

ENSF 380

Exercises - Lesson 20B

The following exercises correspond to the video for Lesson 20B.

Exercise 20B.1

1. Download and extract PostgreSQL connector on your computer
2. Follow along with the demonstration to practice database connections and manipulations

- Download PostgreSQL connector
 - <https://jdbc.postgresql.org/download/>
 - Java 8 version
 - Can use platform-independent version
 - Also uploaded to course repository
- Follow along with the video demonstration
 - Code can be found in the course repository

Tip: Find the code examples in the course repository and follow along with the demonstration.

Lesson20_Databases/01_DemoApplication

Exercise 20B.2

- Follow along with the video demonstration
 - Code can be found in the course repository
 - Compare the differences between the two versions of the insert statement. Why is specifying the ID optional for this particular table?

Tip: Find the code examples in the course repository and follow along with the demonstration.

Lesson20_Databases/02_ModifiedInsert

Exercise 20B.3

1. Download the `competition.sql` file from the course repository.

Lesson20_Databases/03_Registration

2. Create the database and insert the data using the information provided in *competition.sql*.
3. Add the required code to `Registration.java` to enable database access and to update the provided database
4. The functionality of the missing code is explained above each test provided in the main method. Example output is provided in the course repository.

- Connect to the competition database using "oop" as the username and "ucalgary" as the password
- Complete the method `initializeConnection`
 - Creates a connection to the database
 - No arguments, does not return anything
- Complete the method `selectAllNames`
 - Takes in a `String` value representing a database table name
 - Returns a `String` containing any last and first names in the specified table
 - Each last/first name combination should be listed as `LName`, `FName` on its own line (no new line at the beginning or end)
- Complete the method `insertNewCompetitor`
 - Takes in six arguments as they appear in the `Competitor` table
 - If the specified teacher is not included in the `Teacher` table, throw an `IllegalArgumentException`
 - If the competitor is not between the ages of 5 and 18, throw an `IllegalArgumentException`
 - If above conditions are met, insert the competitor into the `Competitor` table
 - No return value
- Complete the method `deleteCompetitor`
 - Takes in a competitor's id as a `String`
 - Deletes the specified competitor from the `Competitor` table
 - No return value
- You may add any private data members and methods you wish
- All database-related try/catch statements should use the `SQLException`
- No required behavior for the catch statement (a print statement or stack trace is fine)
- All resources must be released, either in the `close` method or elsewhere in the code
- Method documentation should explain if/how each method is using the database