

Лабораторная работа № 5.

Объединение таблиц

Цель работы

Целью выполнения данной лабораторной работы является изучение операторов для создания запросов, использующих объединение таблиц; получение навыков их использования.

Теоретические сведения

Для создания запросов, результатом которых должно быть объединение таблиц по некоторым критериям в SQL используется оператор JOIN.

JOIN — оператор языка SQL, позволяющий включить данные из нескольких таблиц. Входит в оператор FROM и отдельно от него не используется.

Синтаксис оператора SELECT с использованием конструкции JOIN.

```
SELECT FIELD [, ... n]
FROM MainTable
{INNER | {LEFT | RIGHT | FULL} OUTER | CROSS } JOIN
JoinTable ON <condition>
```

В большинстве СУБД при указании слов `left`, `right`, `full` слово `outer` можно опустить. Слово `INNER` также в большинстве СУБД можно опустить.

В общем случае СУБД проверяет условие `condition`, на соответствие в соединяемой таблице (`JoinTable`). Для `cross join` условие не указывается.

Для `cross join` в некоторых реализациях SQL используется такой вид оператора:

```
SELECT FIELD [, ... n]
FROM MainTable, JoinTable
```

ИЛИ

```
SELECT FIELD [, ... n]
FROM MainTable
FROM JoinTable
```

Оператор JOIN может использоваться с различными опциями: inner join (по умолчанию), outer join, left outer join, right outer join, full outer join, cross join.

Для пояснений будут использоваться следующие таблицы:

Люди, проживающие в городах (таблица Person)

Name	CityId
Андрей	1
Леонид	2
Сергей	1
Григорий	4

Города (таблица City)

Id	Name
1	Москва
2	Санкт-Петербург
3	Казань

INNER JOIN

Объединяет две таблицы, где каждая строка обеих таблиц в точности соответствует условию. Если для строки одной таблицы не найдено соответствия в другой таблице, строка не включается в набор.

```
SELECT *
```

```
FROM Person
```

```
INNER JOIN City ON Person.CityId = City.Id
```

Person.Name	Person.CityId	City.Id	City.Name
Андрей	1	1	Москва
Леонид	2	2	Санкт-Петербург
Сергей	1	1	Москва

OUTER JOIN

Присоединение таблицы с необязательным присутствием записи в таблице.

LEFT OUTER JOIN

К левой таблице присоединяются все записи из правой, соответствующие условию (по правилам `inner join`), плюс все не вошедшие записи из левой таблицы, поля правой таблицы заполняются значениями `NULL`.

```
SELECT *
FROM Person
LEFT OUTER JOIN City ON Person.CityId = City.Id
```

Person.Name	Person.CityId	City.Id	City.Name
Андрей	1	1	Москва
Леонид	2	2	Санкт-Петербург
Сергей	1	1	Москва
Григорий	4	NULL	NULL

RIGHT OUTER JOIN

Аналогично `left outer join`, но применяется для правой таблицы.

К левой таблице присоединяются все записи из правой соответствующие условию (по правилам `inner join`) плюс все не вошедшие записи из правой таблицы, поля левой таблицы заполняются значениями `NULL`.

```
SELECT *
FROM Person
```

RIGHT OUTER JOIN City ON Person.CityId = City.Id

Person.Name	Person.CityId	City.Id	City.Name
Андрей	1	1	Москва
Леонид	2	2	Санкт-Петербург
Сергей	1	1	Москва
NULL	NULL	3	Казань

FULL OUTER JOIN

К левой таблице присоединяются все записи из правой, соответствующие условию (по правилам `inner join`), плюс все не вошедшие записи из правой таблицы, поля левой таблицы заполняются значениями `NULL` и плюс все не вошедшие записи из левой таблицы, поля правой таблицы заполняются значениями `NULL`

SELECT *

FROM Person

FULL OUTER JOIN City ON Person.CityId = City.Id

Person.Name	Person.CityId	City.Id	City.Name
Андрей	1	1	Москва
Леонид	2	2	Санкт-Петербург
Сергей	1	1	Москва
Григорий	4	NULL	NULL
NULL	NULL	3	Казань

CROSS JOIN

Все возможные сочетания из обеих таблиц. Как уже говорилось выше, условие для этого типа оператора `JOIN` не указывается.

SELECT *

FROM Person

CROSS JOIN City

или

```
SELECT *
```

```
FROM Person, City
```

Набор очевиден и содержит для данного случая 12 записей.

Этот вид соединения нужно использовать с осторожностью, время выполнения растет экспоненциально с ростом количества записей и зачастую содержит избыточную информацию.

Ход выполнения работы

1. Изучить теоретические сведения.
2. Реализовать и протестировать запросы, возвращающие указанный преподавателем набор данных с использованием конструкции JOIN. При необходимости предусмотреть сортировку и группирование записей результата.
3. Составить отчет о выполнении лабораторной работы.
4. Подготовить ответы на контрольные вопросы.

Контрольные вопросы

1. Запишите синтаксис оператора SELECT с использованием конструкции JOIN.
2. Какие бывают виды объединения таблиц?
3. В чем различие между RIGHT JOIN и LEFT JOIN?
4. В чем различие между INNER JOIN и OUTER JOIN?
5. Для чего используется конструкция FULL JOIN?
6. Для чего используется конструкция CROSS JOIN?