# 04. 인증 시스템 구현

`auth.service.ts`

```
// auth.service.ts
import { Injectable } from '@nestjs/common';
import { JwtService } from '@nestjs/jwt';

@Injectable()
export class AuthService {
  constructor(private jwtService: JwtService) {}

  async validateUser(username: string, pass: string): Promise<any> {
    // 사용자 검증 로직
  }

  async login(user: any) {
    const payload = { username: user.username, sub: user.userId };
    return {
      access_token: this.jwtService.sign(payload),
    };
  }
}
```

`jwt.strategy.ts`

```
// jwt.strategy.ts
import { ExtractJwt, Strategy } from 'passport-jwt';
import { PassportStrategy } from '@nestjs/passport';
import { Injectable } from '@nestjs/common';

@Injectable()
export class JwtStrategy extends PassportStrategy(Strategy) {
  constructor() {
    super({
      jwtFromRequest: ExtractJwt.fromAuthHeaderAsBearerToken(),
      ignoreExpiration: false,
      secretOrKey: 'secretKey',
    });
  }

  async validate(payload: any) {
    return { userId: payload.sub, username: payload.username };
  }
}
```

# 1. 트러블 슈팅

- passport-jwt 못불러옴 문제

```
npm install @nestjs/passport passport passport-jwt
```

# 2. 모듈 등록

`app.module.ts`

```
import { Module } from '@nestjs/common';
import { AppController } from './app.controller';
import { AppService } from './app.service';
import { TypeOrmModule } from '@nestjs/typeorm';
import { AuthService } from './auth.service'; // AuthService import
import { JwtModule } from '@nestjs/jwt'; // JwtModule import
import { PassportModule } from '@nestjs/passport'; // PassportModule import
import { JwtStrategy } from './jwt.strategy'; // JwtStrategy import

@Module({
  imports: [
    TypeOrmModule.forRoot({
      type: 'mysql',
      host: 'localhost',
      port: 3306,
      username: 'root',
      password: '1234',
      database: 'opstest',
      entities: [],
      synchronize: true,
    }),
    PassportModule, // PassportModule 등록
    JwtModule.register({ // JwtModule 설정
      secret: '1234', // 비밀 키 설정
      signOptions: { expiresIn: '60s' }, // 토큰 만료 시간 설정
    }),
  ],
  controllers: [AppController],
  providers: [AppService, AuthService, JwtStrategy], // AuthService와 JwtStrategy 등록
})
export class AppModule {}
```