

- 과제 목표 (도출해야 할 결과)

- ➔ Node.js를 이용한 웹서버 구동 및 웹페이지 라우팅 확인

- 코드 설명과 과제 해결 방법

- index.js (메인 파일): require를 통해 필요한 라이브러리들을 불러오고, express 객체를 불러와 세팅한다. view engine을 ejs로 설정하여, views 디렉토리에 있는 ejs 형식의 웹페이지를 읽을 수 있도록 하며 css를 불러오기 위한 설정도 해준다.
(body parser는 중첩된 객체 표현을 허용해주고, methodOverride는 PUT / DELETE method를 허용해주나 이번 과제에서는 실질적으로 사용하지 않았음)

```
const port = 3000;

var express = require('express');
var bodyParser = require('body-parser');
var methodOverride = require('method-override');
var app = express();

app.set('view engine', 'ejs');
app.use(express.static(__dirname + '/public'));
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({extended: true}));
app.use(methodOverride('_method'));
```

routes 폴더에 있는 라우터 설정파일 (home.js)을 require을 통해 불러온다.

그 후, 주소가 / 로 시작하면 Router가 동작하도록 use()를 통해 지정해준다.

```
// 라우터 설정파일 호출
var router = require('./routes/home');

// 주소가 / 로 시작하면 ./routes/home.js 호출하라는 의미
app.use('/', router);
```

이후에는 listen()을 통해 지정한 포트번호로 서버를 가동한다.

```
app.listen(port, () => {
  console.log(`Example app listening at http://localhost:${port}`);
});
```

- home.js : 연결을 home 디렉토리에 저장된 웹페이지로 Routing 해주는 과정이다. express의 Router 객체를 받아온 뒤, get()을 통해 입력받은 URL을 분류하고 render()를 통해 표시할 페이지를 지정한다.

```
var express = require('express');
var router = express.Router();

router.get('/', function(req, res) {
  res.render('./home/welcome');
});

router.get('/about', function(req, res) {
  res.render('./home/about');
});

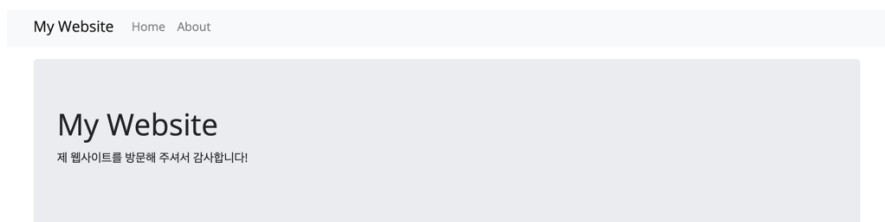
module.exports = router;
```

- 실행 결과

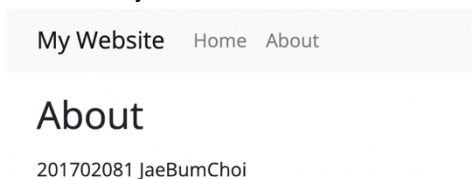
➤ 서버 가동

```
▶ node index.js
Example app listening at http://localhost:3000
```

➤ welcome.ejs 열림



➤ about.ejs 열림



● 과제 느낀 점

과제 수행 방법에 대하여 ppt 설명이 조금 더 상세했으면 좋겠습니다.