Introduction-Code

Purpose

The purpose of this "code" folder is to give the user a full presentation of our code in our project, the code is not only about building the databases, but also include:

- SQL code that creates the database, and that creates and populates the tables, triggers, and stored procedures and functions.
- Java code that creates the database, and that creates and populates the tables, triggers, and stored procedures and functions.
- Java code for the stored procedures and functions for the API and any private stored procedures and functions (Derby only).
- Java or other code that implements any demo software that uses the project database.

Structure

We designed our project strictly following the OODs principles. We separate the data model with their services into different classes in order to make the structure clear. We also encapsulated all the related functions for different tables into different classes to make it easy for the user to implement CRUD operations. (For more detailed description of this, please refer to **/documentation/Project Design.pdf** and **/documentation/Project API.pdf**).

In addition, we create a User Interface to provide convenience for the user. The user can see a menu in the terminal and choose different operations and do the following operations.

So, we put our java files into different parts:

- "model" directory: Java files related to data model
- "service" directory: Java files related to data model services
- "controller" directory: Java files related to the controller of user interface
- "view" directory: Java file that defines the user interface view in the terminal
- InOrder.java: Java file that runs on terminal to show the user interface

In the model directory, we have:

- Customer.java
- InventoryRecord.java
- Order.java
- OrderRecord.java
- Product.java

which represent our five data models.

In the service directory, we have:

- CustomerService.java
- InOrderModel.java
- InOrderFunctions.java
- InOrderTriggers.java
- InventoryRecordService.java
- OrderRecordService.java
- OrderService.java
- ProductService.java

to show the services related to data models

In the controller directory, we have:

- "commands" directory:
  - AddCustomer.java
  - AddInventoryRecord.java
  - AddProduct.java
  - ExamCustomer.java
  - ExamInventoryRecord.java
  - ExamOrder.java
  - ExamProduct.java
  - PlaceOrder.java
- Features.java
- InOrderCommand.java
- InOrderControllerImpl.java

to realize the function of our user interface.

In the view directory, we have:

ConsoleBasedView.java

to define how our user interface looks like.

This is only a brief introduction focusing on the structure of this folder. For more information on how to run this program using these codes, please go to the **Introduction.pdf** at the top level of this project's directory. For the design and structure of these codes, please refer to **/documentation/Project Design.pdf.**