Microservices Architecture and Programming

**Lab Practical and date** – Practical 5, 16th October 2020

**Name and Roll Number**- Het Shah, 17BIT103

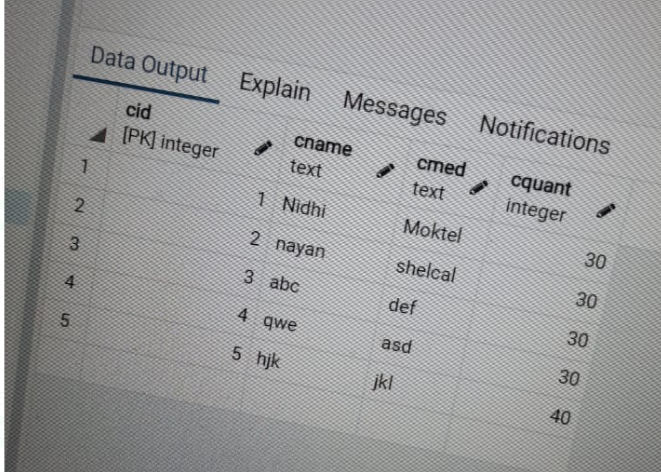
**Practical Objective-** Designing Distributed transaction based Micro-service application

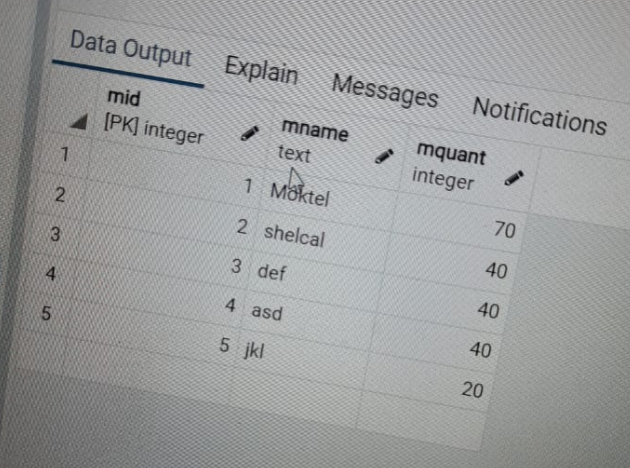
**Steps Involved-**

We implemented SAGA based transcations, based on postgresSQL and CloudAMQP as the messasing platform. The business solution implemented was to order medicine from a client via a service and the another service handle the stock inventory.

**Running this project**

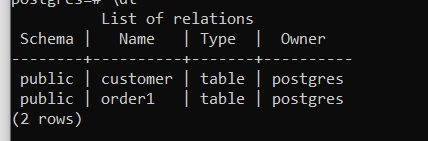
* + Install PostgresSQl and create the two tables

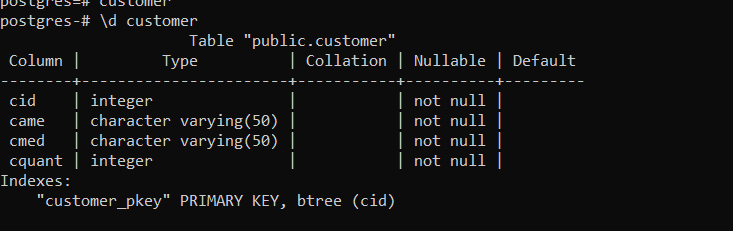


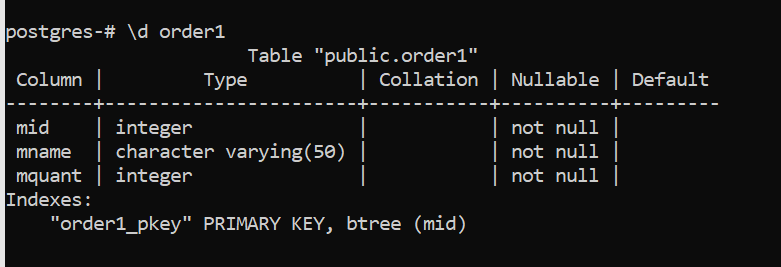


* Run Customer.py and enter the details
* Run Stock.py and enter the details
* Order is either accepted or rejected

DataBase Screenshots

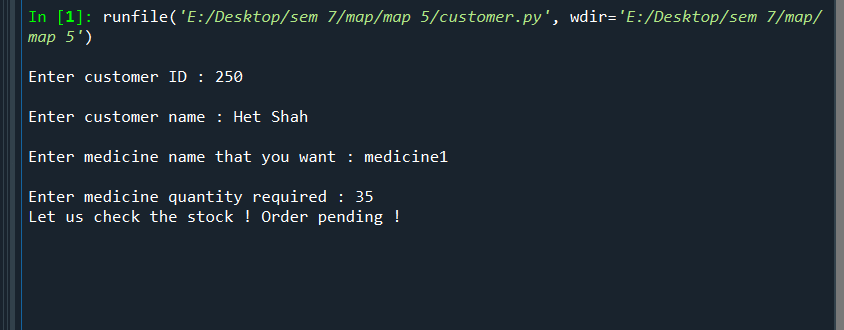


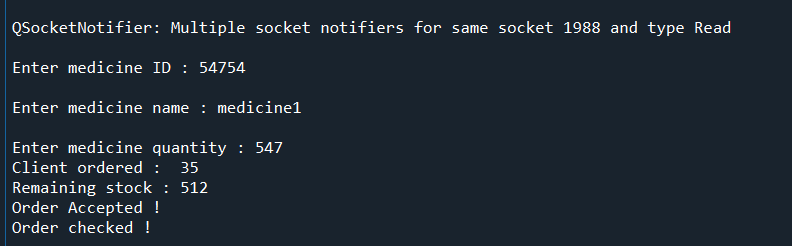


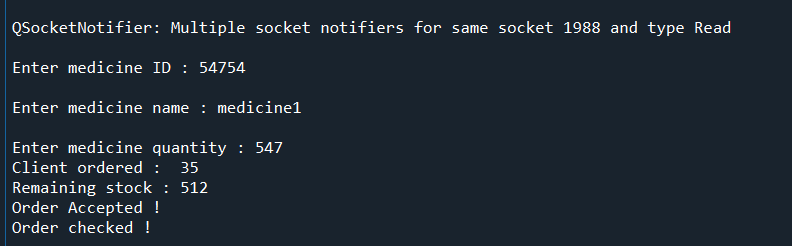


Running Screenshots

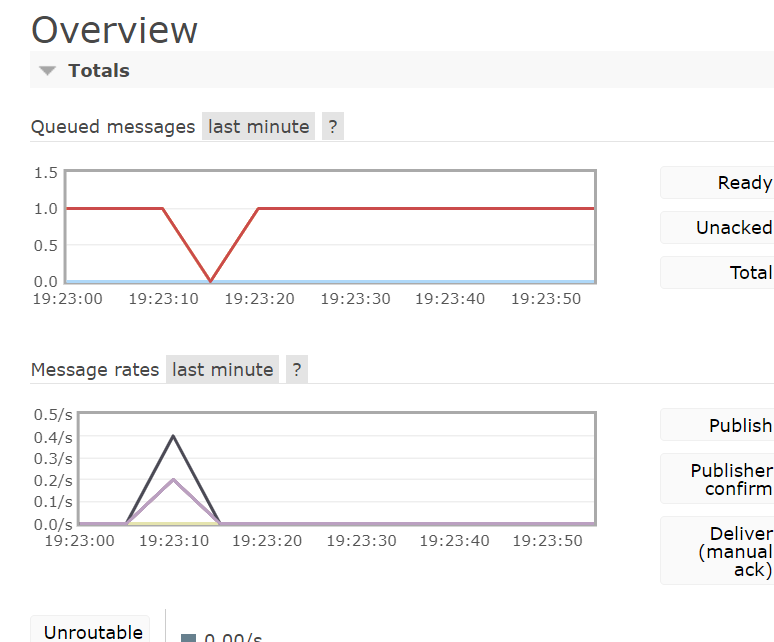
**Order Accepted Scenario**





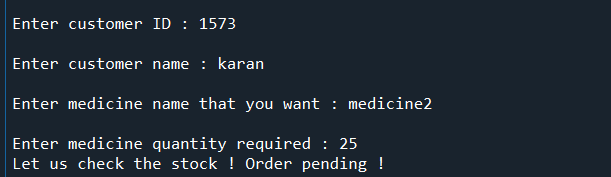


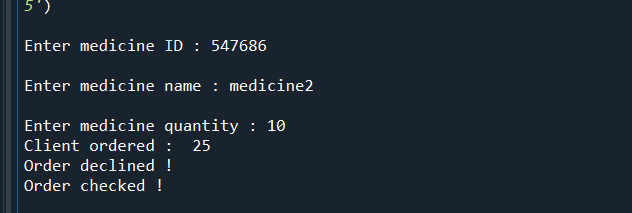
Since order<=stock, the order is accepted



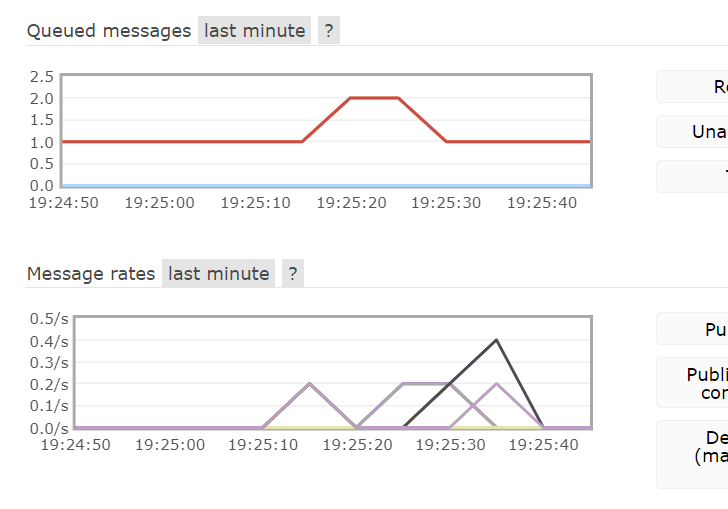
Message Activity

**Order Rejected Scenario**

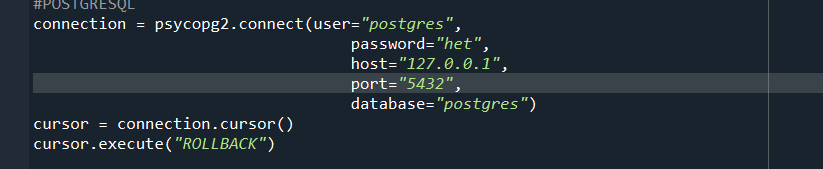




Since the stock is less than order value, it got rejected



Message Activity



Rollback is executed since the transaction did not occur

Conclusion

In this practical we learned about distributed tranactions based on the saga pattern, looking at two use cases consisting of accepted order and rejected order.