JWT Authentication Service - Token Generation

# Objective

To create an authentication service in Spring Boot that accepts user credentials via Basic Auth, validates them, and returns a JWT token.

# Sample Request

curl -s -u user:pwd http://localhost:8090/authenticate

# Sample Response

{"token":"eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJ1c2VyIiwiaWF0IjoxNTcwMzc5NDc0LCJleHAiOjE1NzAzODA2NzR9.t3LRvlCV-hwKfoqZYlaVQqEUiBloWcWn0ft3tgv0dL0"}

# Implementation Steps

## Step 1: Create Authentication Controller

- Create a controller class with endpoint `/authenticate`.  
- Annotate it with `@RestController`.  
- Configure the endpoint in SecurityConfig to allow unauthenticated access.

## Step 2: Read and Decode Authorization Header

- Read the `Authorization` header from the request.  
- Decode the Base64 encoded username and password from the header.  
- Validate the credentials using a custom UserDetailsService or hardcoded values for simplicity.

## Step 3: Generate JWT Token

- If the credentials are valid, generate a JWT using a library like `io.jsonwebtoken` (jjwt).  
- Include user information (e.g., subject) and expiry time in the token.  
- Return the token as a JSON response.