

# БАЗЫ ДАННЫХ

---

Лекция 6  
Соединение таблиц

# Оператор соединения

- **JOIN** - позволяет извлекать данные более чем из одной таблицы

# Варианты синтаксиса соединения

- явный синтаксис соединения (ANSI SQL:1992)
- неявный синтаксис соединения (старого стиля)

# Пример

- **Employee**

- **empid**

- **name**

- **firstname**

- **dep\_no**

- **Department**

- **dep\_no**

- **department**

## Пример

<b>empid</b>	<b>firstname</b>	<b>name</b>	<b>dep_no</b>
18316	John	Barrimore	d1
29346	James	James	d2
25348	Matthew	Smith	d3
10102	Ann	Jones	d3
28559	Sybill	Moser	d1
9031	Elsa	Bertoni	d2

# Пример

<b>dep_no</b>	<b>department</b>
d1	Research
d2	Accounting
d3	Marketing

# ЯВНЫЙ СИНТАКСИС

**SELECT**

**name,**

**firstname,**

**department**

**FROM employee**

**INNER JOIN department ON**

**employee.dep\_no =**

**department.dep\_no;**

# ЯВНЫЙ СИНТАКСИС

- CROSS JOIN;
- [INNER] JOIN;
- LEFT [OUTER] JOIN;
- RIGHT [OUTER] JOIN;
- FULL [OUTER] JOIN.



# Неявный синтаксис

**SELECT**

**name,**

**firstname,**

**department**

**FROM employee, department**

**WHERE**

**employee.dep\_no =  
department.dep\_no;**

# Соединение таблиц

empid	firstname	name	dep_no	dep_no	department
18316	John	Barrimore	d1	d1	Research
18316	John	Barrimore	d1	d2	Accounting
18316	John	Barrimore	d1	d3	Marketing
29346	James	James	d2	d1	Research
29346	James	James	d2	d2	Accounting
29346	James	James	d2	d3	Marketing

# Соединение таблиц

empid	firstname	name	dep_no	dep_no	department
25348	Matthew	Smith	d3	d1	Research
25348	Matthew	Smith	d3	d2	Accounting
25348	Matthew	Smith	d3	d3	Marketing
10102	Ann	Jones	d3	d1	Research
10102	Ann	Jones	d3	d2	Accounting
10102	Ann	Jones	d3	d3	Marketing

# Соединение таблиц

empid	firstname	name	dep_no	dep_no	department
28559	Sybill	Moser	d1	d1	Research
28559	Sybill	Moser	d1	d2	Accounting
28559	Sybill	Moser	d1	d3	Marketing
9031	Elsa	Bertoni	d2	d1	Research
9031	Elsa	Bertoni	d2	d2	Accounting
9031	Elsa	Bertoni	d2	d1	Research

# Соединение таблиц

- Соединяется каждый с каждым
- Остаются только совпадающие
- Реализуются разными стратегиями
- Неявный синтаксис устарел

## Соединение более чем двух таблиц

- Ограничение - 64 таблицы
- `SELECT name, firstname`
- `FROM works_on`
- `JOIN employee ON  
works_on.empid=employee.empid`
- `JOIN department ON  
employee.dep_no=department.dep_no`

# Виды соединений

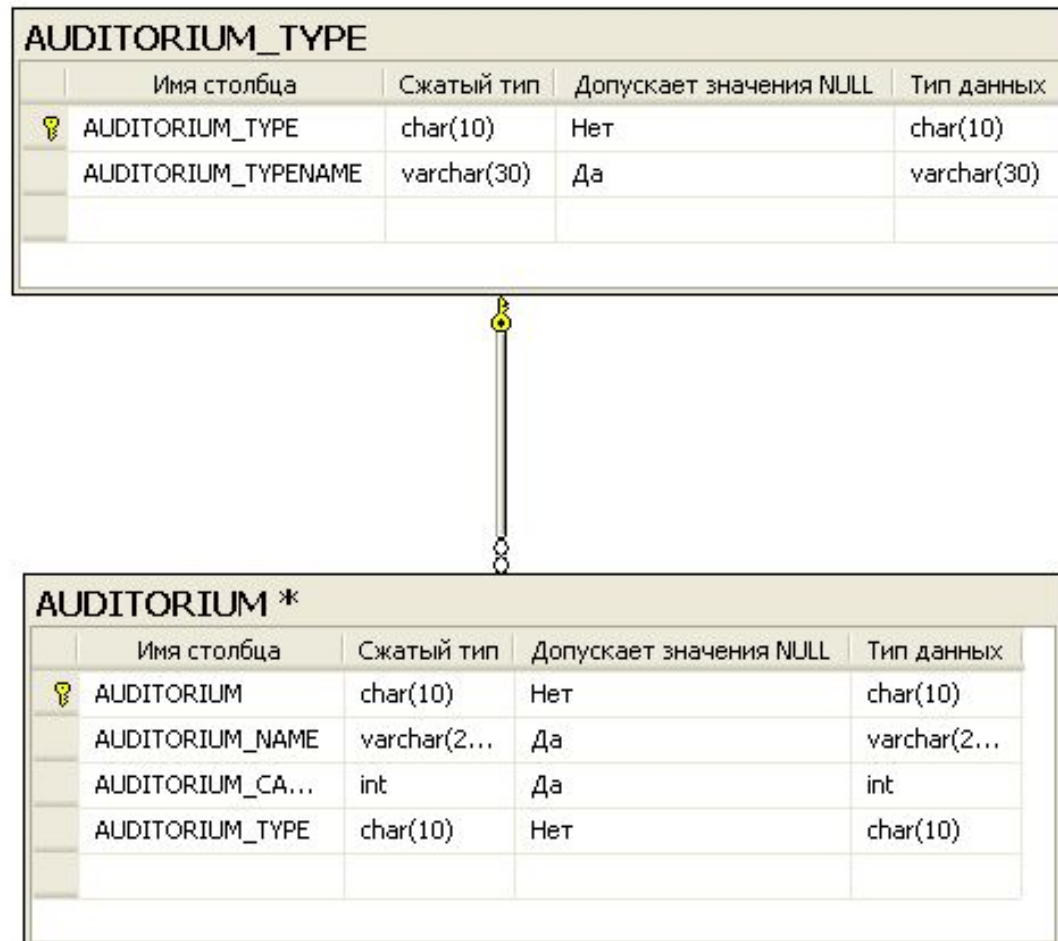
- Естественное соединение
- Декартово произведение (перекрестное соединение)
- Внешнее соединение
- Тета-соединение, самосоединение и полусоединение

# Естественное соединение

- Внутреннее соединение содержит только те строки одной таблицы, для которых имеются соответствующие строки в другой таблице
  - Соединяемые таблицы
  - Условие соединения
  - Столбцы соединения



# Естественное соединение



```

select aa.AUDITORIUM, aa.AUDITORIUM_CAPACITY, at.AUDITORIUM_TYPENAME
  from AUDITORIUM_TYPE at inner join AUDITORIUM aa
    on at.AUDITORIUM_TYPE = aa.AUDITORIUM_TYPE

```

AUDITORIUM	AUDITORIUM_CAPACITY	AUDITORIUM_TYPENAME
?	90	Лекционная
025-4	90	Лекционная
103-4	90	Лекционная
105-4	90	Лекционная
107-4	90	Лекционная
110-4	30	Лекционная
111-4	30	Лекционная
114-4	90	Лекционная с уст. компьютерами
132-4	90	Лекционная
206-1	15	Компьютерный класс
229-4	90	Лекционная
236-1	60	Лекционная
301-1	15	Компьютерный класс
304-4	90	Компьютерный класс
313-1	60	Лекционная
314-4	90	Лекционная
320-4	90	Лекционная
324-1	50	Лекционная
408-2	90	Лекционная
413-1	15	Компьютерный класс
423-1	90	Компьютерный класс
429-4	90	Лекционная

# Естественное соединение

```
select aa.AUDITORIUM, aa.AUDITORIUM_CAPACITY, at.AUDITORIUM_TYPENAME  
       from AUDITORIUM_TYPE at join AUDITORIUM aa  
       on at.AUDITORIUM_TYPE = aa.AUDITORIUM_TYPE
```

# Декартово произведение

- Ортогональное соединение, перекрестное соединение
- CROSS JOIN
- Декартово произведение соединяет каждую строку первой таблицы с каждой строкой второй таблицы
- Количество строк  $n \times m$

# Декартово произведение

```
select * from AUDITORIUM CROSS JOIN AUDITORIUM_TYPE;
```

AUDITORIUM	AUDITORIUM_NAME	AUDITORIUM_CAPACITY	AUDITORIUM_TYPE	AUDITORIUM_TYPE	AUDITORIUM_TYPENAME
?	???	90	ЛК	ЛБ-Х	Химическая лаборатория
025-4	025-4	90	ЛК	ЛБ-Х	Химическая лаборатория
103-4	103-4	90	ЛК	ЛБ-Х	Химическая лаборатория
105-4	105-4	90	ЛК	ЛБ-Х	Химическая лаборатория
107-4	107-4	90	ЛК	ЛБ-Х	Химическая лаборатория
110-4	110-4	30	ЛК	ЛБ-Х	Химическая лаборатория
111-4	111-4	30	ЛК	ЛБ-Х	Химическая лаборатория
114-4	114-4	90	ЛК-К	ЛБ-Х	Химическая лаборатория
132-4	132-4	90	ЛК	ЛБ-Х	Химическая лаборатория
229-4	229-4	90	ЛК	ЛБ-Х	Химическая лаборатория
236-1	236-1	60	ЛК	ЛБ-Х	Химическая лаборатория
301-1	301-1	15	ЛБ-К	ЛБ-Х	Химическая лаборатория
304-4	304-4	90	ЛБ-К	ЛБ-Х	Химическая лаборатория
313-1	313-1	60	ЛК	ЛБ-Х	Химическая лаборатория
314-4	314-4	90	ЛК	ЛБ-Х	Химическая лаборатория

# Декартово произведение

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM CROSS JOIN AUDITORIUM_TYPE) 'к-во строк в CROSS JOIN'
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM	к-во строк в CROSS JOIN
21	5	105

# Внешнее соединение

- При внешнем соединении результирующий набор содержит все строки одной таблицы и те из второй таблицы, для которых имеются соответствующие строки в первой таблице.

# LEFT OUTER JOIN

```
insert into AUDITORIUM_TYPE (AUDITORIUM_TYPE, AUDITORIUM_TYPENAME )  
values ('XX', 'XXXXX');
```

```
select * from AUDITORIUM_TYPE at LEFT OUTER JOIN AUDITORIUM aa  
on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE;
```

AUDITORIUM_TYPE	AUDITORIUM_TYPENAME	AUDITORIUM	AUDITORIUM_NAME	AUDITORIUM_CAPACITY	AUDITORIUM_TYPE
XX	XXXXX	NULL	NULL	NULL	NULL
ЛБ-Х	Химическая лаборатория	NULL	NULL	NULL	NULL
ЛБ-К	Компьютерный класс	301-1	301-1	15	ЛБ-К
ЛБ-К	Компьютерный класс	304-4	304-4	90	ЛБ-К
ЛБ-К	Компьютерный класс	413-1	413-1	15	ЛБ-К
ЛБ-К	Компьютерный класс	423-1	423-1	90	ЛБ-К
ЛБ-СК	Спец. компьютерный класс	NULL	NULL	NULL	NULL
ЛК	Лекционная	?	???	90	ЛК
ЛК	Лекционная	025-4	025-4	90	ЛК
ЛК	Лекционная	103-4	103-4	90	ЛК
ЛК	Лекционная	105-4	105-4	90	ЛК
ЛК	Лекционная	107-4	107-4	90	ЛК
ЛК	Лекционная	110-4	110-4	30	ЛК
ЛК	Лекционная	111-4	111-4	30	ЛК
ЛК	Лекционная	132-4	132-4	90	ЛК



# LEFT OUTER JOIN

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM_TYPE',  
       (select COUNT(*) from AUDITORIUM_TYPE at LEFT OUTER JOIN AUDITORIUM aa  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк в LEFT OUTER JOIN'
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM_TYPE	к-во строк в LEFT OUTER JOIN
21	6	<u>24</u>

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM_TYPE',  
       (select COUNT(*) from AUDITORIUM aa LEFT OUTER JOIN AUDITORIUM_TYPE at  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк в LEFT OUTER JOIN'
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM_TYPE	к-во строк в LEFT OUTER JOIN
21	6	<u>21</u>

# RIGHT OUTER JOIN

```
select * from AUDITORIUM_TYPE at RIGHT OUTER JOIN AUDITORIUM aa  
on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE;
```

AUDITORIUM_TYPE	AUDITORIUM_TYPENAME	AUDITORIUM	AUDITORIUM_NAME	AUDITORIUM_CAPACITY	AUDITORIUM_TYPE
ЛК	Лекционная	?	???	90	ЛК
ЛК	Лекционная	025-4	025-4	90	ЛК
ЛК	Лекционная	103-4	103-4	90	ЛК
ЛК	Лекционная	105-4	105-4	90	ЛК
ЛК	Лекционная	107-4	107-4	90	ЛК
ЛК	Лекционная	110-4	110-4	30	ЛК
ЛК	Лекционная	111-4	111-4	30	ЛК
ЛК-К	Лекционная с уст. комп...	114-4	114-4	90	ЛК-К
ЛК	Лекционная	132-4	132-4	90	ЛК
ЛК	Лекционная	229-4	229-4	90	ЛК
ЛК	Лекционная	236-1	236-1	60	ЛК
ЛБ-К	Компьютерный класс	301-1	301-1	15	ЛБ-К
ЛБ-К	Компьютерный класс	304-4	304-4	90	ЛБ-К
ЛК	Лекционная	313-1	313-1	60	ЛК
ЛК	Лекционная	314-1	314-1	90	ЛК

# RIGHT OUTER JOIN

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE at RIGHT OUTER JOIN AUDITORIUM aa  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM	к-во строк в RIGHT OUTER JOIN
<u>21</u>	6	<u>21</u>

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM aa RIGHT OUTER JOIN AUDITORIUM TYPE at  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM	к-во строк в RIGHT OUTER JOIN
<u>21</u>	6	<u>24</u>

# RIGHT OUTER JOIN

```
select * from AUDITORIUM aa RIGHT OUTER JOIN AUDITORIUM_TYPE at  
on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE  
order by AUDITORIUM;
```

AUDITORIUM	AUDITORIUM_NAME	AUDITORIUM_CAPACITY	AUDITORIUM_TYPE	AUDITORIUM_TYPE	AUDITORIUM_TYPENAME
NULL	NULL	NULL	NULL	ЛБ-СК	Спец. компьютерный класс
NULL	NULL	NULL	NULL	XX	XXXXX
NULL	NULL	NULL	NULL	ЛБ-Х	Химическая лаборатория
?	???	90	ЛК	ЛК	Лекционная
025-4	025-4	90	ЛК	ЛК	Лекционная
103-4	103-4	90	ЛК	ЛК	Лекционная
105-4	105-4	90	ЛК	ЛК	Лекционная
107-4	107-4	90	ЛК	ЛК	Лекционная
110-4	110-4	30	ЛК	ЛК	Лекционная
111-4	111-4	30	ЛК	ЛК	Лекционная
114-4	114-4	90	ЛК-К	ЛК-К	Лекционная с уст. компь...

# FULL OUTER JOIN

```
select * from AUDITORIUM_TYPE at FULL OUTER JOIN AUDITORIUM aa
on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE
order by aa.AUDITORIUM, at.AUDITORIUM_TYPE
```

AUDITORIUM_TYPE	AUDITORIUM_TYPENAME	AUDITORIUM	AUDITORIUM_NAME	AUDITORIUM_CAPACITY	AUDITORIUM_TYPE
XX	XXXXX	NULL	NULL	NULL	NULL
ЛБ-Х	Химическая лаборатория	NULL	NULL	NULL	NULL
ЛБ-СК	Спец. компьютерный класс	NULL	NULL	NULL	NULL
ЛК	Лекционная	?	???	90	ЛК
ЛК	Лекционная	025-4	025-4	90	ЛК
ЛК	Лекционная	103-4	103-4	90	ЛК
ЛК	Лекционная	105-4	105-4	90	ЛК
ЛК	Лекционная	107-4	107-4	90	ЛК
ЛК	Лекционная	110-4	110-4	30	ЛК
ЛК	Лекционная	111-4	111-4	30	ЛК
ЛК-К	Лекционная с уст. компьютерами	114-4	114-4	90	ЛК-К
ЛК	Лекционная	132-4	132-4	90	ЛК
ЛК	Лекционная	229-4	229-4	90	ЛК

# FULL OUTER JOIN

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM_TYPE',  
       (select COUNT(*) from AUDITORIUM_TYPE at FULL OUTER JOIN AUDITORIUM aa  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк в FULL OUTER JOIN'
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM_TYPE	к-во строк в FULL OUTER JOIN
21	6	24

```
select (select COUNT(*) from AUDITORIUM) 'к-во строк в AUDITORIUM',  
       (select COUNT(*) from AUDITORIUM_TYPE) 'к-во строк в AUDITORIUM_TYPE',  
       (select COUNT(*) from AUDITORIUM aa FULL OUTER JOIN AUDITORIUM_TYPE at  
        on aa.AUDITORIUM_TYPE = at.AUDITORIUM_TYPE) 'к-во строк в RIGHT OUTER JOIN'
```

к-во строк в AUDITORIUM	к-во строк в AUDITORIUM_TYPE	к-во строк в RIGHT OUTER JOIN
21	6	24

# Виды соединений

- Тета-соединение
- Самосоединение
- Полусоединение

# Тета-соединение

- Соединение, в котором используется общее условие сравнения столбцов соединения, называется тета-соединением
- `SELECT name, firstname, user_location, dept_location`
- `FROM employee JOIN department`
- `ON user_location < dept_location;`



# Самосоединение

- При самосоединении таблица соединяется сама с собой, столбец таблицы сравнивается сам с собой
- `SELECT t1.dep_no, t1.dep_name, t1.location`
- `FROM department t1 JOIN department t2`
- `ON t1.location = t2.location`
- `WHERE t1.dep_no <> t2.dep_no;`

# Самосоединение

- Иерархия



# Самосоединение

	id_employee	employee_name	department	head
1	1	John Dow	1	NULL
2	2	Jane Smith	1	1
3	3	Judith Perks	1	1

```
select e1.employee_name, e2.employee_name  
from Employee e1 inner join Employee e2  
on e1.head=e2.id_employee;
```

0 %

employee_name	employee_name
Jane Smith	John Dow
Judith Perks	John Dow

# Полусоединение

- Полусоединение возвращает набор всех строк из одной таблицы, для которой в другой таблице есть одно или несколько совпадений
- `SELECT empid, name, e.dep_no`
- `FROM employee e JOIN department d`
- `ON e.dep_no = d.dep_no`

# Операция соединения

- Коммутативность операций соединения?

Вопросы?