

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Систем автоматизированного проектирования**

**ОТЧЕТ**  
**по лабораторной работе №3**  
**по дисциплине «Базы данных»**  
**Тема: Выполнение запросов по нескольким таблицам**

Студенты гр. 2308

\_\_\_\_\_

Рымарь М.И.

\_\_\_\_\_

Мелихов М.А.

\_\_\_\_\_

Придчин В.Е.

Преподаватель

\_\_\_\_\_

Горяинов С.В.

Санкт-Петербург

2024

## Цель работы

Научиться соединять данные из нескольких таблиц.

## Выполнение работы

**Упражнение 1 – создание списка почтовой рассылки с использованием оператора JOIN. Требуется создать список рассылки читателей библиотеки. Список должен включать полное имя и информацию о месте жительства читателя.**

Запрос:

```
SELECT  
  
    dbo.member.firstname + ' ' + dbo.member.middleinitial  
+ ' ' + dbo.member.lastname AS name,  
    dbo.adult.street, dbo.adult.city, dbo.adult.state,  
    dbo.adult.zip  
  
FROM dbo.member  
  
INNER JOIN dbo.adult ON dbo.member.member_no =  
dbo.adult.member_no
```

Результат выполнения запроса показан на рисунке 1.

name	street	city	state	zip
Amy A Anderson	Bowery Estates	Montgomery	AL	36100
Brian A Anderson	Dogwood Drive	Sacramento	CA	94203
Daniel A Anderson	Fir Street	Washington	DC	20510-0001
Eva A Anderson	The Highlands	Atlanta	GA	30026
Gary A Anderson	James Road	Springfield	IL	62700
Joshua A Anderson	Larch Mont Road	Boston	MA	02138-7190

(затронута строк: 5000)

Время выполнения: 2024-10-20T18:08:23.5115995+03:00

Рисунок 1 – Запрос с INNER JOIN без условий

**Упражнение 2 – объединение нескольких таблиц и сортировка результатов. Нужно выполнить запрос по таблицам title, item и copy,**

который возвращал бы 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 должен быть отсортирован по полю isbn.

Запрос:

```
SELECT cp.isbn, cp.copy_no, cp.on_loan, tit.title,
it.translation, it.cover
FROM dbo.title AS tit
INNER JOIN dbo.copy AS cp
ON tit.title_no = cp.title_no
INNER JOIN dbo.item AS it
ON cp.isbn = it.isbn
WHERE cp.isbn = 1 OR cp.isbn = 500 OR cp.isbn = 1000
ORDER BY cp.isbn
```

Результат выполнения запроса представлен на рисунке 2.

isbn	copy_no	on_loan	title	translation	cover
1	1	Y	Last of the Mohicans	ARABIC	HARDBACK
1	2	N	Last of the Mohicans	ARABIC	HARDBACK
1	3	N	Last of the Mohicans	ARABIC	HARDBACK
1	4	N	Last of the Mohicans	ARABIC	HARDBACK
1	5	N	Last of the Mohicans	ARABIC	HARDBACK
1	6	Y	Last of the Mohicans	ARABIC	HARDBACK
1	7	N	Last of the Mohicans	ARABIC	HARDBACK
1	8	N	Last of the Mohicans	ARABIC	HARDBACK
1	9	N	Last of the Mohicans	ARABIC	HARDBACK
1	10	N	Last of the Mohicans	ARABIC	HARDBACK
500	1	N	Frankenstein	SPANISH	HARDBACK
500	2	Y	Frankenstein	SPANISH	HARDBACK
500	3	N	Frankenstein	SPANISH	HARDBACK
500	4	N	Frankenstein	SPANISH	HARDBACK
500	5	N	Frankenstein	SPANISH	HARDBACK
500	6	N	Frankenstein	SPANISH	HARDBACK
500	7	Y	Frankenstein	SPANISH	HARDBACK
500	8	N	Frankenstein	SPANISH	HARDBACK
500	9	N	Frankenstein	SPANISH	HARDBACK
500	10	N	Frankenstein	SPANISH	HARDBACK
1000	1	N	Frankenstein	SPANISH	SOFTBACK
1000	2	Y	Frankenstein	SPANISH	SOFTBACK
1000	3	N	Frankenstein	SPANISH	SOFTBACK

(затронута строк: 30)

Время выполнения: 2024-10-20T18:09:49.8616585+03:00

Рисунок 2 – Результат выполнения запроса с двумя INNER JOIN и условием

**Упражнение 3 – объединение таблиц с использованием OUTER JOIN.**  
Необходимо выполнить 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 отсортировать по member\_no. Показать информацию об этих читателях вне зависимости от того, взяты ими книги или нет.

Запрос:

```
SELECT m.lastname + ' ' + m.firstname + ' ' +  
m.middleinitial AS name,  
r.isbn,  
CONVERT(CHAR(8), r.log_date, 112) AS date  
FROM dbo.member AS m  
LEFT OUTER JOIN dbo.reservation AS r ON m.member_no =  
r.member_no  
WHERE m.member_no IN (250, 341, 1675)  
ORDER BY m.member_no
```

Пояснение к CONVERT – третий аргумент в функции не обязателен обычно, но в данном случае он является обязательным. Третий аргумент – это стиль форматирования, в данном случае стиль применяется для даты, представляет её в формате YYYYMMDD из YYYY-MM-DD.

Результат выполнения запроса представлен на рисунке 3.

name	isbn	date
Hightower Michael A	NULL	NULL
Martin Brian A	43	20060318
Martin Brian A	330	20060318
Martin Brian A	617	20060318
Martin Brian A	904	20060318
LaBrie Joshua B	NULL	NULL

(затронуто строк: 6)

Время выполнения: 2024-10-20T18:11:10.2639291+03:00

Рисунок 3 – Результат выполнения запроса с OUTER JOIN и условием

#### Упражнение 4 – использование оператора UNION для соединения результирующих наборов.

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

```
SELECT a.member_no, COUNT(j.adult_member_no) AS
numkids
FROM dbo.adult AS a
INNER JOIN dbo.juvenile AS j ON a.member_no =
j.adult_member_no
WHERE a.state = 'AZ'
GROUP BY a.member_no
HAVING COUNT(j.adult_member_no) > 2
```

Результат выполнения запроса представлен на рисунке 4.

member_no	numkids
25	4
71	4
117	4
163	4
209	4
255	4
301	4
347	4
393	4
439	4
485	4
531	3
577	3
623	3

(затронуто строк: 21)

Время выполнения: 2024-10-20T18:12:00.1328867+03:00

Рисунок 4 – Результат выполнения запроса

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

```
SELECT a.member_no, COUNT(j.adult_member_no) AS
numkids
FROM dbo.adult AS a
INNER JOIN dbo.juvenile AS j ON a.member_no =
j.adult_member_no
WHERE a.state = 'CA'
GROUP BY a.member_no
HAVING COUNT(j.adult_member_no) > 3
```

Результат выполнения запроса представлен на рисунке 5.

member_no	numkids
3	4
49	4
95	4
141	4
187	4
233	4
279	4
325	4
371	4
417	4
463	4

(затронуто строк: 11)

Время выполнения: 2024-10-20T18:12:44.7364763+03:00

Рисунок 5 – Результат выполнения запроса

Запрос 3: объединить два предыдущих запроса с помощью оператора UNION.

```

SELECT a.member_no, COUNT(j.adult_member_no) AS
numkids, a.state
FROM dbo.adult AS a
INNER JOIN dbo.juvenile AS j
ON a.member_no = j.adult_member_no
WHERE a.state = 'AZ'
GROUP BY a.member_no, a.state
HAVING COUNT(j.adult_member_no) > 2
UNION
SELECT a.member_no, COUNT(j.adult_member_no) AS
numkids, a.state
FROM dbo.adult AS a
INNER JOIN dbo.juvenile AS j ON a.member_no =
j.adult_member_no
WHERE a.state = 'CA'
GROUP BY a.member_no, a.state

```

```
HAVING COUNT(j.adult_member_no) > 3
```

Дополнительно также был выведен штат, чтобы удостовериться, что запрос отработал корректно. Результат выполнения представлен на рисунке 6.

member_no	numkids	state
95	4	CA
117	4	AZ
141	4	CA
163	4	AZ
187	4	CA
209	4	AZ
233	4	CA
255	4	AZ
279	4	CA
301	4	AZ
325	4	CA
347	4	AZ
371	4	CA
393	4	AZ
417	4	CA
439	4	AZ
463	4	CA
485	4	AZ
531	3	AZ
577	3	AZ

(затронута строк: 32)

Время выполнения: 2024-10-20T18:13:34.0230755+03:00

Рисунок 6 – Результат выполнения запроса с UNION

## Выводы

В ходе выполнения лабораторной работы были получены практические навыки работы с операторами соединения данных в SQL, такими как INNER JOIN, LEFT OUTER JOIN и UNION. Изучение этих операторов позволяет эффективно извлекать информацию из разных таблиц и объединять результаты запросов.

Особое внимание было уделено LEFT OUTER JOIN, который позволяет получить все записи из левой таблицы, даже если нет соответствующих записей в правой. Также была изучена функция CONVERT для преобразования типов данных.



Полученные навыки позволят создавать более сложные запросы для решения задач в различных сферах деятельности.