- Node.js 프레임워크 - [hoi-hoon]

- [Express](https://expressjs.com/ko/)

-장점

-Nodejs 웹 프레임워크중 가장 많이 사용.

-내장된 라우터로 코드를 쉽게 재사용 가능.

-많은 써드 파티 플러그인 지원.

-document가 자세함.

-단점

-오류처리기능이 내장되어 있지 않아 오류처리를 직접 구현해야 함.

-koa에 비해 메모리를 많이 차지.

-한 문제를 해결하기 위해 여러 방법으로 접근 가능. Express는 스스로를 완고하다고 표현하는데 초보인 경우에는 단점으로 작용.

- [Koa](https://koajs.com/)

-장점

-가볍고(550line) 표현력이 좋음

-ES6사용 가능

-단점

-아직 불안정(unstable)하다는 의견이 있음.

-개발자인 TJ가 지속적인 유지보수를 하겠다고 하였지만 지금은 Javascript 세계를 떠남.

- [Hapi](https://hapijs.com/)

-장점

-크고 복잡한 어플리케이션에 특화. (월마트 모바일 외 큰 기업들)

-단점

-보일러플레이트(변경없이 계속 재사용 가능한)로 작성해야 할 코드가 많아서 작은 웹앱에서는 쓰기 불편

-Hapi를 사용하는 User pool이 적음. 그만큼 정보 얻기가 어려울 것.

- [Sails](https://sailsjs.com/)

-장점

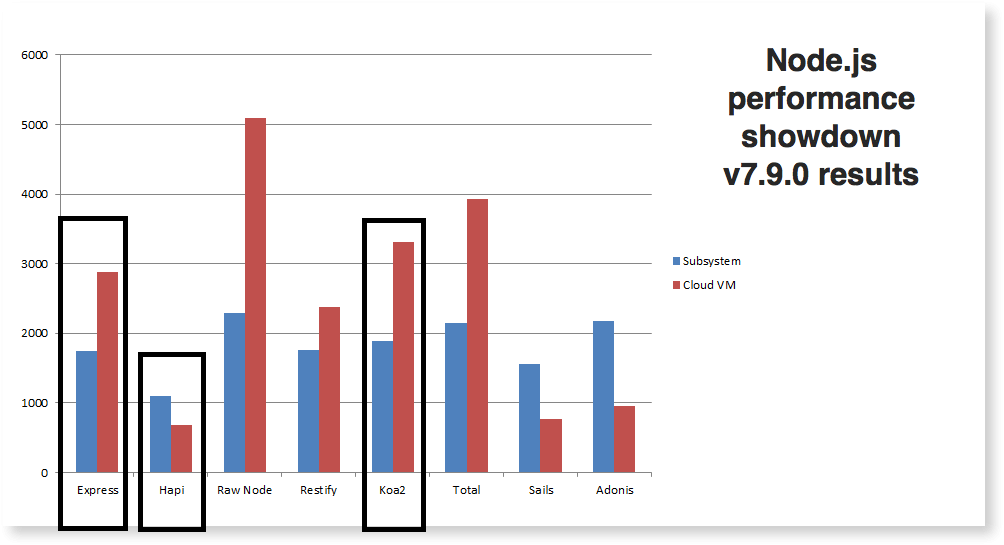
-채팅이나 멀티플레이 게임 같은 실시간 소통(real time communication)에 특화.

-단점

-객체관계매핑(ORM)이 빈약.

- [Ect...](http://nodeframework.com/)

-성능 비교



Frameworks:

Hapi 16.1.1

Express 4.15.2

Restify 4.3.0

Koa 2.0.0

Total.js 2.4.0

Sails 0.12.13

Adonis 3.0.12

Test environments

The Ubuntu subsystem environment running on Windows 10 PC; 32 GB RAM, i7-4790 CPU

A $20 Digital Ocean VM running Ubuntu 16.04 , 2GB Memory, 2 Cores

-출처: <https://raygun.com/blog/nodejs-vs-hapi-express-restify-koa/>

종합

- hapi는 대형 어플리케이션에 최적화 되어있기 때문에 적합하지 않음.

- Express관련 자료가 많아 예제나 정보를 얻기 편함.

- stackoverflow에서 올해 기준 koa 관련질문 약 1300개, express 관련질문 약 4만 7천개

- koa가 성능면에서 아주 약간 우세하고 가볍기까지 함. 프레임워크 자체로만 봤을 때에는 koa를 선택하고 싶지만 User pool과 정보 접근성 같은 외부적 요인이 크게 작용할 것이라 생각함. 따라서 개인적으로 express를 사용하는 것이 좋겠다고 판단하였으나, 이것을 결정하기 위해서는 의견조율이 필요해 보임.