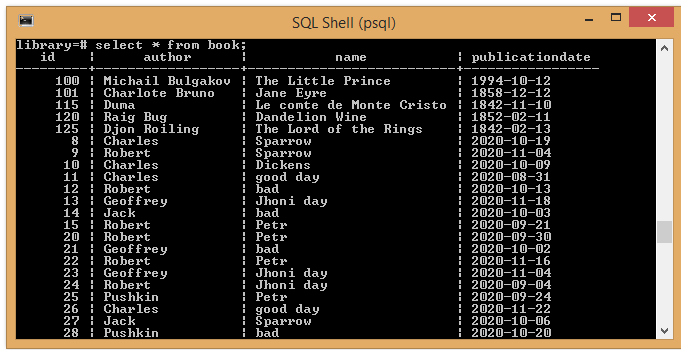
**Мащёнский Дмитрий**

**Задание 1. Для Вашей представленной базы данных сгенерировать тестовые данные**.



Выполним при помощи GENERATE\_SERIES

**Задание 2. Разработать запрос, который будет делать выборку по крайней мере из 3 связных таблиц, использовать фильтрацию, сортировку и вывод конкретного диапазона записей (WHERE, ORDER BY, LIMIT). Запрос должен выполняться несколько секунд (для этого можно увеличить количество тестовых данных или сложность запроса).**

SELECT

Firstname, Surname, phone,u.id

FROM

user\_Profile as ud

JOIN read\_card AS u ON u.user\_profile\_id = ud.id

LEFT JOIN book AS t ON u.book\_id = t.id

where t.name='Dickens'

GROUP BY Firstname, Surname, phone,u.id

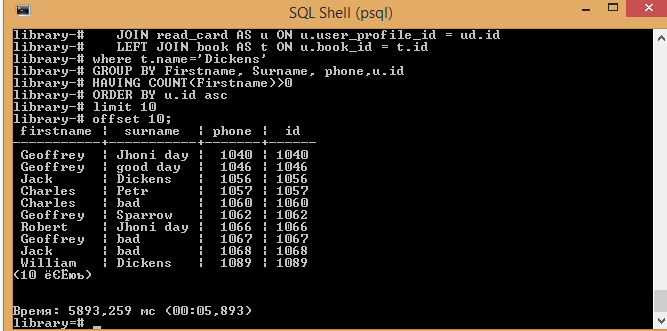
HAVING COUNT(t.name)>0

ORDER BY u.id asc

limit 10

offset 10;

Время выполнения запроса



Запрос выполнялся 5 секунд.

**Задание 3. Вывести план запроса и объяснить последовательность действий при его выполнении.**

SELECT

Firstname, Surname, phone,u.id

FROM

user\_Profile as ud

JOIN read\_card AS u ON u.user\_profile\_id = ud.id

LEFT JOIN book AS t ON u.book\_id = t.id

where t.name='Dickens'

GROUP BY Firstname, Surname, phone,u.id

HAVING COUNT(Firstname)>0

ORDER BY u.id asc

limit 10

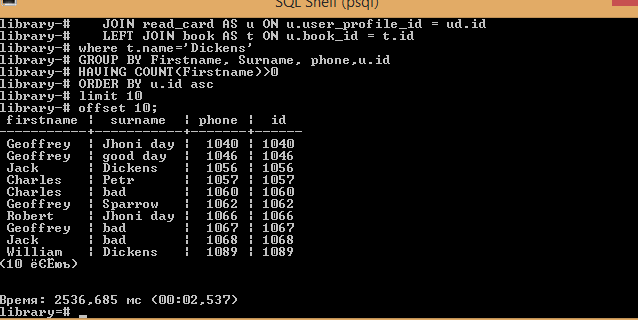
offset 10;

В самом начале мы объединяем таблицу read\_card и user\_Profile по их id, затем добавляем ещё одну таблицу book LEFT JOIN-ом. После того , как у нас создалась одна большая таблица мы начинаем сравнивать каждую книгу которые находятся в таблице с названием “Dickens”. Потом группируем все фамилии , имена , телефоны и id , у которых есть книга “Dickens”.После группировки мы оставляем только тех читателей у которых фамилия повторилась хотя бы раз. Затем происходит сортировка id по возрастанию полученных читателей .Поиск читателей начнётся с 10 позиции и прекратится при нахождении 10 читателей подходящих под наше условие.

Задание 4. Добавить необходимые индексы к Вашему запросу, чтобы скорость запроса была увеличена в несколько раз (возможно предложить его оптимизацию). Вывести план запроса и объяснить, за счет чего была получена прибавка в скорости выполнения запроса.

Добавим индекс в таблицу book

create INDEX idx\_name on book (name);



Заметим, что при добавлении индекса время выполнения индекса уменьшилось в 2 раза.

Это происходит из-за того, что у нас есть слова(название книги) и сначала поиск происходит по первой букве, потом будет искаться по второй букве среди тех отфильтрованных уже по первой букве, ну и так далее. То есть на каждом этапе мы отбрасываем не интересующие нам слова.