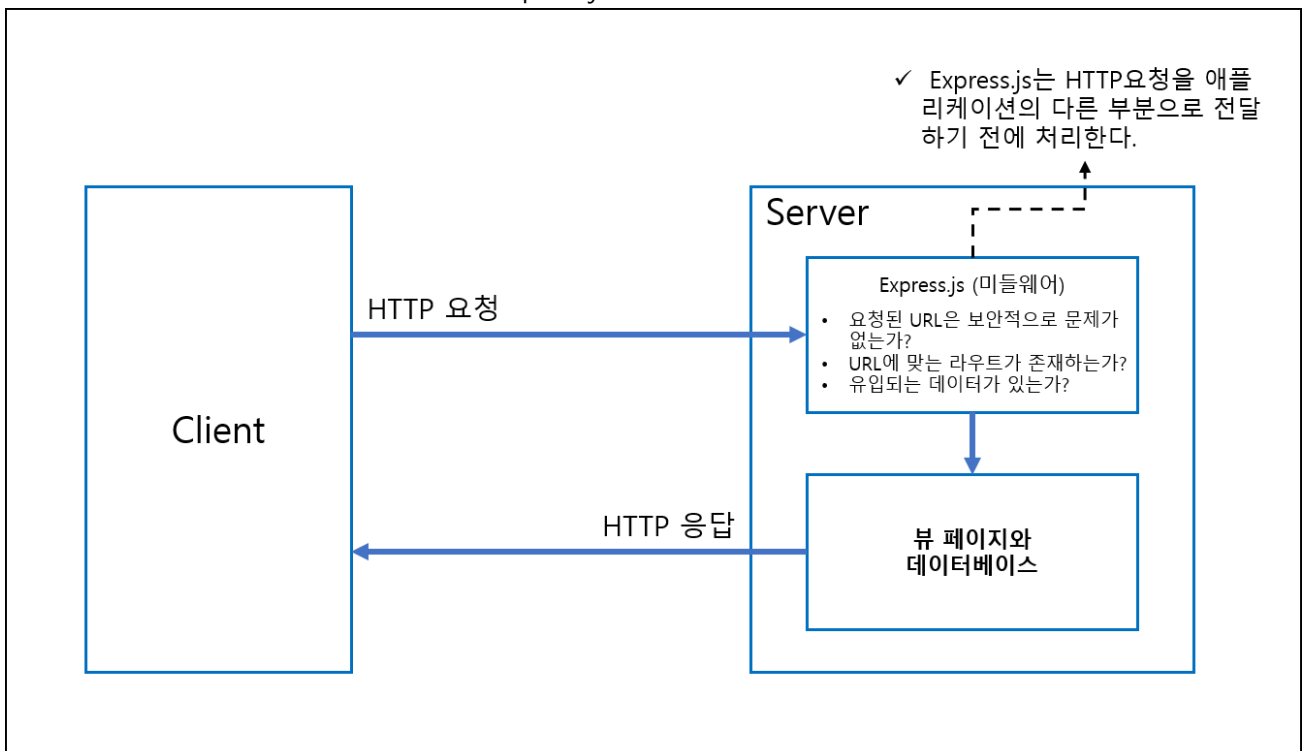


Express

Express

- Express.js는 개발 속도를 올려주고 애플리케이션 개발을 위한 안정적인 구조를 제공하는 웹 프레임워크이다.
- Express.js는 Node.js 커뮤니티에서 가장 많이 쓰이는 웹 프레임워크이며, 다른 프레임워크들에 비해 원하는 것이나 문제에 부딪혔을 때 솔루션을 찾을 확률이 매우 높다. 이것이 Express.js 프레임워크의 큰 장점이라고 할 수 있다.
- 웹 프레임워크란?
 - 웹 개발에 사용될 통상적인 도구를 제공
 - 요청 처리를 도와주고, 동적 및 정적 콘텐츠를 제공.
 - 데이터베이스를 연결하고 사용자 행동을 추적하는 모듈과 메소드를 제공.
- 웹 프레임워크는 보통 미들웨어로 운영된다.
왜냐하면 웹 프레임워크의 위치가 HTTP와 Node.js 플랫폼 사이이기 때문이다.
- 미들웨어란?
 - 애플리케이션 로직과의 데이터 교환 전에 대기하고, 분석하고 필터링하고 HTTP통신을 다루는 코드를 일컫는 일반적인 용어.
- HTTP요청과 애플리케이션 코드 사이의 Express.js의 위치



예제1) Express 프레임워크 기반 간단한 웹 서버

1. mkdir명령어로 웹 루트 디렉터리 생성 (디렉터리 이름은 자유)
2. 생성된 디렉터리로 이동 후 npm init 진행
3. npm init가 끝났다면 Express 프레임워크 설치
\$ npm install express --save
4. main.js에 아래의 내용과 같이 입력
\$ vi main.js

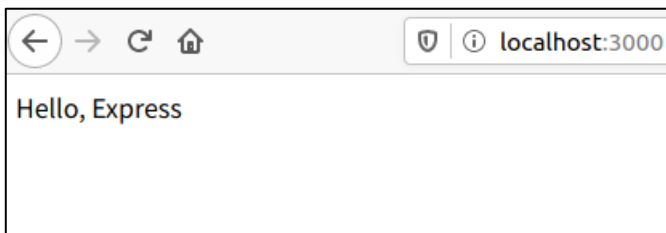
```
"use strict";

const port = 3000;
const express = require("express"); // Express 프레임워크 불러오기
const app = express(); // Express기반 웹 서버 객체 불러오기

app.get("/", function(req, res){ // 홈 페이지("/")의 GET라우트 셋팅
    console.log(req.url); // 요청 URL출력
    console.log(req.params); // URL로부터 ID와 토큰을 추출
    console.log(req.body); // 요청의 Content부분 확인
    console.log(req.query); // URL의 쿼리스트링 출력 (ex. localhost:3000?name=jon)
    res.send("Hello, Express!"); // send로 Client에게 응답
});

app.listen(port, function(){ // 3000번 포트로 웹 서버 실행
    console.log("The Express.js Server has started and is listening on port number : %d", port);
});
```

- ✓ 클라이언트 웹 브라우저 실행화면



- ✓ 서버 콘솔 실행화면

```
wlgns12www@nodejs-test:~/nodejs/9_express$ ls
main.js  node_modules  package-lock.json  package.json
wlgns12www@nodejs-test:~/nodejs/9_express$ node main.js
The Express Server has started and is listening on port : 3000
/   -> req.url
{}  -> req.params
undefined -> req.body
{}  -> req.query
```

참고문헌

- 조나단 웨슬러, 김성준. (2020.01.31). NODEJS로 프로그래밍 시작하기. 페이지. 134-141.