**Chapter 05. 웹 서버 만들기**

* 미들웨어 사용하기
* static 미들웨어

특정 폴더의 파일들을 특정 패스로 일괄 접근할 수 있도록 만들어 준다.   
e.g.) /public/index.html이 있는데, public이란 폴더를 static으로 설정해 줄 경우 <http://localhost:3000/index.html> 로 접근할 수 있게 된다.

(다른 부분에서 좀 더 살펴볼 것)

* body-parser 미들웨어

GET방식 요청 시에는 주소 string에 query string(요청 파라미터)가 들어갔지만(…/?name=mike 의 경우, req.query.name이 mike가 된다), POST 방식일 경우 Body 영역에 query string이 들어가므로 파싱 방법이 다르다.

**클라이언트가 POST방식 요청 시, body 영역의 query string을 파싱하여 req 객체의 body 속성에 넣어준다.**

[/public/login.html]

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>로그인 테스트</title>

</head>

<body>

<h1>로그인</h1>

<br>

<form method="post">

<table>

<tr>

<td><label>아이디</label></td>

<td><input type = "text" name = "id"></td>

</tr>

<tr>

<td><label>비밀번호</label></td>

<td><input type = "password" name = "password"></td>

</tr>

</table>

<input type = "submit" value = "전송" name = "">

</form>

</body>

* query string 전달 방식을 POST로 설정하고, 각 id / password name값으로 전달됨

[app7.js]

var express = require('express');

var http = require('http');

var path = require('path');

// Express 미들웨어

var bodyParser = require('body-parser');

var static = require('serve-static');

// Express 객체 생성

var app = express();

// Express 서버 객체 기본 설정

app.set('port', process.env.PORT || 3000);

// body-parser를 이용해 application/x-www-form-urlencoded 파싱

app.use(bodyParser.urlencoded({ extended: false}));

// body-parser를 이용해 application/json 파싱

app.use(bodyParser.json());

app.use(static(path.join(\_\_dirname, 'public')));

// 미들웨어에서 파라미터 확인

app.use(function(req, res, next){

console.log('첫 번째 미들웨어에서 요청을 처리함');

var paramId = req.body.id || req.query.id;

var paramPassword = req.body.password || req.query.password;

res.writeHead('200', {'Content-Type': 'text/html;charset=utf8'});

res.write('<h1>Express 서버에서 응답한 결과입니다.</h1>');

res.write('<div><p>Param id : ' + paramId + '</p></div>');

res.write('<div><p>Param password : ' + paramPassword + '</p></div>');

res.end();

});

http.createServer(app).listen(3000, function(){

console.log('Express 서버가 3000번 포트에서 시작됨.');

});

* body-parse를 사용함으로써 login.html에서 POST로 전달된 query string을 req.body.id와 req.body.password로 저장이 가능함