

# GNB

## 웹 프로그래밍

첫번째 수업 자료

Made by 박은빈  
Copy from 양한솔

## INDEX

1. 웹이란?

2. Http 요청과 응답

1) HTTP 프로토콜이란?

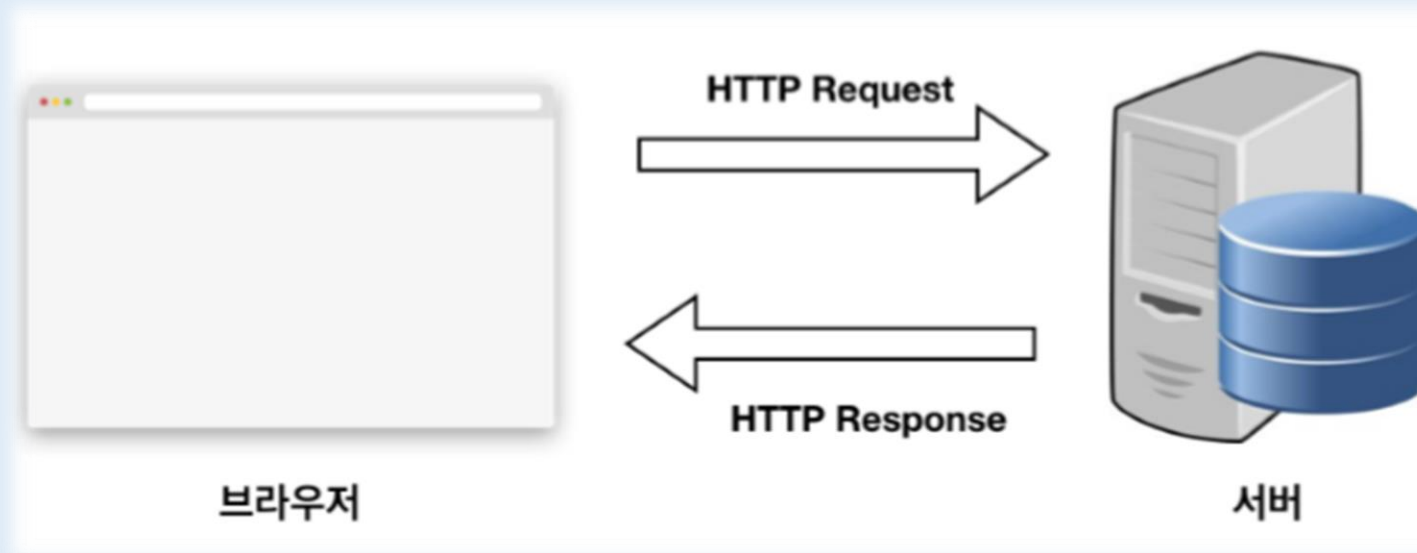
2) HTTP 요청

3) HTTP 응답

3. 실습

## 1. 웹(WEB)이란?

- HTTP(Hyper Text Transfer Protocol) 이라는 통신 규약에 따라, HTML등의 언어로 작성된 정보를 주고 받는 서비스
- 정보를 주고 받기 위해서는 최소 두 대의 컴퓨터가 필요하다.
- 클라이언트 컴퓨터(웹 브라우저), 서버 컴퓨터(웹 서버)가 필요하다.



# 1. 웹(WEB)이란?

- 이해를 위한 예시

In Café

고객: 아메리카노 한 잔 주세요

점원: 주문하신 아메리카노 한 잔 제공 해드릴게요!

In Web

클라이언트: 네이버에 들어가게 해줘

서버: 요청하신 네이버 정보 응답 해드릴게요!

카페에서	웹에서
고객	클라이언트
점원	서버
아메리카노	정보
주문	요청
제공	응답

## 2. HTTP요청과 응답

### 1) HTTP프로토콜이란?

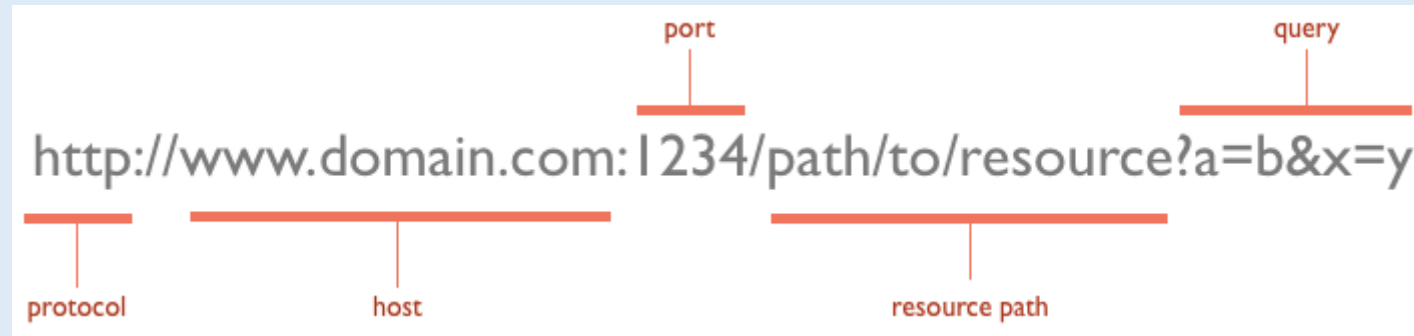
- 통신 프로토콜
- 프로토콜: 상호 간에 정의한 규칙을 의미하며 특정 기기 간에 데이터를 주고받기 위해 정의됨
- “나는 이렇게 줄 테니 넌 이렇게 받고 난 너가 준거 그렇게 받을게”
- HTTP 프로토콜은 상태가 없는(stateless) 프로토콜
- 상태가 없다: 데이터를 주고 받기 위한 각각의 데이터 요청이 서로 독립적으로 관리가 된다. 좀 더 쉽게

말해서 이전 데이터 요청과 다음 데이터 요청이 서로 관련이 없다

## 2. HTTP요청과 응답

### 2) HTTP요청

- URL + 요청 메서드
- URL: (Uniform Resource Locators) 서버에 자원을 요청하기 위해 입력하는 영문 주소
- URL 구조:



## 2. HTTP요청과 응답

### 2) HTTP요청

- URL + 요청 메서드
- HTTP 요청 메서드 : URL로 서버에 특정 데이터를 요청할 때, 그 특정 동작을 수행하고 싶을 때 사용!

Method	Action
POST	Create (생성)
GET	Read (요청)
PUT	Update (변경)
DELETE	Delete (삭제)

## 2. HTTP요청과 응답

### 3) HTTP 응답

- HTTP 상태 코드: 서버에서 설정해주는 응답(Response) 정보
- 요청이 성공적으로 완료되었는가를 나타내 줌
- 참고: <https://developer.mozilla.org/ko/docs/Web/HTTP/Status>

- 예시:

#### 200 OK

요청이 성공적으로 되었습니다. 성공의 의미는 HTTP 메소드에 따라 달라집니다.

GET: 리소스를 불러와서 메시지 바디에 전송되었습니다.

HEAD: 개체 헤더가 메시지 바디에 있습니다.

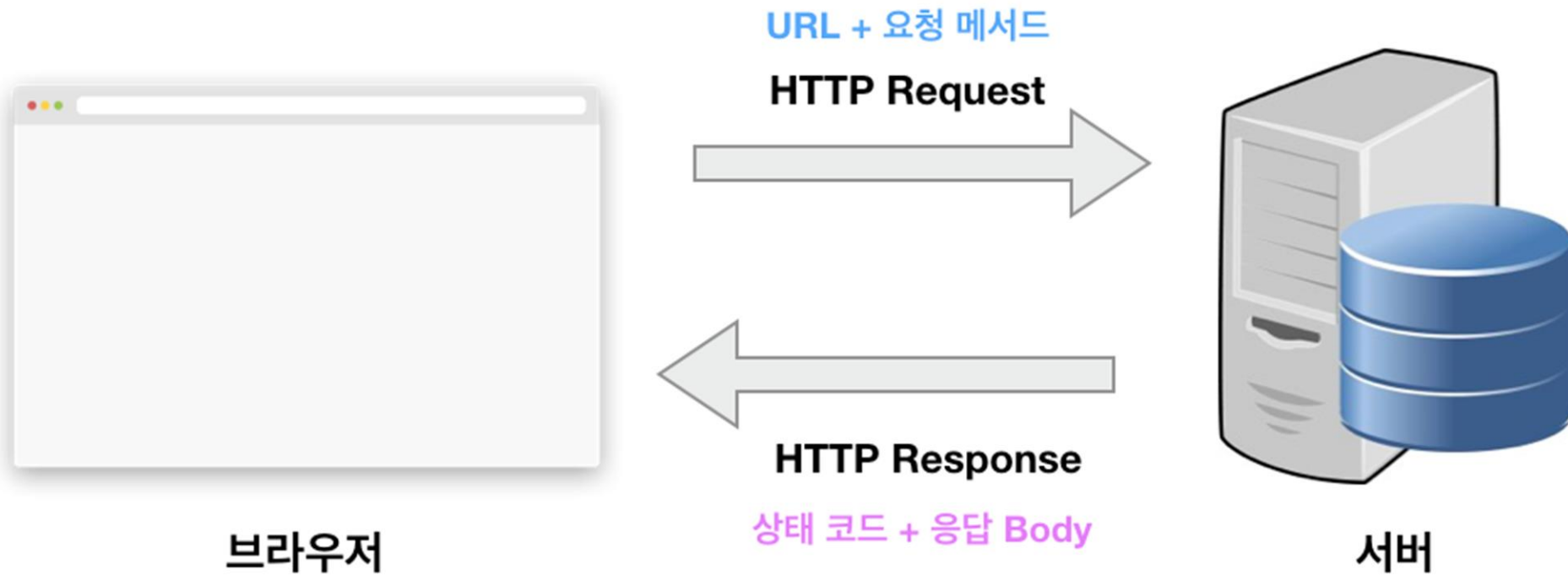
PUT 또는 POST: 수행 결과에 대한 리소스가 메시지 바디에 전송되었습니다.

TRACE: 메시지 바디는 서버에서 수신한 요청 메시지를 포함하고 있습니다.

#### 501 Not Implemented

요청 방법은 서버에서 지원되지 않으므로 처리할 수 없다. 서버가 지원해야 하는 유일한 방법은 GET와 HEAD이다. 이 코드는 반환하면 안됩니다.

## 2. HTTP요청과 응답

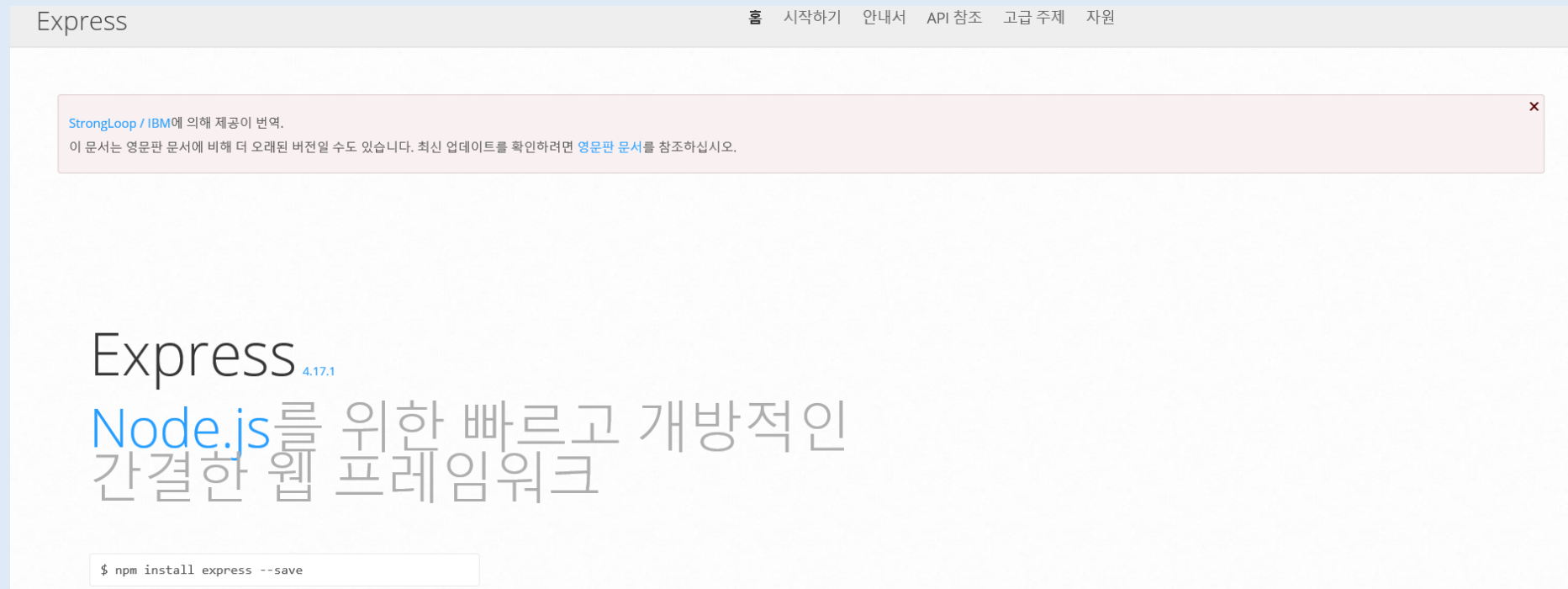




### 3. 실습

-express 사용 : <https://expressjs.com/>

-웹 개발을 위해 필요한 파일을 자동으로 생성해줌



## Express 설치

```
C:\Users\beb99>npm install express-generator -g
```

```
C:\Users\beb99>express -h
```

Usage: express [options] [dir]

Options:

--version	output the version number
-e, --ejs	add ejs engine support
--pug	add pug engine support
--hbs	add handlebars engine support
-H, --hogan	add hogan.js engine support
-v, --view <engine>	add view <engine> support (dust ejs hbs hjs jade pug twig vash) (defaults to jade)
--no-view	use static html instead of view engine
-c, --css <engine>	add stylesheet <engine> support (less stylus compass sass) (defaults to plain css)
--git	add .gitignore
-f, --force	force on non-empty directory
-h, --help	output usage information

```
C:\Users\beb99>express -e hellognb
```

```
warning: option '--ejs' has been renamed to '--view=ejs'
```

```
create : hellognb\
create : hellognb\public\
create : hellognb\public\javascripts\
create : hellognb\public\images\
create : hellognb\public\stylesheets\
create : hellognb\public\stylesheets\style.css
create : hellognb\routes\
create : hellognb\routes\index.js
create : hellognb\routes\users.js
create : hellognb\views\
create : hellognb\views\error.ejs
create : hellognb\views\index.ejs
create : hellognb\app.js
create : hellognb\package.json
create : hellognb\bin\
create : hellognb\bin\www
```

```
change directory:
```

```
> cd hellognb
```

```
install dependencies:
```

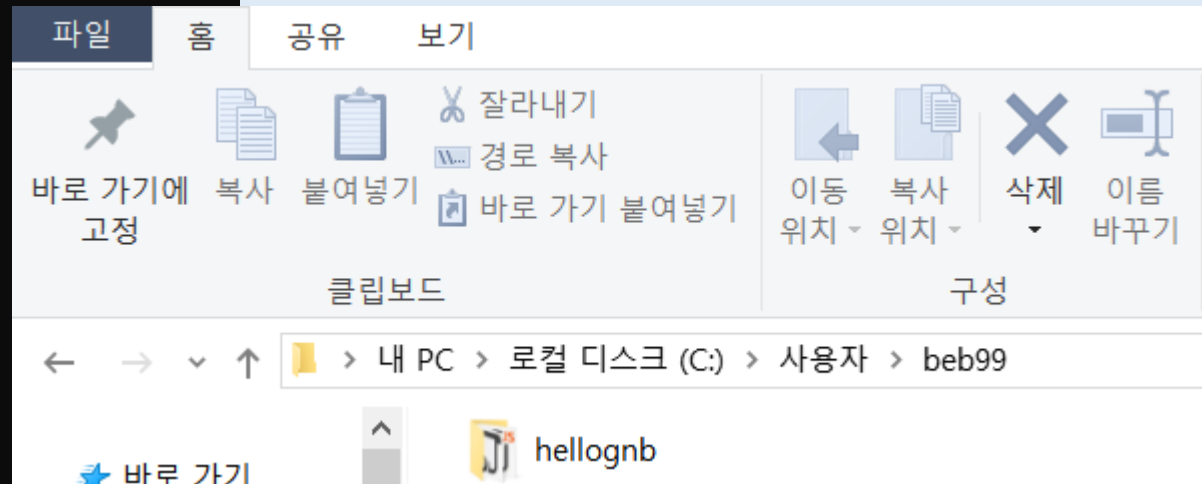
```
> npm install
```

```
run the app:
```

```
> SET DEBUG=hellognb:* & npm start
```

Express 로 hellognb 웹 파일 만들기

폴더 확인하기



- cd 폴더명: 폴더로 들어가기
- cd .. : 상위 폴더로 나가기
- hellognb 폴더로 들어가서 웹을 위한 파일 설치

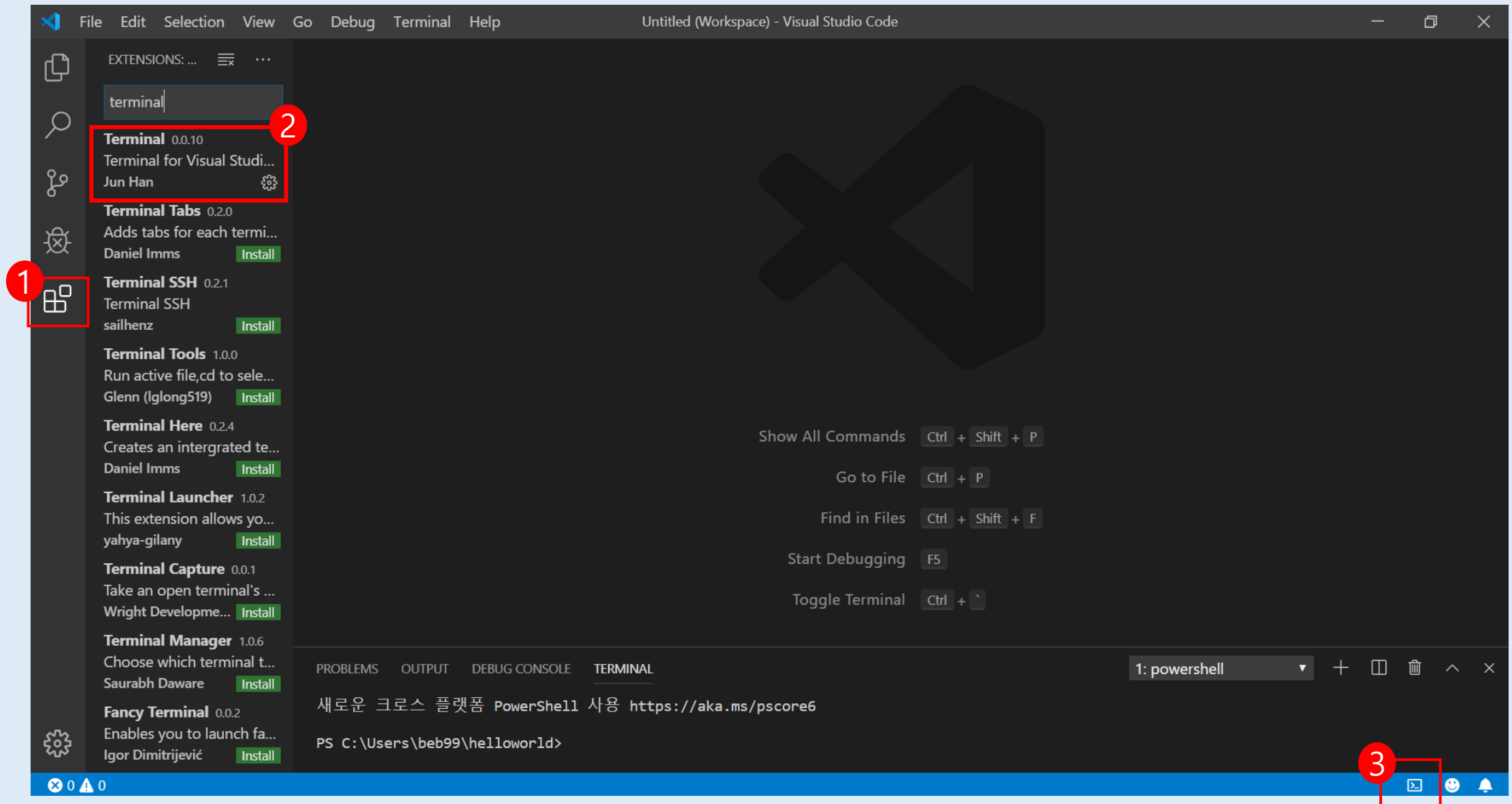
```
C:\Users\beb99>cd hellognb
```

```
C:\Users\beb99\hellognb>npm install
```

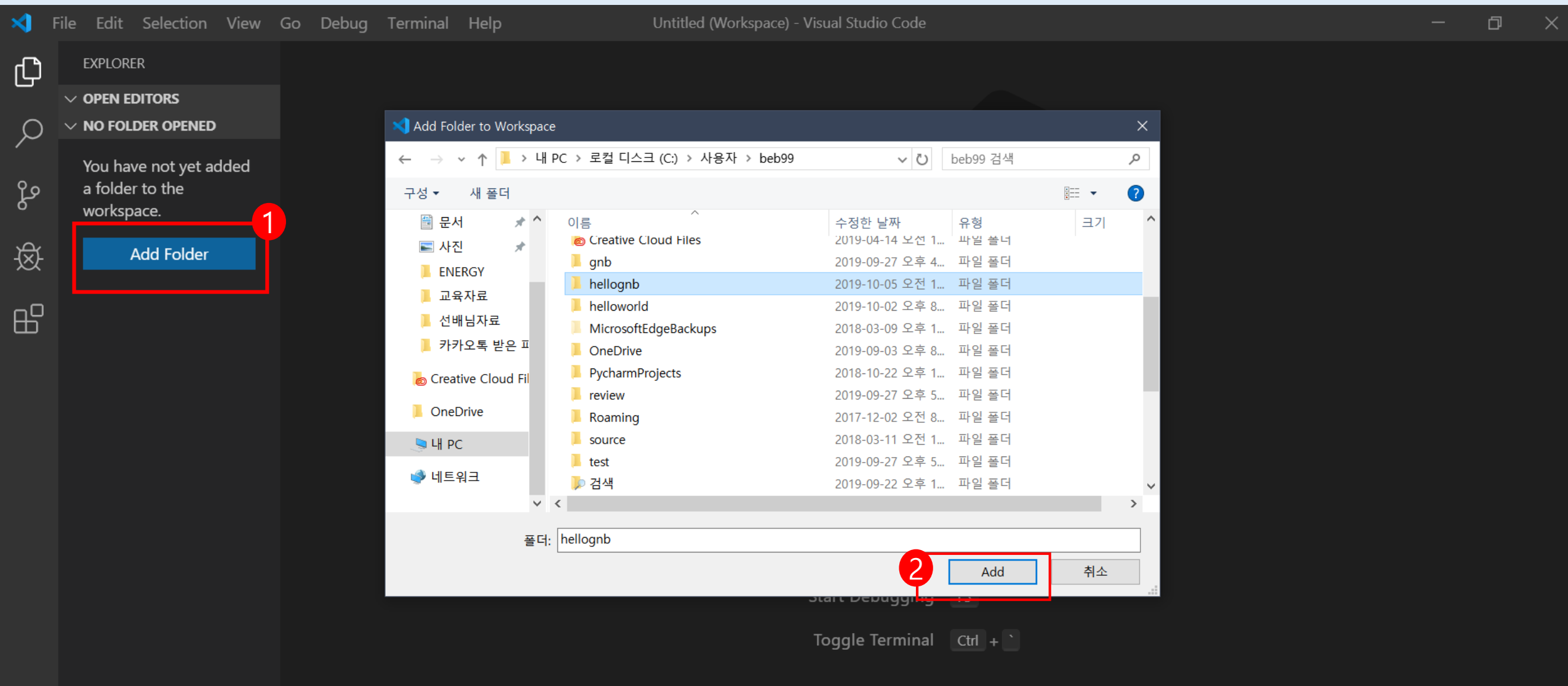
```
npm notice created a lockfile as package-lock.json. You should commit this file.  
added 53 packages from 38 contributors and audited 141 packages in 4.926s  
found 0 vulnerabilities
```

내 PC > 로컬 디스크 (C:) > 사용자 > beb99 > hellognb		
	이름	수정한 날짜
	bin	2019-10-05 오전 12:56
	node_modules	2019-10-05 오전 1:00
	public	2019-10-05 오전 12:56
	routes	2019-10-05 오전 12:56
	views	2019-10-05 오전 12:56
	app.js	2019-10-05 오전 12:56
	package.json	2019-10-05 오전 12:56
	package-lock.json	2019-10-05 오전 1:00

- VS code켜기
- Terminal 설치



## - VS code 에서 폴더 열기



EXPLORER

✓ OPEN EDITORS

- ✕ JS www bin

✓ UNTITLED (WORKSPACE)

✓ hellognb

- ✓ bin
- JS www**
- > node\_modules
- > public
- > routes
- > views

JS app.js

{ } package-lock.json

{ } package.json

JS www

hellognb > bin > JS www > ...

```
1  #!/usr/bin/env node
2
3  /**
4   * Module dependencies.
5   */
6
7  var app = require('../app');
8  var debug = require('debug')('hellognb:server');
9  var http = require('http');
10
11  /**
12   * Get port from environment and store in Express.
13   */
14
15  var port = normalizePort(process.env.PORT || '3000');
16  app.set('port', port);
17
18  /**
19   * Create HTTP server.
20   */
21
22  var server = http.createServer(app);
23
```

- 실행시켜보기
- Chrome에서 실행

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\beb99\hellognb> npm start
```

```
> hellognb@0.0.0 start C:\Users\beb99\hellognb
```

```
> node ./bin/www
```



Express

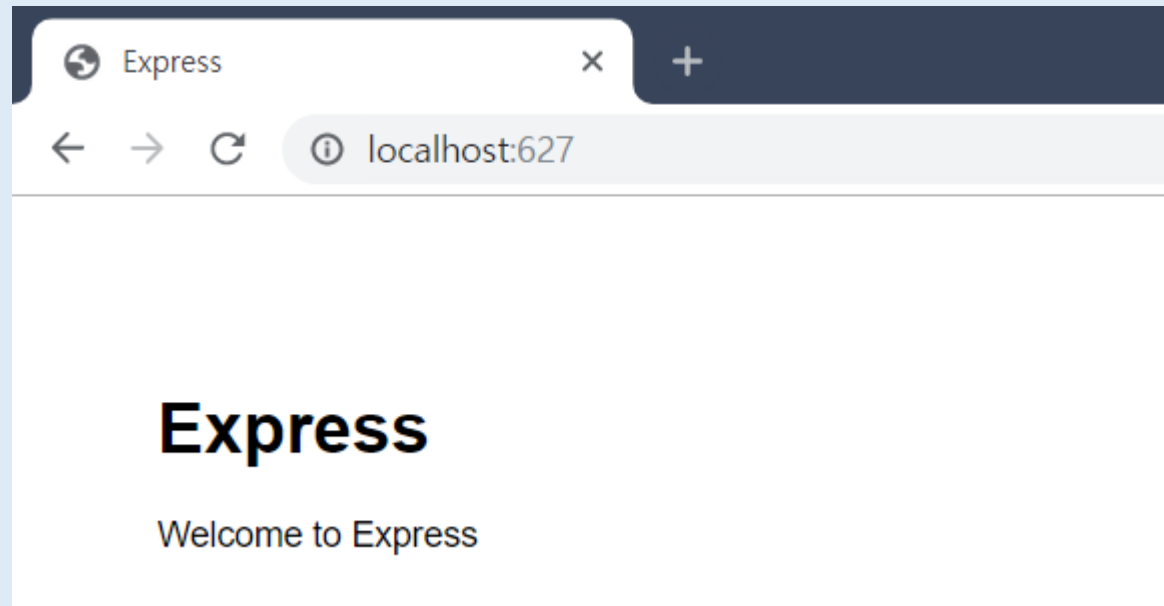
localhost:3000

# Express

Welcome to Express



- 실행 중단: `ctrl+c` -> `y`
- 서버코드를 수정한 후에는 실행 중단 후 다시 실행해야 새로고침 됨
- Html코드 같은 경우는 인터넷창에서만 새로고침 해도 됨
- Misson : portnumber를 본인 생일로 바꿔보자!!



오늘은 여기까지!

수고했어요\*^^\*