# Министерство науки и высшего образования Российской Федерации

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ «ЛЭТИ»  
ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

197376, Санкт-Петербург, ул. проф. Попова, 5.

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

«ЗАЧТЕНО»

\_\_\_\_\_\_\_\_\_\_ О.А. Жирнова

“\_\_” \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

ОТЧЁТ  
по дисциплине «Базы данных»

**Лабораторная работа № 3  
«Выполнение запросов по нескольким таблицам»**

|  |  |
| --- | --- |
| Студент группы 9308 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Н.В. Яловега |

Санкт Петербург 2021

*Цель работы:* научиться соединять данные из нескольких таблиц.

*Используемая база данных (БД):* Library.

**Порядок выполнения**

*Упражнение 1 –* **создание списка почтовой рассылки с использованием оператора “JOIN”**

Запрос для таблиц *member* и *adult*, для получения значения полей *firstname**,* *middleinitial**,* *lastname**,* *street**,* *city**,* *state**,* *zip*. Значения полей *firstname**,**middleinitial**,**lastname* были конкатенированы в один столбец с   
псевдонимом *name:*

SELECT

concat(member.firstname,' ', member.middleinitial,' ', member.lastname)

AS

name, adult.street, adult.city, adult.state, adult.zip

FROM

member JOIN adult ON member.member\_no = adult.member\_no

Результат выполнения запроса представлен на рисунке 1

Рисунок 1



*Упражнение 2 –* **создание списка почтовой рассылки с использованием оператора “JOIN”**

Запрос по таблицам *title, item* и *copy*, который возвращает поля *isbn*, *copy\_no, on\_loan, title, translation* и *cover*, а также строки из таблицы *copy*, где *isbn* равен 1, 500 или 1000. Полученный набор сортируется по полю *isbn*:

SELECT

copy.isbn AS 'Isbn', copy.copy\_no AS 'Copy', copy.on\_loan AS 'Loan', title.title AS 'Title', item.translation AS 'Trans', item.cover AS 'Cov'

FROM

title INNER JOIN (copy INNER JOIN item ON item.isbn = copy.isbn) ON title.title\_no = copy.title\_no

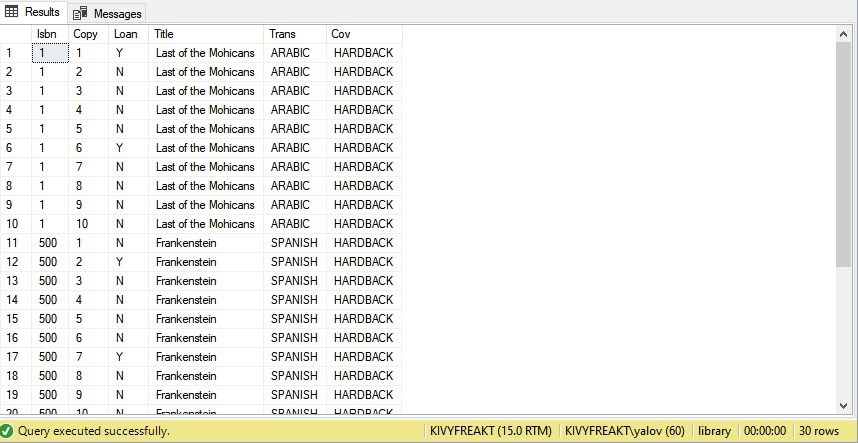
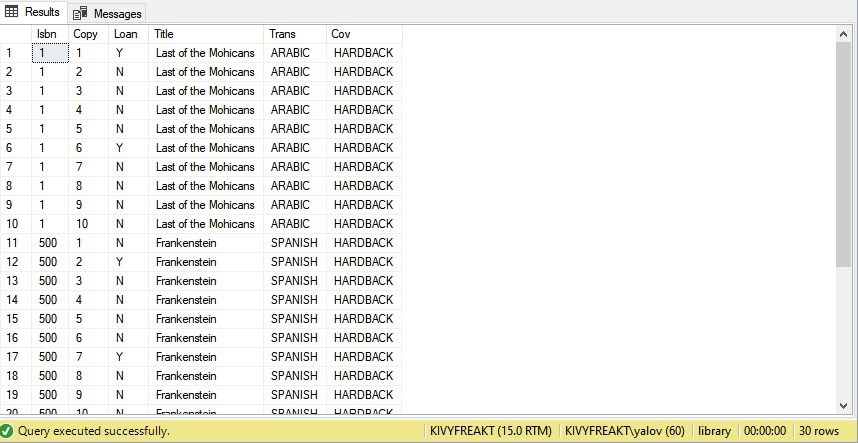
WHERE

item.isbn=1 or item.isbn=500 or item.isbn=1000

ORDER BY isbn ASC

Результат выполнения запроса представлен на рисунке 2

Рисунок 2



*Упражнение 3 –* **объединение нескольких таблиц и сортировка результатов**

Запрос, возвращающий полное имя читателя *member\_no* из таблицы *member, isbn* и *log\_date* из таблицы *reservation* для читателей с номерами *250*, *341* и *1675*. Результат сортируется по *member\_no*. Информация об этих читателях показывается вне зависимости от того, взяты ими книги или нет:

SELECT

member.member\_no, concat(member.lastname,' ', member.firstname,' ', member.middleinitial) AS 'name', reservation.isbn, CONVERT(char(8),reservation.log\_date) AS 'date'

FROM

member LEFT OUTER JOIN reservation ON member.member\_no = reservation.member\_no

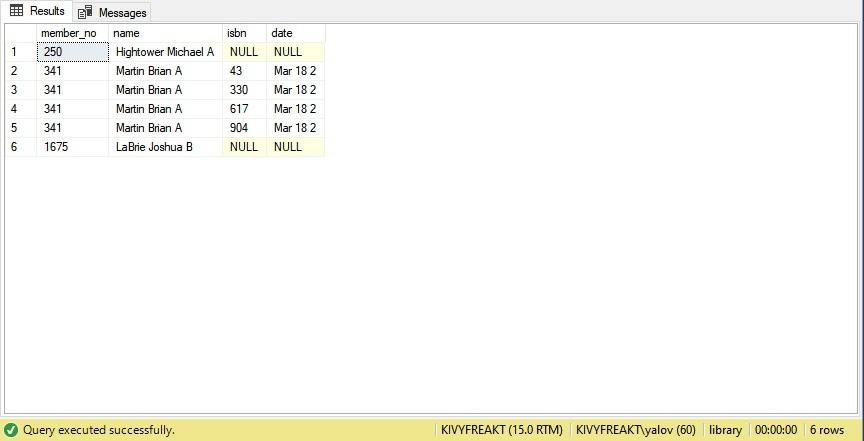
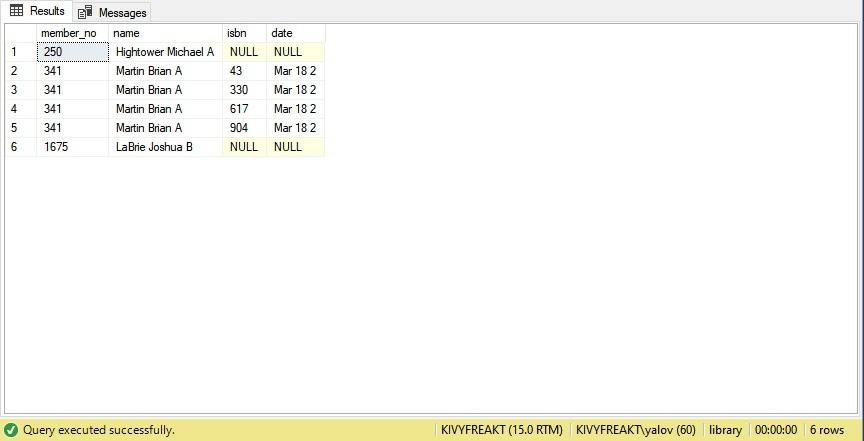
WHERE

member.member\_no =250 or member.member\_no = 341 or member.member\_no = 1675

ORDER BY member\_no ASC

Результат выполнения запроса представлен на рисунке 3

Рисунок 3



*Упражнение 4 –* **использование оператора UNION для соединения результирующих наборов**

Запрос 1. Требуется определить читателей, живущих в Аризоне, у которых более двух детей посещают библиотеку:

SELECT

member.member\_no, COUNT(juvenile.member\_no) AS Numkids

FROM

member INNER JOIN juvenile ON member.member\_no = juvenile.adult\_member\_no

INNER JOIN adult ON member.member\_no = adult.member\_no

WHERE

adult.state = 'AZ'

GROUP BY

member.member\_no

HAVING COUNT(juvenile.member\_no) > 2

Результат выполнения запроса представлен на рисунке 4

Рисунок 4



Запрос 2. Нужно определить читателей, живущих в Калифорнии, у которых более трёх детей ходят в библиотеку:

SELECT

member.member\_no, COUNT(juvenile.member\_no) AS Numkids

FROM

member INNER JOIN juvenile ON member.member\_no = juvenile.adult\_member\_no

INNER JOIN adult ON member.member\_no = adult.member\_no

WHERE

adult.state = 'CA'

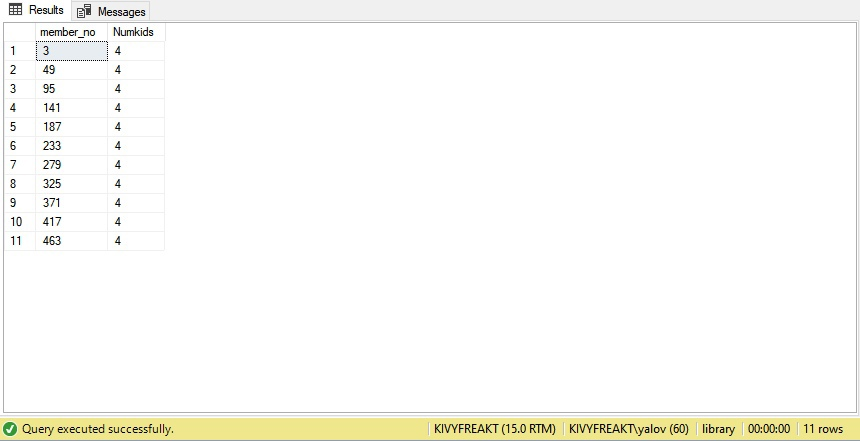
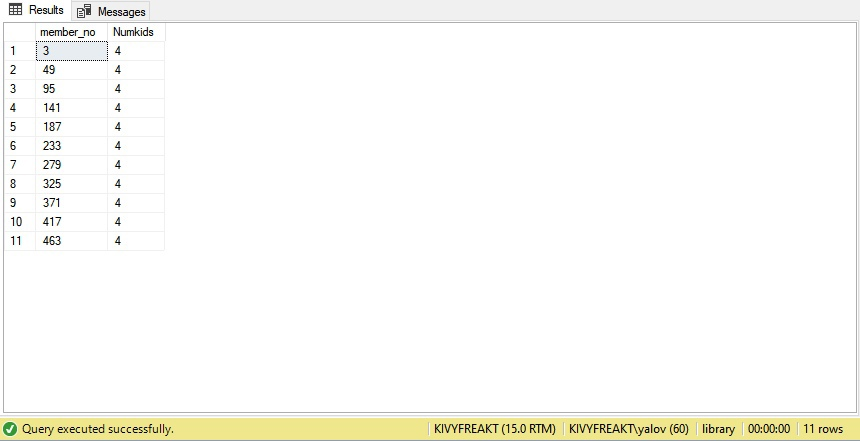
GROUP BY

member.member\_no

HAVING COUNT(juvenile.member\_no) > 3

Результат выполнения запроса представлен на рисунке 5

Рисунок 5



Запрос 3. Объединение двух предыдущих запросов:

SELECT

member.member\_no, COUNT(juvenile.member\_no) AS Numkids

FROM

member INNER JOIN juvenile ON member.member\_no = juvenile.adult\_member\_no

INNER JOIN adult ON member.member\_no = adult.member\_no

WHERE

adult.state = 'AZ'

GROUP BY

member.member\_no

HAVING

COUNT(juvenile.member\_no) > 2

UNION

SELECT

member.member\_no, COUNT(juvenile.member\_no) AS Numkids

FROM

member INNER JOIN juvenile ON member.member\_no = juvenile.adult\_member\_no

INNER JOIN adult ON member.member\_no = adult.member\_no

WHERE

adult.state = 'CA'

GROUP BY

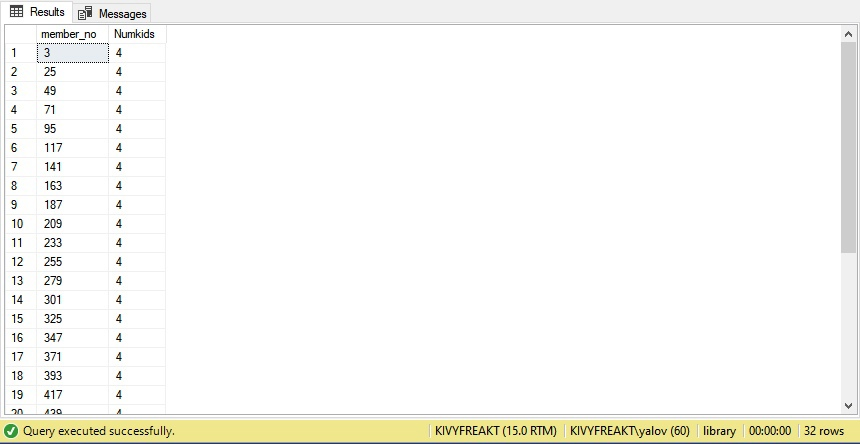
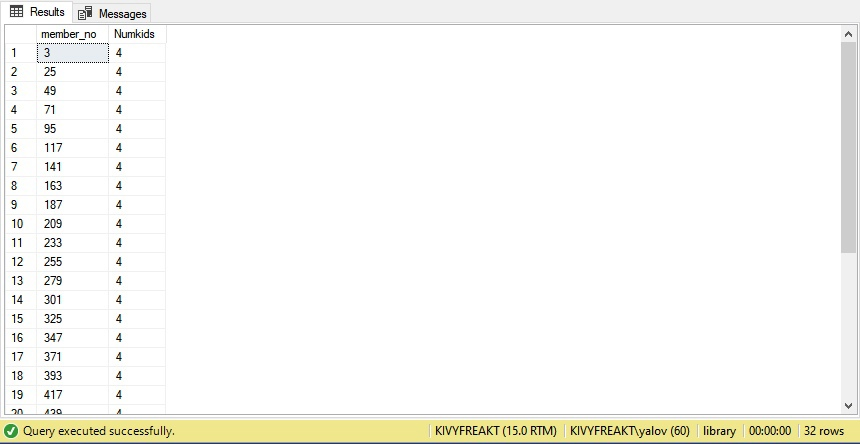
member.member\_no

HAVING

COUNT(juvenile.member\_no) > 3

Результат выполнения запроса представлен на рисунке 6

Рисунок 6



**Выводы**

Научились соединять данные из нескольких таблиц. Для этого более подробно изучены операторы JOIN и UNION, позволяющие объединять данные необходимым образом. Для объединения по определенному критерию данных из нескольких таблиц использовался оператор JOIN. Для объединения результирующих наборов нескольких запросов использовали оператор UNION.

**Список использованных источников**

1. Горячев А. В., Новакова Н. Е. Распределенные базы данных. Мет. указания к лаб. работам., СПб. Изд-во СПбГЭТУ «ЛЭТИ», 2008

2. Горячев А.В, Новакова Н.Е. Особенности разработки и администрирования приложений баз данных: учеб. пособие. СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2016. 68 с.

3. Дейт К. Введение в системы баз данных. : Пер. с англ. – 6-е изд. -К.:Диалектика, 1998.