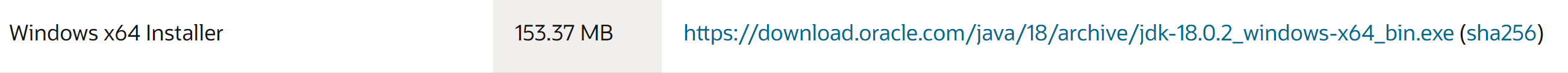
1. BackEnd : Using latest jdk 18 <https://download.oracle.com/java/18/archive/jdk-18.0.2_windows-x64_bin.exe>

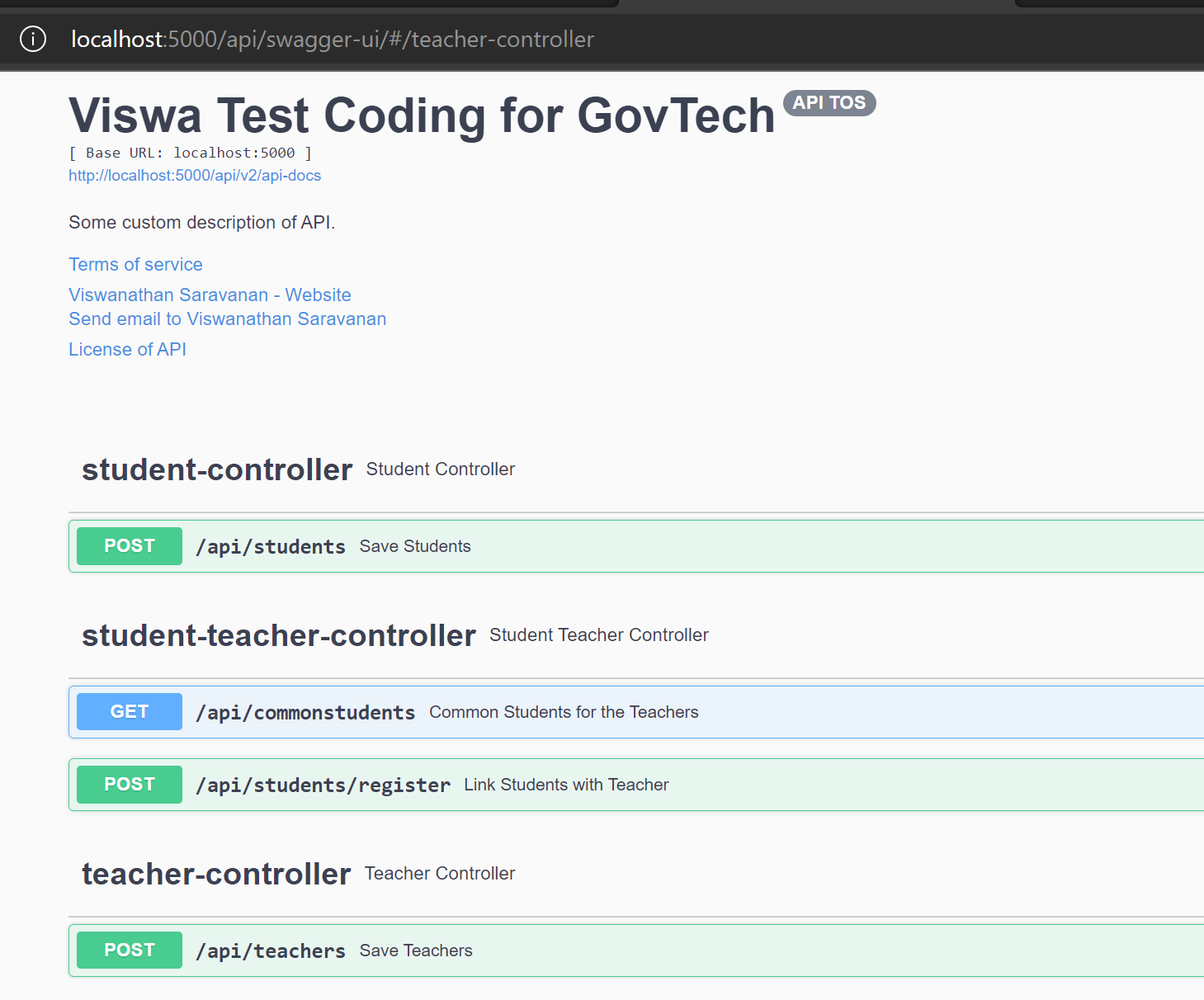


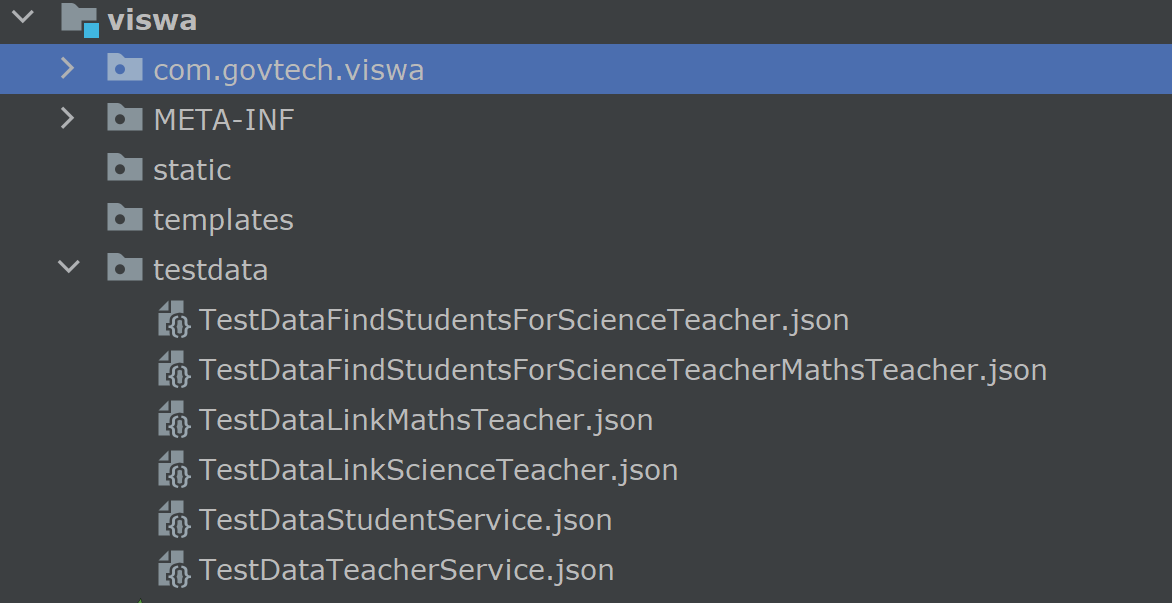
Database mysql 8

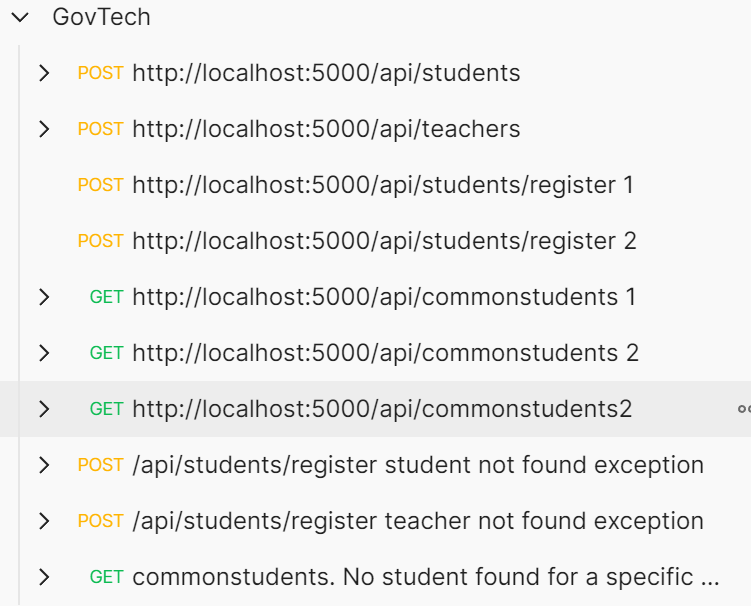
IDE : IntelliJ

Postman

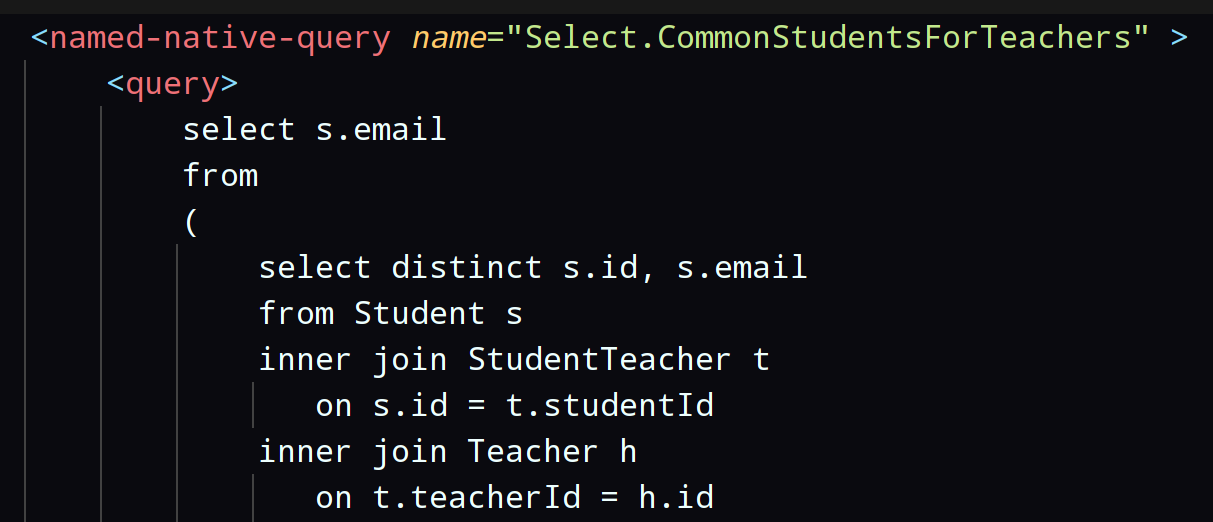
1. Refer swagger url [Swagger UI](http://localhost:5000/api/swagger-ui/) for API spec



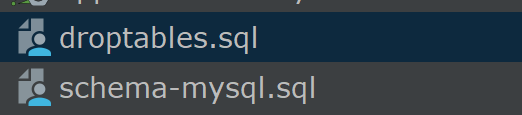
1. Test data  
   
2. Postman collection and attached examples



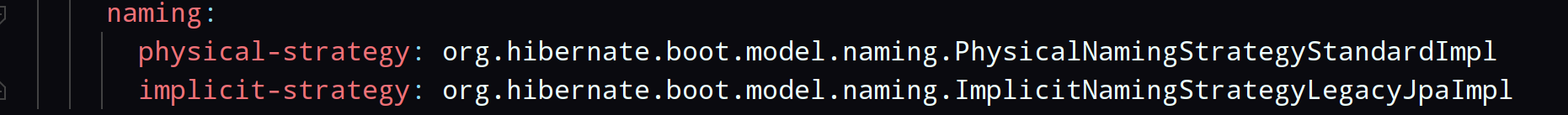
1. Used orm.xml for native queries for freestyle coding with proper indents and avoided lot of string concats. More readability



1. Used schema creations during loading ( since it is a test application, but not good for production)

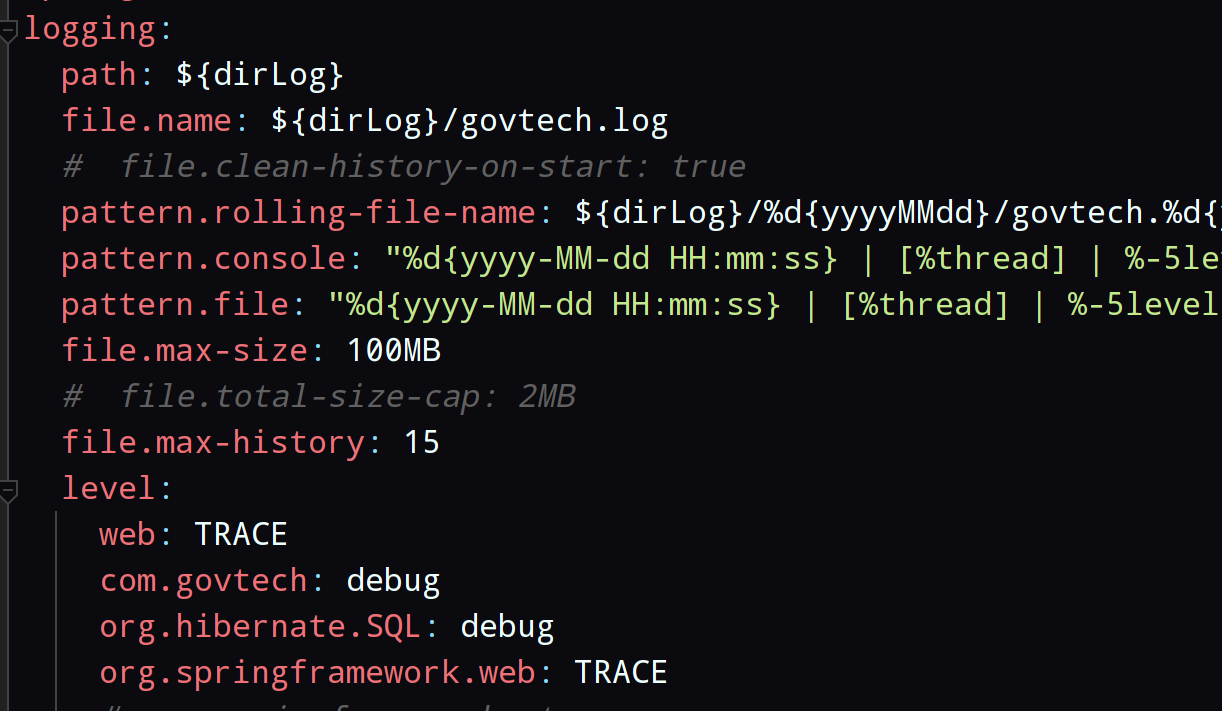


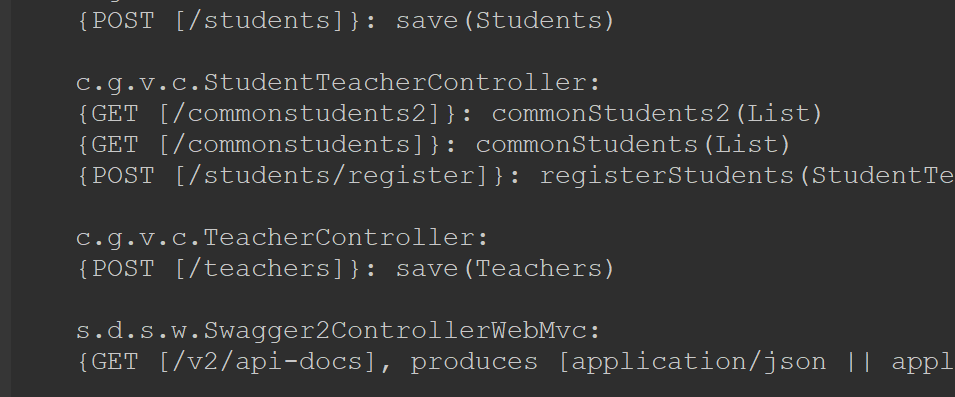
1. Used camel casing for table names to sync with java pattern



1. Detailed logging of the application to monitor and debug the process flow in case of troubleshooting (temporarily turned off running timestamp for greater readability, but not for production. Trace will be turned off in production )

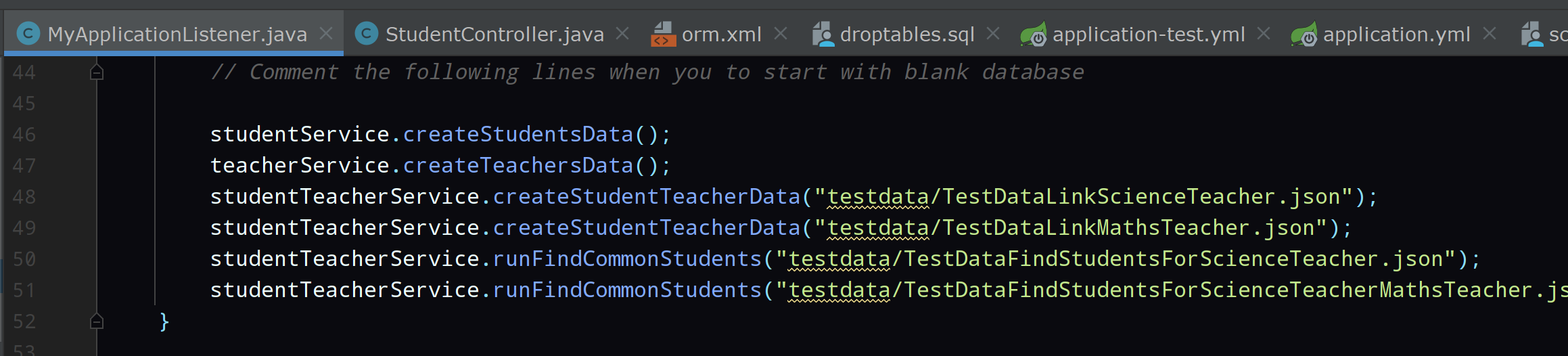
Please take a look a look on log once



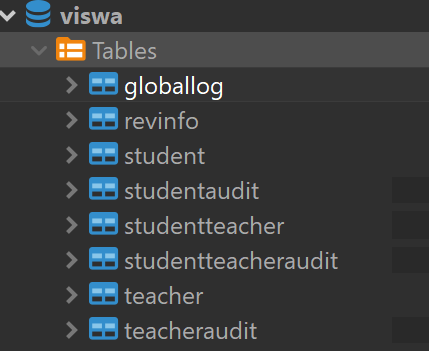


1. Prepared with test data for easy testing that covers most of the use cases and ready to try API commonstudents

If you require to start with blank database, please comment these lines and restart



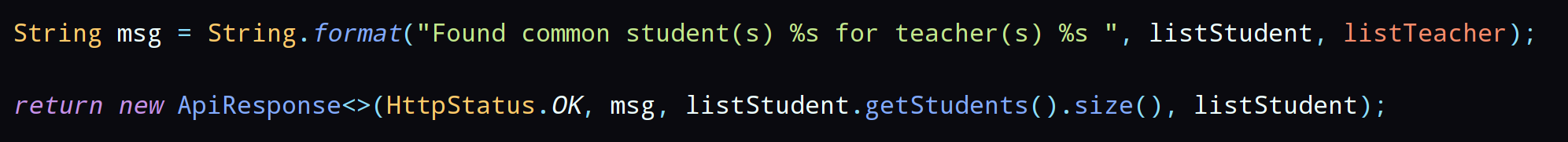
1. Created with auditing tables. All the data changes will be captured

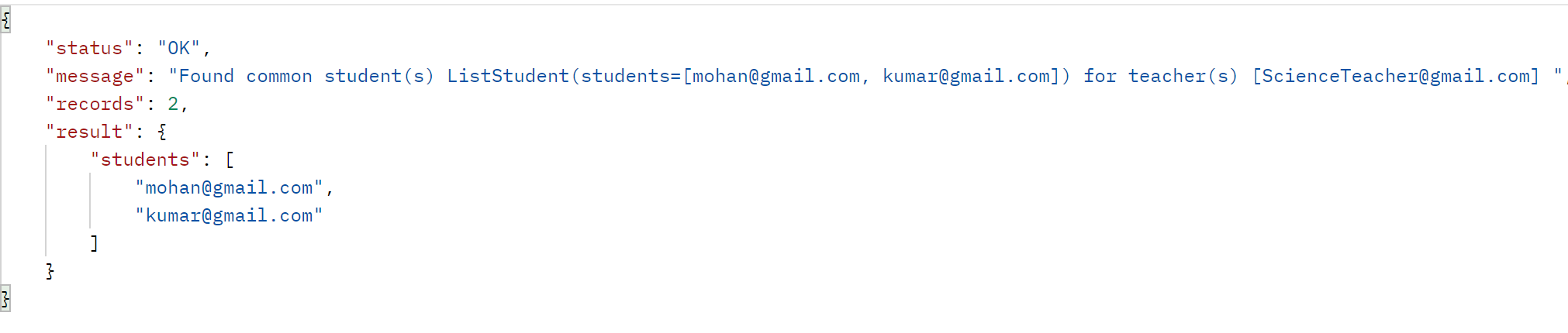


1. Used ApiResponse.

This helps message will be notified in the front end.

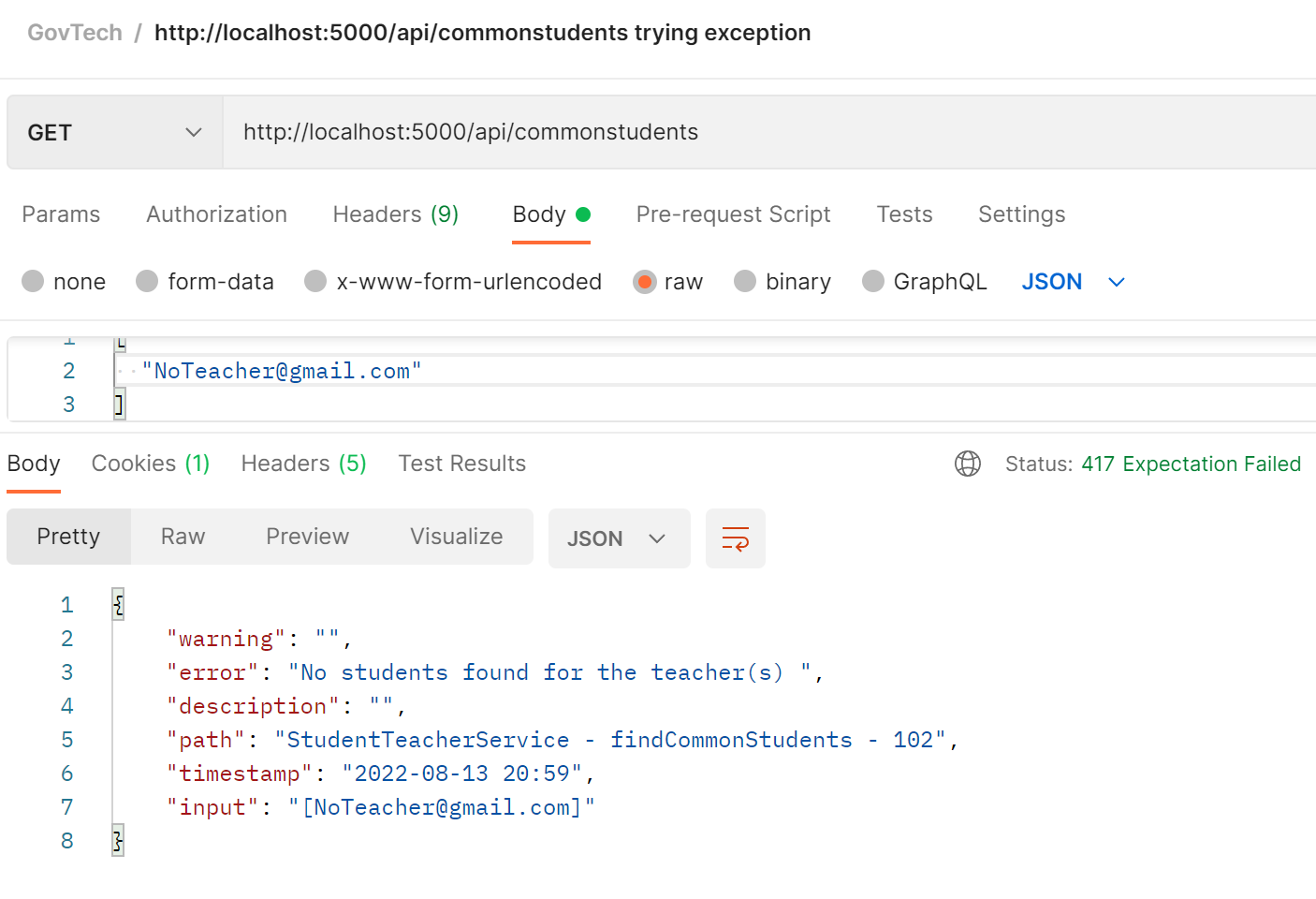
Also returning the response data in result.

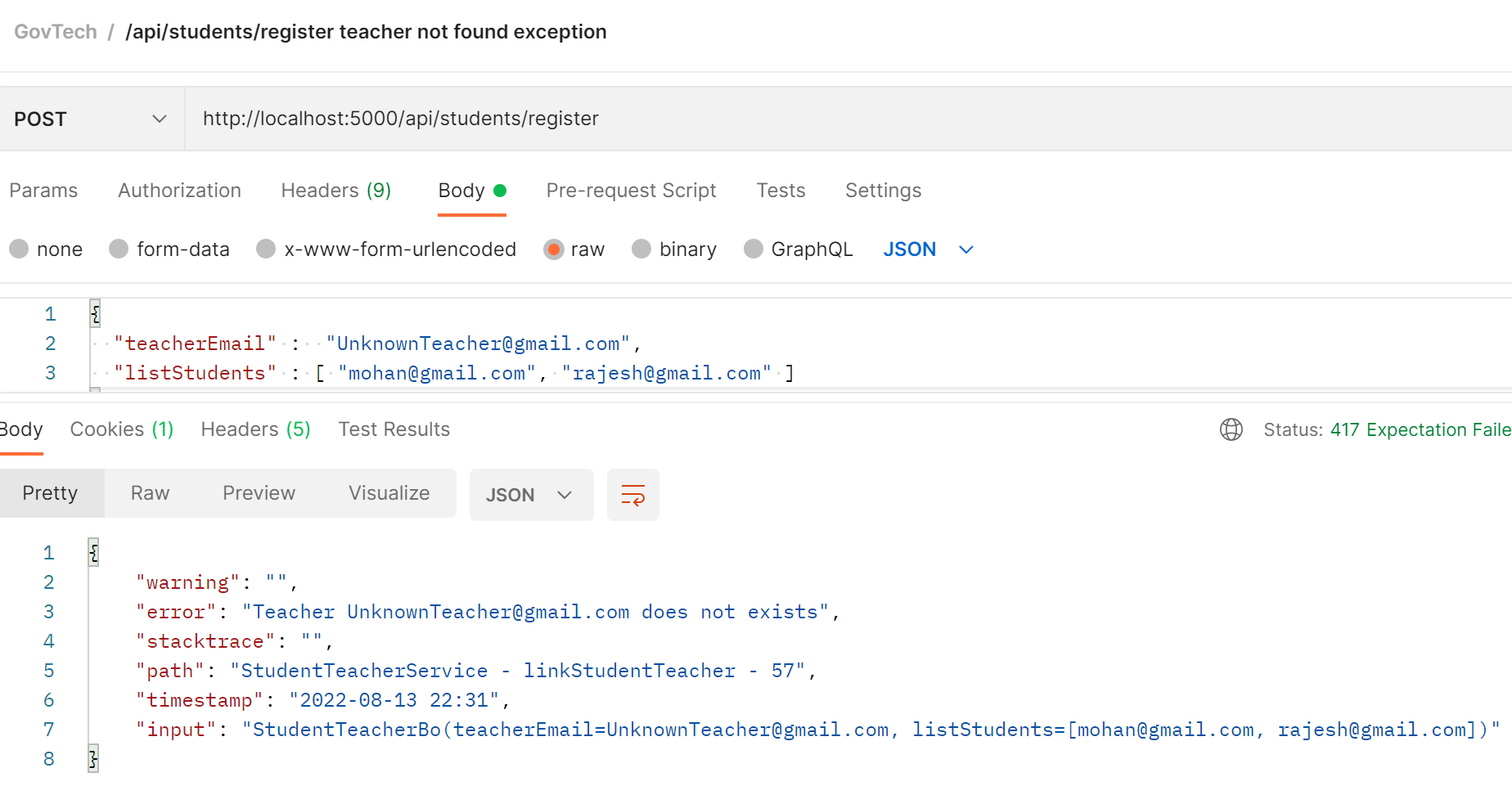




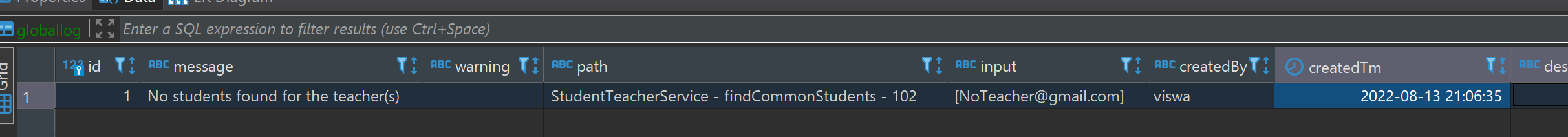
1. Properly handling global exceptions (both user defined and runtime)

And providing clear details about error or warning, path and line number, input data provided, etc

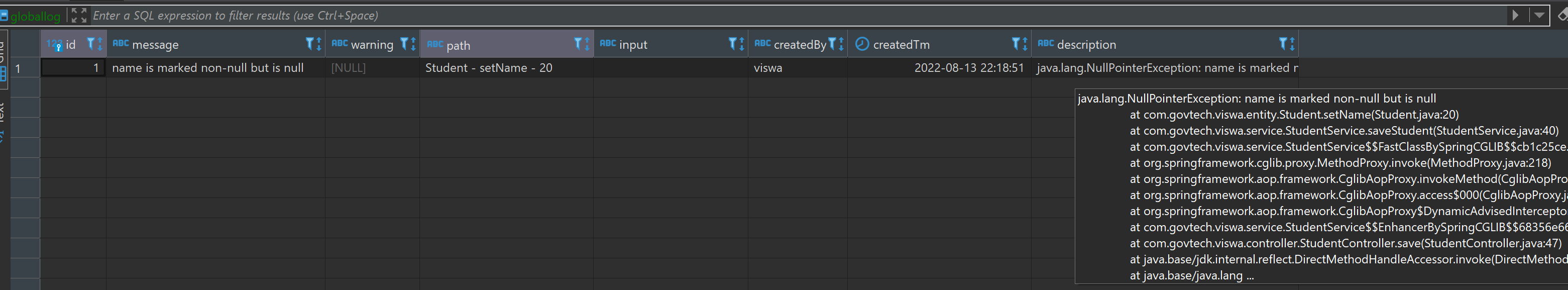




1. Capturing user defined exception in GlobalLog table. No need to open and debug the log file



1. Capturing runtime exception in GlobalLog table



1. Capturing login user (not fully implemented) and updating in audit fields

