

## Лекция 7. Реляционные базы данных. SQL-запросы.

### Реляционные таблицы.

Рассмотрим конечное семейство доменов  $D_i = \{d_i\}$ .

Отношение  $R = R(D_1, \dots, D_n) \subset D_1 \times \dots \times D_n = \{(d_1, \dots, d_n)\}$ .  $D_i$  – домены отношения  $R$ .

Атрибуты отношения  $D_i^R = \{d_i \mid (d_1, \dots, d_i, \dots, d_n) \in R\}$ .

Реляционная таблица:

$D_1^R$	...	$D_n^R$
$d_{11}$	...	$d_{n1}$
$d_{12}$	...	$d_{n2}$
...	...	...
$d_{1k}$	...	$d_{nk}$
...	...	...

Характеристическая функция отношения  $R$ :

$F(x_1, \dots, x_n) = 1 \Leftrightarrow (d_1, \dots, d_n) \in R, x_1 = d_1, \dots, x_n = d_n; F(x_1, \dots, x_n) \in \{0, 1\}$ .

### Реляционная алгебра.

Реляционная алгебра  $RA(T, O)$ , где  $T$  – множество реляционных таблиц,  $O$  – множество операций над таблицами.

Теоретико-множественные операции: Union, Intersection, Difference, Symmetrical Difference.

Для операций 1-4 требуется совместимость по операции  $R, S \subset D_1 \times \dots \times D_n$ .

1. Объединение (Union).  
 $R \cup S$ .
2. Пересечение (Intersection).  
 $R \cap S$ .
3. Разность (Difference).  
 $R \setminus S$ .
4. Симметрическая разность (Symmetrical Difference).  
 $R \nabla S$ .

Реляционные операции: Selection, Product, Join, Projection.

5. Выборка (Selection).  
 $R' = \sigma_F(R); R' = \{(d_1, \dots, d_n) \in R \mid F(d_1, \dots, d_n), F - \text{высказывание}\}$
6. Произведение (Product).  
 $R \times S = \{(d_1, \dots, d_n, s_1, \dots, s_m) \mid (d_1, \dots, d_n) \in R, (s_1, \dots, s_m) \in S\}$
7. Проекция (Projection).

$$\downarrow \\ \frac{}{i_1, \dots, i_k} R = \{(d_{i_1}, \dots, d_{i_k}) \mid (d_1, \dots, d_n) \in R, \{i_1, \dots, i_k\} \subset \{1, \dots, n\}\}$$

одинаковые картежи удаляются из проекции.

8. Соединение (Join).

$R \parallel S = \{(c_1, \dots, c_{n1}, d_1, \dots, d_{n2}, e_1, \dots, e_{n3}) \mid (c_1, \dots, c_{n1}, d_1, \dots, d_{n2}) \in R, (d_1, \dots, d_{n2}, e_1, \dots, e_{n3}) \in S\}$ .

Пример соединения.

SN    PN    AC			ACC   SUM	
SP = -----			PAY = -----	
Джон	гайки	№11	#11	100
Джон	болты	№12	#14	50
Браун	клещи	№14		

  

SN    PN    ACC   SUM				
SP    PAY = -----				
Джон	гайки	№11	100	
Браун	клещи	№14	50	

9. Тета-соединение (Teta-Join)

$$R \bowtie_F S = \sigma_F(R \times S)$$

Этот набор не полон и избыточен.

### Базисный язык SQL (Structured Query language)

#### I. Подъязык запросов.

1. Выборка данных из таблицы: `select * from <имя таблицы>`
2. Выбор различных записей: `select distinct * from <имя таблицы>`
3. Аналог операции select реляционной алгебры:  
`select S.SN, S.City from S where City = "Париж" and ...`  
`select * from SN where JST > 50;`
4. Предикаты in и between:  
`select * from S where City in ('Осло', 'Париж')`  
`select * from S where Pay between 2,5 and 3,5`
5. Предикаты like, is NULL поиска подстрок в строке по шаблону,  
где символ \_ означает любой символ, а % – последовательность символов:  
`select * from S where SN like 'Mac%'`  
`select * from S where SN is not NULL`
6. Реализация операции соединения Join  
`select * from R, S where ...`

#### II. Подъязык манипуляции данными.

7. Оператор insert into:  
ввести строку в таблицу `insert into S values ('s1', 'Смит', NULL, 12)`  
`insert into SW select * from S where City = 'Париж'`  
таблицы SW и S должны быть совместимы по атрибутам;
8. Оператор delete:  
`delete from S where City = 'Париж'`  
`delete from S (все строки)`
9. Оператор update:  
`update S set JST = 50`  
`update S set JST = 50 where S# = 's1'`  
`update S set Pay = Pay*2`

#### III. Подъязык определения данных.

10. Оператор create table  
`create table S (S# integer, SN char(10), City char(10))` – создание пустой таблицы.  
Заполняется таблица операторами манипулирования данными.

11. Оператор create index JS# on J(S#)
12. Оператор alter table (изменить таблицу)  
alter table S delete SPD  
alter table S add PD decimal

#### IV. Операции реляционной алгебры.

Операции реляционной алгебры – проекция, селекция и соединение выражаются одной из форм оператора select

13. Проекция: select SN, Pay from S
14. Селекция: select SN, City from S where F (выражение)
15. Соединение, тета-соединение:  
select J.JN, S.SN, S.City from S, J where F (выражение)

Операторы:

16. Оператор union:  
<запрос> union [all] <запрос>  
select S#, SN from S where ... union select J#, JN from J where ... ;  
типы и размеры данных должны совпадать
17. Оператор intersect (пересечения):  
select S#, SN from S where ... intersect select J#, JN from J where ... ;  
типы и размеры данных должны совпадать.