




시작

2025. 06. 16

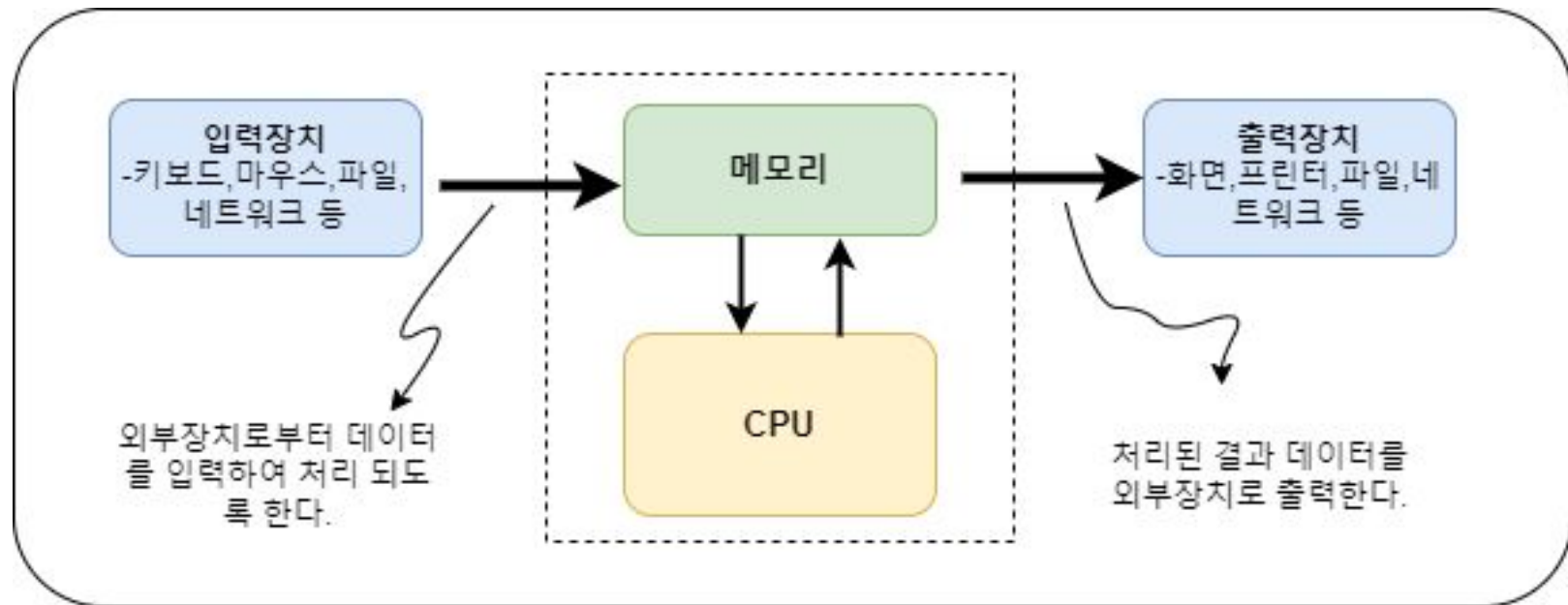


컴퓨터 시스템

- **하드웨어 (Hardware):** 물리적으로 존재하는 모든 컴퓨터 장치를 의미합니다. 예를 들어, 중앙처리장치(CPU), 메모리(RAM), 저장장치(SSD/HDD), 키보드, 마우스, 모니터 등이 포함됩니다. 쉽게 말해, 만질 수 있는 모든 컴퓨터 부품을 하드웨어라고 합니다.
- **소프트웨어 (Software):** 컴퓨터에서 실행되는 프로그램이나 데이터를 의미합니다. 운영체제(OS), 응용 프로그램(예: 웹 브라우저, 게임, 워드 프로세서), 드라이버 등이 포함됩니다. 하드웨어가 제대로 작동하도록 돕고, 사용자가 다양한 작업을 수행할 수 있도록 합니다.

하드웨어 hardware

- **입력 (Input):** 사용자가 데이터를 입력할 수 있도록 하는 기능입니다. 키보드, 마우스, 터치스크린, 마이크 등의 장치가 포함됩니다.
- **처리 (Processing):** 입력된 데이터를 처리하는 기능입니다. 중앙처리장치 (CPU)와 그래픽처리장치 (GPU)가 주요 역할을 수행하며, 연산과 논리적 처리를 담당합니다.
- **저장 (Storage):** 데이터를 저장하는 기능입니다. 하드디스크 (HDD), 솔리드 스테이트 드라이브 (SSD), 램 (RAM), 플래시 메모리 등이 포함되며, 장기 및 단기 저장을 담당합니다.
- **출력 (Output):** 처리된 데이터를 사용자에게 출력하는 기능입니다. 모니터, 프린터, 스피커 등이 해당합니다.
- **통신 (Communication):** 네트워크를 통해 다른 장치와 데이터를 주고받는 기능입니다. Wi-Fi, 블루투스, 이더넷 등이 포함되며, 인터넷 연결과 장치 간 통신을 가능하게 합니다.



소프트웨어 software

시스템 소프트웨어 : 하드웨어와 직접 연결되어 컴퓨터를 운영하고 관리하는 역할

- 운영체제(OS): Windows, macOS, Linux 등
- 드라이버: 하드웨어를 제어하는 프로그램
- 유틸리티 프로그램: 보안 소프트웨어, 백업 프로그램 등

응용 소프트웨어 : 사용자가 특정 작업을 수행할 수 있도록 돕는 프로그램

- 생산성 소프트웨어: 워드 프로세서, 스프레드시트, 프레젠테이션 소프트웨어 등
- 멀티미디어 소프트웨어: 그래픽 디자인 프로그램, 음악 및 동영상 편집 소프트웨어 등

프로그래밍 소프트웨어 : 개발자가 소프트웨어를 만들고 수정하는 데 사용하는 도구

- 프로그래밍 언어: Python, Java, C++ 등
- 개발 도구(IDE): Visual Studio, Eclipse 등

운영체제의 기능 (Windows)

컴퓨터의 하드웨어와 소프트웨어를 관리하여 원활한 작동을 돕는 중요한 역할을 합니다.

- **프로세스 관리**: 여러 프로그램을 동시에 실행할 수 있도록 **CPU** 사용을 조정하고, 프로세스 간의 작업 우선순위를 결정합니다.
- **메모리 관리**: 실행 중인 프로그램이 **RAM**을 효율적으로 사용할 수 있도록 조정하며, 불필요한 메모리 사용을 방지합니다.
- **파일 시스템 관리**: 데이터 저장과 검색을 위해 파일 및 폴더를 구성하고, 읽기/쓰기 권한을 관리합니다.
- **입출력 장치 관리**: 키보드, 마우스, 모니터, 프린터 등 여러 장치가 원활하게 작동하도록 제어합니다.
- **보안 및 접근 제어**: 사용자 계정 관리, 암호 보호, 바이러스 및 악성 소프트웨어 방지 등의 보안 기능을 제공합니다.
- **네트워크 관리**: 인터넷 및 로컬 네트워크 연결을 관리하여 데이터 전송과 통신이 원활하게 이루어지도록 합니다.

윈도우즈 Windows

Microsoft 운영체제. 기본 기능 사용 방법은 다음과 같습니다.

시작 메뉴 활용: 시작 버튼을 클릭하면 프로그램 목록과 설정 옵션을 확인할 수 있습니다.

파일 탐색기 사용: 폴더와 파일을 정리하고 검색할 수 있는 도구입니다. 윈도우 로고키 + E

설정 메뉴 : 시스템, 네트워크, 개인 설정 등을 변경할 수 있습니다.

단축키 활용: Ctrl + C(복사), Ctrl + V(붙여넣기), Alt + Tab(창 전환) 등 다양한 단축키가 있습니다. Ctrl + S(파일 저장), F2(이름 변경)

작업 관리자 실행: Ctrl + Shift + Esc를 눌러 실행 중인 프로그램을 관리할 수 있습니다.


😊 이모지 입력은 윈도우 로고키 + ; (세미콜론)

🔄 시스템이 응답하지 않을 때 Ctrl + Alt + Del



서버와 클라이언트

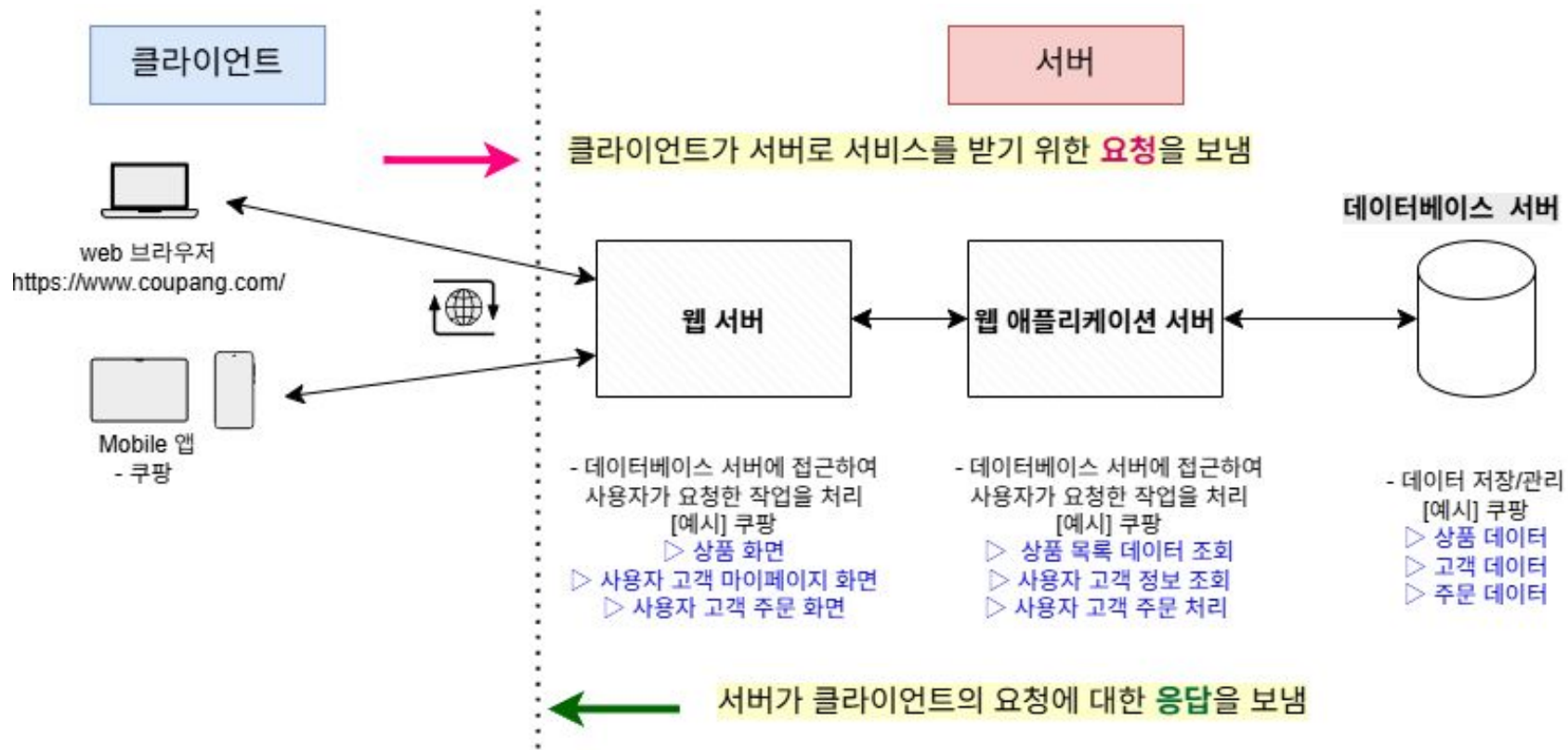
과정 소개



서버(Server)와 클라이언트(Client)는 네트워크 환경에서 서로 다른 역할을 수행하는 두 개념입니다.

- **서버(Server)**: 데이터를 저장하고 관리하며, 클라이언트의 요청을 처리하는 **컴퓨터** 또는 **프로그램**입니다. 웹 서버, 데이터베이스 서버, 게임 서버 등 다양한 유형이 존재하며, 클라이언트에게 필요한 정보를 제공하는 역할을 합니다.
- **클라이언트(Client)**: 서버에 요청을 보내고 응답을 받아 사용하는 **컴퓨터** 또는 **프로그램**입니다. 웹 브라우저, 모바일 앱 등이 클라이언트의 대표적인 예이며, 서버가 제공하는 서비스를 이용하는 역할을 합니다.

비유하자면, 서버는 음식점의 주방처럼 주문을 받고 음식을 준비하는 역할을 하며, 클라이언트는 손님처럼 주문하고 음식을 받는 역할을 합니다.



프론트엔드 (Front-end)

프론트엔드는 사용자가 직접 보는 화면과 상호작용하는 부분을 담당합니다. 웹사이트의 디자인, 버튼, 메뉴, 애니메이션 등을 구현하며, 사용자 경험(UX)과 인터페이스(UI)를 최적화하는 역할을 합니다.

- **주요 프로그래밍 언어 및 기술:**
 - **HTML:** 웹 페이지의 구조를 정의하는 마크업 언어
 - **CSS:** 웹 페이지의 스타일을 지정하는 스타일 시트 언어
 - **JavaScript:** 웹 페이지의 동적 기능을 구현하는 프로그래밍 언어
 - **프레임워크 및 라이브러리:** React, Vue.js, Angular 등

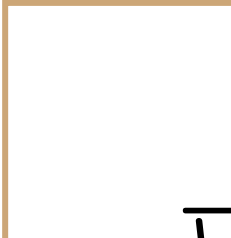
백엔드 (Back-end)

백엔드는 사용자가 직접 볼 수 없는 서버, 데이터베이스, 애플리케이션 로직을 관리하는 부분입니다. 사용자의 요청을 처리하고 데이터를 저장 및 관리하며, 보안과 성능 최적화를 담당합니다.

- **주요 프로그래밍 언어 및 기술:**

- **Java:** 안정성과 확장성이 뛰어난 백엔드 언어
- **Python:** 데이터 처리 및 AI 개발에도 활용되는 범용 언어
- **Node.js:** 서버 기능을 구현 할 수 있는 자바스크립트 백엔드 개발 플랫폼
- **데이터베이스:** 오라클, MySQL, PostgreSQL, MongoDB 등

프론트엔드와 백엔드는 서로 협력하여 웹사이트와 애플리케이션이 원활하게 작동하도록 합니다.



프로그램 개발 도구

Visual Studio Code (VS Code)



IDE

- 통합 개발 환경, **Integrated Development Environment**
- 소프트웨어 개발을 돕기 위한 도구들을 하나의 프로그램에 **통합**한 환경



코드 편집기

문법 강조, **자동완성**, 코드 하이라이팅 등 지원



디버거

코드 오류 추적, 변수 감시, 브레이크포인트 설정



빌드 도구

코드 컴파일 또는 실행 자동화 도구



버전 관리 시스템 연동

Git 등으로 코드 변경 이력 관리



테스트 도구

단위 테스트, 통합 테스트 실행 및 결과 확인

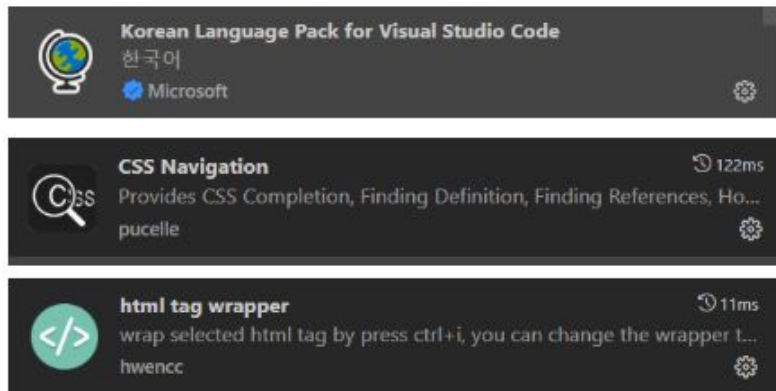


패키지/의존성 관리

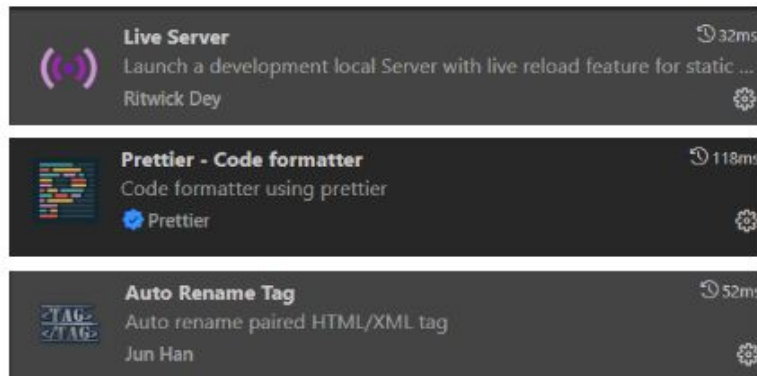
라이브러리 설치 및 업데이트 (ex. npm, pip 등) - 프로젝트 관리

VS Code

- 마이크로소프트에서 개발한 **코드 편집기**
- 설치와 실행이 빠름. **기본 기능에 추가 기능을 제공하는 확장 프로그램 설치**
- 자동완성 등 코드 작성에 도움을 주는 코딩 지원 기능 (**IntelliSense**)
- 프론트엔드, 백엔드 등 모든 언어에서 사용할 수 있는 **IDE** 기능 제공
- 프론트엔드 개발에 유용한 확장 기능
 - VS Code 다운로드하여 설치 후 추가합니다.



(단축키 Ctrl + i)



(단축키 Shift+ Alt + F)

단축키 (1)

파일 및 탐색

- `Ctrl + P` → 빠른 파일 열기
- `Ctrl + G` → 특정 줄로 이동
- `Ctrl + B` → 사이드바 토글
- `Ctrl + Shift + N` → 새 VS Code 창 열기

코드 편집

- `Ctrl + L` → 현재 줄 선택
- `Alt + ↑ / ↓` → 현재 줄 이동
- `Ctrl + Shift + K` → 현재 줄 삭제
- `Alt + Shift + F` → 코드 자동 정렬

검색 및 바꾸기

- `Ctrl + F` → 현재 파일에서 검색
- `Ctrl + H` → 현재 파일에서 텍스트 바꾸기
- `Ctrl + Shift + F` → 전체 프로젝트에서 검색
- `Ctrl + Shift + H` → 전체 프로젝트에서 바꾸기

멀티 커서 및 선택

- `Alt + 클릭` → 클릭한 위치에 커서 추가
- `Ctrl + D` → 같은 단어 선택
- `Ctrl + Shift + L` → 같은 단어 모두 선택
- `Shift + Alt + 드래그` → 세로로 여러 줄 선택 (마우스 사용)
- `Ctrl + Alt + ↑ / ↓` → 세로로 여러 줄 선택 (키보드 사용)

단축키 (2)

메뉴 및 인터페이스

- **Ctrl + Shift + P** → 명령 팔레트 열기 (모든 기능 검색 가능)
- **Ctrl + B** → 사이드바 토글 (탐색기, Git, 확장 등)
- **Ctrl + Shift + X** → 확장 프로그램 메뉴 열기
- **Ctrl + (백틱)** → 터미널 열기/닫기
- **Ctrl + K Z** → **Zen Mode** 활성화 (전체 화면 집중 모드)

탭 및 창 관리

- **Ctrl + Tab** → 열린 파일 간 탭 전환
- **Ctrl + W** → 현재 탭 닫기
- **Ctrl + Shift + N** → 새 **VS Code** 창 열기
- **Ctrl + Shift + W** → 현재 **VS Code** 창 닫기

파일 및 프로젝트 관리

- **Ctrl + N** → 새 파일 생성
- **Ctrl + O** → 파일 열기
- **Ctrl + S** → 현재 파일 저장
- **Ctrl + K S** → 모든 열린 파일 저장
- **Ctrl + K Ctrl + W** → 모든 열린 파일 닫기