

## 12. 요구사항-2(2)

### 1. 로그인

- 예제 코드를 사용하자
- `/auth/auth.controller.ts` 를 사용하겠다.

```
import {
  Body,
  Controller,
  Get,
  HttpStatusCode,
  HttpStatus,
  Post,
  Request,
} from '@nestjs/common';
import { AuthService } from '../auth.service';
import { Public } from '../decorators/public.decorator';
import { LoginDto } from 'src/dto/login.dto';

@Controller('auth')
export class AuthController {
  constructor(private authService: AuthService) {}

  @Public()
  @HttpCode(HttpStatus.OK)
  @Post('login')
  signIn(@Body() signInDto: LoginDto) {
    return this.authService.signIn(signInDto.email, signInDto.password);
  }

  @Get('profile')
  getProfile(@Request() req) {
    return req.user;
  }
}
```

- `auth.service.ts`

```
async signIn(email: string, pass: string) {
  const user = await this.usersService.findByEmail(email);
  if (!user || !(await this.comparePasswords(pass, user.password))) {
    throw new UnauthorizedException();
  }
  const payload = { username: user.email, sub: user.id };
  return {
    access_token: await this.jwtService.signAsync(payload),
  };
}
```

## 2. 로그인 확인

- 요청주소 : `http://localhost:3000/auth/login`
- Body :

```
{
  "email" : "test2@gmail.com",
  "password" : "1234"
}
```

- `access_token`:

```
{
  "access_token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6InRlc3QyQGdtYWlsLmNvbSIsInN1YiI6I6NSwiaWF0IjoxNzA2MTYxNzI4LCJleHAiOjE3MDYxNjIzMjh9.132b7CpzWZ39XWl7A33f41pegGdMHBvQHxr6dnin6x8"
}
```