

Лекция 13. 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 .

Реляционная таблица:

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

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

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 Т – множество реляционных таблиц, О – множество операций над таблицами.

Теоретико-множественные операции: Union, Intersection, Difference, Symmetrical Difference. для операций 1-4 требуется совместимость по операции.

1. Объединение (Union).
 $R \cup S, R, S \subset D_1 \times \dots \times D_n$.
2. Пересечение (Intersection).
 $R \cap S, R, S \subset D_1 \times \dots \times D_n$.
3. Разность (Difference).
 $R \setminus S, R, S \subset D_1 \times \dots \times D_n$.
4. Симметрическая разность (Symmetrical Difference).
 $R \Delta S, R, S \subset D_1 \times \dots \times D_n$.

Реляционные операции: Selection, Join, Projection, Division.

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

$$\downarrow \\ \underbrace{R}_{i_1, \dots, i_k} = \{(d_{i_1}, \dots, d_{i_k}) \mid (x_1, \dots, x_n) \in R, x_{i_1} = d_{i_1}, \dots, x_{i_k} = d_{i_k}\}$$

одинаковые картежи удаляются из проекции.

8. Соединение (Join).

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

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 SN, City from S where City = “Париж” and ...
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 ...
15. Соединение: select J.JN, S.SN, S.City from S, J where S.City = J.City
- Операторы:
16. Оператор union:
<запрос> union [all] <запрос>
select S#, SN from S where ... union select J#, JN from J where ... ;
домены должны совпадать
17. Оператор intersect (пересечения):
select S#, SN from S where ... intersect select J#, JN from J where ... ;
домены должны совпадать