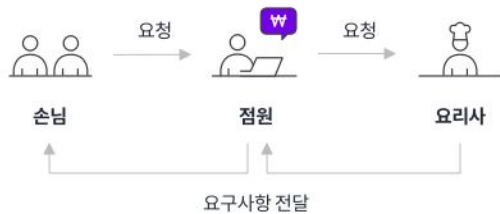


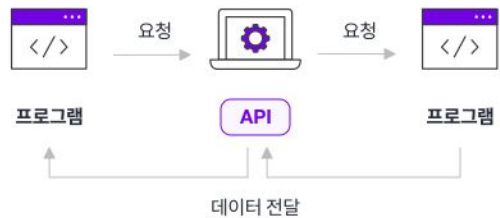
API vs Library vs Framework

API

- 점원의 역할



- API의 역할



API

Application Programming Interface

- 애플리케이션 소프트웨어 및 서비스를 통합하는 정의, 프로토콜의 세트
- 두 응용 프로그램이 서로 통신할 수 있도록 도와주는 소프트웨어 중개자

역할

- 서버와 데이터베이스에 대한 출입구 역할을 한다.
- 애플리케이션과 기기가 원활하게 통신할 수 있도록 한다.
- 모든 접속을 표준화한다.

예

- 오픈 API

라이브러리

- 프로그램을 만드는 데 자주 쓰일 법한 단순 활용가능한 여러 함수들의 모음집

사용 목적

- 프로그램에 필요한 기능, 혹은 그 일부를 이미 만들어진 함수로 사용함으로 해당 함수를 만드는 시간과 노력을 줄이는 것

예

- jQuery

API vs 라이브러리

구현 로직의 유무

프레임워크

프로그래밍의 뼈대가 이미 준비

최소한의 노력으로 프로그램 완성 가능!

프레임워크

특징

- 어떠한 프로그램 제작의 시작부터 끝까지 필요한 모든 기능들을 제공한다.
- 코딩 방법 역시 제공한다.
- 문제를 해결하기 위해 같이 작업하는 여러 클래스나 컴포넌트로 구성되어 있습니다.
- 컴포넌트들은 재사용이 가능합니다.
- 높은 수준에서 패턴들을 조작할 수 있습니다.

예

- Django

라이브러리 vs 프레임워크

제어 흐름에 대한 주도성이 누구에게 있는가



Node.js

프레임워크? 실행 환경!

Express, koa, NestJS