Securing an API with Basic Authentication

Assignment: Implement AppController for Basic Authentication

Objective

Your task is to implement a Spring Boot REST API controller named AppController that authenticates API requests using Basic Authentication with RestTemplate. Your implementation should pass all functional and structural tests.

Provided Files

1. ApiAuthenticatorApplication.java: Spring Boot main class.
2. AppConfig.java: Configuration class providing a RestTemplate bean and custom error handler.
3. Blank AppController.java: You need to implement this file.
4. Pre-written test cases (ApiControllerTest) to validate your implementation.

Task Details

1. Class must be annotated with RestController annotation.
2. Constructor-based injection for RestTemplate must be used.
3. Implement a method named authenticateApiRequest of type public.
4. Method must be annotated with GET type with value as /authenticateApi.
5. Method must accept three parameters:
    - request parameter named apiUrl using annotation.
    - request parameter named username using annotation.
    - request parameter named password using annotation.
6. Encode username and password into Base64 format for Basic Authentication.
7. Set the Authorization header using Basic Authentication with encoded credentials.
8. Create an HttpEntity with the headers set.
9. Send a GET request using RestTemplate.exchange(apiUrl, HttpMethod.GET, entity, String.class) with the authentication header as:
    headers.set("Authorization", "Basic " + encodedAuth);
10. Return the response body as String.

Note: You can use the endpoint **https://reqres.in/api/users/2** in Postman and pass it into the **localhost:8081/authenticateApi** URL parameter for testing, along with username as “user” and password as “password”.

Execution Steps to Follow

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers need to go to the Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal.
3. cd into your backend project folder.
4. To build your project use command:

   ```
   mvn clean package -Dmaven.test.skip
   ```

5. To launch your application, move into the target folder (cd target). Run the following command to run the application:

   ```
   java -jar <your application jar file name>
   ```

6. To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN. Please use 127.0.0.1 instead of localhost to test rest endpoints.
7. Default credentials for MySQL:
   a. Username: root
   b. Password: pass@word1
8. To login to mysql instance: Open new terminal and use following command:
   a. sudo systemctl enable mysql
   b. sudo systemctl start mysql

   NOTE: After typing any of the above commands you might encounter any warnings.

   >> Please note that this warning is expected and can be disregarded. Proceed to the next step.

   c. mysql -u root -p

   The last command will ask for password which is 'pass@word1'

9. Mandatory: Before final submission run the following command:

   ```
   mvn test
   ```

10. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.