Contents

- Minor Project (Online Library Management System) [CONTD]
- Project Flowchart

Minor Project (Online Library Management System) [CONTD]

TASK : Request to Transaction creation

Note : One request has been placed , it's the admin's responsibility to process it . Once processed and approved , a PENDING transaction will be created which will become SUCCESS once entity level changes take place successfully and a transaction row is created in the Transaction table.

HINT :

```
@PostMapping("/admin/process_request/{requestId}")

    public void processRequest(@Valid @RequestBody AdminProcessRequest
adminProcessRequest)

    {

        //check if request is valid
```

```
        //check if admin is valid to process this request
        //check if request is not already process

        //If REJECTED
            //change request status
            //set admin comment in request if its not null
            //save request [NO TRANSACTION CREATED]

        // If APPROVED

                //change request status
                //set admin comment in request if its not null
                //save request
                //create Transaction
                                -save PENDING transaction
                                -case ISSUE
                                        --make Entity level changes ie assign student in book
table and save book

                                -case RETURN
                                        --make Entity level changes ie assign student in book
table and save book
                                        --calculate fine
```

-update Transaction status as complete and save transaction

}

Note :

1. Make sure an Admin should not be able to process a request which is not assigned to them .

2. Make sure Admin is allowed to process only PENDING requests .

3. For every ADMIN APPROVED request there will be a transaction created after being processed by admin . Requests can be approved or rejected but transactions will be created only if Request is APPROVED by the admin. In the rejection case , we just update the request status and add an admin comment in the request object .

4. When Admin processes a request , he should get a JSON Process Request response back telling him request final status , requestId and admin comment.

Q. If we are adding Student object to Book object , then do we need to book to Student's bookList ?

HINT : We only do the change for the referencing table (ie the one which has foreign key and the changes will reflect in referenced table through back referencing)

TASK : Complete the logic for calculateFine function in transaction service class .

-> Here we will get the book from the return request and get the list of successful transactions for this book . The last successful transaction should be an ISSUE one and it will give us the issue date . Then we will find the number of days between issue date and return date . If this number exceeds the allowed number of issue days , then there will be a fine calculated else fine will be 0.

HINT :

We will need to write our own query in TransactionRepository .

select t.id , t.created_on , t.external_transaction_id , t.fine, t.transaction_status ,t.request_id from transaction t JOIN request r on t.request_id = r.id where r.book_id=?1 and t.transaction_status='SUCCESS' order by t.created_on desc;

MAGIC FORMULA :

long noOfDaysPassed = TimeUnit.DAYS.convert(timeDiff, TimeUnit.MILLISECONDS);

TASK : Can we do something to accommodate these 4 tasks in one api :

TASK 1 : Create an API to get all books of a particular genre

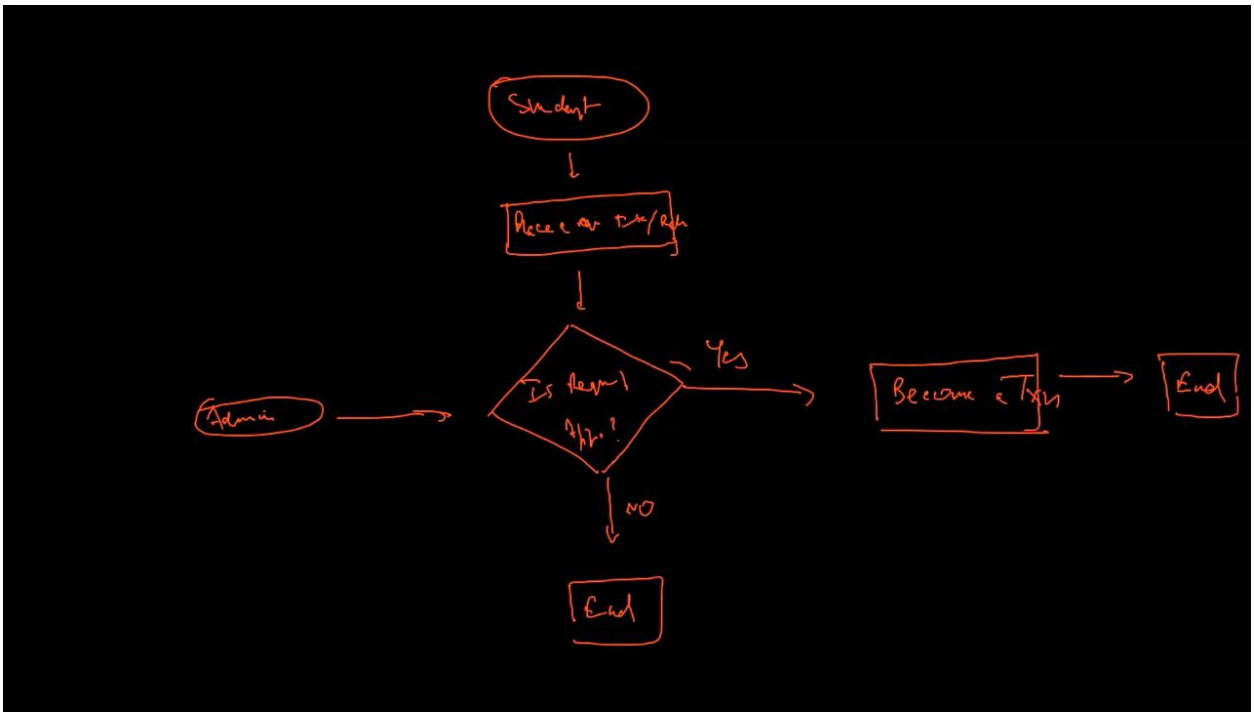
TASK 2 : Create an API to get all books written by a particular author

TASK 3 : Create an API to get all available books which are not yet issued to anyone .

TASK 4 : Create an API to get all books with a given book id .

Awesome Work . The final step is to test the application and check if we have built a fine app .

COMPLETE FLOWCHART :



HOMEWORK : What is Unit Testing ?

https://www.geeksforgeeks.org/what-is-unit-testing-and-why-developer-should-learn-it/

https://www.geeksforgeeks.org/unit-testing-software-testing/

What is Mockito and Junit ?

www.geeksforgeeks.org/unit-testing-in-spring-boot-project-using-mockito-and-junit/


LIVE
Java
Backend
Development