**Create authentication service that returns JWT** 

package com.cognizant.jwt.controller;

import org.springframework.web.bind.annotation.\*;

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

import io.jsonwebtoken.security.Keys;

import io.jsonwebtoken.JwtBuilder;

import java.security.Key;

import java.util.Base64;

import java.util.Date;

import java.util.HashMap;

import java.util.Map;

*@RestController*

public class AuthenticationController {

*@GetMapping*("/authenticate")

public Map<String, String> authenticate(*@RequestHeader*("Authorization") String authHeader) {

System.***out***.println("Authorization Header: " + authHeader);

String user = getUser(authHeader);

String token = generateJwt(user);

Map<String, String> response = new HashMap<>();

response.put("token", token);

return response;

}

private String getUser(String authHeader) {

String base64Credentials = authHeader.substring("Basic ".length());

byte[] decodedBytes = Base64.*getDecoder*().decode(base64Credentials);

String credentials = new String(decodedBytes);

String[] parts = credentials.split(":", 2);

return parts[0];

}

private String generateJwt(String user) {

Key key = Keys.*secretKeyFor*(*SignatureAlgorithm*.***HS256***);

return Jwts.*builder*()

.setSubject(user)

.setIssuedAt(new Date())

.setExpiration(new Date(System.*currentTimeMillis*() + 20 \* 60 \* 1000))

.signWith(key)

.compact();

}

}

