Debugging

What are the objectives?

- Evaluate learners' ability to identify, analyze, and resolve issues in Java code.
- By the end of the exam, learners will have enhanced their ability to effectively debug Java applications, ensuring higher quality and more maintainable code.

MySQL Workbench Set up

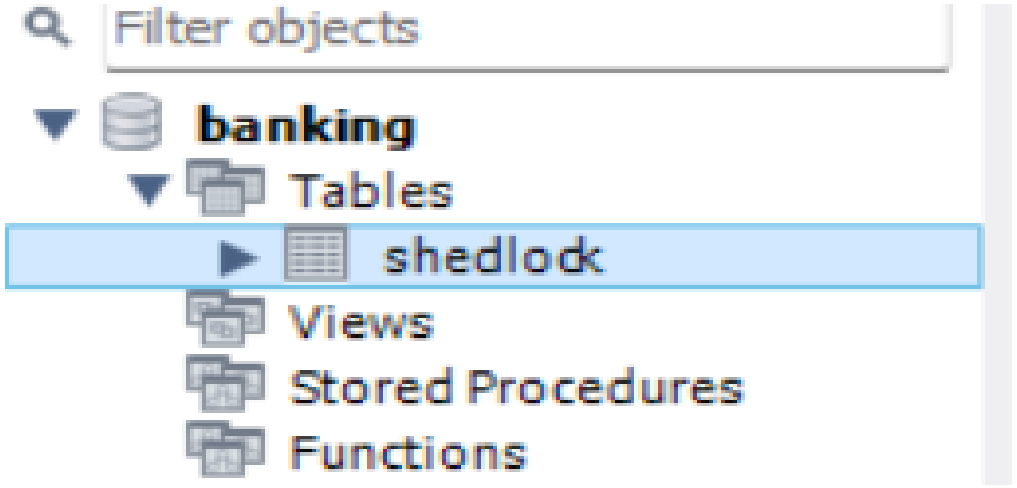
- This will create the shedlock table in the banking database
- This table is necessary solely to avoid errors related to its existence or usage in the system.

```
USE banking;

CREATE TABLE shedlock (
   name VARCHAR(64) NOT NULL,
   lock_until TIMESTAMP NOT NULL,
   locked_at TIMESTAMP NOT NULL,
   locked_by VARCHAR(255) NOT NULL,
   PRIMARY KEY (name)
);
```

MySQL Workbench Set up

- This will create the shedlock table in the banking database
- This table is necessary solely to avoid errors related to its existence or usage in the system.

```
USE banking;

CREATE TABLE shedlock (
   name VARCHAR(64) NOT NULL,
   lock_until TIMESTAMP NOT NULL,
   locked_at TIMESTAMP NOT NULL,
   locked_by VARCHAR(255) NOT NULL,
   PRIMARY KEY (name)
);
```



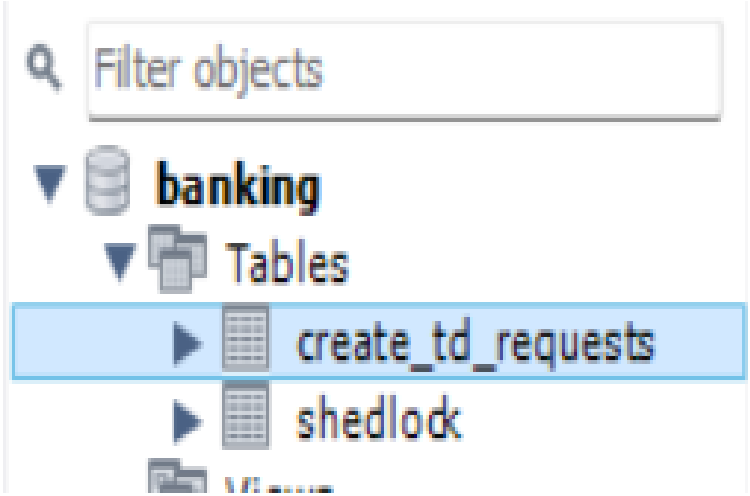
1st Task

Start the application and check the console after it runs. Check the error message as shown below.

2nd Task

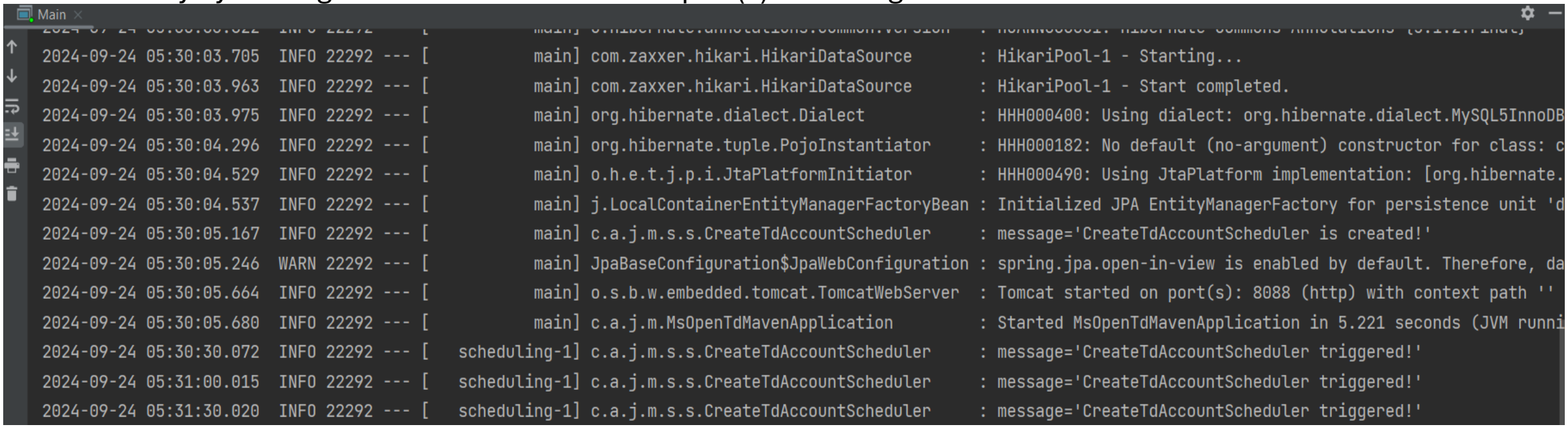
If the issue in your first task has been resolved, the expected error will appear below.
Additionally, verify that the `create_td_requests` table has been created. You can check this using Workbench.

```
Main
***************************
APPLICATION FAILED TO START
***************************

Description:

Parameter 0 of constructor in com.accenture.java.msopentdmaven.controller.OpenTdController required a bean of type 'com.accenture.java.msopentdmaven.service.Crea


Action:

Consider defining a bean of type 'com.accenture.java.msopentdmaven.service.CreateTdAccountService' in your configuration.
```



3rd Task

If the issue in your second task has been resolved, check your console to confirm that Spring Boot is running successfully by looking for the 'Tomcat started on port(s): 8088' log.

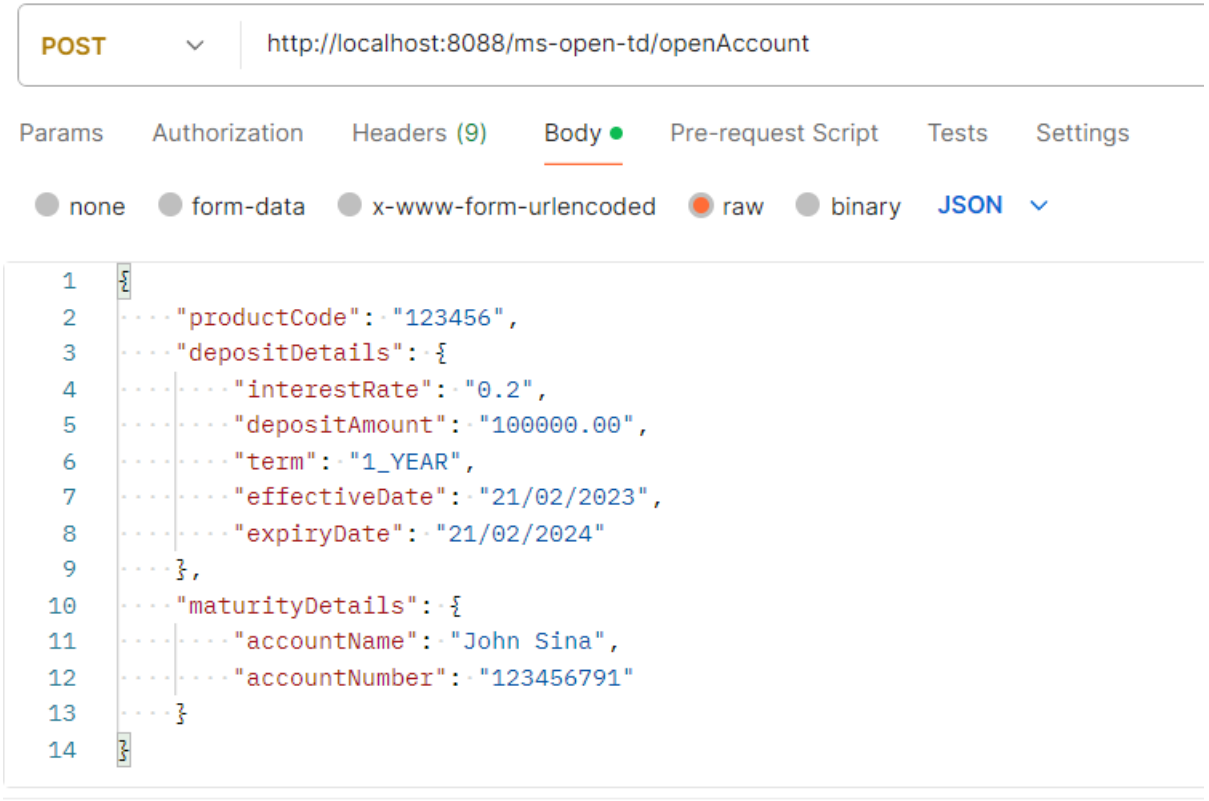


3rd Task - continuation

Use Postman to execute this endpoint: http://localhost:8088/ms-open-td/openAccount

Use the JSON payload provided below to fill in the body of Postman.

```
{
    "productCode": "123456",
    "depositDetails": {
        "interestRate": "0.2",
        "depositAmount": "100000.00",
        "term": "1_YEAR",
        "effectiveDate": "21/02/2023",
        "expiryDate": "21/02/2024"
    },
    "maturityDetails": {
        "accountName": "John Sina",
        "accountNumber": "123456791"
    }
}
```

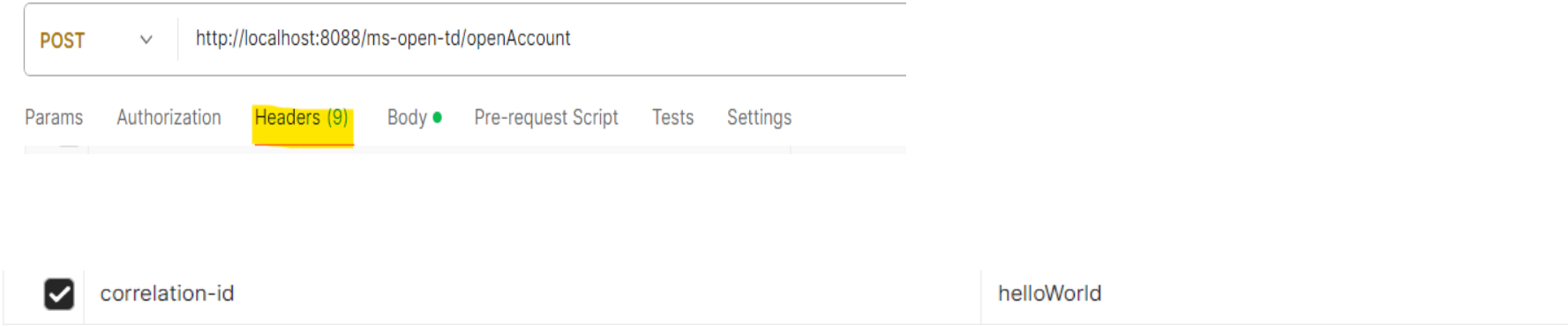


3rd Task - continuation

Use Postman to execute this endpoint: http://localhost:8088/ms-open-td/openAccount

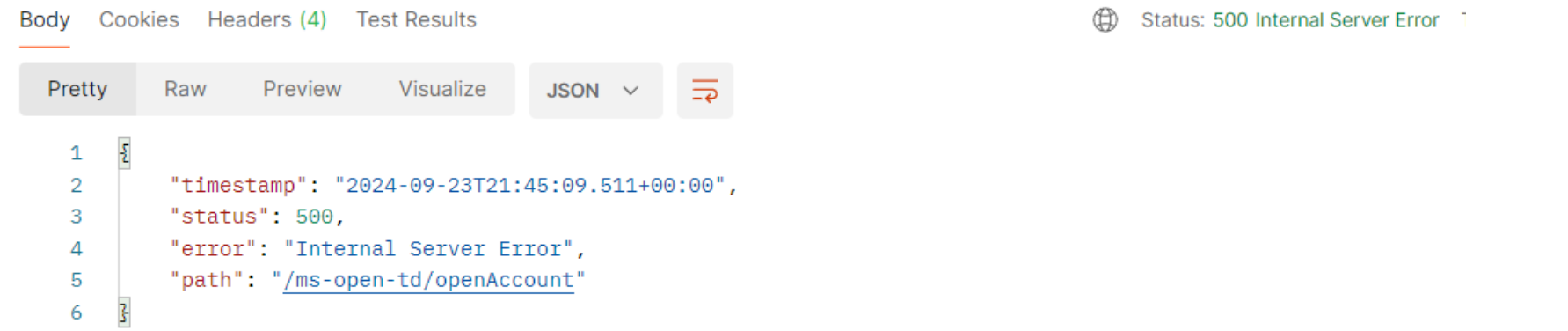
Populate the Header with the key-value pair below

Key: correlation-id
Value: helloWorld



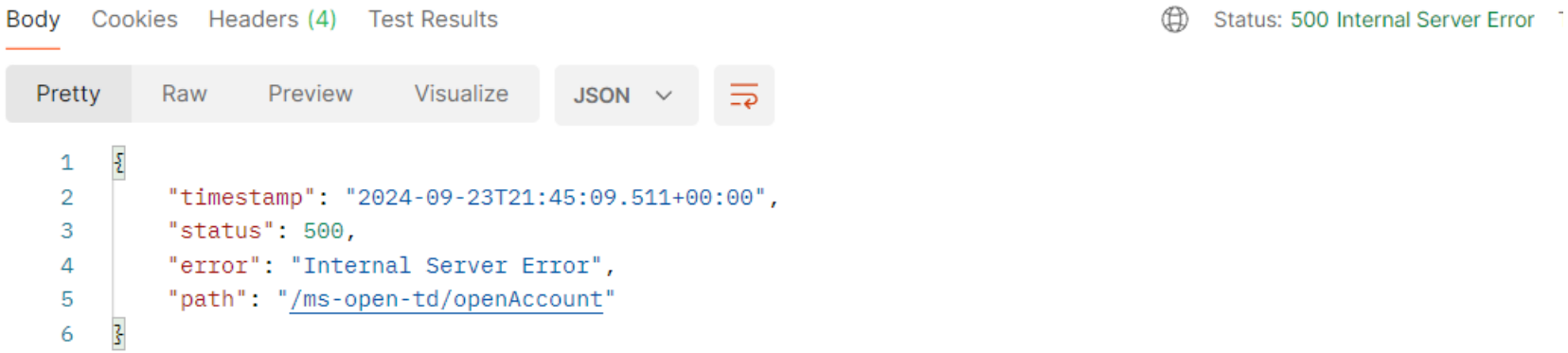
3rd Task - continuation

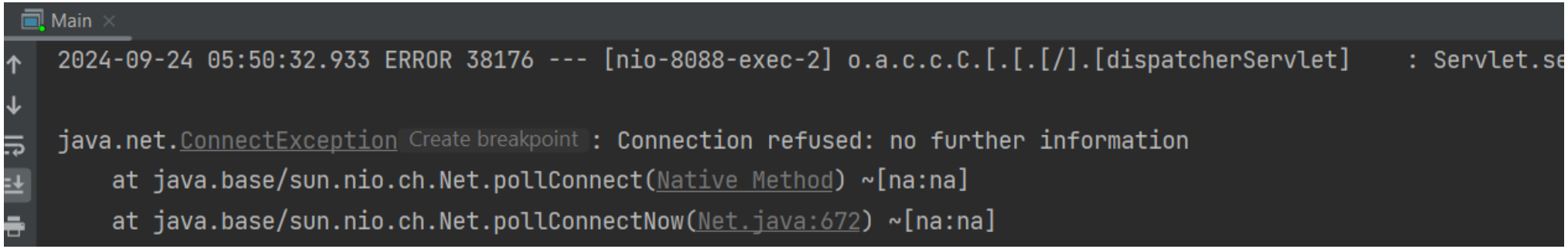
You may encounter a 500 Internal Server Error in Postman, but you can resolve it by checking your console for more information.



4th Task

If the issue in your 3rd task has been resolved, you may continue to see a 500 Internal Server Error in Postman. The console will display “Connection refused: no further information,” indicating that your other services are not running. You can ignore this message.

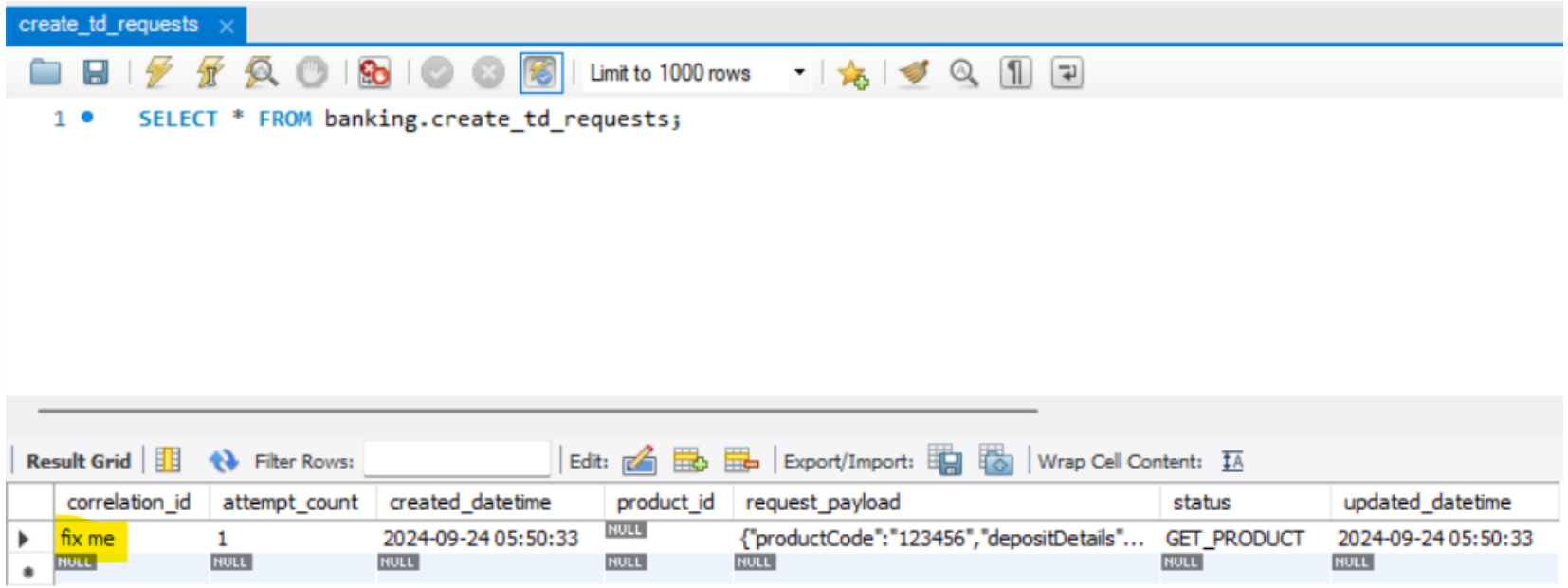




4th Task - continuation

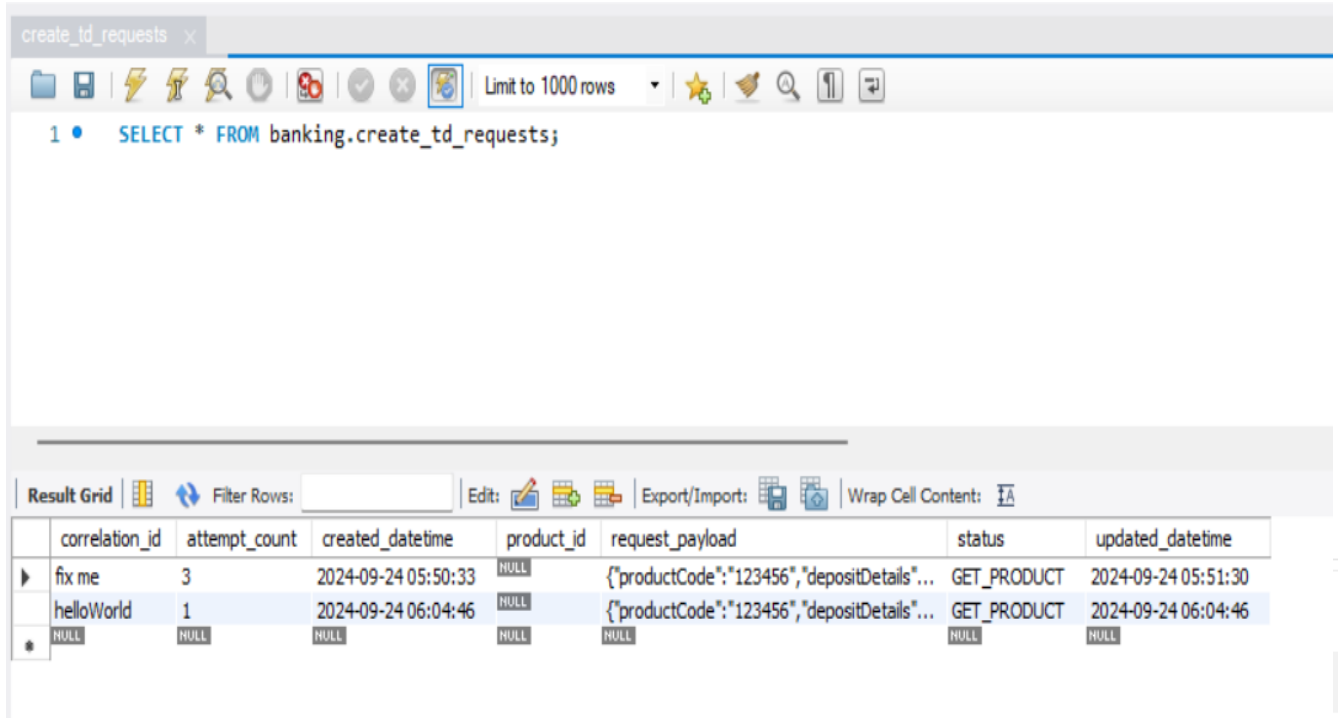
Review your workbench to find the record in the created_td_requests table.
Check the value in the correlation_id column, which currently shows “**fix me**.” This is incorrect; the value should be obtained from the header key in your previous task, and the correct value must be '**HelloWorld**’.

You can debug your code using your IDE.



5th Task

If you think you resolved 4th task, run Postman again. Check if correlation_id is populated with the correct value. If it is, run postman again, and you will see the error message “Correlation id already exists.”



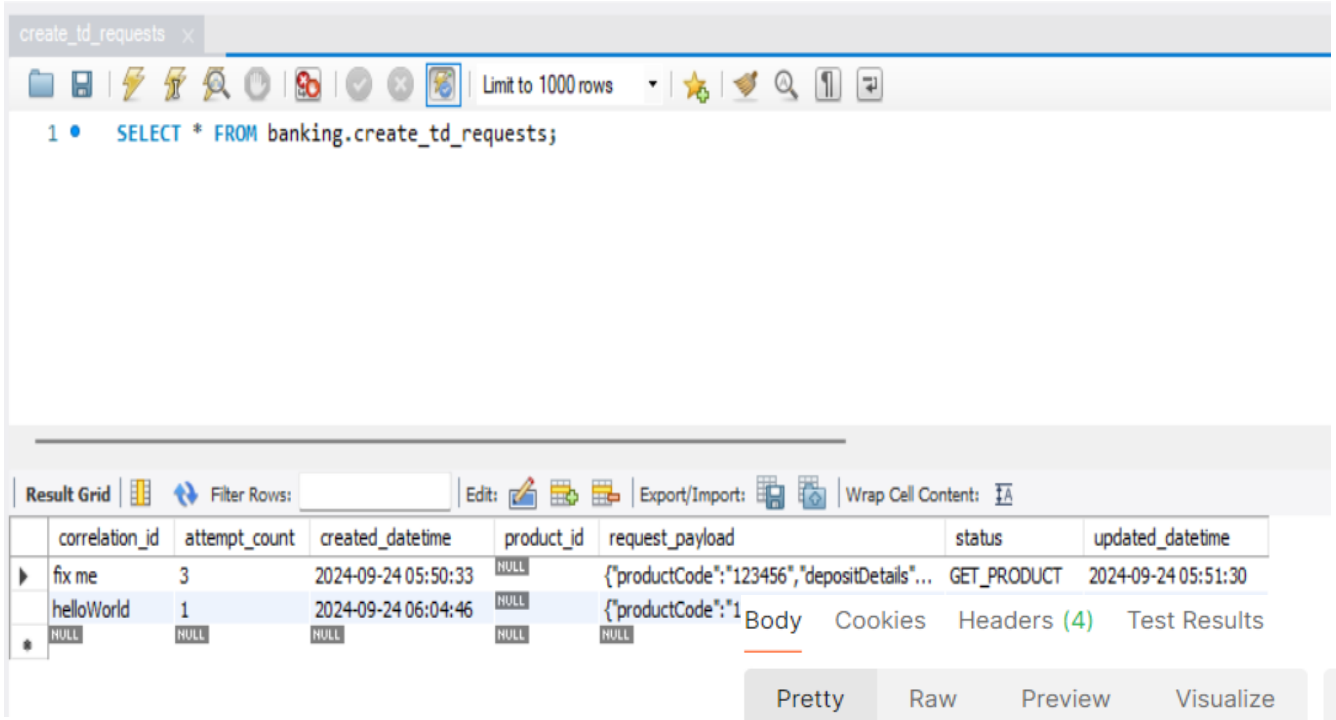
Status: 400 Bad Request

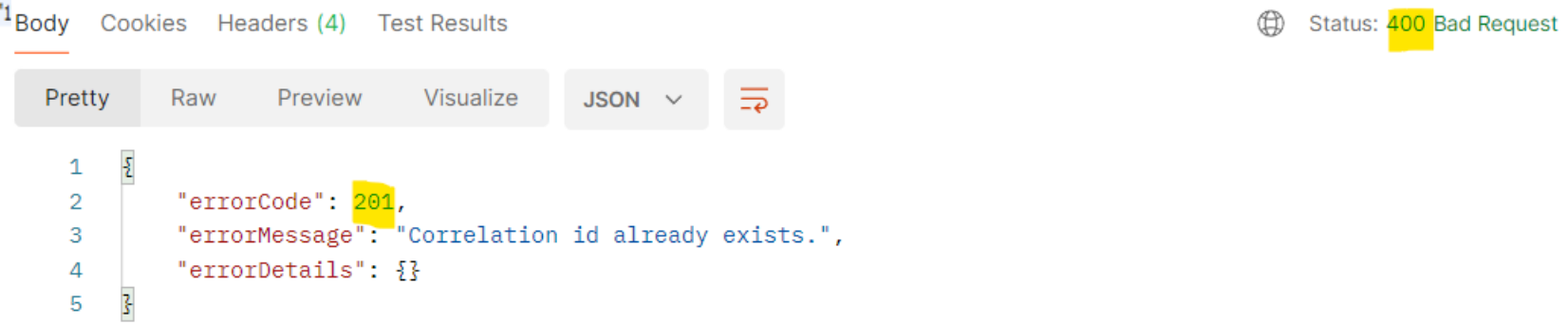
JSON

```
{
    "errorCode": 201,
    "errorMessage": "Correlation id already exists.",
    "errorDetails": {}
}
```

5th Task

If you think you resolved 4th task, run Postman again. Check if correlation_id is populated with the correct value. If it is, run postman again, and you will see the error message “Correlation id already exists.”





5th Task - continuation

The expected result is that the error code should be 400.

Note: Avoid using the hardcoded value 400.