**Hands-on: Creating an Authentication Service that Returns JWT in Spring Boot**

**Objective**

To build a Spring Boot-based REST service that generates a **JWT (JSON Web Token)** when a valid user sends their credentials using **Basic Authentication**

**Tools & Technologies**

Java 17+

Spring Boot

Spring Security

JJWT (Java JWT library)

Maven

**Functional Requirement**

Expose a /authenticate endpoint that:

* Accepts HTTP Basic Authentication
* Validates the username/password
* If valid, generates and returns a JWT token

**Step-by-Step Implementation**

1. Add Required Dependencies in pom.xml

Spring Boot Starter Web

Spring Boot Starter Security

JWT

JAXB (for Java 11+ compatibility with JJWT)

2. Define Server Port and User Credentials in application.properties

server.port=8083

spring.security.user.name=user

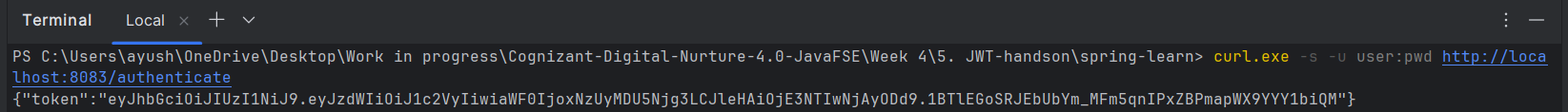
spring.security.user.password=pwd

3. Create JwtUtil.java – Token Generator Utility

4. Create AuthenticationController.java – REST Endpoint

5. Configure Security in SecurityConfig.java

**Output**

****

**Final Outcome**

You successfully built a Spring Boot-based JWT authentication service that:

* Accepts Basic Auth credentials
* Validates them
* Returns a signed JWT token