**DO NOT SHARE THIS EXAM WITH ANYONE ELSE**

**FAILING DOING THIS WILL RESULT IN A SERIOUS PENALTY**

**Submission: Students can submit up to 10 minutes late without penalty. After that, every 1 hour late will result in 30% deduction.**

**Final Test**

**Total hours:  3 hours**

Rules

1. This is an **open book and INDIVIDUAL.** Students can use the Internet and Canvas to support their answers.
2. Exchanging answers or ideas between students, online chatting or posting questions to get support from other people, family, friends, or verbal chat during the test time is strictly prohibited, and will be considered as cheating
3. Use your own words. Do not just copy and paste from the Internet. Failing doing this can be considered as plagiarism.

Notes

1. Read through the test carefully and plan your answers. Don’t jump to answer them right away. Students can ask for clarity via Teams or email (I prefer Teams).
2. There are 2 parts of the final test: Theory and Programming.
3. Students provide answers for theoretical questions in a doc file named StudentName\_StudentID\_Theory.doc
4. For the programming question, provide the answer in a folder with all source code and related files. Zip the folder and doc into a folder name StudentName\_StudentID.zip.
5. Submit the zip file to Canvas.

**Theory (60 marks)**

1. **(10 marks)** Question 1. What is Dependency Injection? And why do we call it Inversion of Control? Why do we need Dependency Injection? List a few frameworks that support Dependency Injection in Java.
2. **(10 marks)** Question 2. What is Object-relational Mapping (ORM)? List some popular ORM tools. Compare between ORM framework and traditional JDBC.
3. **(25 marks)** Question 3. List at least 5 design patterns that we have learned in this course, and provide an example for each of them (For the snippets of codes, you can take screenshots, edit, and include them in your doc)
4. **(10 marks)** Question 4. What are Java IO streams? List some stream implementations and discuss about their abilities.
5. **(5 marks)** Question 5: Tell me some topics in this course that you like, and the field of Computer Science you are into, and why.

**Programming (40 marks)** 10 + 10 + 10 + 10

Library system is a very important tool to support researching. Use Spring framework to build a backend component for that system. Students can use Spring with or without SpringBoot, Hibernate or JPA, and SpringMVC RESTful API architecture.

The system allows users to manage sub-libraries. Each sub-library contains a list of authors. Each author is associated with 1 or more books. For the sake of simplicity, each book belongs to only 1 author.

Information about a sub-library is id (integer), subject (string)

Information of an author is id (integer), and name (string), and academic credentials (string)

Information of a book is id (integer), name (string), and date of creation (date)

* 1. Build entities class for sub-library, author, and book. Make sure the relationships between authors and books are **one to many and bi-directional.** The relationship between sub-library and author is **one to many and uni-directional.**
  2. Build a REST controller named **LibraryController**to perform **add and update** operations on sub-libraries. Also, a sub-library can be searched by subject and the search results will be ordered (descending or ascending)
  3. Build a REST controller named **AuthorController**to perform **add, update,** and **delete** operations on authors. Also, an author can be searched by name or academic credentials and the search results will be ordered (descending or ascending)
  4. Build a REST controller named **BookController**to perform **add, update,** and **delete** operations on books. Also, a book can be searched by name or the created date and the search results will be ordered (descending or ascending)

Students must define necessary request params or path variables in the controllers in order to provide above-mentioned features.

Students must build necessary repositories and service classes to be called by these controllers.

Test your system using Postman and show me the testing results in a doc.

\*\*\*\*\*