	"Alt + Up"	줄 위로 이동
"Alt + Down"	"Alt + Down"	줄 아래로 이동

VS Code – 주요 단축키



"Shift + Alt + UpArrow"	"Shift + Alt + UpArrow"	위에 줄 복사
"Shift + Alt + DownArrow"	"Shift + Alt + DownArrow"	아래 줄 복사
"Tab"	"Tab"	들여쓰기
"Shift + Tab"	"Shift + Tab"	내어쓰기
"Ctrl + PageUp"	"Cmd + Shift + ["	이전 편집기 열기(좌측 창으로 전환)
"Ctrl + PageDown"	"Cmd + Shift +]"	다음 편집기 열기(우측 창으로 전환)
"Ctrl + \"	"Cmd + \"	편집기 분할(백슬래쉬)
"Ctrl + 숫자"	"Cmd + 숫자"	숫자 번째 분할된 편집기 그룹에 포커스
"Ctrl + W"	"Cmd + W"	편집기 닫기

웹에서 사용하는 이미지



비트맵과 벡터 이미지

이미지(그래픽)는 크게 비트맵과 벡터로 구분됩니다.

비트맵(Bitmap)은 각 픽셀이 모여 만들어진 정보의 집합으로 **레스터 (Raster)** 이미지라고도 부릅니다.

픽셀 단위로 화면에 렌더링합니다.(렌더링: 컴퓨터가 화면에 그림을 그려서 우리가 볼 수 있게 합니다)

우리가 일반적으로 사용하는 대부분의 이미지가 비트맵 형식입니다.

그림판, 포토샵과 같은 툴로 편집할 수 있습니다.

웹에서 사용하는 이미지



벡터(Vector)는 수학적 정보의 형태(Shape)들이 만들어내는 결과물입니다.

이미지가 가지고 있는 점, 선, 면의 위치(좌표), 색상 등의 정보를 온전히 가지고 있으며 그를 화면에 렌더링합니다.

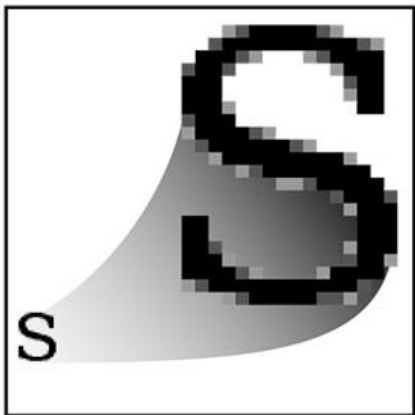
따라서 좀 더 많은 연산을 해야 하지만, 대신 해상도(픽셀)에 영향을 받는 비트맵 이미지와 달리 해상도로부터 자유롭게 렌더링할 수 있습니다.

쉽게 말하면 확대 및 축소를 해도 이미지가 깨지지 않죠.

또한 수학적 정보만을 가지고 있기 때문에 이미지 확대/축소에 따른 용량 변화가 없습니다.

일러스트 같은 툴로 편집할 수 있습니다.

웹에서 사용하는 이미지



Raster
.jpeg .gif .png



Vector
.svg

웹에서 사용하는 이미지



이미지 종류	장점	단점
비트맵	정교하고 다양한 색상을 자연스럽게 표현	확대/축소 시 계단 현상, 품질 저하
벡터	확대/축소에서 자유로움, 용량 변화가 없음	정교한 이미지(인물, 풍경 사진 같은)를 표현하기 어려움

웹에서 사용하는 이미지

JPG(JPEG)

JPG(Joint Photographic coding Experts Group) Full-color와 Gray-scale의 압축을 위해 만들어졌으며 압축률이 훌륭해 사진이나 예술 분야에서 많이 사용됩니다.

- 손실 압축
- 표현 색상도(24비트, 약 1600만 색상) 뛰어나 고해상도 표시장치에 적합
- 이미지의 품질과 용량을 쉽게 조절 가능
- 가장 널리 쓰이는 이미지 포맷

웹에서 사용하는 이미지



GIF

GIF(Graphics Interchange Format)는 이미지 파일 내에 이미지 및 문자열 같은 정보들을 저장할 수 있습니다.

- 비손실 압축
- 여러 장의 이미지를 한 개의 파일에 담을 수 있음(움짤, 애니메이션)
- 8비트 컬러만 지원(다양한 색상을 표현하는 작업에는 적합하지 않음)
- 동영상 같은 이미지!(애니메이션)

웹에서 사용하는 이미지



PNG

PNG(Portable Network Graphics)는 Gif의 대체 포맷으로 개발되었습니다.

- 비손실 압축
- 8비트(256 색상) / 24비트(약 1600만 색상) 컬러 이미지 처리
- Alpha Channel 지원(투명도)
- W3C 권장 포맷
- 투명도 지원!

웹에서 사용하는 이미지

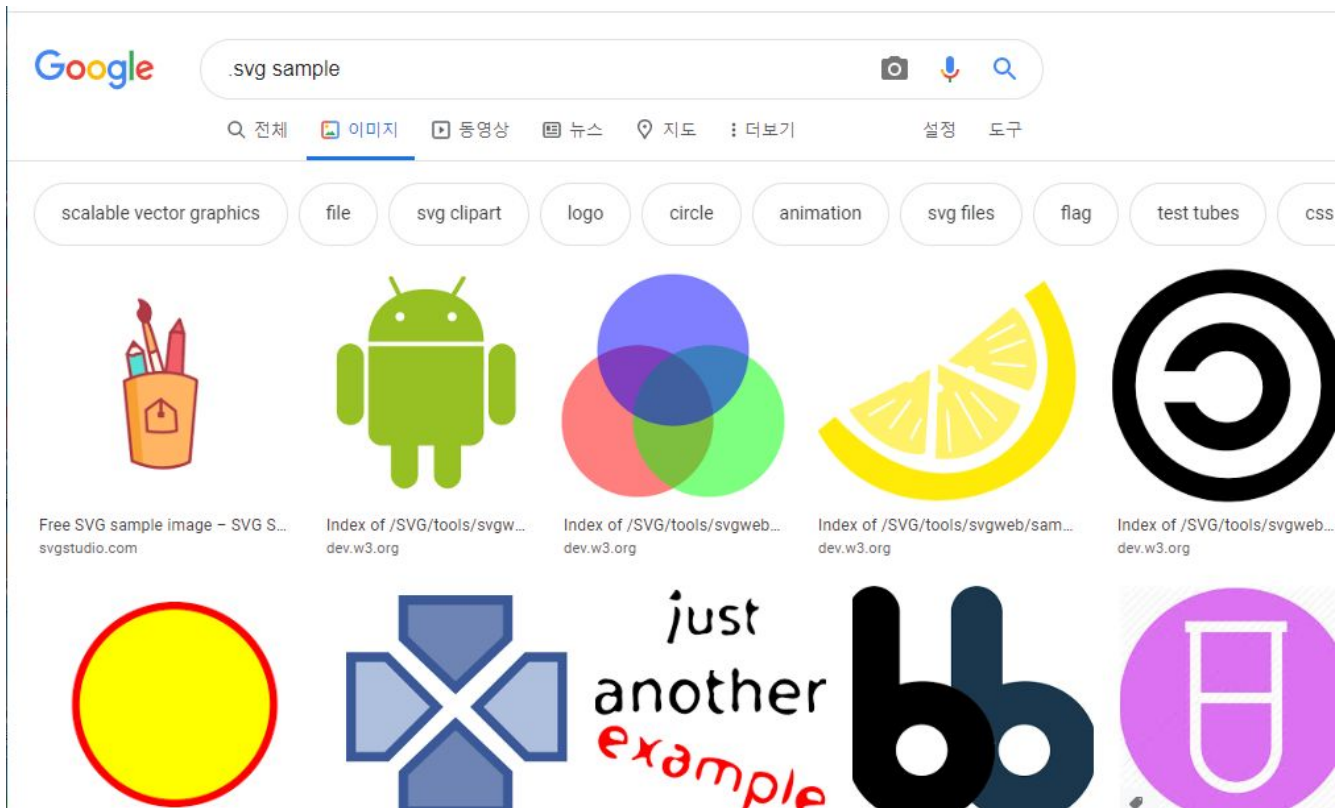


SVG

SVG(Scalable Vector Graphics)는 마크업 언어(HTML/XML) 기반의 벡터 그래픽을 표현하는 포맷입니다.

- 해상도의 영향에서 자유로움
- CSS로 Styling 가능
- JavaScript로 Event Handling 가능
- 코드 혹은 파일로 사용 가능
- 벡터 이미지에 익숙하지 않다면 다루기 까다로움

웹에서 사용하는 이미지



특수문자 용어 정리

Backtick, Grave
백틱, 그레이브



특수문자 용어 정리

~

Tilde

틸드, 물결 표시



특수문자 용어 정리

@

At sign
앳, 골뱅이



특수문자 용어 정리



Caret
캐럿



특수문자 용어 정리

&

Ampersand

앰퍼센드



특수문자 용어 정리

*

Asterisk
에스터리스크, 별표



특수문자 용어 정리

|

Vertical bar
버티컬 바



특수문자 용어 정리

{ }

Brace
브레이스, 중괄호



특수문자 용어 정리

[]

Bracket
브래킷, 대괄호



특수문자 용어 정리

기호	영어(발음)	한글
`	Grave(그레이브)	-
~	Tilde(틸드)	물결표시
!	Exclamation(엑스클러메이션) mark	느낌표
@	At(앳) sign	골뱅이
#	Number(넘버) sign, Sharp(샵)	샵, 우물 정
\$	Dollar(달러) sign	달러

특수문자 용어 정리

%	Percent(퍼센트) sign	퍼센트
^	Caret(캐럿)	-
&	Ampersand(앰퍼샌드)	-
*	Asterisk(에스터리스크)	별표
-	Hyphen(하이픈), Dash(대쉬)	마이너스
_	Underscore(언더스코어), Low dash(로대쉬)	밑줄
=	Equals(이퀄) sign	이꼬르
“	Quotation(쿼테이션) mark	큰 따옴표
'	Apostrophe(아포스트로피)	작은 따옴표

특수문자 용어 정리

:	Colon(콜론)	땡땡이
;	Semicolon(세미콜론)	털 달린 땡땡이
,	Comma(콤마)	쉼표
.	Period(피리어드), Dot(닷)	점, 마침표
?	Question(퀘스천) mark	물음표
/	Slash(슬래쉬)	-
	Vertical bar(버티컬 바)	-

특수문자 용어 정리



\	Backslash(백슬래쉬)	-
()	Parenthesis(퍼렌서시스)	(소)괄호
{ }	Brace(브레이스)	중괄호
[]	Bracket(브래킷)	대괄호
< >	Angle Bracket(앵글 브래킷)	꺾쇠괄호