**Task 04 || Spring Boot, ORM (JPA, Hibernate)**

1. **Duration:** 3 working days
2. **Description:**

* This task has 2 parts.
  + Connect with a Relational Database (MySQL) in place of H2 DB.
  + In Task 02, DTO and Entity/Domain object has the same properties. Then what is the difference between them? why we need separate objects? Read these articles:

1. <https://www.baeldung.com/java-dto-pattern>
2. <https://www.linkedin.com/pulse/difference-between-entity-dto-what-use-instead-omar-ismail/?trk=pulse-article_more-articles_related-content-card>

In this task we will add some properties to the BookEntity to trace and version our data.

* Add below properties in *application.properties* file

*spring.datasource.url=jdbc:h2:mem:testdb*

*spring.datasource.driverClassName=org.h2.Driver*

*spring.datasource.username=sa*

*spring.datasource.password=*

*spring.jpa.database-platform=org.hibernate.dialect.H2Dialect*

*spring.h2.console.enabled=true*

* Add Below dependency in *build.gradle → dependencies*

*implementation 'org.springframework.boot:spring-boot-starter-data-jpa'*

*runtimeOnly 'com.h2database:h2'*

* Hit the URL: <http://localhost:8084/book-info-manager-1.0/h2-console>
* Below console will be displayed on browser:
* Press the **Connect** button and see your database like:

1. **Learning metatrails**
2. <https://www.baeldung.com/spring-boot-h2-database>
3. <https://www.youtube.com/watch?v=9SGDpanrc8U&ab_channel=Amigoscode>
4. <https://www.baeldung.com/learn-jpa-hibernate>
5. <https://www.baeldung.com/transaction-configuration-with-jpa-and-spring>
6. **Helper Project**

* This is an already done project to help you.
* Simple CRUD operations to manage student info is done in this project using H2 Database and JPA in 4 layers.

1. You have already cloned this project in task 02

2. fetch branches: *git fetch*

3. checkout branch: *git checkout hp/student-info-manage-h2-jpa*

4. Description can be found in README.md file.

5. 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 these two questions:

* What is ORM?
* Describe Persistence Life Cycle.
* Why JPA?
* How Builder Pattern work?
* 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:

@Builder, @Repository, @Entity, @Table, @Id, @Column, @Enumerated

* Submit your code for task 03 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\_03.docx

* Upload your doc/pptx in documents folder
* commit your changes in your task 02 feature branch:

*git commit -m "[bim] task 03"*

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

Ex: *feature/bim/atique-202015/task-03 → bim/atique-202015/release-1.0*

* Follow up instructors’ feedback on GitHub merge request.