**Домашнее задание №3**

1. Примените пожалуйста [скрипт](https://drive.google.com/file/d/15CZE97IH7N9mgAvNSseu_61vxMf-wGqD/view?usp=sharing) для базы данных PostgreSQL. Если есть необходимость, то вы можете применить его заново, предварительно удалив таблицы командой

DROP TABLE *имя\_таблицы;*

Применить скрипт - это значит выполнить его через ваш используемый IDE (DataGrip, DBeaver , pgAdmin)

1. Необходимо написать SQL запросы к следующим 5 задачам ниже. SQL запрос пишется в вашем IDE и прикладывается (как текст) вместе со скрином ответа в отдельном doc документе - который в свою очередь необходимо приложить к домашнему заданию и отправить преподавателю. Задание считается выполненным - если SQL запрос написан синтаксически корректно и возвращает ожидаемые данные на условие задачи.
   1. Напишите SQL запрос который возвращает имена студентов и название курсов университетов в одном списке. Результат отсортируйте в убывающем порядке. Пример части результата представлен ниже

|  |
| --- |
| **name** |
| Цифровая трансформация |
| Сергей Петров |
| … |

* 1. Напишите SQL запрос который возвращает имена университетов и название курсов в одном списке, но с типом что запись является или “университет” или “курс”. Результат отсортируйте в убывающем порядке по типу записи и потом по имени. Пример части результата представлен ниже

|  |  |
| --- | --- |
| **name** | **object\_type** |
| Иннополис | университет |
| КФУ | университет |
| … | … |
| Data Mining | курс |
| … | … |

* 1. Напишите SQL запрос который возвращает название курса и количество заявленных студентов в отсортированном списке по количеству слушателей в возрастающем порядке, **НО** запись с количеством слушателей равным 300 должна быть на первом месте. Ограничьте вывод данных до 3 строк. Пример результата представлен ниже

|  |  |
| --- | --- |
| **name** | **amount\_of\_students** |
| Введение в РСУБД | 300 |
| Data Mining | 10 |
| Актерское мастерство | 15 |

**Подсказка**: используйте в ORDER BY синтаксический

элемент CASE … END. Саму конструкцию можно посмотреть вот [здесь](https://postgrespro.ru/docs/postgrespro/14/functions-conditional)

* 1. Напишите DML запрос который создает новый **offline** курс со следующими характеристиками:
* id = 60
* название курса = Machine Learning
* количество студентов = 17
* курс проводится в том же университете что и курс Data Mining

Предоставьте INSERT выражение которое заполняет необходимую

таблицу данными (Не используйте вложенные подзапросы!).

Приложите скрин результата запроса к данным курсов после

выполнения команды INSERT к таблице которая была изменена.

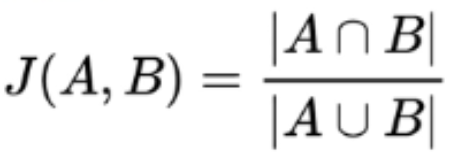
* 1. Напишите SQL скрипт который подсчитывает симметрическую разницу множеств A и B.

(A \ B) ⋃ (B \ A)

где A - таблица **course,** B - таблица **student\_on\_course,** “\” - это разница множеств, “⋃” - объединение множеств. Необходимо подсчитать на основании атрибута **id** из обеих таблиц. Результат отсортируйте по 1 столбцу. Пример результата представлен ниже.

|  |
| --- |
| **id** |
| 70 |
| 80 |
| 90 |
| 100 |
| … |

* 1. (\*\* - **задача опциональная**) Напишите SQL запрос который подсчитывает коэффициент Жаккара (близости множеств) между двумя таблицами A (**course**) и B (**student\_on\_course**) по атрибуту id на основании формулы:



где A - таблица **course,** B - таблица **student\_on\_course,** “⋃” - объединение множеств, “∩” - пересечение множеств, | A ∩ B | - мощность множества (количество записей) по пересечению, | A ⋃ B | - мощность множества (количество записей) по объединению

**Подсказка**: чтобы подсчитать количество элементов используйте

конструкцию SELECT count(\*) FROM …

Пример результата представлен ниже

|  |
| --- |
| **val** |
| 0.46153846153846153846 |