**Chapter 05. 웹 서버 만들기**

* 쿠키와 세션 관리

사용자의 로그인 상태를 확인하고 싶을 때 쿠키나 세션을 사용함. 쿠키는 클라이언트 웹 브라우저에 저장되는 정보 / 세션은 웹 서버에 저장되는 정보를 의미

* 쿠키처리

쿠키는 웹 브라우저에 저장되는 정보로, 일정기간 저장함(cookie-parser사용)

[app11.js]

// 라우팅 함수 등록

router.route('/process/showCookie').get(function(req, res) {

console.log('/process/showCookie 호출됨');

// 서버에서 클라이언트로 쿠키 객체를 전달함

res.send(req.cookies);

});

router.route('/process/setUserCookie').get(function(req, res){

console.log('/process/setUserCookie 호출됨');

// 쿠키 설정을 하면 클라이언트 웹 브라우저에 쿠키가 설정

res.cookie('user', {

id: 'izone',

name: '아이즈원',

authorized: true

});

// redirect로 응답

res.redirect('/process/showCookie');

});

// 라우터 객체를 app 객체에 미들웨어로 등록

app.use('/', router);

* localhost:3000/setUserCookie를 호출하면 res.cookie( ) 메소드를 통해 웹 브라우저에 쿠키 값이 저장된다. 그리고 /process/showCookie로 redirect 되면 req.cookies 객체에 저장된 브라우저 쿠키 값을 res한다.(브라우저에서 확인 가능)
* 세션 처리하기

세션은 상태 정보를 서버에 저장하는 방식. 대표적인 용례는 로그인 여부 체크를 통한 다른 페이지 redirect 기능이다.

[app12.js] – 코드 양 많으므로 별도 확인

1. /process/product 요청 패스로 상품 페이지 처리 요청(GET) 🡪 2) /process/product 라우팅 처리하면서 ‘user’세션 유무 확인 🡪 3-1) 세션 있으면 /public/product.html 응답으로 전송 3-2) 세션 없으면 public/login.html 전송