

Лекция 11. Манипулирование реляционными данными.

11.1 Операции обновления реляционных данных	1
11.2 Реляционная алгебра.....	3
11.3. Теоретико-множественные операции	3
11.4 Специальные реляционные операции	6
11.5 Определение реляционной алгебры	11

Получив представление о том, как создать набор нужных реляционных отношений, приступим к изучению способов получения информации из данных. Т.к. реляционная модель является формальной системой, то для манипулирования данными применяется набор операций, имеющих четкое формальное определение. Этот набор операций вместе со множеством всех возможных реляционных отношений называется реляционной алгеброй.

Алгебра – это некоторое множество M вместе с заданной на нем совокупностью операций $\Omega = \{\varphi_1, \dots, \varphi_m, \dots\}$.

$\varphi_i : M^n \rightarrow M$ - n -арная операция на множестве M .

Множество M называется **носителем** алгебры.

Особое место среди операций манипулирования данными занимают операции обновления кортежей. В отличие от всех рассматриваемых ниже реляционных операций, действующих на уровне переменных отношений в целом, операции обновления имеют дело с множествами переменных кортежей, т.е. формально не относятся к реляционной алгебре.

11.1 Операции обновления реляционных данных

Обновление кортежей включает в себя добавление, изменение и удаление кортежей.

11.1.1 Добавление кортежей

Эта операция имеет вид

ADD (target; A_1=a_1, A_2=a_2, ..., A_n=a_n),

где target (A_1, A_2, ..., A_n) – реляционное отношение, $a_i \in \text{dom}(A_i)$, $i = 1, \dots, n$, n – степень отношения.

Результат операции негативный, если:

1. Добавляемый кортеж не соответствует схеме target.
2. Значения кортежа не принадлежат соответствующим доменам.
3. Кортеж совпадает по ключу с кортежем, уже существующем в target.

На языке SQL:

```
INSERT INTO имя_отношения
[VALUES (список значений атрибутов через запятую, согласно порядку их следования в
схеме) ] | [SELECT...FROM...WHERE... ] ;
```

Пример 1.

Ввод нового кортежа в отношение СОТРУДНИКИ

```
INSERT INTO СОТРУДНИКИ
VALUES (5053, "Ващенко", "психолог", 1500);
```

Пример 2.

Ввод кортежа с неполными данными

```
INSERT INTO ДЕТИ_СОТРУДНИКИ
VALUES (5053, "Алена", NULL);
```

NULL-значение допускается лишь в том случае, когда при описании отношения данному полю было разрешено иметь NULL-значения.

11.1.2 Удаление кортежей

Эта операция имеет вид

```
DEL (target; A_1=a_1, A_2=a_2, ...,A_n=a_n)
```

Или, если $K=\{B_1, B_2, \dots, B_r\}$ – ключ отношения target ($r < n$), то

```
DEL (target; B_1=b_1, B_2=b_2, ...,B_r=b_r)
```

На SQL:

```
DELETE FROM имя_отношения
[WHERE логическое условие отбора записей];
```

Пример 3.

Удаление всех значений из таблицы

```
DELETE FROM СОТРУДНИКИ ;
```

Пример 4.

Удалить информацию о конкретном сотруднике (указываем ключ записи)

```
DELETE FROM СОТРУДНИКИ
WHERE таб# = 5051 ;
```

Пример 5.

Удалить несколько кортежей

```
DELETE FROM СОТРУДНИКИ
WHERE зарплата < 500;
```

11.1.3 Обновление кортежей

Эта операция имеет вид

```
CH (target; A_1=a_1, A_2=a_2, ...,A_k=a_k; C_1=c_1, ...,C_m=c_m)
```

где $\{C_1, \dots, C_m\} \subseteq \{A_1, \dots, A_n\}$, $n=k+m$, n – степень отношения target

Или, если $K=\{B_1, B_2, \dots, B_r\}$ – ключ отношения target ($r < n$), то

```
CH (target; B_1=b_1, B_2=b_2, ...,B_r=b_r; C_1=c_1, ...,C_m=c_m)
```

На SQL:

```
UPDATE имя_отношения
SET имя_столбца = новое значение
[WHERE логическое условие отбора записей для обновления]
```

Пример 6.

Обновить все значения определенного столбца

```
UPDATE ЛУЧШИЕ_СОТРУДНИКИ
SET зарплата=зарплата + 200;
```

Пример 7.

Обновить значения определенного столбца, удовлетворяющие условию

```
UPDATE СОТРУДНИКИ
SET зарплата=зарплата + 200
WHERE должность = "психолог"
AND зарплата=600;
```

11.2 Реляционная алгебра

Рассмотрим теперь реляционные операции над множеством всех реляционных отношений.

Всякая операция из алгебры по определению отображает декартово произведение множества M на само множество M . Это свойство называется **свойством замкнутости алгебры**.

Все реляционные операции были определены так, чтобы применяться к реляционным отношениям и возвращать в результате реляционное отношение. Это дает возможность использовать композиции реляционных операций, т.е. результат одной операции может использоваться в качестве исходных данных для другой операции.

Все реляционные операции принято делить на две группы:

1. Теоретико-множественные (или булевы),
2. Специальные реляционные операции

11.3. Теоретико-множественные операции

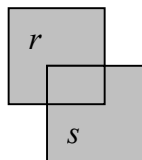
Работают с отношением как с множеством кортежей (см. определение [реляционного отношения](#)). Это – традиционные операции над множествами: объединение, пересечение, разность, декартово произведение. Однако, в отличие от обычных математических множеств, операции над отношениями имеют такую особенность:

Реляционные отношения, к которым применяются операции объединения, пересечения, разности, должны быть **совместимы по типу**, т.е.

1. Оба отношения имеют одинаковую схему.
2. Соответствующие атрибуты определены на одном и том же домене.

11.3.1 Объединение

Пусть $r(R)$, $s(R)$ – отношения, совместимые по типу.



Отношение $q(R) = r \cup s$ – **объединение отношений** r и s – отношение, состоящее из всех кортежей, принадлежащих хотя бы одному из исходных отношений.

Пример 8.

Дано:

r	A	B	C	D
	$a1$	$b1$	$c1$	$d1$
	$a2$	$b2$	$c2$	$d1$

s	A	B	C	D
	$a1$	$B2$	$c1$	$d1$
	$a2$	$b2$	$c2$	$d1$

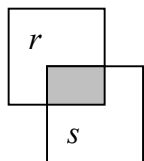
$r \cup s$	A	B	C	D
	$a1$	$b1$	$c1$	$d1$
	$a2$	$b2$	$c2$	$d1$
	$a1$	$b2$	$c1$	$d1$

На языке SQL:

r UNION s ;

11.3.2 Пересечение

Пусть $r(R)$, $s(R)$ – отношения, совместимые по типу.



Отношение $q(R) = r \cap s$ – **пересечение отношений** r и s – отношение, состоящее из всех кортежей, принадлежащих обоим исходным отношениям одновременно.

Пример 9.

Дано:

r	A	B	C	D
	$a1$	$b1$	$c1$	$d1$
	$a2$	$b2$	$c2$	$d1$

s	A	B	C	D
	$a1$	$B2$	$c1$	$d1$
	$a2$	$b2$	$c2$	$d1$

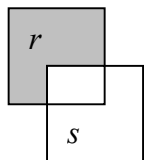
$r \cap s$	A	B	C	D
	$a2$	$b2$	$c2$	$d1$

На языке SQL:

r INTERSECT s ;

11.3.3 Разность

Пусть $r(R)$, $s(R)$ – отношения, совместимые по типу.



Отношение $q(R) = r \setminus s$ – **разность отношений** r и s – отношение, состоящее из всех кортежей, принадлежащих отношению r и не принадлежащих отношению s .

Замечание: как и в обычной арифметике, $r \setminus s$ не то же самое, что и $s \setminus r$

Пример 10.

Для отношений r и s из примера 9

$r \setminus s$	A	B	C	D
	$a1$	$b1$	$c1$	$d1$

$s \setminus r$	A	B	C	D
-----------------	-----	-----	-----	-----

	<i>a1</i>	<i>b2</i>	<i>c1</i>	<i>d1</i>
--	-----------	-----------	-----------	-----------

На языке SQL:

r MINUS *s*;

11.3.4 Декартово произведение

Пусть $r(R)$, $s(S)$ – два реляционных отношения. $R=\{A_1, \dots, A_n\}$, $S=\{B_1, \dots, B_m\}$. R и S не имеют общих атрибутов.

Отношение $q(R \cup S) = r * s$ – **декартово произведение отношений** r и s – отношение, состоящее из всех кортежей t вида
 $t = \langle A_1: a_1, \dots, A_n: a_n, B_1: b_1, \dots, B_m: b_m \rangle$

Замечание: в математике декартово произведение множеств r и s является множеством упорядоченных пар элементов $\langle t(R), t(S) \rangle$, в нашем случае – пар кортежей $t(R)$ и $t(S)$.

Поэтому приведенное выше определение отличается от общепринятого, для поддержания свойства замкнутости реляционных отношений относительно реляционных операций.

Пример 11.

Дано:

<i>r</i>	<i>A</i>
	<i>a1</i>
	<i>a2</i>

<i>s</i>	<i>B</i>
	<i>b1</i>
	<i>b2</i>

$r * s$	<i>A</i>	<i>B</i>
	<i>a1</i>	<i>b1</i>
	<i>a1</i>	<i>b2</i>
	<i>a2</i>	<i>b1</i>
	<i>a2</i>	<i>b2</i>

На SQL: *r* TIMES *s*;

11.3.5 Операция переименования

Реляционная алгебра требует, чтобы все имена атрибутов во всех отношениях были уникальными.

Для выполнения этого используется операция переименования

Пусть $r(R)$ - реляционное отношение. Атрибут $A \in R$, атрибут $B \in (R \setminus A)$.

Пусть $R' = (R \setminus A) \cup B$ - схема, в которой вместо имени атрибута A стоит имя атрибута B .

Тогда операция $\delta_{A \leftarrow B}(r)$ возвращает новую копию отношения r , в которой атрибут A имеет другое заданное имя B .

Пример 12.

<i>r</i>	<i>A</i>	<i>C</i>	<i>D</i>
	<i>a1</i>	<i>c1</i>	<i>d1</i>
	<i>a2</i>	<i>c2</i>	<i>d2</i>

$$q = \delta_{A \leftarrow B}(r)$$

q	B	C	D
	$a1$	$c1$	$d1$
	$a2$	$c2$	$d2$

На языке SQL: r [RENAME] A AS B

Имея оператор переименования, можно, например, выполнить декартово произведение отношения с самим собой:

Пример 13

r	A
	$a1$
	$a2$

$$q = r * \delta_{A \leftarrow B}(r)$$

q	A	B
	$a1$	$a1$
	$a1$	$a2$
	$a2$	$a1$
	$a2$	$a2$

11.3.6 Свойства теоретико-множественных операций

Свойства реляционных операций используются в оптимизации выражений запросов.

1. Объединение, пересечение, декартово произведение – коммутативны

$$r \cup s = s \cup r$$

2. Объединение, пересечение, декартово произведение – ассоциативны

$$r \cup s \cup q = (r \cup s) \cup q = r \cup (s \cup q)$$

3. Разность некоммутативна

$$r \setminus s \neq s \setminus r$$

С помощью разности можно представить пересечение

$$r \cap s = r \setminus (r \setminus s)$$

11.4 Специальные реляционные операции

Пусть множество $\Theta = \{=, \neq, <, >, \leq, \geq\}$ - множество операций сравнения элементов из двух доменов.

Если $\theta \in \Theta$, а A и B – атрибуты, то говорят, что A – **тега-сравним с B** , если оба атрибута из одного домена или домен одного атрибута является подмножеством другого.

11.4.1 Выборка (селекция)

Пусть $r(R)$ - реляционное отношение. Атрибут $A \in R$, значение $a \in \text{dom}(A)$.

Операция выборки $\sigma_{A=a}(r)$ возвращает отношение $r'(R)$, для которого значение атрибута A равно a .
В общем виде,

$$\sigma_{A=a}(r) = r'(R) = \{t \in r : t(A) \theta a\}$$

На языке SQL операция выборки записывается так:

R WHERE A θ a

Допускается сравнение не с константой a , а с другим атрибутом.

Пусть атрибут $A \in R$, атрибут $B \in R$ – совместим по типу с атрибутом A . Тогда

$$\sigma_{A \theta B}(r) = r'(R) = \{t \in r : t(A) \theta t(B)\}$$

На языке SQL:

R WHERE A θ B

В операции выборки может одновременно проверяться несколько логических условий, в которых допускаются логические связки AND, OR, NOT.

Пример 14.

$\sigma_{(A=a) \text{ AND } (B < b) \text{ OR } (A >= B)}(r)$. В данном случае приоритет выполнения операций стандартный.

Пример 15.

Дано отношение «ПОСТАВЩИКИ»(п_номер, п_адрес, п_название)

<i>r</i>	<i>П номер</i>	<i>П адрес</i>	<i>П название</i>
	S001	Киев	Оболонь
	S002	Киев	Сепро
	S003	Львов	Свиточ
	S004	Запорожье	Славутич

Результат запроса «выбрать всех поставщиков из Киева»:

$\sigma_{n_адрес="Киев"}(r)$	<i>П номер</i>	<i>П адрес</i>	<i>П название</i>
	S001	Киев	Оболонь
	S002	Киев	Сепро

На SQL:

r WHERE п_адрес= «Киев»;

Выборка по нескольким логическим условиям C_1, C_2 производится так:

1. $\sigma_{C_1 \text{ AND } C_2}(r) = \sigma_{C_1}(r) \cap \sigma_{C_2}(r)$
2. $\sigma_{C_1 \text{ OR } C_2}(r) = \sigma_{C_1}(r) \cup \sigma_{C_2}(r)$
3. $\sigma_{\text{NOT } C_2}(r) = r \setminus \sigma_{C_2}(r)$

11.4.2 Проекция

Пусть $r(R)$ - реляционное отношение. $R = X \cup Y$, X, Y – подмножества схемы R

Операция проекции $\pi_X(r)$ возвращает отношение $r'(X)$, полученное из исходного отношения вычеркиванием столбцов Y и исключением из оставшихся столбцов повторяющихся строк.

$$\pi_X(r) = r'(X) = \{t(X) : t \in r\}$$

На языке SQL операция проекции записывается так:

```
SELECT [DISTINCT] X FROM r;
```

Или, если представить множество $X=A_1 \cup \dots \cup A_n$, то

```
SELECT [DISTINCT] A_1, ..., A_n FROM r;
```

Ключевое слово DISTINCT явно указывается, когда нужно получить неповторяющиеся строки.

Пример 16.

Для отношения «ПОСТАВЩИКИ» из примера 15 перечислить адреса всех поставщиков, с которыми работает фирма.

```
SELECT DISTINCT п_адрес FROM ПОСТАВЩИКИ;
```

Результатом будет отношение

$\pi_{п_адрес}(r)$	<i>П адрес</i>
	<i>Киев</i>
	<i>Львов</i>
	<i>Запорожье</i>

11.4.3 Соединение и его разновидности

В отличие от проекции и выборки, операция соединения использует два и более отношений. В общем случае операция соединения комбинирует два отношения по всем их общим атрибутам. Критерий комбинирования – либо по равенству значений в соответствующих столбцах (тогда соединение называется **естественным**), либо по операциям сравнения $<$, $<=$, $>$, $>=$ (тогда это **тета-соединение**)

Рассмотрим сначала естественное соединение двух отношений

Пусть $r(R)$, $s(S)$ – два реляционных отношения.

Пусть $T = R \cup S$ – схема, состоящая из всех атрибутов и схемы R , и схемы S .

Пусть подсхема $B = R \cap S \neq \emptyset$, т.е. схемы имеют общие атрибуты.

Операция естественного соединения $r \bowtie s$ возвращает отношение
 $q(T) = r \bowtie s = \{t \in T : \exists t_r \in r, t_s \in s, \text{ такие, что } t_r = t(R), t_s = t(S) \text{ и } t_r(B) = t_s(B)\}$

Отношение q имеет схему T , состоящую из всех атрибутов схем R и S . Каждый кортеж из q является комбинацией кортежа из r и кортежа из s с равными значениями подкортежей из $R \cap S$.

Если R и S не имеют общих атрибутов, т.е. $R \cap S = \emptyset$, то результат естественного соединения совпадает с декартовым произведением отношений r и s :

$$r \bowtie s = r * s$$

На языке SQL:

Вариант 1. `r NATURAL JOIN s;`

Вариант 2. `SELECT R, S
FROM r, s
WHERE r.B = s.B;`

Если множество $B = B_1 \cup \dots \cup B_m$, то


```

SELECT R, S
FROM r, s
WHERE r.B_1 = s.B_1
      AND r.B_2 = s.B_2 ...
      AND r.B_n = s.B_n;

```

Пример 17.

Дано:

<i>r</i>	<i>A</i>	<i>B</i>
	<i>a1</i>	<i>b1</i>
	<i>a2</i>	<i>b1</i>
	<i>a3</i>	<i>b2</i>

<i>s</i>	<i>B</i>	<i>C</i>
	<i>b1</i>	<i>c1</i>
	<i>b2</i>	<i>c1</i>
	<i>b3</i>	<i>c2</i>

$r \triangleright \triangleleft s$	<i>A</i>	<i>B</i>	<i>C</i>
	<i>a1</i>	<i>b1</i>	<i>c1</i>
	<i>a2</i>	<i>b1</i>	<i>c1</i>
	<i>a3</i>	<i>b2</i>	<i>c1</i>

В данном случае два отношения соединяются по атрибуту *B*. В обоих отношениях есть одинаковые значения этого атрибута (*b1*, *b2*).

Пример 18.

Пусть помимо отношения «ПОСТАВЩИКИ» в базе данных есть отношение «ПОСТАВКИ». Выполнить запрос: включить в информацию о поставках информацию о поставщиках.

<i>s</i>	<i>П_номер</i>	<i>Код_пст</i>	<i>Код_товара</i>	<i>дата</i>	<i>Кол-во</i>
	<i>S001</i>	<i>P025</i>	<i>517</i>	<i>01.04.00</i>	<i>100</i>
	<i>S004</i>	<i>P026</i>	<i>266</i>	<i>01.04.00</i>	<i>500</i>

Результат запроса такой:

$r \triangleright \triangleleft s$	<i>П номер</i>	<i>П адрес</i>	<i>П название</i>	<i>Код пст</i>	<i>Код товара</i>	<i>дата</i>	<i>Кол-во</i>
	<i>S001</i>	<i>Киев</i>	<i>Оболонь</i>	<i>P025</i>	<i>517</i>	<i>01.04.00</i>	<i>100</i>
	<i>S004</i>	<i>Запорожье</i>	<i>Славутич</i>	<i>P026</i>	<i>266</i>	<i>01.04.00</i>	<i>500</i>

При использовании оператора естественного соединения отношения могут комбинироваться только по **однoименным** столбцам, и должны комбинироваться по всем таким столбцам. Однако отношения могут соединяться по столбцам с разными именами атрибутов, но с равными доменами.

Пусть $r(R)$, $s(S)$ – два реляционных отношения. Атрибуты $A_i \in R$, $B_i \in S$, $dom(A_i) = dom(B_i)$, $i=1, \dots, m$

Операция эквисоединения отношений r и s по атрибутам A_1, \dots, A_m , и B_1, \dots, B_m возвращает отношение

$$q(R \cup S) = r[A_1 = B_1, \dots, A_m = B_m]s = \{t \in R \cup S : \exists t_r \in r, t_s \in s, t_r = t(R), t_s = t(S) \text{ и } t(A_i) = t(B_i)\}$$

Пример 19.

Если при проектировании отношения «ПОСТАВКИ» столбец «П_номер» (номер поставщика) был назван, например, «Код_поставщика», то получить результат запроса из примера 18 можно будет эквисоединением отношений «ПОСТАВЩИКИ» и «ПОСТАВКИ».

Обобщая случай сравнения значений из равных доменов, получим

Операция тета-соединения отношений r и s по атрибутам A и B ($dom(A) = dom(B)$) возвращает отношение

$$q(R \cup S) = r[A \theta B]s = \{t \in R \cup S : \exists t_r \in r, t_s \in s, t_r = t(R), t_s = t(S) \text{ и } t(A) \theta t(B)\}$$

На языке SQL:

Вариант 1.

```
(r TIMES s) WHERE A theta B
```

Вариант 2.

```
SELECT R, S
FROM r, s
WHERE r.B_1 theta_1 s.B_1
      AND r.B_2 theta_2 s.B_2 ...
      AND r.B_n theta_m s.B_n;
```

(если θ_i принадлежит к операциям сравнения на доменах)

Пример 20.

Вполне допустимо такое тета-соединение:

$q = r[A < D, B = C]$, если $r = r(AB)$, $s = s(CD)$, и соответствующие домены равны.

11.4.4 Соединение и NULL-значения

От соединений, перечисленных выше, **внешнее соединение** отличается тем, что кортежи одного отношения, которые не имеют соответствующих значений во втором отношении, появляются в результирующем отношении со значениями NULL в позициях атрибутов второго отношения, вместо того, чтобы быть просто проигнорированными (как это делается в обычных соединениях).

Внешнее соединение имеет такие разновидности:

- левое соединение – «сохраняет» информацию из левого отношения
- правое соединение – «сохраняет» информацию из правого отношения
- полное соединение – «сохраняет» информацию из обоих отношений.

Пример 21

Дано:

r	A	B
	$a1$	$b1$
	$a2$	$b1$
	$a3$	$b2$

<i>s</i>	<i>B</i>	<i>C</i>
	<i>b1</i>	<i>c1</i>
	<i>b2</i>	<i>c1</i>
	<i>b3</i>	<i>c2</i>

Результат правого соединения такой:

<i>q</i>	<i>A</i>	<i>B</i>	<i>C</i>
	<i>a1</i>	<i>b1</i>	<i>c1</i>
	<i>a2</i>	<i>b1</i>	<i>c1</i>
	<i>a3</i>	<i>b2</i>	<i>c1</i>
	<i>null</i>	<i>b3</i>	<i>c2</i>

11.4.5 Деление

Пусть $r(R)$, $s(S)$ – два реляционных отношения, причем $S \in R$

Пусть $R' = R \setminus S$

Операция деления $r \div s$ формирует отношение r' – частное от деления r на s
 $r' = r \div s = \{t : \forall t_s \in s \exists t_r \in r, t_r(R') = t, \text{ и } t_r(S) = t_s\}$

Пример 22.

Для отношения «ПОСТАВЩИКИ» (r) выбрать всех поставщиков, имеющих филиалы в Киеве, Одессе и Запорожье одновременно.

<i>r</i>	<i>П номер</i>	<i>П адрес</i>	<i>П название</i>
	<i>S001</i>	<i>Киев</i>	<i>Оболонь</i>
	<i>S002</i>	<i>Киев</i>	<i>Сепро</i>
	<i>S005</i>	<i>Запорожье</i>	<i>Оболонь</i>
	<i>S006</i>	<i>Одесса</i>	<i>Оболонь</i>

Для решения задачи разделим отношение r на отношение s :

<i>s</i>	<i>П адрес</i>
	<i>Киев</i>
	<i>Львов</i>
	<i>Запорожье</i>

Результат будет таким (обратите внимание на то, что в результирующем отношении нет атрибутов из схемы S – по определению частного от деления)

$r \div s$	<i>П номер</i>	<i>П название</i>
	<i>S001</i>	<i>Оболонь</i>
	<i>S005</i>	<i>Оболонь</i>
	<i>S006</i>	<i>Оболонь</i>

11.5 Определение реляционной алгебры

Итак. Теперь дадим строгое определение реляционной алгебры.

Пусть U – множество имен атрибутов.

Пусть D – множество всевозможных доменов.

Пусть $dom:U \rightarrow D$ - отображение, переводящее множество имен атрибутов во множество доменов.

Пусть $R = \{R_1, \dots, R_p\}$ - множество различных схем отношений: $R_i \subseteq U, i = 1, \dots, p$

Пусть d - множество отношений со схемами из R .

Пусть $\Theta = \{=, \neq, <, >, \leq, \geq\}$ - множество операций сравнения на доменах.

O - множество реляционных операций: $O = \{\cup, \cap, \setminus, *, \delta, \sigma, \pi, \triangleright, \triangleleft\}$ и операция деления отношений.

Тогда реляционная алгебра $B = \langle U, D, dom, R, d, \Theta, O \rangle$ - семиместный кортеж, в котором носитель алгебры - $\langle U, D, dom, R, d, \Theta \rangle$, а множество операций - множество O .