Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа №3

по дисциплине "Распределенные программные системы"

ТЕМА РАБОТЫ:

Работа с реляционными СУБД средствами JDBC API

Выполнил:

студент гр. ПРИ-120

Парахин К.В.

Приняла:

Проскурина Г.В.

Владимир 2022 г.

Цель работы:

Изучить средства JDBC API для работы с реляционными СУБД и получить навыки их практического использования.

### Задание

1. Разработать настольное (stand-alone) Java-приложение, выполняющее т.н. CRUD-операции (вставка, получение, изменение, удаление) с записями таблицы реляционной БД.
2. Реляционная таблица должна содержать столбцы символьного, числового и временного типов. В качестве первичного ключа должен использоваться автоматически генерируемый идентификатор.
3. Операции с записями должны выполняться с использованием параметризованных запросов.
4. Реализовать подключение к базе данных с помощью менеджера JDBC-драйверов и с помощью источника данных.
5. Выводить при вставке записи сгенерированное значение идентификатора.
6. В качестве клиента готового приложения должны выступать Junit тесты.
7. Переработать готовое приложение в соответствии с шаблоном DAO.

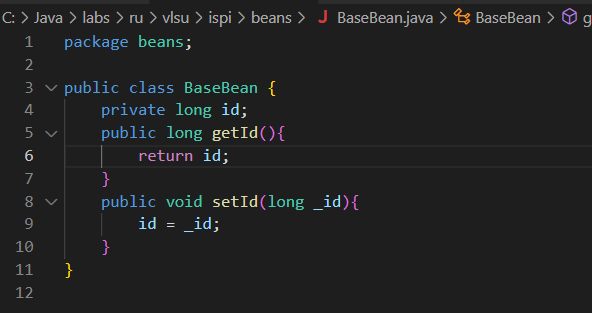
Выполнение работы:

Создаем окружение на файлов лабораторной работы № 3.

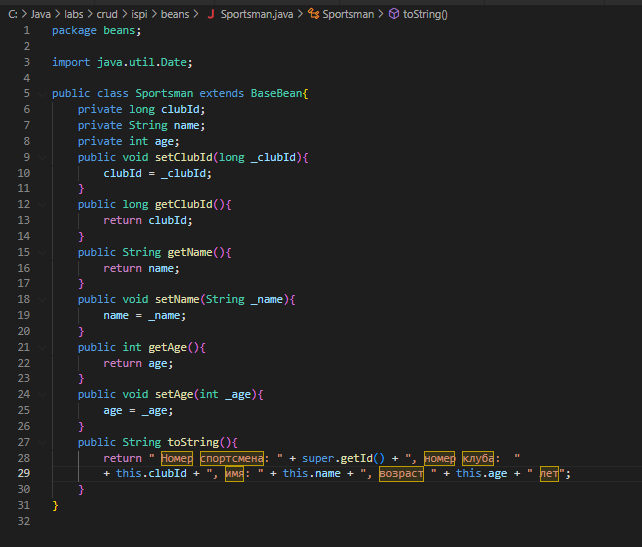
В данной работе будет использоваться подход: один DAO – класс на каждый Value-object

Создадим модели в папке beans:

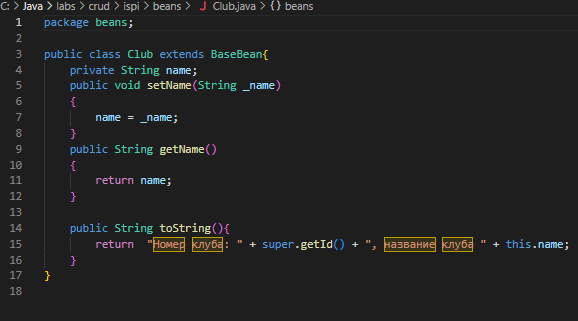
Модель BaseBean.java:



Модель SportsMan.java:

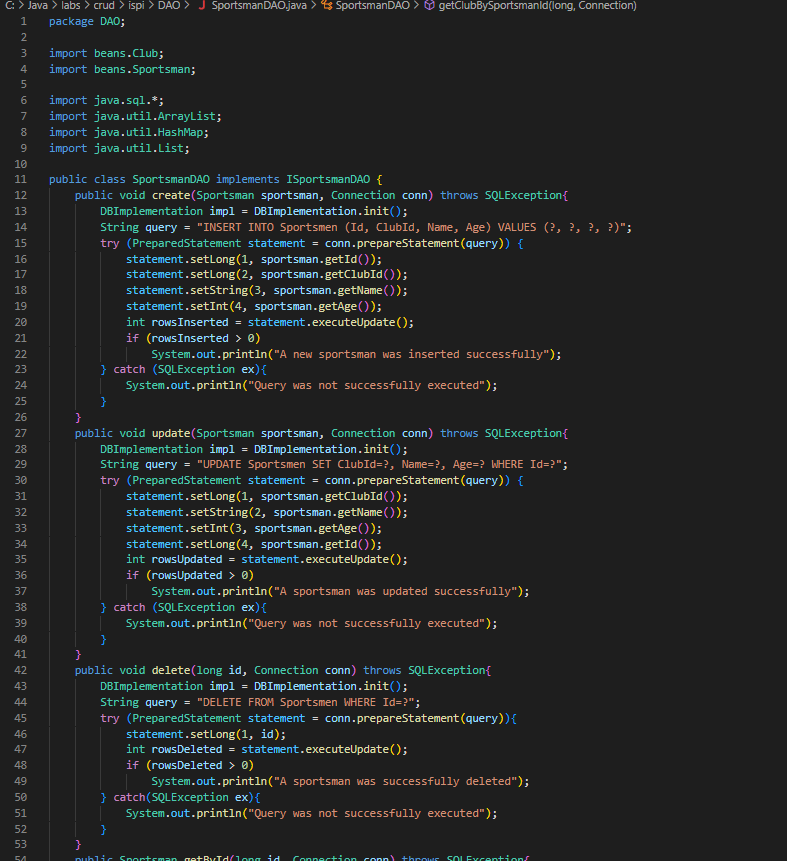


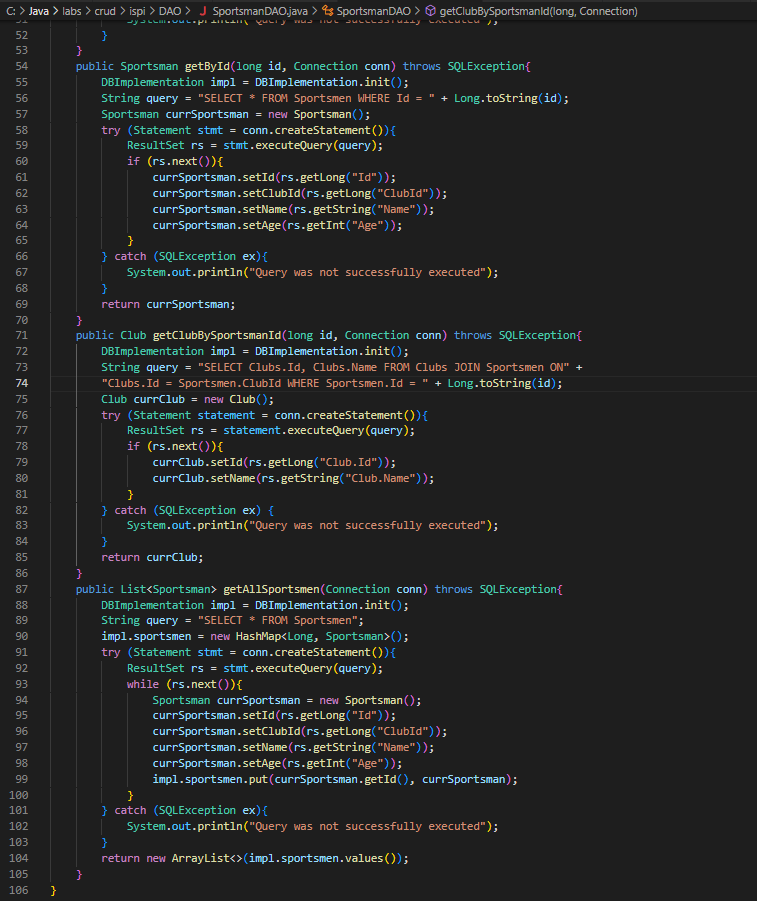
Модель Club.java:



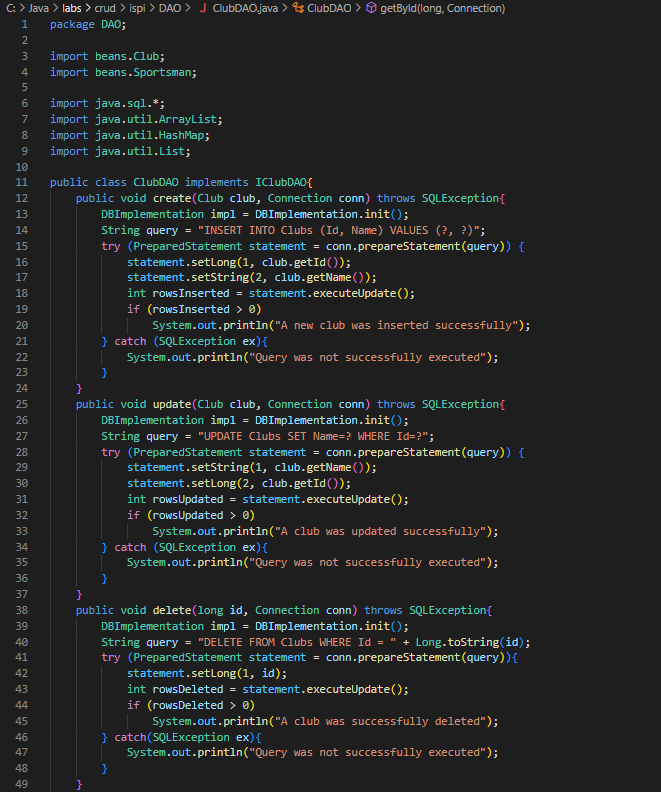
Создадим DAO-классы для каждой из модели (для каждого Value-object):

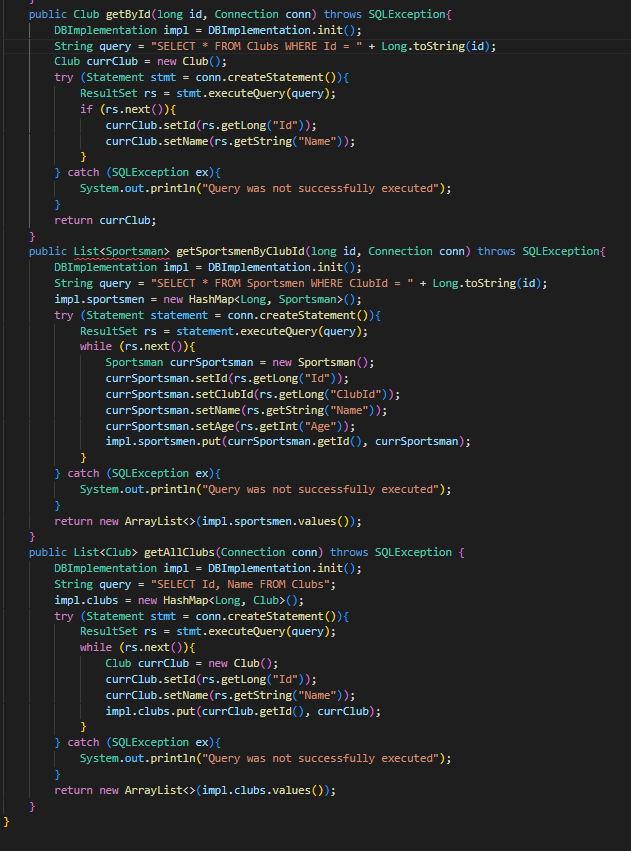
Класс DAO для модели Sportsman.java под названием SportsmanDAO.java:





Класс DAO для модели Club.java под названием ClubDAO.java:





Класс DAO, представлющий реализацию DB, под названием DBImplementation.java:

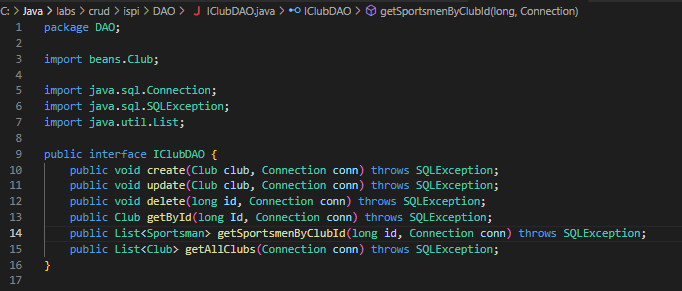


Создадим интерфейсные классы для существующих DAO – классов:

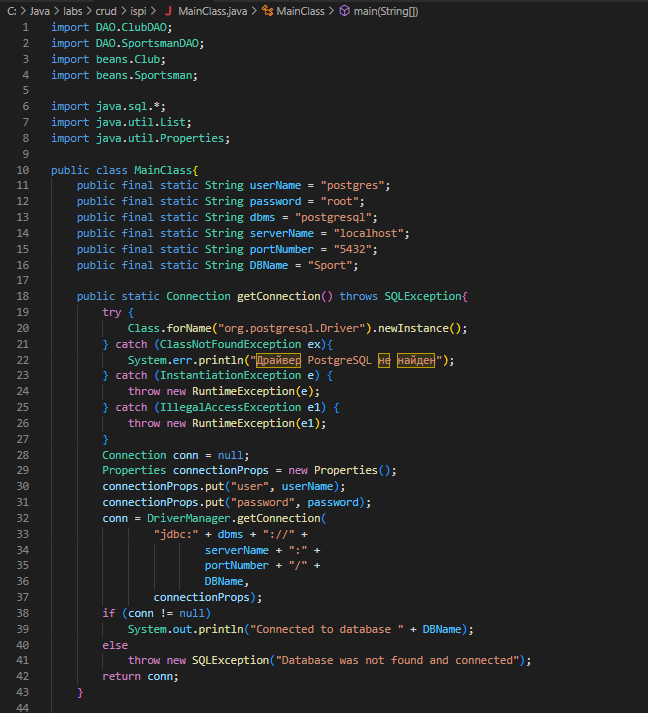
Интерфейс ISportsmanDAO.java для DAO – класса Sportsman.java:

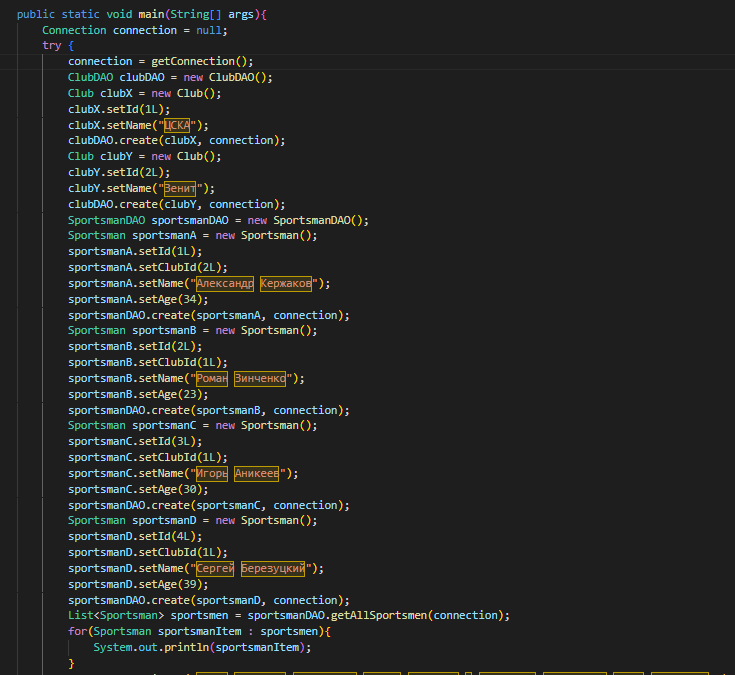


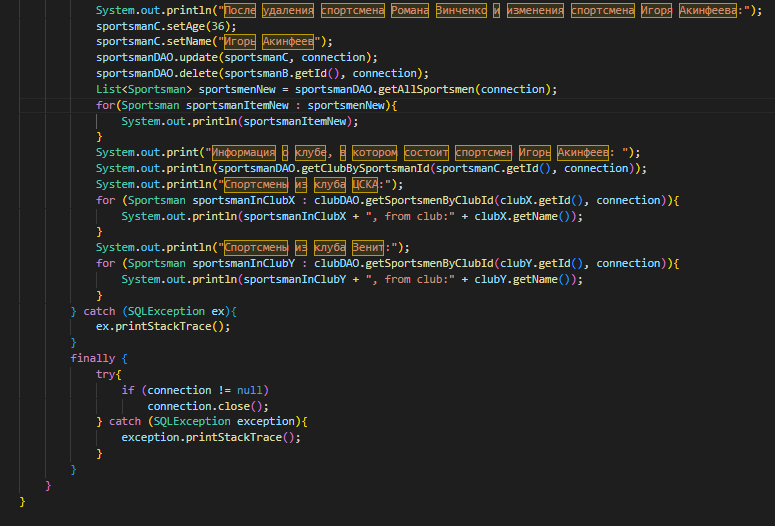
Интерфейс IClubDAO.java для DAO – класса Club.java:



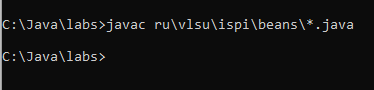
Создадим основной класс программы MainClass.java:

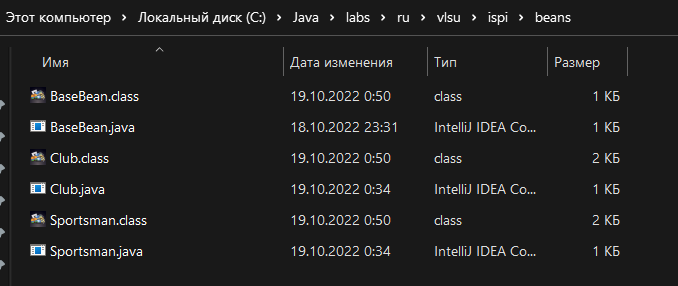




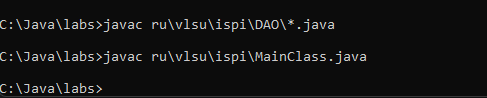


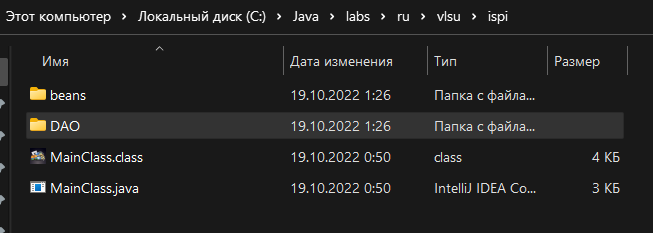
Сначала компилируем файлы-модели в папке \beans:

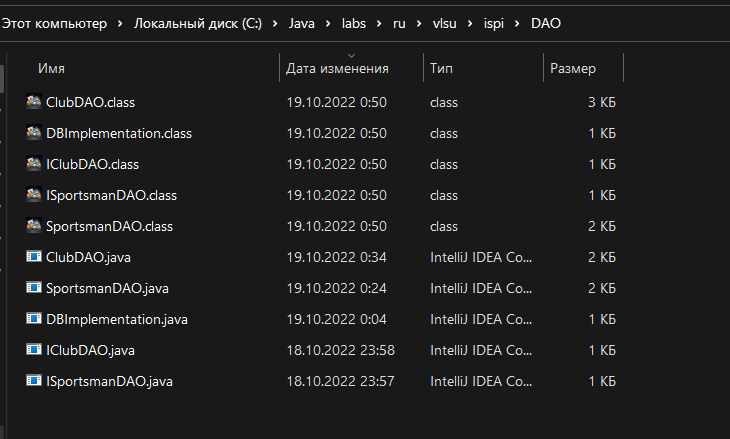




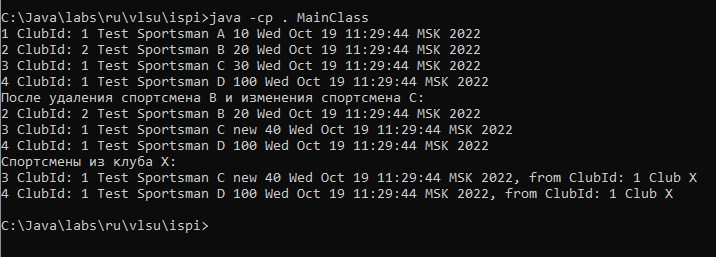
Аналогично компилируем файлы-DAO в папке \DAO, а затем компилируем основной класс MainClass.java и запускаем его:



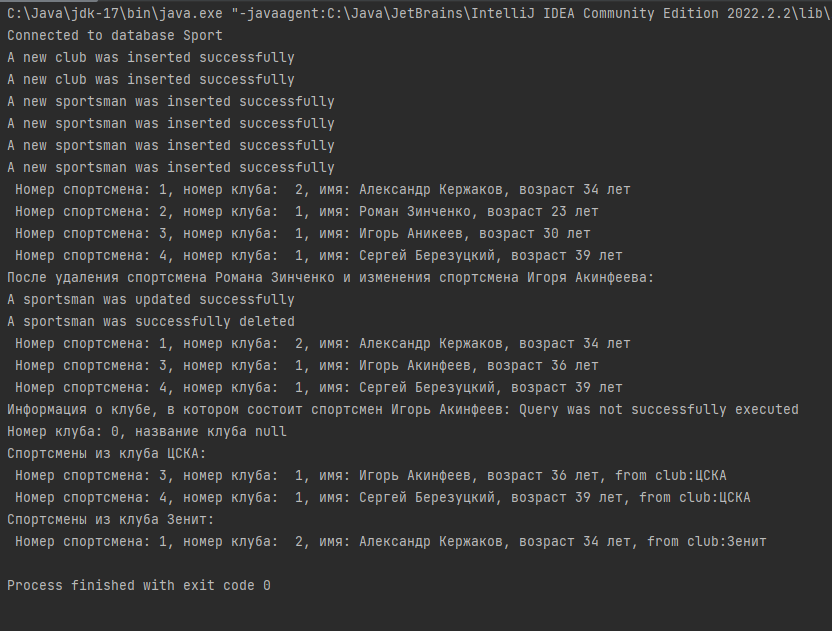




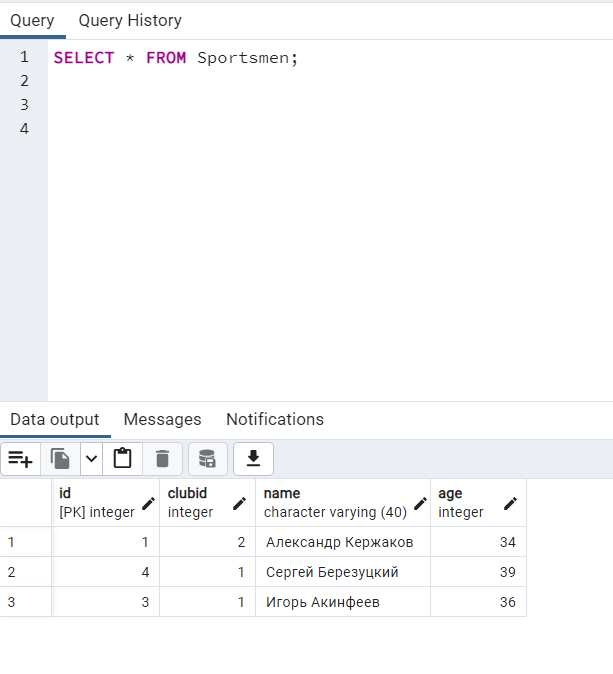
После запуска программы (класса MainClass.class) получаем следующий результат:

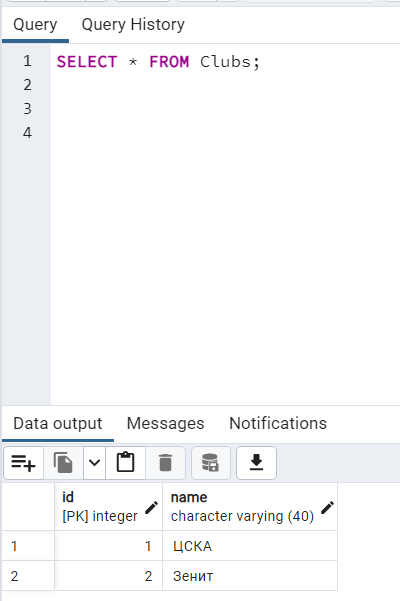


Запускаем новую версию основного класса программы MainClass.cs в IDE Intellij Idea и получаем следующий результат:



Изменения в таблицах БД Sport СУБД Postgre SQL:





Вывод

В результате выполнения работы я изучил средства JDBC API для работы с реляционными СУБД и получить навыки их практического использования.