INTRODUCTION

Overview

This **InOrder** program is an e-commerce program that enables three functions of a business.

- Managing products: Manage information about products that can be sold to customers.
- Tracking inventory: Track current inventories of products
- Processing orders: Process orders for products from customers.

Team Members

Our Team consists of four members: Wenxuan Guo, Lei Cao, Suyue Jiang and Yibao Hu.

Tools and Environment

The basic tools and environment we were using:

- Project Building: Java, Apache Derby
- Project Testing: JUnit5
- Project IDE: IntelliJ

Project Structure

There are three main parts in this repository: **Docs**, **Code** and **Tests.**

- Docs: Contains 4 parts to describe our project: Projects Functionality, Project design, Project API, Test plan.
- Code: Includes all of the code that implements our project
- Tests: Includes all of the tests that can be run to verify the functionality of the project.

You can find a document that describes the purpose of each folder and the files and directories that it contains. Also, we generated the Javadoc of the program and it is located in the documentation folder. You can start reading it from index.html and it can be read offline.

How to Run Our Project?

- **Get IntelliJ IDE:**

We use IntelliJ as our IDE. You can get IntelliJ through: https://www.jetbrains.com/idea/. The community version is free and open to everyone and the unlimited version can be used by verifying your university email address.

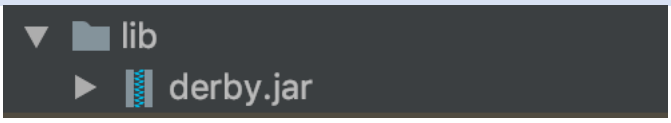
- **Import and set up the project:**

After importing the project into IntelliJ, you may see some errors regarding JUnit5 in the test files, this is because you are not installing Junit5. Click on the red bulb and it will assist you to add JUnit5 to the build path.

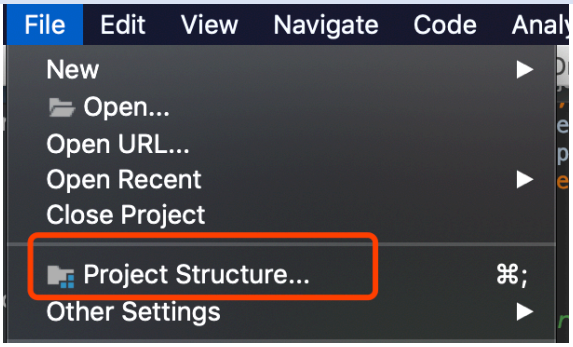
- **Configure derby**

There are many ways to configure derby in IntelliJ. One way we are using is to:

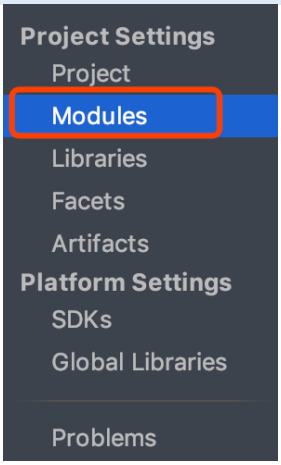
1. Create a new repository named 'lib' into the project
2. Paste "derby.jar" into "lib"

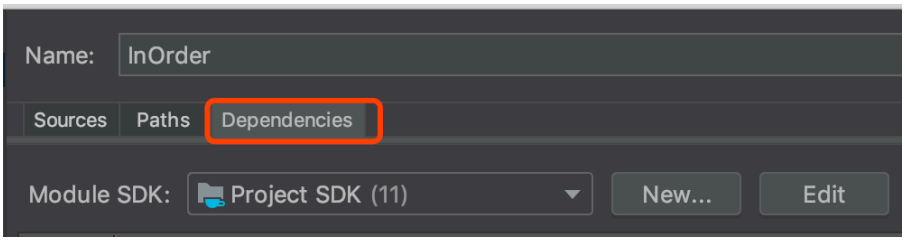


3. Click on File -> Project Structure.

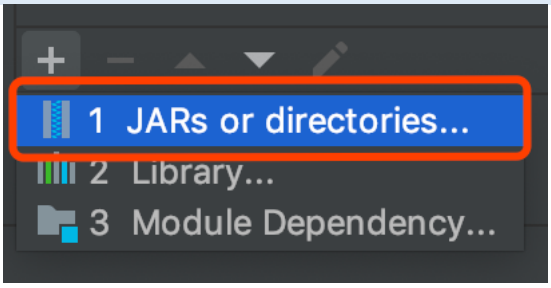


4.Click on Modules->Dependencies

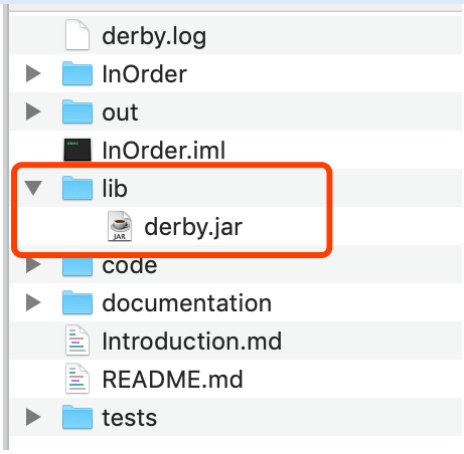




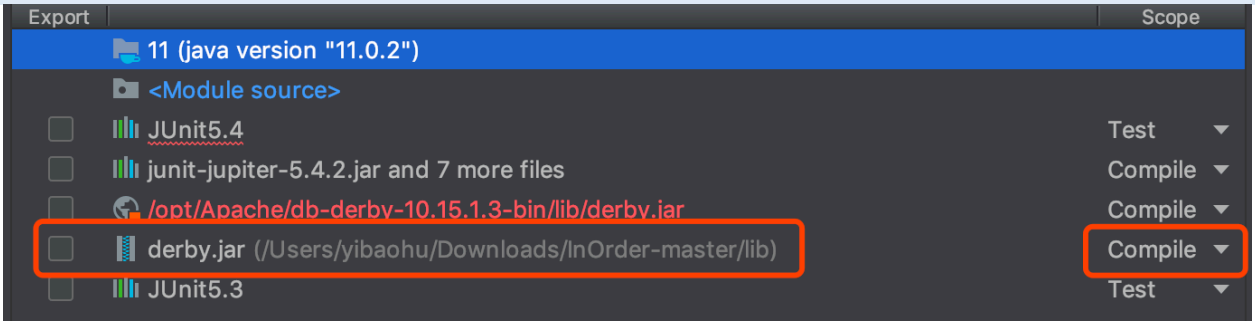
5. Click on the "+" button and choose "JARs or directories…"



6. Choose "derby.jar" which we just put in the "lib" directory



7.You will find “derby.jar” has been added to the list and make sure the Scope is “Compile” and click Ok or apply



- **Run the program**

This is a backend program without user interface. Basically, you can do two things in the program, one is running the tests. You can go to the test directory and find different tests that exam whether our services files work fine. We use Junit5 for our tests, if you hit run, you will see results at the button of your IDE, it will tell you how many tests that have passed.

The other thing you can do is to add data to the database we created. You can call APIs in each services file. You can find detailed description of the APIs is the **/doc/Project API.pdf** and go from there.