

Лабораторный практикум

МДК 11.01 Технология разработки
и защиты баз данных

SQL

Куропаткина О.П.

ЛАБОРАТОРНАЯ РАБОТА №7

SQL

Structured Query Language

(язык структурированных запросов)

Многотабличные запросы с использованием JOIN

SQL Server выполняет операции сортировки, пересечения, объединения и поиска различий при помощи технологий хэш-соединений и сортировки в оперативной памяти.

С помощью соединения можно получать данные из двух или нескольких таблиц на основе логических связей между ними. Соединения позволяют указать, как в SQL Server должны использоваться данные из одной таблицы для выбора строк из другой таблицы.

Соединение определяет способ связывания двух таблиц в запросе следующим образом:

- для каждой таблицы указываются столбцы, используемые в соединении. В типичном условии соединения указывается внешний ключ из одной таблицы и связанный с ним ключ из другой таблицы;
- указывается логический оператор (например, = или \lt), для сравнения значений столбцов.

SQL Server реализует операции логического объединения в соответствии с синтаксисом Transact-SQL:

- внутреннее соединение,
- левое внешнее соединение,
- правое внешнее соединение,
- полное внешнее соединение,
- перекрестное соединение.

INNER JOIN (внутреннее объединение) - объединение, при котором в запросе все записи из таблицы на левой и правой стороне операции INNER JOIN добавляются в результирующий набор записей, при соответствии условию значений в связанных полях.

Для примера создадим две таблицы. Заполним тестовыми данными. Обратите внимание, что в таблице JoinTest1 отсутствуют строки с номерами 6, 7, 8. В JoinTest2 – 9, 10.

33	select * from JoinTest1;	33	select * from JoinTest1;
34	select * from JoinTest2;	34	select * from JoinTest2;

108 %	Результаты	Сообщения
	id_jt1	name
1	1	one
2	2	two
3	3	three
4	4	four
5	5	five
6	9	nine
7	10	ten

108 %	Результаты	Сообщения
	id_jt2	name
1	1	one
2	2	two
3	3	three
4	4	four
5	5	five
6	6	six
7	7	seven
8	8	eight

Осуществим внутреннее объединение и получим следующий результирующий набор данных

LEFT OUTER JOIN (левое внешнее объединение) - внешнее объединение, при котором в запросе все записи из таблицы на левой стороне операции LEFT JOIN в инструкции SQL добавляются в результирующий набор записей, даже если в таблице на правой стороне отсутствуют совпадающие значения в связанных полях.

```
69 SELECT * FROM JoinTest2 -- Левая таблица JoinTest2
70     LEFT OUTER JOIN JoinTest1 -- LEFT JOIN
71     ON id_jt1=id_jt2;
72 GO
73
```

108 %

Результаты Сообщения

	id_jt2	name	id_jt1	name
1	1	one	1	one
2	2	two	2	two
3	3	three	3	three
4	4	four	4	four
5	5	five	5	five
6	6	six	NULL	NULL
7	7	seven	NULL	NULL
8	8	eight	NULL	NULL

RIGHT OUTER JOIN (правое внешнее объединение) - внешнее объединение, при котором в запросе все записи из таблицы на правой стороне

операции RIGHT JOIN в инструкции SQL добавляются в результирующий набор записей, даже если в таблице на левой стороне отсутствуют совпадающие значения в связанных полях.

```
83 SELECT * FROM JoinTest2
84 RIGHT OUTER JOIN JoinTest1 -- Правая таблица JoinTest2
85 ON id_jt1 = id_jt2;
86 GO
```

108 %

Результаты Сообщения

	id_jt2	name	id_jt1	name
1	1	one	1	one
2	2	two	2	two
3	3	three	3	three
4	4	four	4	four
5	5	five	5	five
6	NULL	NULL	9	nine
7	NULL	NULL	10	ten

FULL OUTER JOIN (полное объединение) внешнее объединение, при котором в запросе все записи из таблицы на левой и правой стороне операции FULL JOIN добавляются в результирующий набор записей, при соответствии условию значений в связанных полях, а также:

- 1) значения из правой таблицы, не имеющие соответствий в левой таблице;
- 2) значения из левой таблицы, не имеющие соответствий в правой таблице.

```
100 SELECT *
101 FROM JoinTest2
102 FULL OUTER JOIN JoinTest1 --FULL JOIN
103 ON id_jt1 = id_jt2;
104 GO
```

108 %

Результаты Сообщения

	id_jt2	name	id_jt1	name
1	1	one	1	one
2	2	two	2	two
3	3	three	3	three
4	4	four	4	four
5	5	five	5	five
6	6	six	NULL	NULL
7	7	seven	NULL	NULL
8	8	eight	NULL	NULL
9	NULL	NULL	9	nine
10	NULL	NULL	10	ten

CROSS JOIN (перекрестное объединение) - выполняет декартово произведение таблиц, вовлеченных в объединение. В CROSS JOIN не используется конструкция ON.

```
117 SELECT * FROM JoinTest1
118 CROSS JOIN JoinTest2 -- CROSS JOIN
119 -- ON - не используется
120 GO
```

	id_jt1	name	id_jt2	name
1	1	one	1	one
2	1	one	2	two
3	1	one	3	three
4	1	one	4	four
5	1	one	5	five
6	1	one	6	six
7	1	one	7	seven
8	1	one	8	eight
9	2	two	1	one
10	2	two	2	two
11	2	two	3	three
12	2	two	4	four
13	2	two	5	five
14	2	two	6	six
15	2	two	7	seven
16	2	two	8	eight
17	3	three	1	one
18	3	three	2	two

Запрос успешно выполнен.

Рассмотрим пример реализации запроса:

Необходимо вывести список товаров (наименование, стоимость, наименование производителя), дату продажи и количество за второе полугодие 2019 года, отсортированный по дате продажи.

```
1 SELECT Product.Title, Product.Cost, ProductSale.Quantity, Manufacturer.Name, ProductSale.SaleDate
2 FROM Product INNER JOIN
3     Manufacturer ON Manufacturer.ID = Product.ManufacturerID INNER JOIN
4     ProductSale ON ProductSale.ProductID = Product.ID
5 WHERE SaleDate BETWEEN '01.07.2019' AND '31.12.2019'
```

	Title	Cost	Quantity	Name	SaleDate
1	English Grammar in Use. Book with Answers	1950.00	1	Cambridge	2019-10-13 18:32:00.000
2	Английская грамматика. Самое важное. Учебное посо...	80.00	1	Проспект	2019-08-10 12:07:00.000
3	Английский для малышей и мам @my_english_baby. К...	390.00	2	АСТ	2019-07-13 14:32:00.000

ЗАДАНИЕ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ:

1. Выведите список всех клиентов с указанием ФИО (в один столбец, разделенные пробелами) и контактных данных, а также количество оказанных

ему услуг и дату последней услуги. Обратите внимание, что в списке должны быть выведены все клиенты, даже те, которым услуги не были оказаны ни разу. Отсортируйте по количеству предоставленных услуг.

2. Выведите сведения о востребованности товаров. А именно: наименование, общая стоимость продаж товара, общее количество проданного товара, количество продаж товара, дата последней продажи. Отсортируйте – сначала самые востребованные товары.

3. Выведите сведения об общей выручке от оказанных услуг (с учетом скидки на услугу), количестве оказанных услуг и средней продолжительности услуги в минутах за первое и второе полугодие 2019 года.

4. Сформируйте статистику о самых востребованных услугах среди клиентов от 18 до 25 лет, и от 26 до 35 лет. Выведите наименование услуг, количество предоставленных услуг указанному диапазону клиентов.

5. Для формирования системы лояльности для клиентов, выведите сведения о клиентах, день рождения которых в текущем месяце. Список должен содержать ФИО, дату рождения, Email, количество совершенных заказов на предоставление услуги, общую стоимость предоставленных услуг.