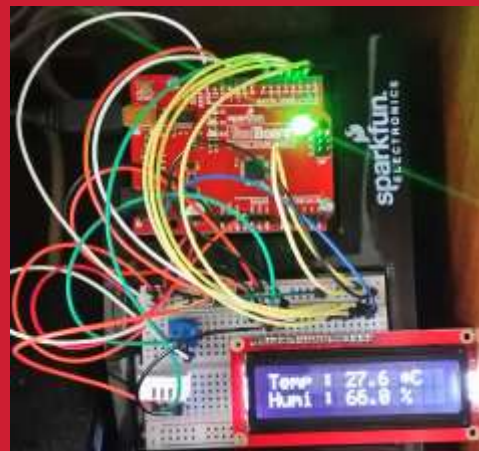




Arduino-IOT

[wk03]

Node.js express



Visualization of Signals using Arduino,
Node.js & storing signals in MongoDB
& mining data using Python

Comsi, INJE University

2nd semester, 2019

Email : chaos21c@gmail.com





My ID

ID	성명
AA01	김관용
AA02	백동진
AA03	김도훈
AA04	김희찬
AA05	류재현
AA06	문민규
AA07	박진석
AA08	이승협
AA09	표혜성
AA10	김다영
AA11	성소진
AA12	김해인
AA13	신송주
AA14	윤지훈



[Review]

◆ [wk02]

- Node module : **aanninfo.js**
- Upload folder : **Aann_Rpt02**



[practice] local module : aanninfo.js

index_aann.js uses local module **aanninfo.js** in start subfolder.

```
1 // index_aann.js
2
3 var myinfo = require('./aanninfo');
4
5 myinfo("aa00", "Redwoods", '010-1234-5678');
6
7 myinfo("aa55", "Comsi", '010-5678-1234');
```

My Info
ID : aa00
Name : Redwoods
Phone : 010-1234-5678

My Info
ID : aa55
Name : Comsi
Phone : 010-5678-1234

[Finished in 0.2s]

Save as
AAnn_info.png



[practice] local module : aanninfo.js

How to make aanninfo.js in start subfolder.

1. Make local module – aanninfo.js
2. Call aanninfo.js from index_aann.js.
3. Capture your result.

```
index_aann.js  x  aanninfo.js  x
1 // aanninfo.js
2
3 module.exports = function(id, name, phone) {
4     console.log("My Info");
5     console.log("ID : " + id);
6     console.log("Name : " + name);
7     console.log("Phone : " + phone + "\n");
8 }
```

[\[참고\] Node local module 만들기](#)

◆ [Target of this week]

My Info using node module – aanninfo.js

Upload folder : AAnn_Rpt02

- 제출할 파일들

- ① **AAnn_package.png**
- ② AAnn_HTTP.png
- ③ AAnn_TCP_Log.png
- ④ AAnn_Upload.png
- ⑤ **AAnn_info.png**
- ⑥ aanninfo.js
- ⑦ index_aann.js



Using node.js

in Sublime text 3

→ Node build system



3.1 Using node.js in Sublime text 3

- Node.js 연결 : Tools > Build System > New Build System

```
{  
    "cmd": ["node.exe path", "$file"],  
    "selector": "source.js"  
}
```

Node.sublime-build 로 저장

D:\Portable\Sublime Text Build 3176\Data\Packages\User\Node.sublime-build (aa00, aann) - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

▶ aa00
▼ aann
▶ start

1 {
2
3
4 }
5

```
"cmd": ["D:\\Portable\\NodeJSPortable\\App\\node.exe", "$file"],  
"selector": "source.js"
```




Node.js Project

npm init



4.2 Node project : npm init

```
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.  
npm WARN npm You can find the latest version at https://nodejs.org/  
This utility will walk you through creating a package.json file.  
It only covers the most common items, and tries to guess sensible defaults.
```

```
See 'npm help json' for definitive documentation on these fields  
and exactly what they do.
```

```
Use 'npm install <pkg>' afterwards to install a package and  
save it as a dependency in the package.json file.
```

```
Press ^C at any time to quit.
```

```
package name: (start)
```

```
version: (1.0.0)
```

```
description: start node project
```

```
entry point: (index.js)
```

```
test command:
```

```
git repository:
```

```
keywords: test
```

```
author: aa00
```

```
license: (ISC) MIT
```

```
About to write to D:\Portable\NodeJSPortable\Data\ann\start\package.json:
```

```
{  
  "name": "start",  
  "version": "1.0.0",  
  "description": "start node project",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \"Error: no test specified\" && exit 1"  
  },  
  "keywords": [  
    "test"  
  ],  
  "author": "aa00",  
  "license": "MIT"  
}
```

```
Is this OK? (yes)
```

// node project

npm init

→ package.json



5. Node Apps

Node Apps



5.2.3 Using local module – hello_mymodule.js

FOLDERS

- aa00
 - aann
 - start
 - hello.js
 - hello_function.js
 - hello_module.js
 - hello_mymodule.js
 - package.json

hello_mymodule.js x

```
1 // hello_mymodule.js
2
3 module.exports = function(what) {
4     console.log("Hello " + what + " !");
5 }
6
```

사용자 정의 모듈

FOLDERS

- aa00
 - aann
 - start
 - hello.js
 - hello_call_module.js
 - hello_function.js
 - hello_module.js
 - hello_mymodule.js
 - package.json

hello_mymodule.js x hello_call_module.js x

```
1 // hello_call_module.js
2 var olleh = require('./hello_mymodule.js');
3
4 olleh("Node");
5 olleh("aa00");
6
7
```

```
Hello Node !
Hello aa00 !
[Finished in 0.2s]
```

Call local module 'hello_mymodule.js' from hello_call_module.js



[extra code] local module : **circle.js**

circle_info.js uses local module **circle.js**.

```
1 // circle.js
2 var PI = Math.PI;
3
4 module.exports.area = function (r) {
5     return PI * r * r;
6 };
7
8 module.exports.circumference = function (r) {
9     return 2 * PI * r;
10 };
```

circle.js



```
1 // circle_info.js
2 var circle = require('./circle.js');
3 console.log( 'The area of a circle of radius 4 is '
4     + circle.area(4).toFixed(2));
5 console.log( 'The circumference of a circle of radius 4 is '
6     + circle.circumference(4).toFixed(2));
```

circle_info.js

```
The area of a circle of radius 4 is 50.27
The circumference of a circle of radius 4 is 25.13
[Finished in 0.2s]
```



Node.js Server

1. http, tcp, file

2. Express



6. Node Server

Node Server I.

- 1. HTTP server**
- 2. TCP server**
- 3. File upload**



Node Server II.

- 1. Express server**
- 2. Full Express App**
- 3. My Express App**



7.1.1 Express server test

Step 1 : npm init

Step 2 : npm install --save express

Step 3 : Write Express code

Step 4 : Run app.js

Step 5 : http://localhost:3000



7.1.2 Express server test

1. Make folder expressTest

2. Go to the folder,
"expressTest"

3. npm init

```
D:\Portable\NodeJSPortable\Data>dir
D 드라이브의 볼륨: DATA
볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data 디렉터리

2018-09-10 오후 12:40 <DIR> .
2018-09-10 오후 12:40 <DIR> ..
2018-09-03 오전 11:12 <DIR> aa00
2018-09-03 오후 04:33 <DIR> aann
2018-09-10 오후 12:40 89 PortableApps.comLauncherRuntimeData-
e.ini
2018-09-03 오전 10:23 <DIR> settings
                        1개 파일 89 바이트
                        5개 디렉터리 851,459,661,824 바이트 남음

D:\Portable\NodeJSPortable\Data>cd aann
D:\Portable\NodeJSPortable\Data\aaann>md expressTest
D:\Portable\NodeJSPortable\Data\aaann>cd expressTest
D:\Portable\NodeJSPortable\Data\aaann\expressTest>npm init
```



7.1.2 Express server test: npm init

FOLDERS

▶ aa00

▼ aann

▼ expressTest

/* package.json

▼ server

▶ File

▶ HTTP

▶ TCP

▶ start

npm

About to write to D:\Portable\NodeJSPortable\Data\ann\expressTest\package.json:

```
{
  "name": "expresstest",
  "version": "1.0.0",
  "description": "test express server",
  "main": "app.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "aa00",
  "license": "MIT"
}
```

Is this OK? (yes)

D:\Portable\NodeJSPortable\Data\ann\expressTest>dir

D 드라이브의 볼륨: DATA

볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\ann\expressTest 디렉터리

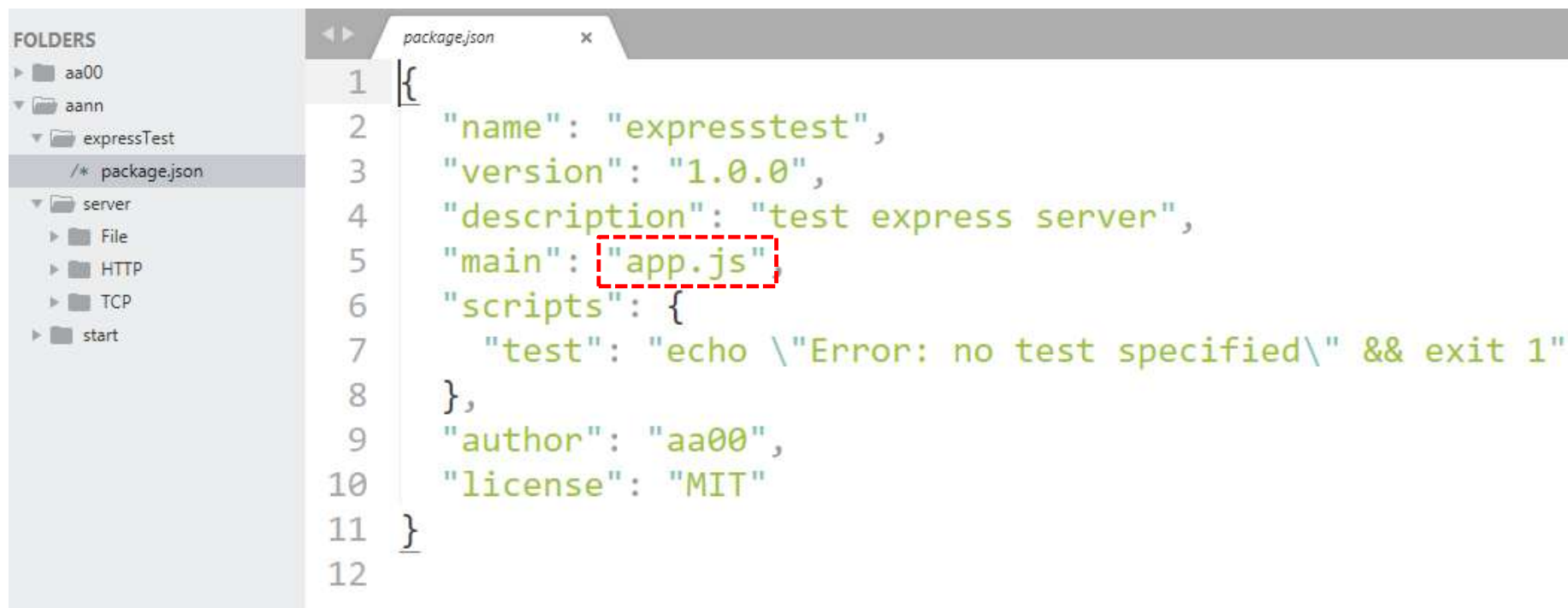
```
2018-09-10 오후 12:43 <DIR> .
2018-09-10 오후 12:43 <DIR> ..
2018-09-10 오후 12:43                228 package.json
                        1개 파일                228 바이트
                        2개 디렉터리  851,459,645,440 바이트 남음
```

D:\Portable\NodeJSPortable\Data\ann\expressTest>



7.1.3 Express server test: package.json

package.json



```
1 {
2   "name": "expresstest",
3   "version": "1.0.0",
4   "description": "test express server",
5   "main": "app.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "author": "aa00",
10  "license": "MIT"
11 }
12
```



7.1.4 Express server test: express module install

npm install --save express

```
npm
D:\Portable\NodeJSPortable\Data\ann\expressTest>npm install --save express
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN express@1.0.0 No repository field.

+ express@4.16.3
added 50 packages from 47 contributors in 5.159s

D:\Portable\NodeJSPortable\Data\ann\expressTest>dir
D 드라이브의 볼륨: DATA
볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\ann\expressTest 디렉터리

2018-09-10 오후 12:48 <DIR> .
2018-09-10 오후 12:48 <DIR>
2018-09-10 오후 12:48 <DIR> node_modules
2018-09-10 오후 12:48 13,427 package-lock.json
2018-09-10 오후 12:48 278 package.json
2개 파일 13,705 바이트
3개 디렉터리 851,460,452,352 바이트 남음
```



7.1.5 Express server test: express install

package.json

```
1 {
2   "name": "expresstest",
3   "version": "1.0.0",
4   "description": "",
5   "main": "app.js",
6   "scripts": {
7     "test": "echo \"Error: no test specified\" && exit 1"
8   },
9   "author": "aa00",
10  "license": "MIT",
11  "dependencies": {
12    "express": "^4.17.1"
13  }
14 }
15
```

expressTest 폴더 내의 **node_modules** subfolder에 **express** **server modules**들이 저장되어 서버 기능을 지원.
그리고 **package.json**에 **express** 모듈 정보가
“**dependencies**” 속성에 저장.



7.1.6 Express server test: app.js

FOLDERS

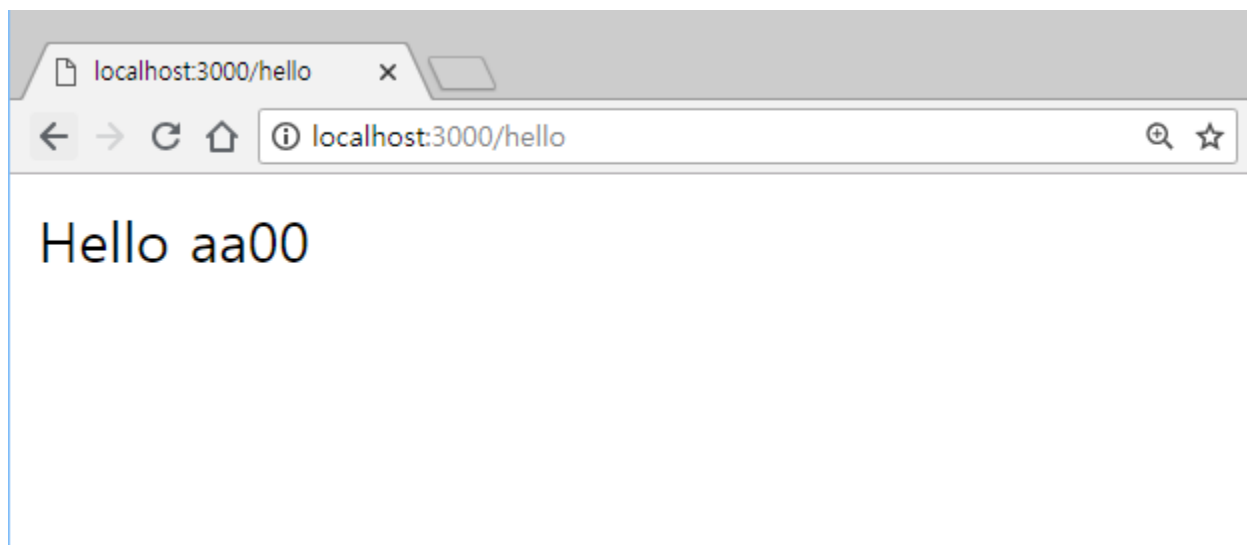
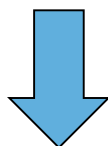
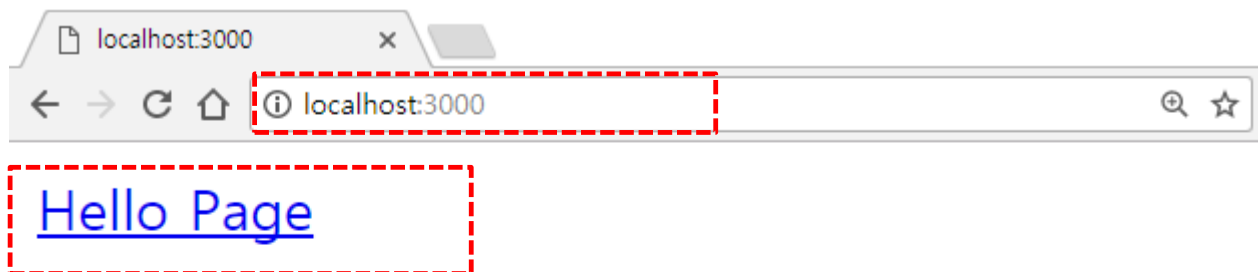
- ▶ aa00
- ▼ aann
 - ▶ expressTest
 - ▶ node_modules
 - /* app.js
 - /* package-lock.json
 - /* package.json
 - ▶ server
 - ▶ File
 - ▶ HTTP
 - ▶ TCP
 - ▶ start

```
app.js
1 // app.js
2 var express = require('express');
3 var app = express();
4 var port = 3000;
5
6 app.get('/', function(req, res) {
7   res.send('<a href="/hello">Hello Page</a>');
8 });
9
10 app.get('/hello', function(req, res) {
11   res.send('Hello aa00');
12 });
13
14 app.get('/comsi', function(req, res) {
15   res.send('Hello Comsi!');
16 });
17
18
19
20 var server = app.listen(port, function() {
21   console.log('Listening on port %d', server.address().port);
22 });
23
24
```

Listening on port 3000

7.1.7 Express server test: run `app.js`

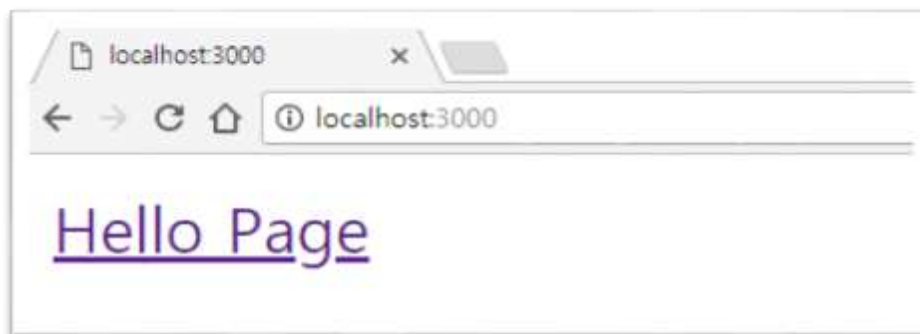
^B (실행 전 Tools > Cancel Build 해제 확인!)



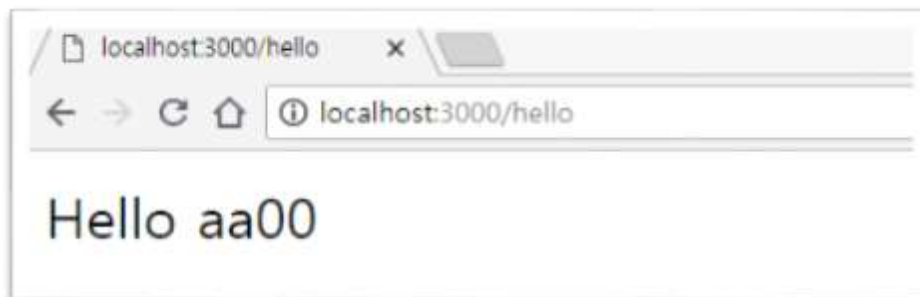


7.1.8 Express server test: test server **routing**

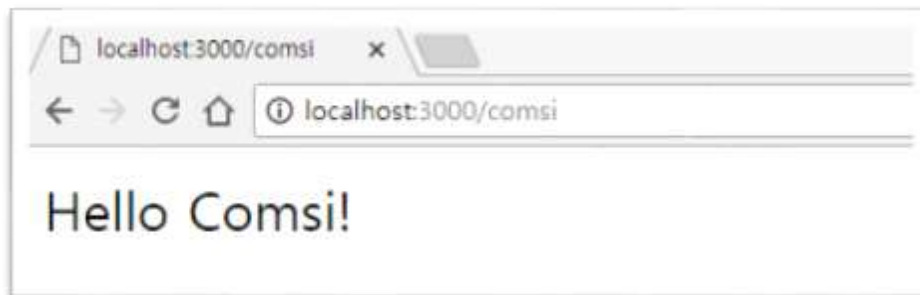
localhost:3000



localhost:3000/hello



localhost:3000/comsi



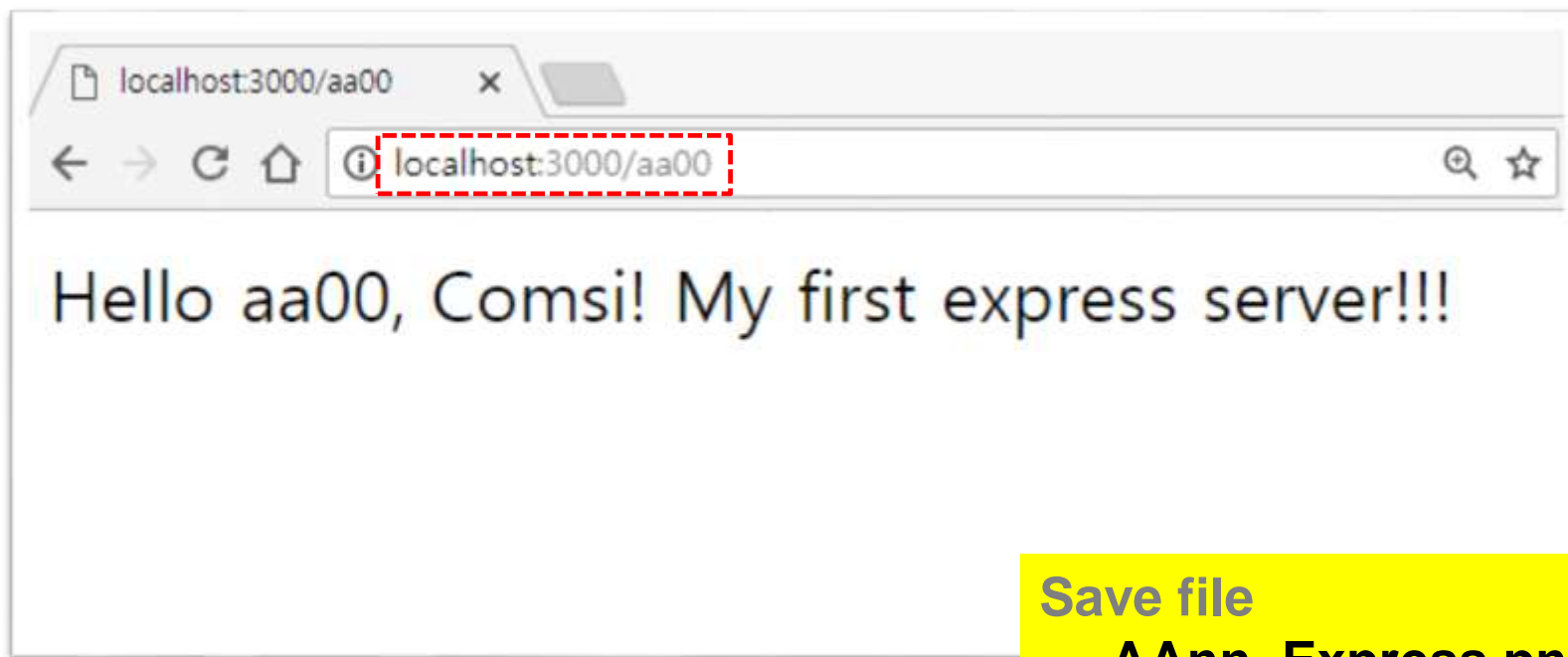
^B (실행 전 Tools > Cancel Build 해제 확인!)



7.1.9 Express server test: run `app.js` – DIY

[DIY] My ID routing → `localhost:3000/aann`

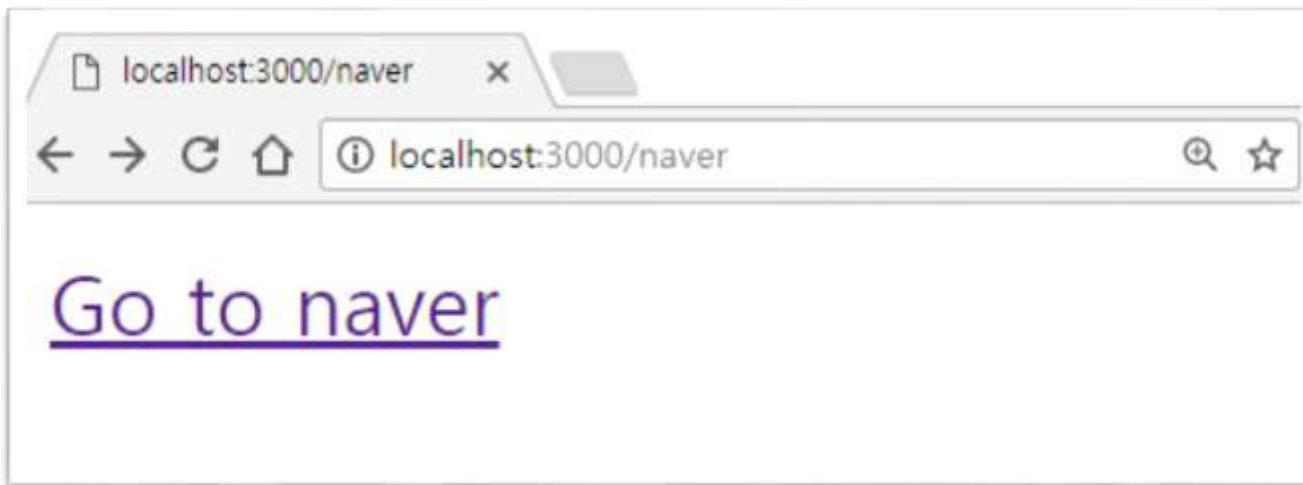
```
app.get('/aa00', function(req, res) {  
  res.send('Hello aa00, Comsi! My first express server!!!');  
});
```



Save file
AAnn_Express.png

Routing: 라우팅은 애플리케이션 엔드 포인트(**URI**)의 정의, 그리고 **URI**가 클라이언트 요청에 응답하는 방식

[DIY] Go to naver.com



[Hint] `Go to naver`



Full Express App using generator



7.2.1 Full Express App using generator

Step 1 : make folder 'express'

Step 2 : go to the folder

Step 3 : `npm install -g express-generator`

Step 4 : `express`

Step 5 : `npm install`

Step 6 : `node ./bin/www`

Step 6 : <http://localhost:3000>



7.2.2 Install Express-generator

1. Make folder express
2. Go to the folder, "express"
3. **npm install -g express-generator**

```
npm
D:\Portable\NodeJSPortable\Data\ann\expressTest>cd ..
D:\Portable\NodeJSPortable\Data\ann>md express
D:\Portable\NodeJSPortable\Data\ann>cd express
D:\Portable\NodeJSPortable\Data\ann\express>npm install -g express-generator
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
D:\Portable\NodeJSPortable\AppData\express -> D:\Portable\NodeJSPortable\AppData\node_modules\exp
ess-generator\bin\express-cli.js
+ express-generator@4.16.0
added 10 packages from 13 contributors in 1.801s

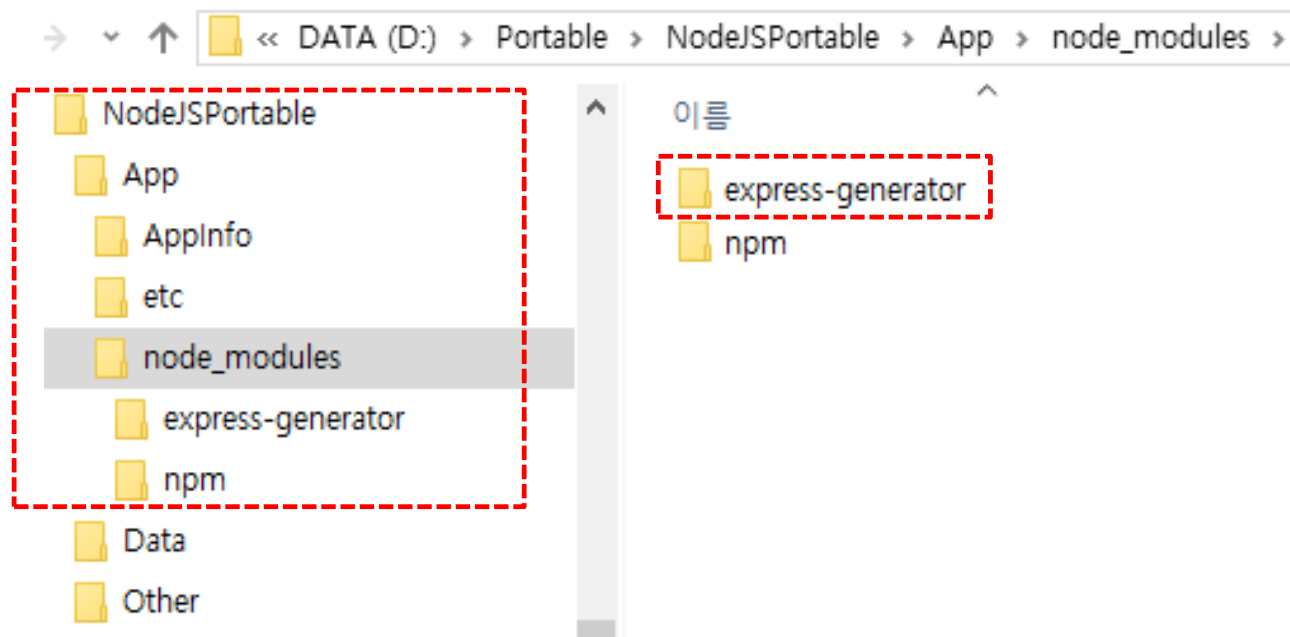
D:\Portable\NodeJSPortable\Data\ann\express>dir
D 드라이브의 볼륨: DATA
볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\ann\express 디렉터리

2018-09-10 오후 03:33 <DIR> .
2018-09-10 오후 03:33 <DIR> ..
                0개 파일                0 바이트
                2개 디렉터리  851,458,920,448 바이트 남음

D:\Portable\NodeJSPortable\Data\ann\express>
```

Global folder





7.2.3 Run Express-generator : express

ca. npm

```
D:\Portable\NodeJSPortable\Data\aan\express>express
```

```
warning: the default view engine will not be jade in future releases  
warning: use '--view=jade' or '--help' for additional options
```

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

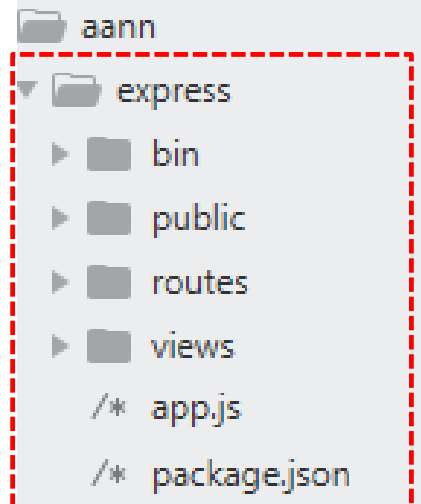
```
install dependencies:
```

```
> npm install
```

```
run the app:
```

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

```
D:\Portable\NodeJSPortable\Data\aan\express>■
```



‘express’ 명령은 현재 폴더에 Express 서버를 구성해서 초기화한다.



7.2.4 package.json

FOLDERS

- ▶ aa00
- ▼ aann
 - ▼ express
 - ▶ bin
 - ▶ public
 - ▶ routes
 - ▶ views
 - /* app.js
 - /* package.json
 - ▶ expressTest
- ▼ server
 - ▶ File
 - ▶ HTTP
 - ▶ TCP
 - ▶ start

```
package.json
1 {
2   "name": "express",
3   "version": "0.0.0",
4   "private": true,
5   "scripts": {
6     "start": "node ./bin/www"
7   },
8   "dependencies": {
9     "cookie-parser": "~1.4.4",
10    "debug": "~2.6.9",
11    "express": "~4.16.1",
12    "http-errors": "~1.6.3",
13    "jade": "~1.11.0",
14    "morgan": "~1.9.1"
15  }
16 }
```

1. Install node modules that is defined in package.json

2. npm install



7.2.5 install node modules: npm install

```

D:\Portable\NodeJSPortable\Data\ann\express>npm install
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN deprecated jade@1.11.0: Jade has been renamed to pug, please install the latest version of pug instead of jade
npm WARN deprecated constantinople@3.0.2: Please update to at least constantinople 3.1.1
npm WARN deprecated transformers@2.1.0: Deprecated, use jstransformer
npm WARN notice [SECURITY] uglify-js has the following vulnerability: 2 low. Go here for more details: https://nodesecurity.io/advisories?search=uglify-js&version=2.2.5 - Run 'npm i npm@latest -g' to upgrade your npm version, and then 'npm audit' to get more info.
npm WARN notice [SECURITY] constantinople has the following vulnerability: 1 critical. Go here for more details: https://nodesecurity.io/advisories?search=constantinople&version=3.0.2 - Run 'npm i npm@latest -g' to upgrade your npm version, and then 'npm audit' to get more info.
npm notice created a lockfile as package-lock.json. You should commit this file.
added 101 packages from 150 contributors in 5.17s
  
```

```
D:\Portable\NodeJSPortable\Data\ann\express>dir
```

D 드라이브의 볼륨: DATA
볼륨 일련 번호: 7A01-106A

```
D:\Portable\NodeJSPortable\Data\ann\express 디렉터리
```

```

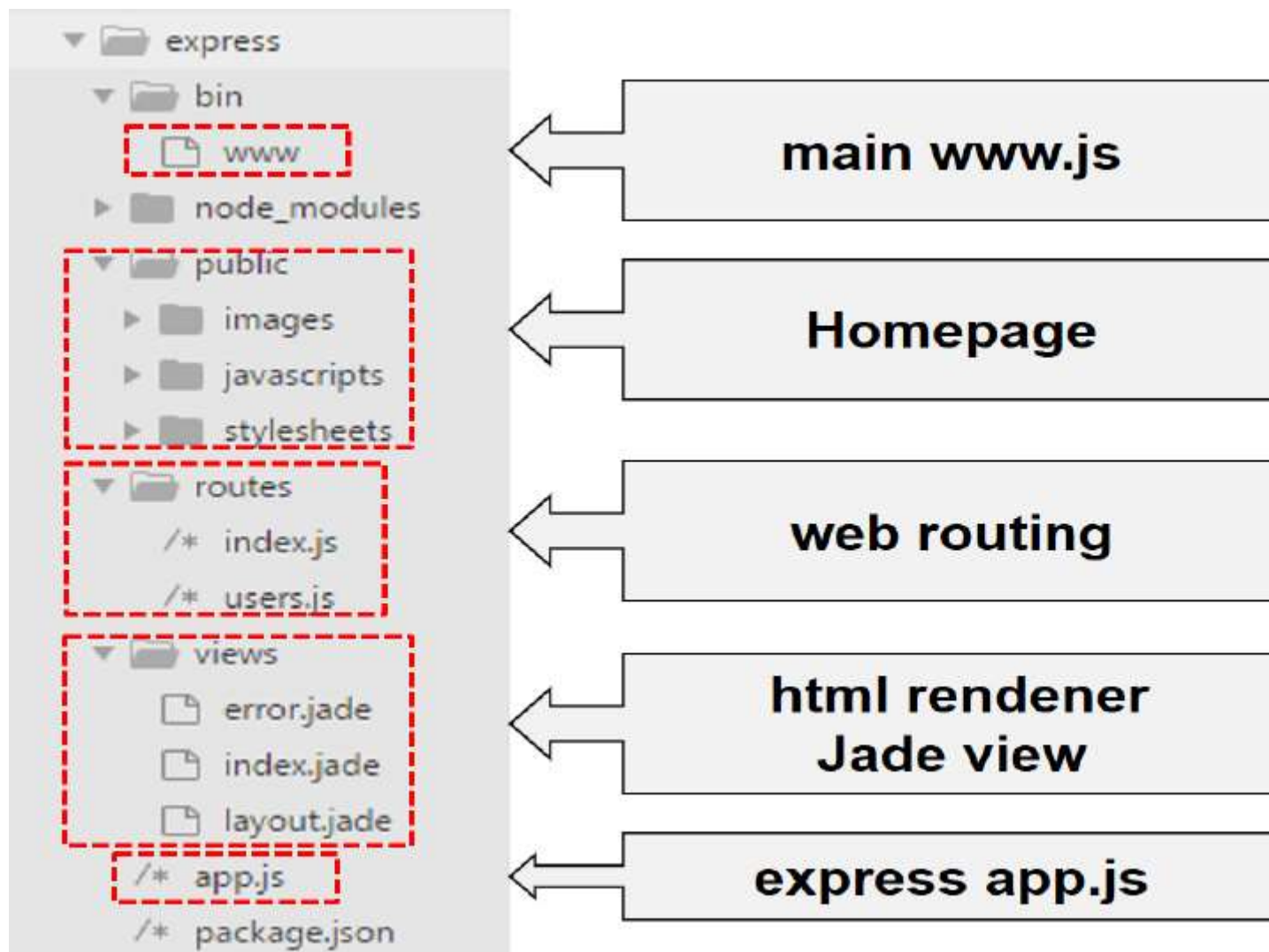
2018-09-10 오후 03:47 <DIR>      .
2018-09-10 오후 03:47 <DIR>      ..
2018-09-10 오후 03:42          1,075 app.js
2018-09-10 오후 03:42 <DIR>      bin
2018-09-10 오후 03:47 <DIR>      node_modules
2018-09-10 오후 03:47          25,938 package-lock.json
2018-09-10 오후 03:42          295 package.json
2018-09-10 오후 03:42 <DIR>      public
2018-09-10 오후 03:42 <DIR>      routes
2018-09-10 오후 03:42 <DIR>      views
                3개 파일          27,308 바이트
                7개 디렉터리  851,450,929,152 바이트 남음
  
```

1. Install node modules that is defined in package.json

2. npm install

3. Modules were installed in node_modules subfolder.

7.2.6 Inner structure of express server





7.2.7 run bin/www

1. Install node modules that is defined in package.json
2. npm install
3. Modules were installed in node_modules subfolder.

4. Run node express web

node ./bin/www

or

npm start

```
28 server.listen(port);
29 server.on('error', onError);
30 server.on('listening', onListening);
31 console.log("Server Running on " + port +
32   ".\nLaunch http://localhost:" + port);
```



7.2.8 test app

npm - node ./bin/www

D:\Portable\NodeJSPortable\Data\aaann\express>node ./bin/www

Server Running on 3000.

Launch http://localhost:3000

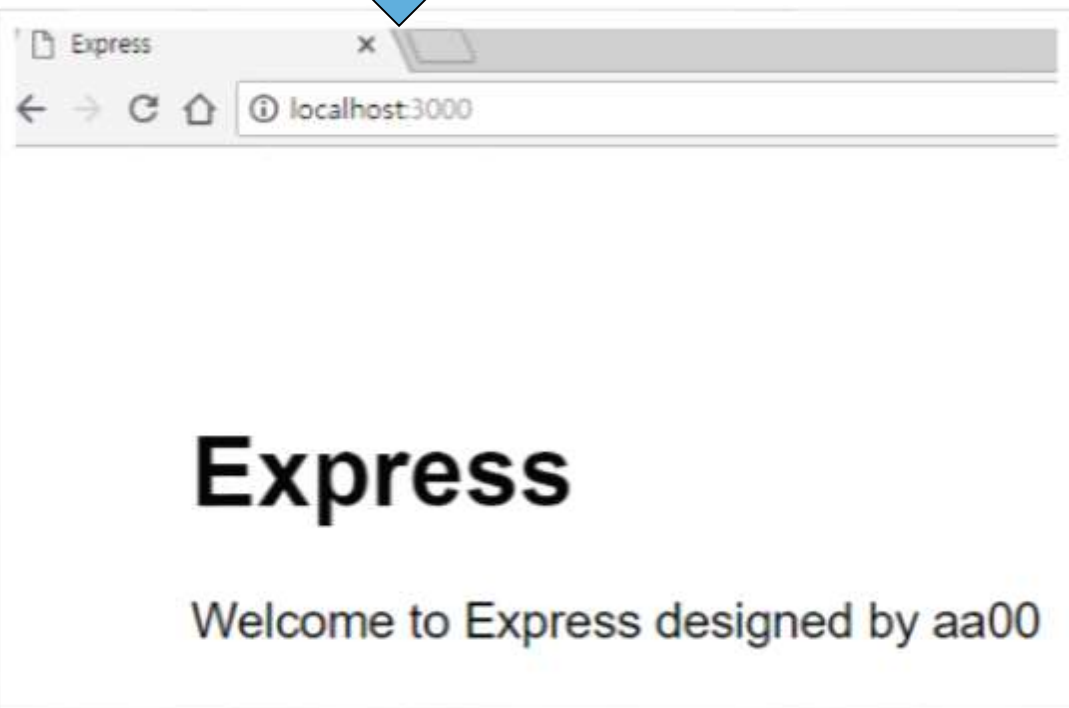
D:\Portable\NodeJSPortable\Data\aaann\express>
D:\Portable\NodeJSPortable\Data\aaann\express>

Server Running on 3000.

Launch http://localhost:3000

GET / 304 281.411 ms - -
GET /stylesheets/style.css 304 2.185 ms - -
GET / 304 14.013 ms - -
GET /stylesheets/style.css 304 0.433 ms - -
GET / 304 24.983 ms - -
GET /stylesheets/style.css 304 0.334 ms - -

Web log



Save file

AAnn_Express_App.png



7.2.9 app.js - 1

```
app.js x
1 var createError = require('http-errors');
2 var express = require('express');
3 var path = require('path');
4 var cookieParser = require('cookie-parser');
5 var logger = require('morgan');
6
7 var indexRouter = require('./routes/index');
8 var usersRouter = require('./routes/users');
9
10 var app = express();
11
12 // view engine setup
13 app.set('views', path.join(__dirname, 'views'));
14 app.set('view engine', 'jade');
15
16 app.use(logger('dev'));
17 app.use(express.json());
18 app.use(express.urlencoded({ extended: false }));
19 app.use(cookieParser());
20 app.use(express.static(path.join(__dirname, 'public')));
21
22 app.use('/', indexRouter);
23 app.use('/users', usersRouter);
24
```

routing

View engine
jade

Homepage

Set routing



7.2.9 app.js - 2

```
app.js x
25 // catch 404 and forward to error handler
26 app.use(function(req, res, next) {
27   next(createError(404));
28 });
29
30 // error handler
31 app.use(function(err, req, res, next) {
32   // set locals, only providing error in development
33   res.locals.message = err.message;
34   res.locals.error = req.app.get('env') === 'development' ? err : {};
35
36   // render the error page
37   res.status(err.status || 500);
38   res.render('error');
39 });
40
41 module.exports = app;
42
```



7.2.10 ./bin/www.js - 1

```
1 #!/usr/bin/env node
2
3 /**
4  * Module dependencies.
5  */
6
7 var app = require('../app');
8 var debug = require('debug')('express: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
```

app.js

모듈로

구성된

http

서버를

만든다.



7.2.10 ./bin/www.js - 2

```
24 ▾ /**
25  * Listen on provided port, on all network interfaces.
26  */
27
28  server.listen(port);
29  server.on('error', onError);
30  server.on('listening', onListening);
31  console.log("Listening on port: " + port);
32
33 ▾ /**
34  * Normalize a port into a number, string, or false.
35  */
36
37 ▾ function normalizePort(val) {
38   var port = parseInt(val, 10);
39
40   if (isNaN(port)) {
41     // named pipe
42     return val;
43   }
44
45   if (port >= 0) {
46     // port number
47     return port;
48   }
49   return false;
50 }
51 }
```

실행 하면 출력을 추가

```
53 /**
54  * Event listener for HTTP server "error" event.
55  */
56
57 function onError(error) { ...
79 }
80
81 /**
82  * Event listener for HTTP server "listening" event.
83  */
84
85 function onListening() { ...
92 }
93
```



7.2.11 routes/index.js

웹 브라우저에서 “**http://localhost:3000**”으로 연결하면 기본 웹 링크인 ‘/’에 지정된 **javascript** 코드인 **index.js**가 실행된다.

routes/index.js

```
index.js x
1 var express = require('express');
2 var router = express.Router();
3
4 /* GET home page. */
5 router.get('/', function(req, res, next) {
6   res.render('index', { title: 'Express' });
7 });
8
9 module.exports = router;
10
```

index.js에는 기본 웹페이지에서 보여주는 페이지에 대한 시각화(**renderer**)가 **views** 폴더의 **index.jade**로 지정된다

7.2.12 views/index.jade

index.jade는 웹페이지 화면의 기본 레이아웃 설정 파일인 **layout.jade**를 이용한다.

그리고 오류가 발생되면 오류를 표현하는 설정은 **error.jade**를 이용한다.

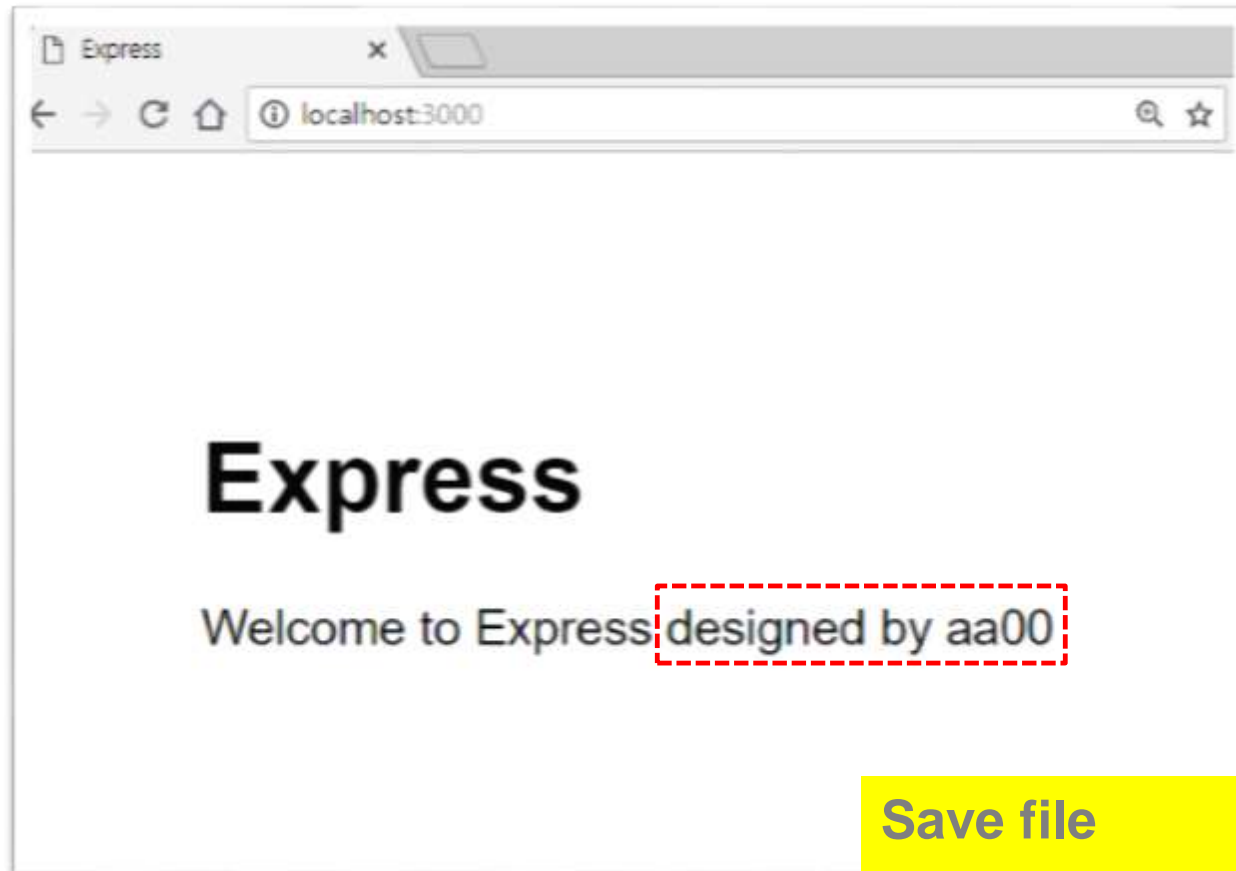
```

index.jade
1 extends layout
2
3 block content
4   h1= title
5   p Welcome to #{title} designed by aa00
6

layout.jade
1 doctype html
2 html
3   head
4     title= title
5     link(rel='stylesheet', href='/stylesheets/style.css')
6   body
7     block content
  
```



7.2.13 result ← localhost:3000

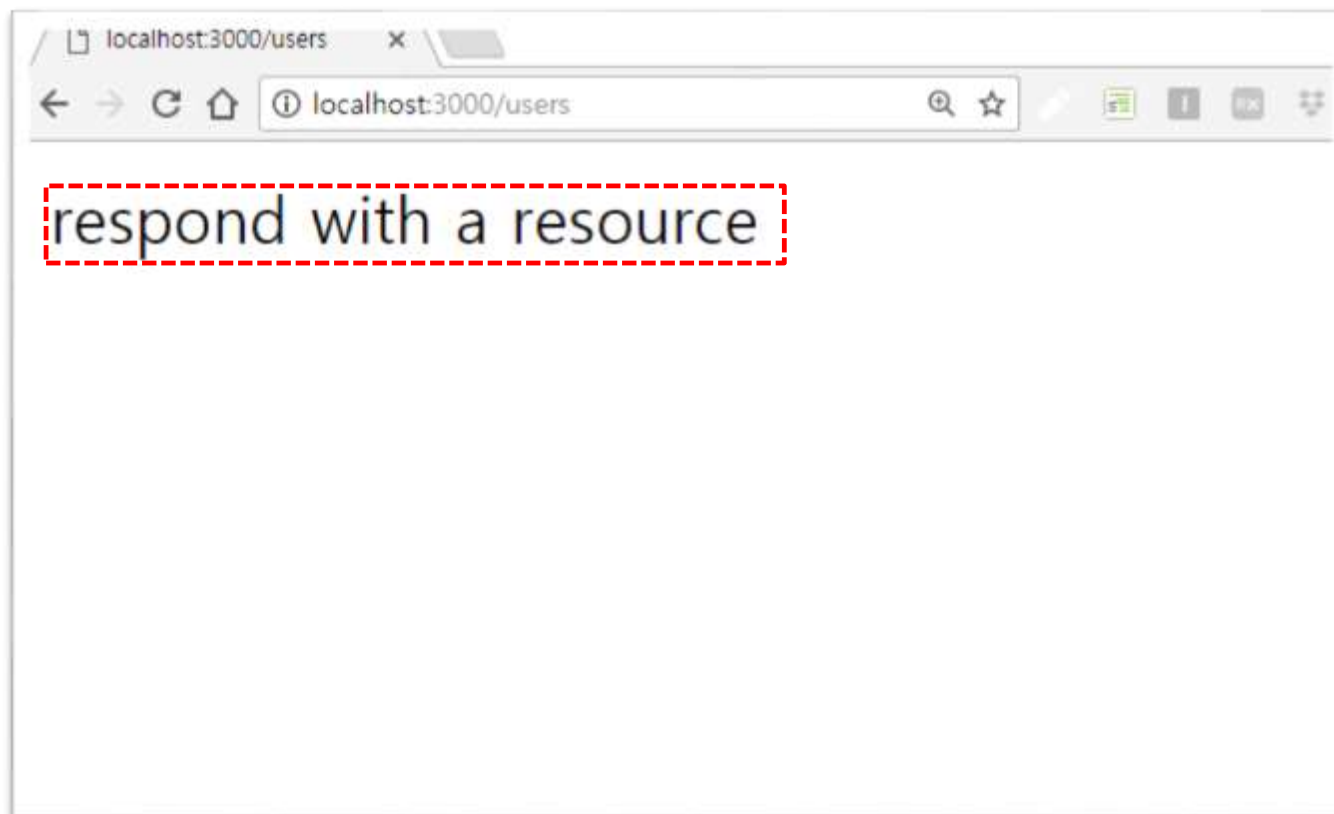


Save file

AAnn_Express_App.png



7.2.14 result ← localhost:3000/users





My Express App using generator



Express options

- e, --ejs** add ejs engine support (defaults to **jade**)
- hbs** add handlebars engine support
- H, --hogan** add hogan.js engine support
- c, --css <engine>** add stylesheet <engine> support (less|stylus|compass)
- f, --force** force on non-empty directory<engine></engine>



7.3.1 Templating using Express-generator

express aa00App -e -c less

npm

```
D:\Portable\NodeJSPortable\Data\aan\express>cd ..
```

```
D:\Portable\NodeJSPortable\Data\aan>express aa00App -e -c less
```

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

```
create : aa00App\
create : aa00App\public\
create : aa00App\public\javascripts\
create : aa00App\public\images\
create : aa00App\public\stylesheets\
create : aa00App\public\stylesheets\style.less
create : aa00App\routes\
create : aa00App\routes\index.js
create : aa00App\routes\users.js
create : aa00App\views\
create : aa00App\views\error.ejs
create : aa00App\views\index.ejs
create : aa00App\app.js
create : aa00App\package.json
create : aa00App\bin\
create : aa00App\bin\www
```

change directory:

```
> cd aa00App
```

install dependencies:

```
> npm install
```

run the app:

```
> SET DEBUG=aa00app:* & npm start
```

```
D:\Portable\NodeJSPortable\Data\aan>
```

npm

```
D:\Portable\NodeJSPortable\Data\aan>dir
```

D 드라이브의 볼륨: DATA

볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\aan 디렉터리

날짜	시간	타입	이름
2018-09-10	오후 04:12	<DIR>	.
2018-09-10	오후 04:12	<DIR>	..
2018-09-10	오후 04:12	<DIR>	aa00App
2018-09-10	오후 03:47	<DIR>	express
2018-09-10	오후 03:07	<DIR>	expressTest
2018-09-03	오후 04:33	<DIR>	server
2018-09-03	오후 05:37	<DIR>	start
		0개 파일	0 바이트
		7개 디렉터리	851,448,971,264 바이트 남음

```
D:\Portable\NodeJSPortable\Data\aan>cd aa00App
```

```
D:\Portable\NodeJSPortable\Data\aan\aa00App>dir
```

D 드라이브의 볼륨: DATA

볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\aan\aa00App 디렉터리

날짜	시간	타입	이름
2018-09-10	오후 04:12	<DIR>	.
2018-09-10	오후 04:12	<DIR>	..
2018-09-10	오후 04:12		1,180 app.js
2018-09-10	오후 04:12		bin
2018-09-10	오후 04:12		326 package.json
2018-09-10	오후 04:12	<DIR>	public
2018-09-10	오후 04:12	<DIR>	routes
2018-09-10	오후 04:12	<DIR>	views
		2개 파일	1,506 바이트
		6개 디렉터리	851,448,971,264 바이트 남음

```
D:\Portable\NodeJSPortable\Data\aan\aa00App>
```


7.3.2 Templating using Express-generator

express aa00App -e -c less

The screenshot shows a code editor with a file explorer on the left and the `package.json` file open in the main editor. The file explorer shows the project structure for `aa00`, with `aa00App` selected. The `package.json` file contains the following content:

```

1 {
2   "name": "aa00app",
3   "version": "0.0.0",
4   "private": true,
5   "scripts": {
6     "start": "node ./bin/www"
7   },
8   "dependencies": {
9     "cookie-parser": "~1.4.3",
10    "debug": "~2.6.9",
11    "ejs": "~2.5.7",
12    "express": "~4.16.0",
13    "http-errors": "~1.6.2",
14    "less-middleware": "~2.2.1",
15    "morgan": "~1.9.0"
16  }
17 }
18

```

The `dependencies` section of the `package.json` file is highlighted with a red dashed box, and the `scripts` section is also highlighted with a red dashed box.



7.3.3 Templating using Express-generator

npm install

```

D:\Portable\NodeJSPortable\Data\aan\aa00App>npm install
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN npm npm does not support Node.js v5.12.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
npm WARN deprecated hoek@2.16.3: The major version is no longer supported. Please update to 4.
x or newer
npm WARN notice [SECURITY] hoek has the following vulnerability: 1 moderate. Go here for more
details: https://nodesecurity.io/advisories?search=hoek&version=2.16.3 - Run 'npm i npm@latest
-g' to upgrade your npm version, and then 'npm audit' to get more info.
npm notice created a lockfile as package-lock.json. You should commit this file.
added 121 packages from 145 contributors in 6.86s

D:\Portable\NodeJSPortable\Data\aan\aa00App>dir
D 드라이브의 볼륨: DATA
볼륨 일련 번호: 7A01-106A

D:\Portable\NodeJSPortable\Data\aan\aa00App 디렉터리

2018-09-10 오후 04:17 <DIR> .
2018-09-10 오후 04:17 <DIR> ..
2018-09-10 오후 04:12 1,180 app.js
2018-09-10 오후 04:12 <DIR> bin
2018-09-10 오후 04:17 <DIR> node_modules
2018-09-10 오후 04:17 32,507 package-lock.json
2018-09-10 오후 04:12 326 package.json
2018-09-10 오후 04:12 <DIR> public
2018-09-10 오후 04:12 <DIR> routes
2018-09-10 오후 04:12 <DIR> views
                3개 파일                34,013 바이트
                7개 디렉터리 851,435,470,848 바이트 남음

D:\Portable\NodeJSPortable\Data\aan\aa00App>
  
```



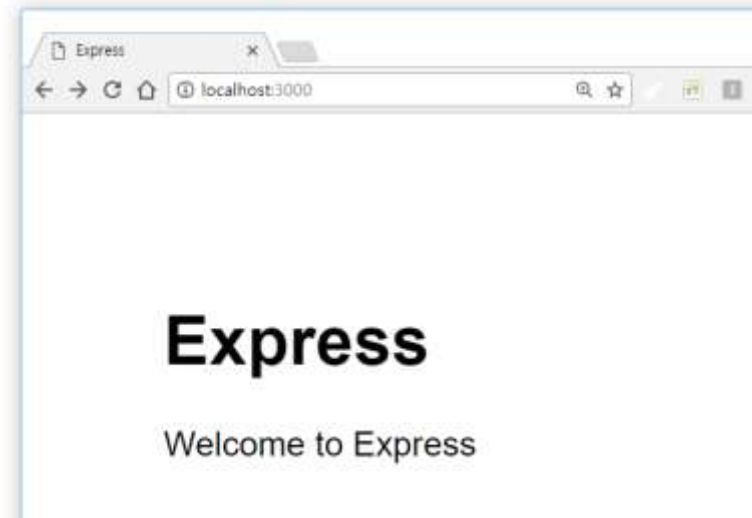
7.3.4 Run template: ejs

npm start or **node ./bin/www**

npm - node ./bin/www

```
D:\Portable\NodeJSPortable\Data\vaann\aa00App>node ./bin/www
GET / 200 12.758 ms - 207
GET /stylesheets/style.css 200 28.347 ms - 87
GET / 304 1.926 ms - -
GET /stylesheets/style.css 304 1.884 ms - -
GET / 304 0.965 ms - -
GET /stylesheets/style.css 304 0.779 ms - -
```

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title><%= title %></title>
5     <link rel='stylesheet' href='/stylesheets/style.css' />
6   </head>
7   <body>
8     <h1><%= title %></h1>
9     <p>Welcome to <%= title %></p>
10  </body>
11 </html>
12
```





[Practice]

◆ [wk03]

- Express App (**7.2 Express project**)
- Add a new route in **routes/index.js**
- Add a new view: **views/aann.jade**
- Upload folder: AAnn_Rpt03



[Practice-1] Modify routes/index.js

Add a new route `/aann` in routes/index.js .

FOLDERS

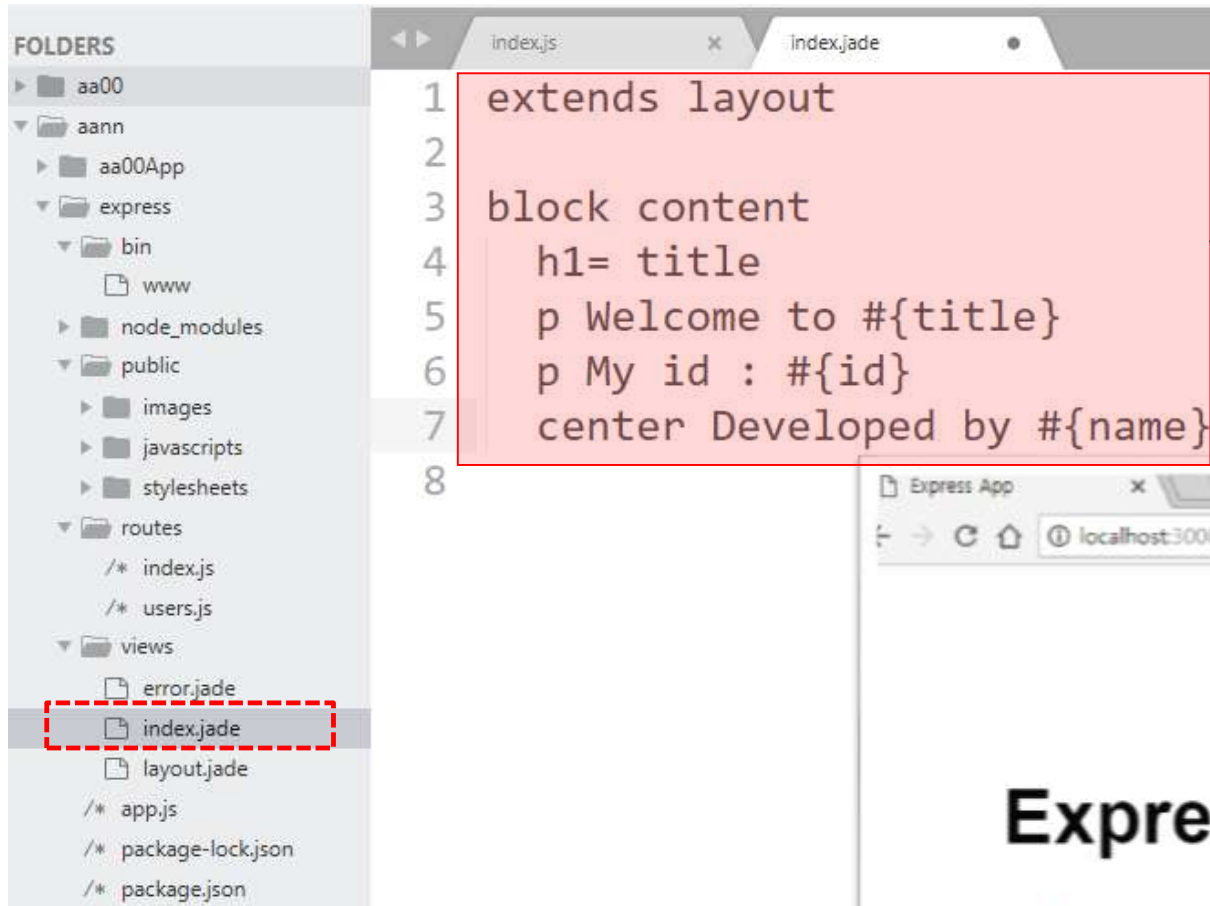
- ▶ aa00
 - ▼ aann
 - ▶ aa00App
 - ▼ express
 - ▼ bin
 - www
 - node_modules
 - ▼ public
 - images
 - javascripts
 - stylesheets
 - ▼ routes
 - /* index.js**
 - /* users.js
 - ▼ views
 - error.jade
 - index.jade
 - layout.jade
 - /* app.js
 - /* package-lock.json
 - /* package.json
 - ▶ expressTest
 - ▼ server
 - File

```
index.js x
1 var express = require('express');
2 var router = express.Router();
3
4 /* GET home page. */
5 router.get('/', function(req, res, next) {
6   res.render('index', { title: 'Express' });
7 });
8
9 /* GET my page by /aann. -> multi-routing*/
10 router.get('/aann', function(req, res, next) {
11   res.render('aann', { title: 'Express App',
12                      id: 'AA00',
13                      name: 'Redwoods' });
14   // views/aann.jade
15 });
16
17 module.exports = router;
```



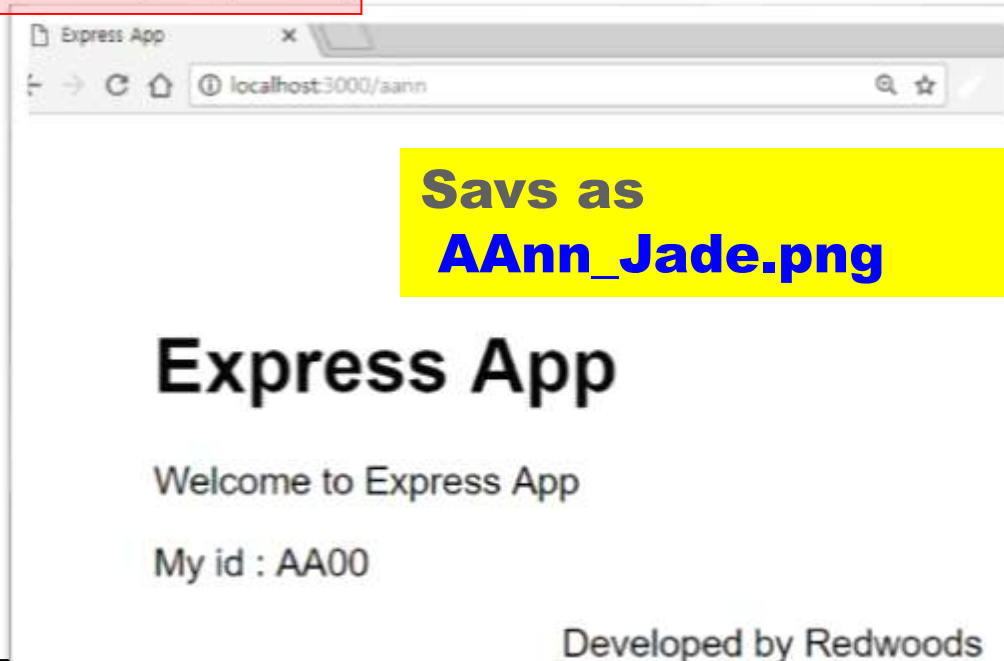
[Practice-2] Add aann.jade in views folder

Add views/aann.jade in views folder



Save as
aann.jade

Saves as
AAnn_Jade.png



◆ [Target of this week]

- Complete your works
- Save your outcomes and 7 figures

Upload folder : AAnn_Rpt03

제출할 파일들

- ① **AAnn_HTTP.png**
- ② **AAnn_TCP_Log.png**
- ③ **AAnn_Upload.png**
- ④ **AAnn_Express.png**
- ⑤ **AAnn_Express_App.png**
- ⑥ **AAnn_Jade.png**
- ⑦ **aann.jade**
- ⑧ **index.js**

[Upload to github]

◆ [wk03]

- upload all work of this week
- Use repo “aann” in github
- upload folder “aann_rpt03” in your github.

● References & good sites

- ✓ <http://www.arduino.cc> Arduino Homepage
- ✓ <http://www.nodejs.org/ko> Node.js
- ✓ <https://plot.ly/> plotly
- ✓ <https://www.mongodb.com/> MongoDB
- ✓ <http://www.w3schools.com> By w3schools
- ✓ <http://www.github.com> GitHub



주교재 및 참고도서

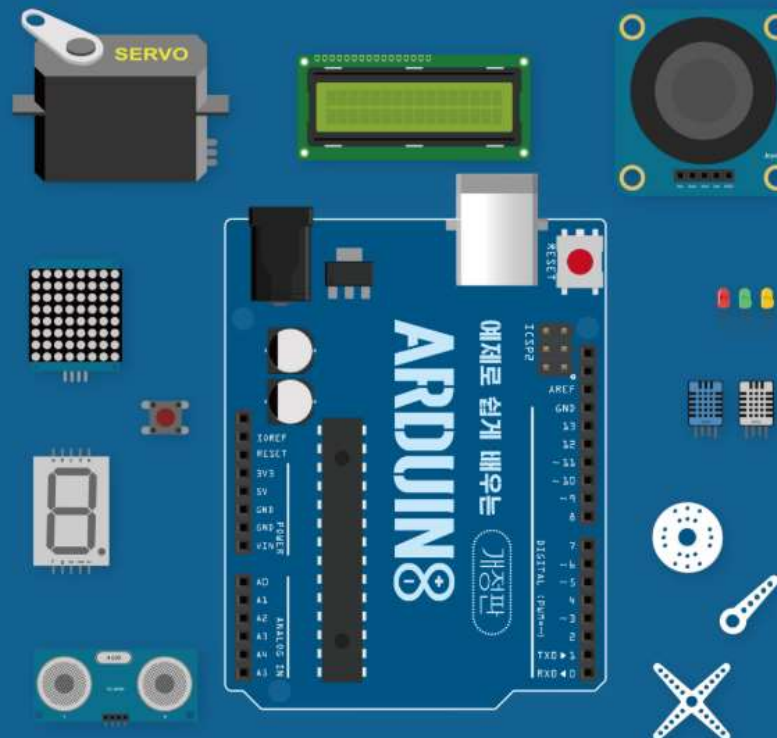
아두이노와 Node.js에 기반한 IOT 신호 시각화 | 저자 이 상 훈 | 인제대학교 출판부

아두이노와 Node.js에 기반한 IOT 신호 시각화

| 저자 이 상 훈 |



인제대학교 출판부



예제로 쉽게 배우는

아두이노

개정판

장성용 · 김진환 지음



Target of this class

Real-time Weather Station from sensors



on Time: 2018-01-22 17:58:31.012



Another target of this class

PPG with rangeslider

