**Task 05 || Spring Boot, ORM (JPA, Hibernate)**

Not Completed

1. **Duration:** 3 working days
2. **Description:**
3. **Learning metatrails**
4. **Helper Project**

* Simple CRUD operations to manage student info is done in this project using MySQL Database and JPA in 4 layers.
* You have already cloned this project in task 02
* fetch branches: *git fetch*
* Migration from H2 DB to MySQL is done in branch “hp/student-info-manage-mysql-jpa-part-01”

To see checkout branch: *git checkout hp/student-info-manage-mysql-jpa-part-01*

* Separate DB primary key and created, updated, version column related work done in branch “hp/student-info-manage-mysql-jpa-part-02”

To see checkout branch: *git checkout hp/student-info-manage-mysql-jpa-part-02*

* Description can be found in README.md file.
* Postman collection kept in *documents* package.
* You can see the project structure and browse the code.

1. **What to submit**

Prepare a doc with the answer of bellow questions:

* In part 02 we have created a new table “BOOK\_V2” for book info to keep things simple. Imagine your book-info-manager-1.0 application already on live and BOOK table has huge amount data.

So, what will be the problem if we create new table?

What problems we have to deal with if we want to proceed with existing “BOOK” table?

Suggest a solution to this migration problem.

* If you work with a live application, which option would you choose for the "spring.jpa.hibernate

.ddl-auto" property? and Why?

* How derived queries in JPA repository works?
* Write about the newly added dependencies in *build.gradle* file.
* Write about the newly added properties in *application.properties* file.
* Write about new annotations we used:

@GeneratedValue, @PrePersist, @Version

* Submit your code for task 04 in GitHub. See how to submit section.

1. **How to submit**

* Name your doc as: <nick name>\_<emp\_id>\_<task id>.

Ex: atique\_202015\_task\_04.docx

* Upload your doc in documents folder
* commit your changes in your task 04 feature branch:

*git commit -m "[bim] task 04"*

* Push the changes to your feature branch: *git push*
* Create a merge req in GitHub [your feature branch→ your release branch]

Ex: *feature/bim/atique-202015/task-04 → bim/atique-202015/release-1.0*

* Follow up instructors’ feedback on GitHub merge request.