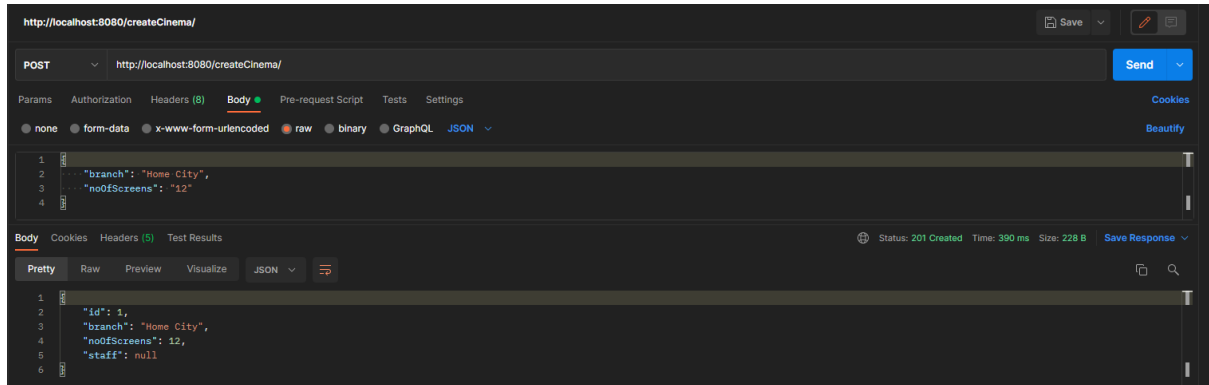


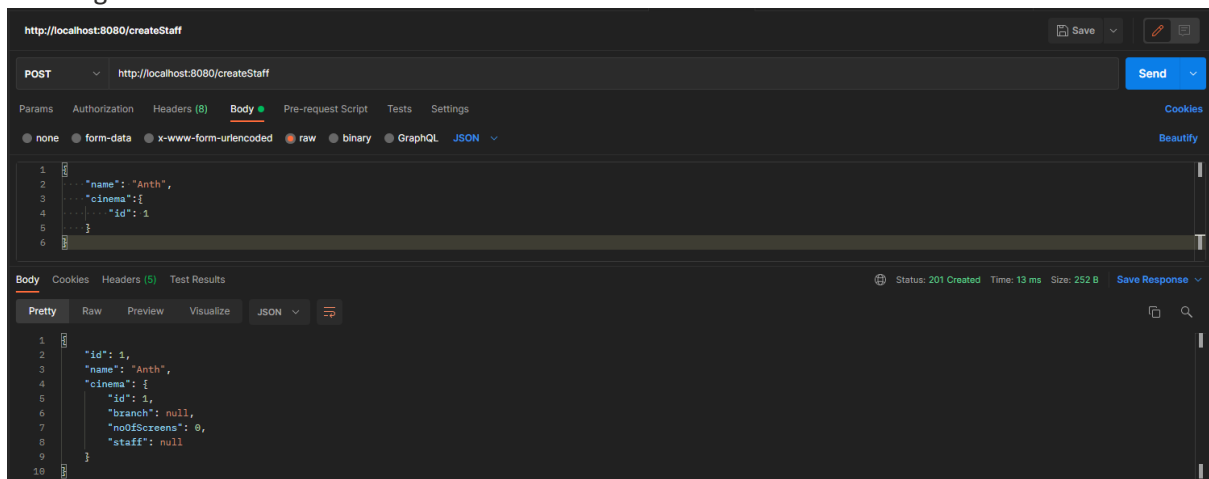
QA Project

- Screenshots showing your postman requests and the output from the API.

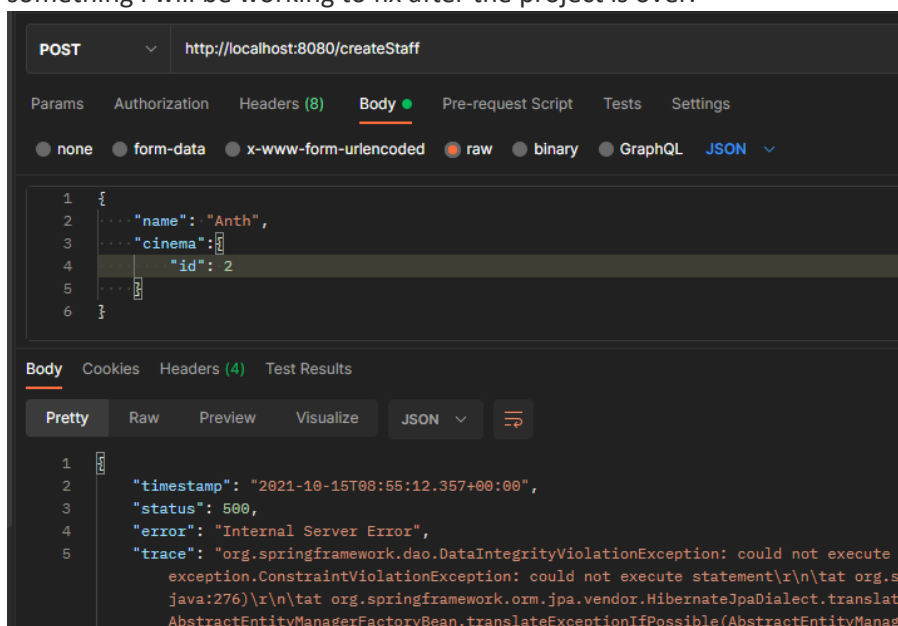
While working on this project I only managed to implement relations and DTOs with cinema and staff with many staff at a cinema.



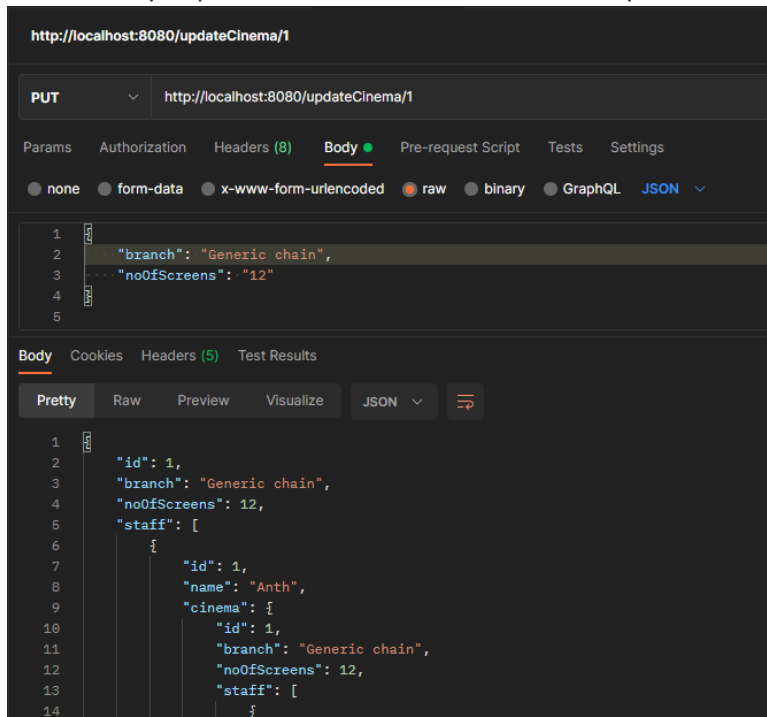
Creating cinema with status code 201 returned



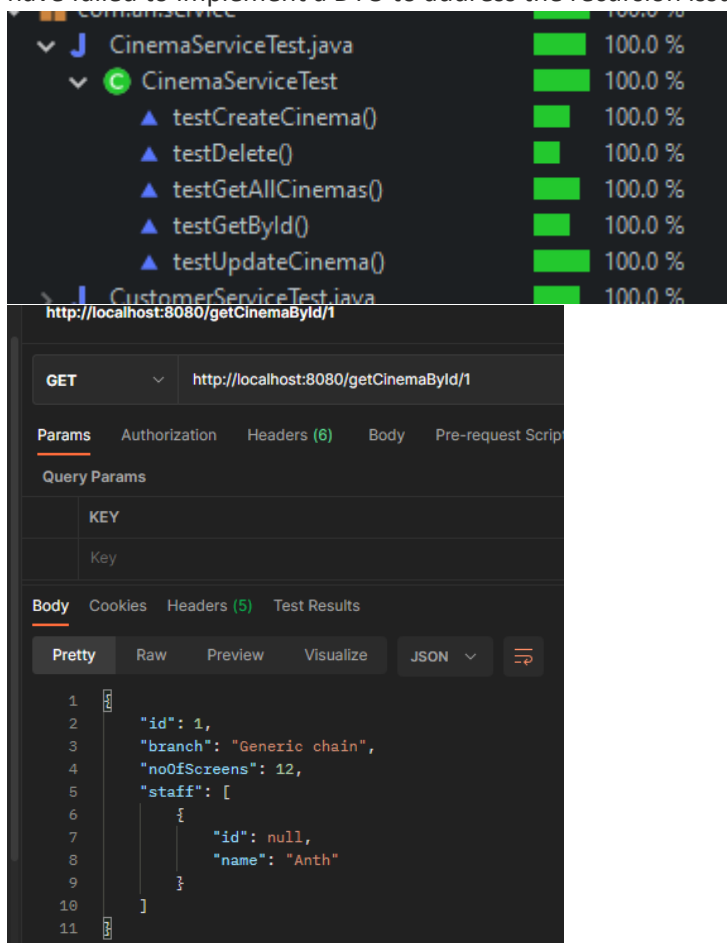
Creating staff and assigning it the cinema with "id 1" no data is returned for the cinema and this is something I will be working to fix after the project is over.



Creating a staff member with a cinema that does not exist returns the status code of 500 as I have not correctly implemented a cinema not found exception.

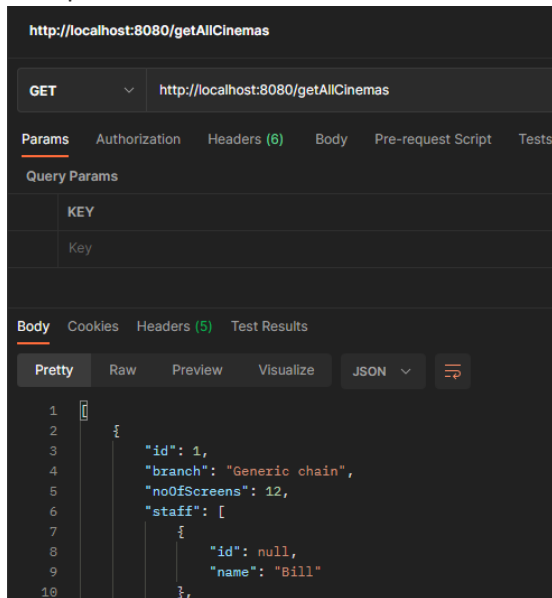


Update cinema does update the branch as expected but even though the tests pass as seen below. I have failed to implement a DTO to address the recursion issue.

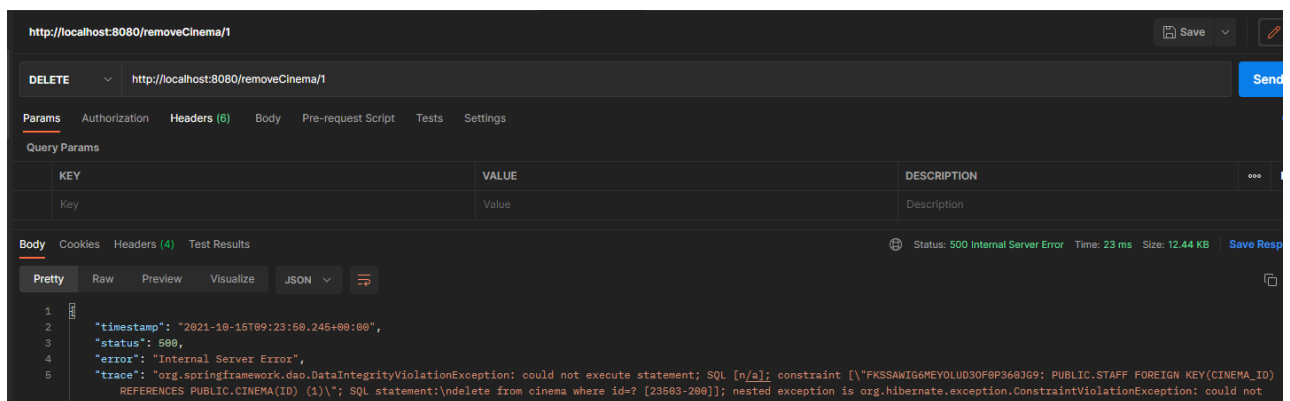


Anthony Harrison

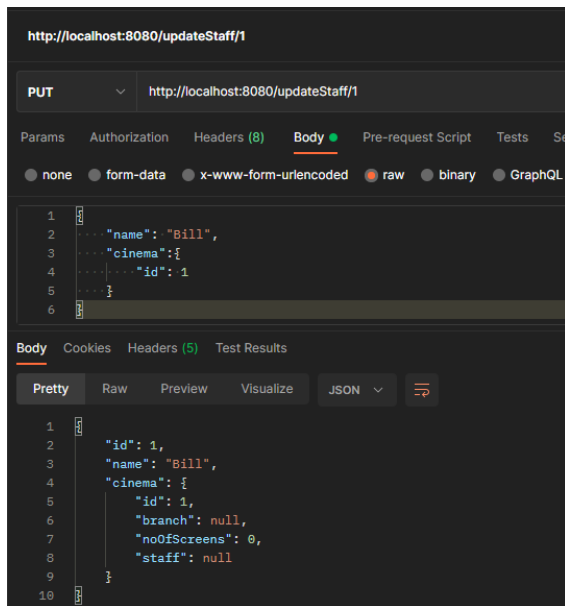
Getting a cinema by id does show all the staff in that cinema however it does not show the staff ids' as expected



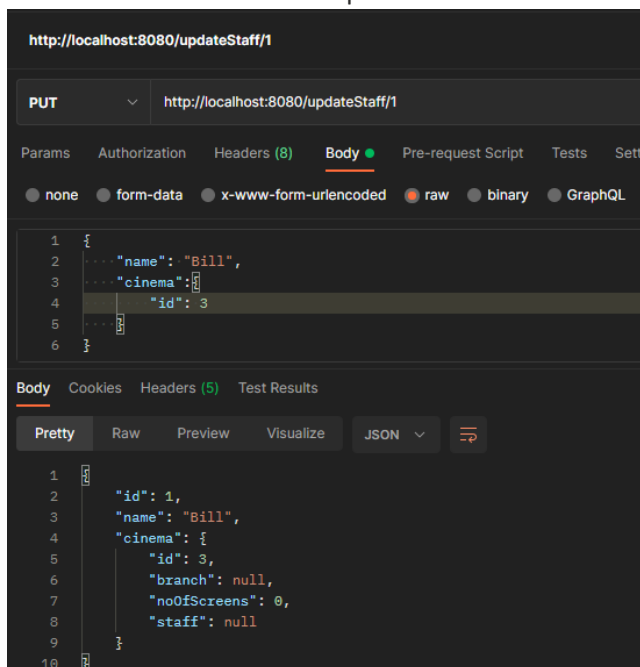
Getting all cinemas shows the same.



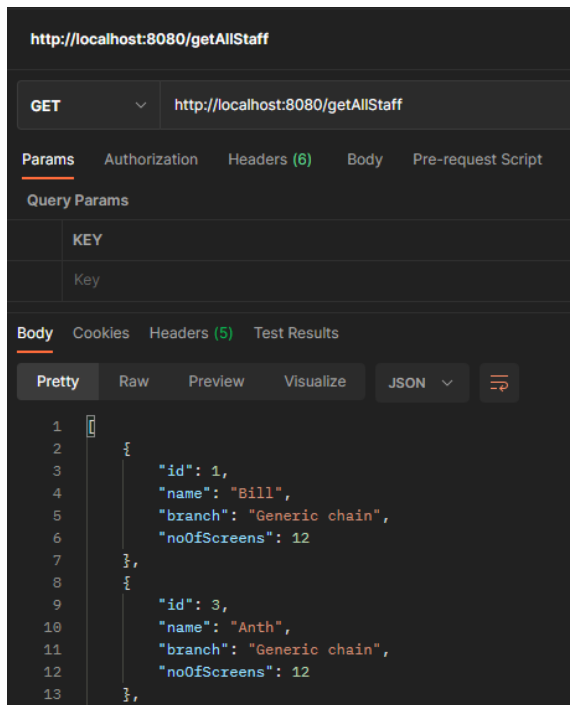
Removing cinema does not work if it has a staff member assigned it must be completely empty. This is another case of not checking because it passed the integration test.



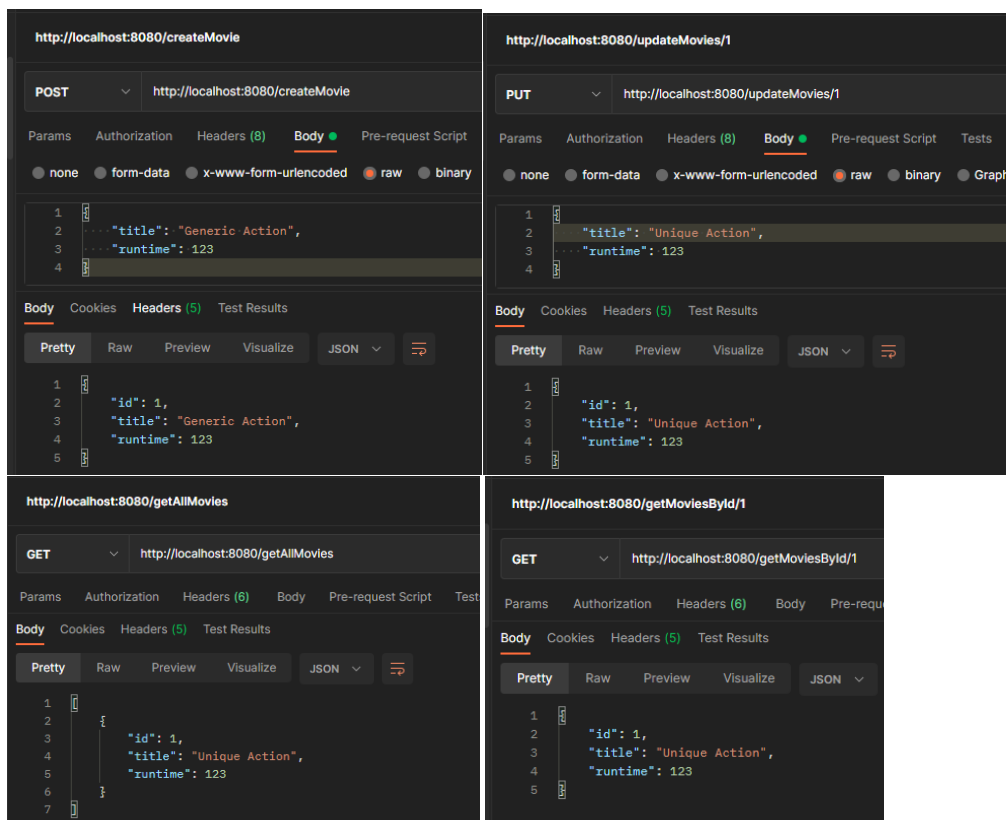
Staff can have their names updated.



And the cinema they are assigned to.



The next 4 images show the movies class these work as intended and the same can be said for the customer class as these 2 are in a state before relations and DTOs.



- o Screenshots of your database to prove that data is being persisted.

The screenshot shows the H2 console interface. On the left, a tree view displays the database schema for 'jdbc:h2:mem:testdb'. The 'STAFF' table is selected, showing its columns: ID, NAME, and CINEMA_ID. The 'Indexes' section for STAFF shows a primary key 'PRIMARY_KEY_4' and a foreign key 'FKSSAWIG6MEYOLU'. The main panel on the right shows the SQL statement 'SELECT * FROM STAFF' and its execution results. The results table has 4 rows and 3 columns: ID, NAME, and CINEMA_ID. The data is as follows:

| ID | NAME | CINEMA_ID |
|----|------|-----------|
| 1 | Anth | 1 |
| 3 | Anth | 1 |
| 4 | Anth | 1 |
| 5 | Anth | 1 |

Below the table, it indicates '(4 rows, 2 ms)' and an 'Edit' button is present.

Staff shown in the h2 console including the foreign key cinema id.













The screenshot shows the H2 console interface. On the left, the 'CINEMA' table is selected in the tree view, showing columns: ID, BRANCH, and NO_OF_SCREEN. The main panel on the right shows the SQL statement 'SELECT * FROM Cinema' and its execution results. The results table has 2 rows and 3 columns: ID, BRANCH, and NO_OF_SCREEN. The data is as follows:

| ID | BRANCH | NO_OF_SCREEN |
|----|---------------|--------------|
| 1 | Generic chain | 12 |
| 2 | Home City | 12 |

Below the table, it indicates '(2 rows, 2 ms)' and an 'Edit' button is present.

Cinema shown in the h2 console

- Screenshot of your test results, including coverage report.

| QA_Project (15 Oct 2021 09:13:27) | | | | | |
|-----------------------------------|---|----------------------|---------------------|--------------------|--|
| Element | Coverage | Covered Instructions | Missed Instructions | Total Instructions | |
| ▼ QA_Project |  89.2 % | 2,780 | 335 | 3,115 | |
| ▼ src/main/java |  76.4 % | 1,082 | 335 | 1,417 | |
| > com.ah.data |  70.0 % | 322 | 138 | 460 | |
| > com.ah.dto |  68.3 % | 226 | 105 | 331 | |
| ▼ com.ah.exception |  0.0 % | 0 | 46 | 46 | |
| > CinemaNotFoundException.java |  0.0 % | 0 | 23 | 23 | |
| > StaffNotFoundException.java |  0.0 % | 0 | 23 | 23 | |
| > com.ah.rest |  83.6 % | 189 | 37 | 226 | |
| > com.ah |  37.5 % | 3 | 5 | 8 | |
| > com.ah.service |  98.8 % | 335 | 4 | 339 | |
| > com.ah.config |  100.0 % | 7 | 0 | 7 | |
| > src/test/java |  100.0 % | 1,698 | 0 | 1,698 | |

76.4% is higher than the required 60% but I feel it is missing some essential testing as the exceptions are not covered at all and some of the staff controller tests are not in a working state and had to be commented out to build the jar file.