
**Реляционная модель.
Реляционная алгебра и
реляционное исчисление**

Рассматриваемые вопросы:

1. Реляционная модель

- Краткий обзор истории реляционной модели
- Используемая терминология
- Альтернативная терминология
- Математические отношения
- Отношения и их свойства в базе данных
- Реляционные ключи
- Представление схем в реляционной базе данных
- Реляционная целостность

2. Реляционные языки

3. Реляционная алгебра

- Унарные операции реляционной алгебры
- Операции с множествами
- Операции соединения
- Деление

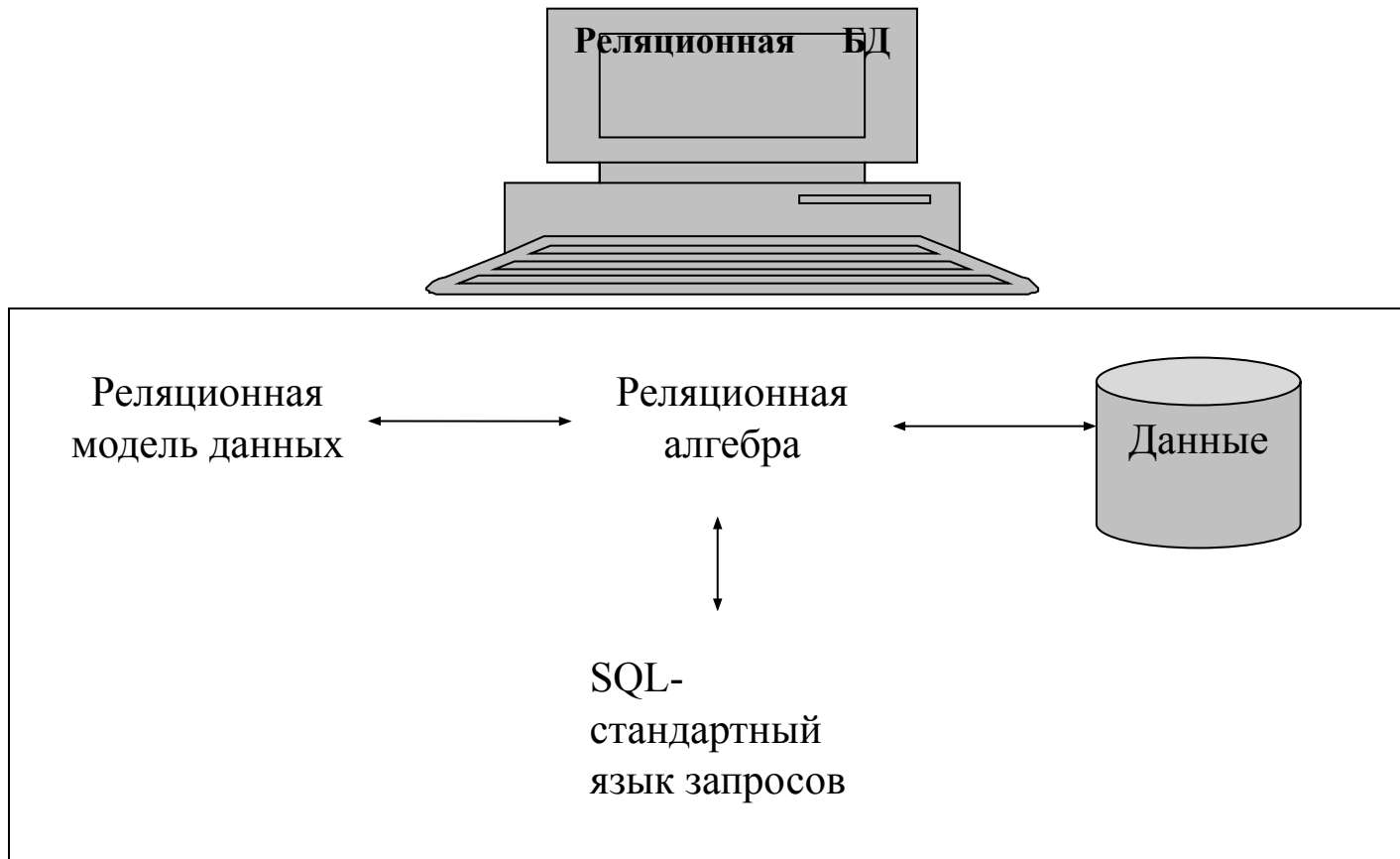
4. Реляционное исчисление

- Реляционное исчисление кортежей
- Реляционное исчисление доменов

5. Другие языки

Реляционная модель

Структура обработки информации в реляционной БД



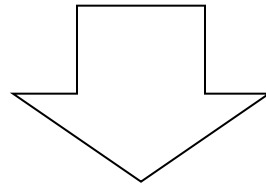
Цели создания реляционной модели:

- 1) Обеспечение высокой степени независимости от данных.
- 2) Нормализация отношений, т.е. создание отношений без повторяющихся групп.
- 3) Расширение языков управления данными за счет включения операций над множествами.

Реляционная модель

Используемая терминология

По сколько Кодд, будучи опытным математиком, широко использовал математическую терминологию теории множеств и логики предикатов.



Реляционная модель основана на математическом понятии отношения, физическим представлением которого является таблица.

Используемая терминология

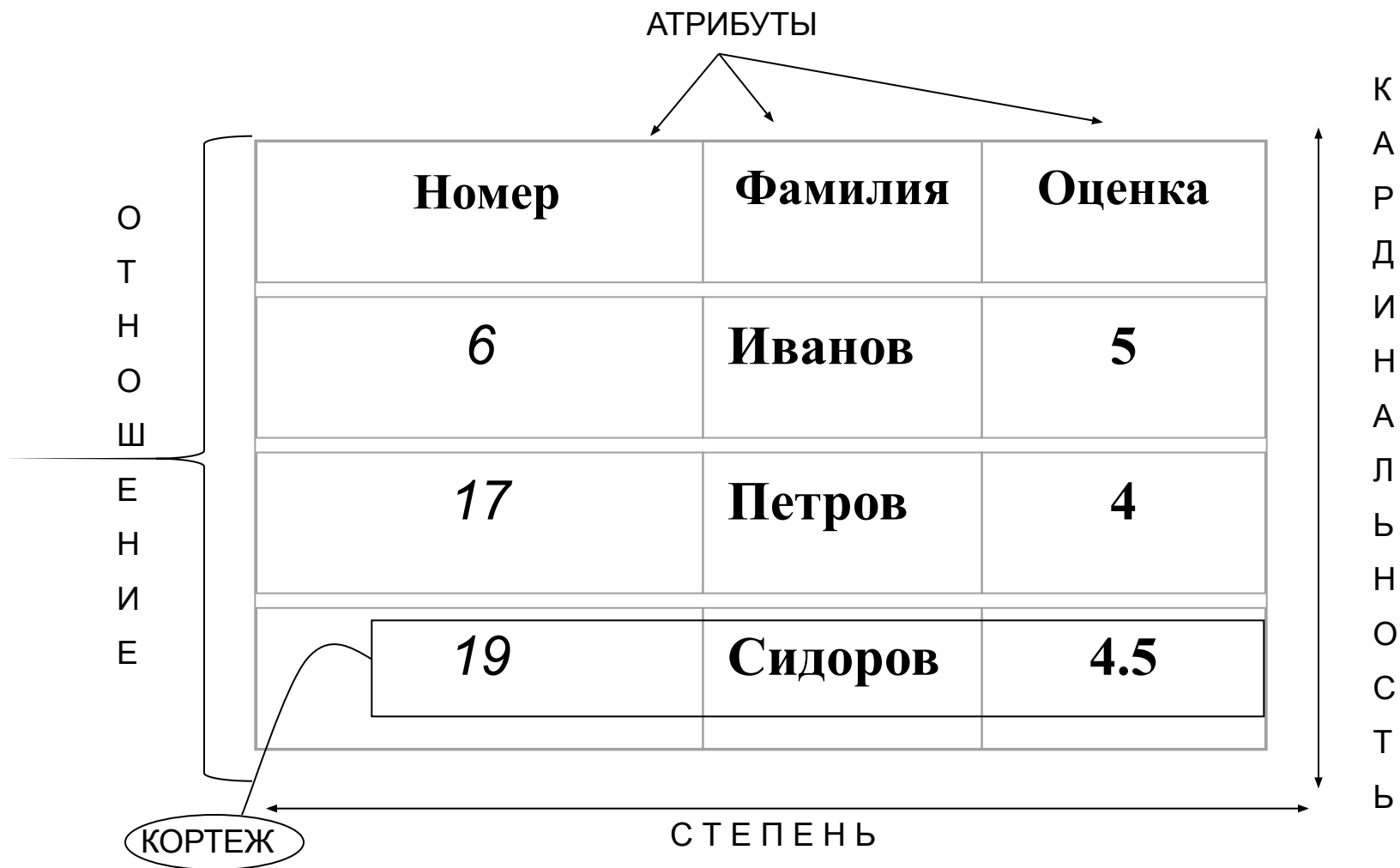
Структура реляционных данных

- Отношение
 - Атрибут
 - Домен
 - Кортеж
- ★ Кардинальность
 - ★ Степень отношения

Таблицы данных

Реляционная база данных

Структура реляционных данных



Отношение - это плоская таблица (двумерная), состоящая из столбцов и строк.

Атрибут - это поименованный столбец отношения.

Домен - это набор допустимых значений для одного или нескольких атрибутов, благодаря ему пользователь может централизованно определять смысл и источник значений, которые могут получать атрибуты.

Кортеж - это строка отношения.

Кортежи называются **расширением, состоянием или телом отношения**, которое постоянно меняется.

Описание структуры отношения вместе со спецификацией доменов и любыми другими ограничениями возможных значений атрибутов иногда называют его **заголовком** (или **содержанием** (intension)).

Степень отношения *определяется количеством атрибутов, которое оно содержит.*

Кардинальность - *это количество кортежей, которое содержит отношение.*

Кардинальность - свойство тела отношения (меняется при каждом добавлении или удалении кортежей).

Реляционная база данных - *это набор нормализованных отношений.*

Реляционная база данных состоит из отношений, структура которых определяется с помощью особых методов, называемых *нормализацией*.

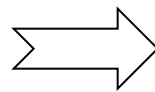
Альтернативная терминология

Официальные термины	Альтернативный вариант 1	Альтернативный вариант 2
Отношение	Таблица	Файл
Кортеж	Строка	Запись
Атрибут	Столбец	Поле

Отношения и их свойства в базе данных

Реляционная схема – это имя отношения, за которым следует множество пар атрибутов и доменов.

атрибуты A_1, A_2, \dots, A_n
домены D_1, D_2, \dots, D_n



реляционная схема:
 $\{A_1:D_1 \dots A_n:D_n\}$

В реляционной модели отношение можно представить как произвольное подмножество декартового произведения, а таблица – это физическое представление такого отношения.

ДАННЫХ

Свойства отношений:

- Отношение имеет неповторимое *имя*.
- Каждая ячейка отношения содержит только *атомарное (неделимое) значение*.
- Каждый атрибут имеет *уникальное имя*.
- Значения атрибута берутся из одного и того же домена.
- Порядок следования атрибутов не имеет никакого значения.
- Каждый кортеж является *уникальным*, т.е. дубликатов кортежей быть не может.
- Порядок следования кортежей в отношении не имеет никакого значения.

Реляционные ключи

Суперключ (*superkey*) – атрибут или множество атрибутов, которое единственным образом идентифицирует кортеж данного отношения.

Потенциальный ключ – это суперключ, который не содержит подмножества, также являющегося суперключом данного отношения.

Потенциальный ключ K для данного отношения R обладает двумя свойствами:

- **Уникальность.** В каждом кортеже отношения R значение ключа K единственным образом идентифицируют этот кортеж.
- **Неприводимость.** Никакое допустимое подмножество ключа K не обладает свойством уникальности.

Реляционные ключи

Наличие значений-дубликатов в конкретном существующем наборе кортежей доказывает то, что некоторая комбинация атрибутов не может быть потенциальным ключом.

*Если ключ состоит из нескольких атрибутов, то он называется **СОСТАВНЫМ КЛЮЧОМ**.*

Первичный ключ – это потенциальный ключ, который выбран для уникальной идентификации кортежей внутри отношения.

Поскольку отношение не содержит кортежей-дубликатов, всегда можно уникальным образом идентифицировать каждую его строку.

Это значит, что *отношение всегда имеет первичный ключ.*

Реляционные ключи

Потенциальные ключи, которые не выбраны в качестве первичного ключа, называются

альтернативными ключами.

Внешний ключ – это атрибут или множество атрибутов внутри отношения, которое соответствует потенциальному ключу некоторого (может быть, того же самого) отношения.

Представление схем в реляционной базе данных

Реляционная база данных может состоять из произвольного количества отношений.

**Концептуальной моделью, или
концептуальной схемой, называется
множество всех реляционных баз данных.**

Реляционная целостность

Модель данных имеет две части:

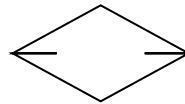
- *управляющую часть*, которая определяет типы допустимых операций с данными,
- *набор ограничений целостности*, которые гарантируют корректность данных.

Определитель NULL *вводится в связи с поддержанием правил целостности и указывает, что значение атрибута в настоящий момент неизвестно или неприемлемо для этого кортежа.*

Нули и пробелы представляют собой некоторые значения, тогда как ключевое слово *NULL* призвано обозначать отсутствие какого-либо значения.

Реляционная целостность

Целостность сущностей означает, что в отношении ни один атрибут первичного ключа не может содержать отсутствующих значений, обозначаемых определителем *NULL*.



Если будет определитель *NULL* в любой части первичного ключа, это утверждает, что не все его атрибуты необходимы для уникальной идентификации кортежей. Это противоречит определению первичного ключа.

Реляционная целостность

Ссылочная целостность.

Если в отношении существует внешний ключ, то значение внешнего ключа должно либо соответствовать значению потенциального ключа некоторого кортежа в его базовом отношении, либо задаваться определителем NULL.

Корпоративные ограничения целостности данных
это дополнительные правила поддержки целостности, определяемые пользователями или администраторами базы данных.

Реляционные языки

Реляционная алгебра - *(высокоуровневый) процедурный язык.*

Использование: сообщение СУБД о том, как следует построить требуемое отношение на базе одного или нескольких существующих в базе данных отношений.

Реляционное исчисление - *непроцедурный язык.*

Использование: определения того, каким будет некоторое отношение, созданное на основе одного или нескольких других отношений базы данных.

Реляционно-полный язык

Использование: получение любого отношения, которое можно вывести с помощью реляционного исчисления.

Реляционная алгебра

Реляционная алгебра - теоретический язык операций, который на основе одного или нескольких отношений позволяет создавать другое отношение без изменения самих исходных отношений.

Реляционная алгебра

Основные операции реляционной алгебры:

- выборка (selection)
- проекция (projection)
- декартово произведение (cartesian product)
- объединение (union)
- разность (set difference)

Дополнительные операции:

- соединения (join)
- пересечения (intersection)
- деления (division)

Унарные операции реляционной алгебры

Операция выборки:

Работает с одним отношением **R**.

Определяет результирующее отношение с тем же заголовком, что и отношение **R**, и телом, состоящим из кортежей, значения атрибутов которых при подстановке в условие (предикат) дают значение истина.

Унарные операции реляционной алгебры

Простейший случай:

$X \Theta Y$ - условие (предикат),

Θ – один из операторов сравнения ($=$, \neq , $<$, $>$ и т.д.),

X и Y - атрибуты отношения R или скалярные значения.

Синтаксис операции выборки:

R where σ , или R where $(X \Theta Y)$

Синтаксис на языке SQL:

select * from R where $(X \Theta Y)$

Унарные операции реляционной алгебры

Пример операции выборки

Отношение **R** (информация о студентах)

Номер студента	Фамилия	Средний балл
6	Иванов	5
17	Петров	4
19	Сидоров	4,5

Результат выборки **R where Средний балл < 5**

Номер студента	Фамилия	Средний балл
17	Петров	4
19	Сидоров	4,5

Унарные операции реляционной алгебры

Операция проекции:

Работает с одним отношением **R**.

Определяет новое отношение с заголовком (X, \dots, Z) , содержащее вертикальное подмножество отношения **R**, создаваемое посредством извлечения значений указанных атрибутов из результата строк-дубликатов.

Синтаксис операции проекции:

R [X, ..., Z]

Синтаксис на языке SQL:

Select X, Y, ..., Z from R

Унарные операции реляционной алгебры

Пример операции проекции

Отношение **R** (информация о преподавателях)

Табельный номер	Фамилия	Предмет
4587	Бондаренко	География
2136	Воронин	История
5496	Анисимова	История
5496	Анисимова	Философия

Проекция **R** [Предмет]

Предмет
География
История
Философия

Операции с множествами

Декартово произведение $R \times S$

определяет новое отношение, которое является результатом конкатенации

(т.е. сцепления) каждого кортежа из отношения R с каждым кортежем из отношения S .

Синтаксис операции декартового произведения:

R times S

Синтаксис на языке SQL :

Select * from R, S

Операции с множествами

Пример декартового произведения

Отношение **R** (Студенты)

Номер студента	Фамилия
6	Иванов
17	Петров
19	Сидоров

Отношение **S** (Предметы)

Код предмета	Название
101	Физика
102	Математика
103	Информатика

Операции с множествами

Отношение **R TIMES S**

Номер студента	Фамилия	Код предмета	Название
6	Иванов	101	Физика
6	Иванов	102	Математика
6	Иванов	103	Информатика
17	Петров	101	Физика
17	Петров	102	Математика
17	Петров	103	Информатика
19	Сидоров	101	Физика
19	Сидоров	102	Математика
19	Сидоров	103	Информатика

Операции с множествами

Операция объединения $R \bowtie S$

получается в результате конкатенации R и S , с образованием одного отношения с тем же заголовком, что и у отношений R и S и телом, состоящим из кортежей, принадлежащих или R , или S , или обоим отношениям (с максимальным количеством кортежей), если кортежи-дубликаты исключены.

Синтаксис операции объединения:

$R \text{ union } S$.

Синтаксис на языке SQL:

(Select * from R) union (select * from S)

Операции с множествами

Пример операции объединения

Отношение **R** (информация о студентах)

Номер студента	Фамилия	Средний балл
6	Иванов	5
17	Петров	4
19	Сидоров	4,5

Отношение **S** (информация о студентах)

Номер студента	Фамилия	Средний балл
6	Иванов	5
18	Пушников	3,5
19	Сидоров	4,5

Операции с множествами

Объединение отношений **R** и **S**

Номер студента	Фамилия	Средний балл
6	Иванов	5
17	Петров	4
19	Сидоров	4,5
18	Пушников	3,5

Операции с множествами

Операция разности R-S

определяет отношение с тем же заголовком, что и у отношений **R** и **S**, и телом, состоящим из кортежей, принадлежащих отношению **R** и не принадлежащих отношению **S**, таких, которые имеются в отношении **R**, но отсутствуют в отношении **S**.

Синтаксис операции разности:

R minus S

Синтаксис на языке SQL:

(select * from R) except (select * from S)

Операции с множествами (щшибка)

Пример операции разности

Отношение R

Номер студента	Фамилия	Средний балл
6	Иванов	5
17	Петров	4
19	Сидоров	4,5

Отношение S

Номер студент	Фамилия	Средний балл
6	Петров	4
18	Сидоров	4,5
20	Пушников	3,5

Операции с множествами

Отношение **R MINUS S**

Номер студента	Фамилия	Средний балл
17	Петров	4
19	Сидоров	4,5

Операции с множествами

Операция пересечения $R \cap S$

определяет отношение, которое содержит кортежи, присутствующие как в отношении **R**, так и в отношении **S**.

Синтаксис операции пересечения:

R intersect S

Синтаксис на языке SQL:

(Select * from R) intersect (select * from S)

Операции с множествами

Пример операции пересечения

Отношение **R** (информация о студентах)

Номер студента	Фамилия	Средний балл
6	Иванов	5
17	Петров	4
19	Сидоров	4,5

Отношение **S** (информация о студентах)

Номер студента	Фамилия	Средний балл
6	Иванов	5
18	Пушников	3,5
20	Сидоров	4,5

Операции с множествами

Отношение **R INTERSECT S**

Номер студента	Фамилия	Средний балл
6	Иванов	5

Операции соединения

Операция соединения - комбинация декартового произведения и выборки, эквивалентна операции выборки из декартового произведения двух операндов отношений тех кортежей, которые удовлетворяют условию, указанному в предикате соединения в качестве формулы выборки.

Соединением отношений **R** и **S** по условию **F** называется отношение **(R times S) where F**

Синтаксис на языке SQL:

Select R.*, S.* from R, S where f

Операции соединения

Типы операций соединения:

- тета-соединение
 - соединение по эквивалентности (частный случай тета-соединения)
 - естественное соединение
 - внешнее соединение
 - полусоединение
-

Операции соединения

Тета-соединение $R \bowtie_F S$ определяет отношение, которое содержит кортежи из декартового произведения отношений R и S , удовлетворяющие предикату F .

Предикат F имеет вид $R.a_i \Theta S.b_j$, где Θ один из операторов сравнения ($<$, \leq , $>$, \geq , $=$ или \neq).

Θ -соединением отношения R по атрибуту X с отношением S по атрибуту Y называют отношение $(R \text{ times } S) \text{ where } (X \Theta Y)$.

Синтаксис на языке SQL:

Select * from R, S where (R.X Θ S.Y)

Операции соединения

Пример тета-соединения

В базе данных хранится информация о:

- преподавателях;
- предметах.

Примечание: преподаватели имеют право преподавать предметы, статус которых не выше статуса преподавателя.

Отношение R (Преподаватели)

Табельный номер	Фамилия	X(Статус преподавателя)
4587	Бондаренко	4
2136	Воронин	1
5496	Анисимова	2

Отношение S (Предметы)

Код предмета	Название	Y(Статус предмета)
101	История	3
102	География	2
103	Философия	1

Операции соединения

Ответ на вопрос: "Какие преподаватели имеют право преподавать какие предметы?" дает Θ -соединение $R[X \geq Y]S$:

Отношение "Какие преподаватели преподают какие предметы?"

Табельный номер	Фамилия	X(Статус преподавателя)	Код предмета	Название	Y(Статус предмета)
4587	Бондаренко	4	101	История	3
4587	Бондаренко	4	102	География	2
4587	Бондаренко	4	103	Философия	1
2136	Воронин	1	103	Философия	1
5496	Анисимова	2	102	География	2
5496	Анисимова	2	103	Философия	1

Операции соединения

Экви-соединение (соединение по эквивалентности) - частный случай Θ -соединения, когда Θ есть просто равенство (предикат F содержит только оператор равенства (=)).

Синтаксис экви-соединения:

$R[X=Y]S$

Синтаксис на языке SQL:

select R.*, S .* from R, S where (R.X = S.Y)

Операции соединения(Ошибка)

Пример экви-соединения

Отношение S (Студенты)

Номер студента SNUM	Фамилия студента SNAME
6	Иванов
17	Петров
19	Сидоров

Отношение P (Предметы)

Код предмета PCOD	Название предмета PNAME
101	Физика
102	Математика
103	Информатика

Операции соединения

Отношение SP (Изучение)

Номер студента SNUM	Код предмета PCOD	Средний бал по предмету SRBALL
6	101	4,5
6	102	4
6	103	5
17	101	3,5
17	102	4
19	101	4,5

Ответ на вопрос: "Какие предметы изучаются студентами?", дает эквивалентное соединение $S[SNUM = SNUM]SP$. Т.к. в отношениях имеются одинаковые атрибуты, то требуется сначала их переименовать.

Получаем: **$(S \text{ rename } SNUM \text{ as } SNUM1)[SNUM1 = SNUM2](SP \text{ rename } SNUM \text{ as } SNUM2)$** .

Операции соединения

Отношение "Какие предметы изучаются какими студентами?"

Номер студента SNUM1	Фамилия студента SNAME	Номер студента SNUM2	Код предмета PCOD	Средний балл по предмету SRBALL
6	Иванов	6	101	4,5
6	Иванов	6	102	4
6	Иванов	6	103	5
17	Петров	17	101	3,5
17	Петров	17	102	4
19	Сидоров	19	101	4,5

Операции соединения

Естественное соединене $R \bowtie S$ -
соединение по эквивалентности двух отношений R и S , выполненное по всем общим атрибутам, из результатов которого исключается по одному экземпляру каждого общего атрибута.

Синтаксис естественного соединения:

$R \text{ join } S$.

Естественное соединение производится по всем одинаковым атрибутам.

Операции соединения

Пример естественного соединения

Упрощенная запись:

Ответ на вопрос "Какие предметы изучаются какими студентами?" в виде естественного соединения трех отношений **S join SP join P**:

Отношение **S JOIN SP JOIN P**

Номер студента SNUM	Фамилия студента SNAME	Код предмета PCOD	Название предмета PNAME	Средний балл по предмету SRBALL
6	Иванов	101	Физика	4,5
6	Иванов	102	Математика	4
6	Иванов	103	Информатика	5
17	Петров	101	Физика	3,5
17	Петров	102	Математика	4
19	Сидоров	101	Физика	4,5

Операции соединения

Операция внешнего соединения

используется при соединении двух отношений, столбцы которых имеют несовпадающие значения.

Внешнее соединение : *левое и правое.*

Левое внешнее соединение $R \supset \boxtimes S$:

кортежи отношения R , не имеющие совпадающих значений в общих столбцах отношения S , также включаются в результирующее отношение.

Обозначения отсутствующих значений во втором отношении - определитель NULL.

Операции соединения

Пример левого внешнего соединения

Примечание: студент может принимать участие в олимпиадах по предметам, установленный общий балл которых не больше среднего бала студента .

Задание: на основе отношений R и S создать список, в котором указаны студенты и предметы, по которым они участвуют в олимпиадах.

Отношение R

Номер студента	Фамилия	Средний бал
6	Иванов	5
17	Петров	3
19	Сидоров	4

Операции соединения

Отношение S

Код предмета	Название	Общий балл
101	Физика	4,5
102	Химия	4

Таблица $((\Pi (R)) \supseteq S)$

Номер студента	Фамилия	Средний балл	Код предмета	Название	Общий балл
6	Иванов	5	101	Физика	4,5
6	Иванов	5	102	Химия	4
17	Петров	3	NULL	NULL	NULL
19	Сидоров	4	102	Химия	4

Операции соединения

Правое внешнее соединение:

в результирующем отношении содержатся все кортежи правого отношения.

Полное внешнее соединение:

в результирующее отношение помещаются все кортежи из обоих отношений и для обозначения несовпадающих значений кортежей используются определители NULL.

Операции соединения

Операция полусоединения $R \bowtie_F S$:

определяет отношение, которое содержит те кортежи отношения R , которые входят в соединение отношений R и S .

Формулировка операции полусоединения с помощью операторов проекции и соединения:

$$R \bowtie_F S = \Pi_A (R \bowtie S)$$

где A - набор всех атрибутов в отношении R .

Операции соединения

Пример операции полусоединения

Номер студента	Фамилия	Средний балл	Код предмета	Название	Общий балл
6	Иванов	5	101	Физика	4,5
6	Иванов	5	102	Химия	4
19	Сидоров	4	102	Химия	4

Операция деления

Пусть:

- отношение R определено на множестве атрибутов A ;
- отношение S — на множестве атрибутов B ;
- $B \subseteq A$;
- $C = A - B$ (C является множеством атрибутов отношения R , которые не являются атрибутами отношения S).

Результат деления $R \div S$ - набор кортежей отношения R , определенных на множестве атрибутов C , которые соответствуют комбинации всех кортежей отношения S .

Операция деления

Пример операции деления

Отношение R

Номер группы	Количество студентов	ФИО куратора	Код предмета	Название предмета
ТМ-31	20	Иванов	01	Математика
ТМ-32	22	Петров	01	Математика
ТИ-31	13	Сидоров	01	Математика
ТМ-31	20	Иванов	02	Физика
ТМ-32	22	Петров	02	Физика

Операция деления

Отношение S

Номер группы	Количество студентов	ФИО куратора
ТМ-31	20	Иванов
ТМ-32	22	Петров
ТИ-31	13	Сидоров

T1:Select'Код предмета', 'Название предмета' from R

Код предмета	Название предмета
01	Математика
02	Физика

Операция деления

T21:Select * from T1,S

Код предмета	Название предмета	Номер группы	Количество студентов	ФИО куратора
01	Математика	ТМ-31	20	Иванов
01	Математика	ТМ-32	22	Петров
01	Математика	ТИ-31	13	Сидоров
02	Физика	ТМ-31	20	Иванов
02	Физика	ТМ-32	22	Петров
02	Физика	ТИ-31	13	Сидоров

Операция деления

T22: (Select * from T21) except (Select * from R)

Код предмета	Название предмета	Номер группы	Количество студентов	ФИО куратора
02	Физика	ТИ-31	13	Сидоров

T2: Select 'Код предмета', 'Название предмета' from T22

Код предмета	Название предмета
02	Физика

R=T1-T2

Код предмета	Название предмета
01	Математика

Реляционное исчисление

Происхождение названия “*реляционное исчисление*”: от части символической логики, которая называется исчислением предикатов.

Реляционное исчисление существует в двух формах:

- реляционного исчисления кортежей;
- реляционного исчисления доменов.

Реляционное исчисление

Предикат в логике первого порядка — истинностная функция с параметрами.

Суждение — выражение, которое принимает функция после подстановки значений вместо параметров.

Суждение: *истинное* и *ложное*.

Пусть: P - предикат;
 x - переменная.

Тогда: $\{x \mid P(x)\}$ - множество всех значений x , при которых суждение P — истина.

Предикаты могут соединяться с помощью логических операторов: \wedge (AND), \vee (OR) и \neg (NOT) с образованием составных предикатов.

Реляционное исчисление кортежей

Задача реляционного исчисления кортежей:
*нахождение кортежей, для которых предикат
является истинным.*

Исчисление основано на переменных кортежа.

Переменными кортежа - переменные, областью
определения которых является указанное отношение.

Реляционное исчисление кортежей

Пример

Запрос:

“Выбрать атрибуты № склада, адрес,идент. код, дата,ФИО заказчика для заказов с количеством >60”

Запись запроса:

$$\{ S \mid R(S) \wedge S.\text{количество} > 60 \}$$

Пояснение:

Выражение “S.количество часов” - значение атрибута количество часов для кортежа.

Реляционное исчисление кортежей

Два типа кванторов, используемых для указания количества экземпляров, к которым должен быть применен предикат:

- **квантор существования** \exists (символ “существует”): используется в формуле, которая должна быть истинной хотя бы для одного экземпляра;
- **квантор общности** \forall (символ “для всех”): используется в выражениях, которые относятся ко всем экземплярам.

Реляционное исчисление кортежей

Пример применения квантора существования

$\text{Студент}(S) \wedge \{Зв\} (\text{Год рождения}(V) \wedge (V.\text{имя}=S.\text{имя}) \wedge V.\text{группа}='ТИ-31')$

Выражение означает: в отношении Год рождения существует кортеж, который имеет такое же значение атрибута имя, что и значение атрибута имя в текущем кортеже S из отношения Студент, а атрибут группа из кортежа V имеет значение 'ТИ-31'.

Пример применения квантора общности

$\forall (V) (V.\text{группа} \neq 'ТИ-31')$

Выражение означает: ни в одном кортеже отношения Год рождения значение атрибута группа не равно 'ТИ-31'.

Реляционное исчисление кортежей

Свободные переменные - переменные кортежа, которые не квалифицируются кванторами, в противном случае они называются **связанными переменными**.

В реляционном исчислении допустимые формулы – только недвусмысленные и небесмысленные последовательности.

Реляционное исчисление кортежей

Правила построения формулы в исчислении предикатов:

- Если P - n -арная формула (предикат с n аргументами),
 t_1, t_2, \dots, t_n - константы или переменные,
то $P(t_1, t_2, \dots, t_n)$ - правильно построенная формула.
- Если t и t_2 - константы или переменные из одного домена,
 \ominus - один из операторов сравнения ($<$, \leq , $>$, \geq , \neq),
то $t_1 \ominus t_2$ - правильно построенная формула.
- Если F_1, F_2 - формулы,
то $F_1 \wedge F_2$ - конъюнкция формул,
 $F_1 \vee F_2$ - дизъюнкция,
 \overline{F} - отрицание.
- Если F_1 - формула со свободной переменной X ,
то $\exists F(X)$ и $\forall F(X)$ - также формулы.

Реляционное исчисление доменов

Значения переменных, используемых в **реляционном исчислении доменов** берутся из доменов, а не из кортежей отношений.

Путь: $P(d_1, d_2, \dots, d_n)$ - предикат;
 d_1, d_2, \dots, d_n - переменные.

Тогда: $\{d_1, d_2, \dots, d_n \mid P(d_1, d_2, \dots, d_n)\}$ - множество всех переменных домена, для которых предикат - истина.

Выражение $R(x, y)$ - **истинное** тогда и только тогда, когда в отношении R имеется кортеж со значениями x и y в двух его атрибутах.

Реляционное исчисление доменов

Пример

Найти: имена всех менеджеров, зарплата которых превышает 2500 гривен.

{Имя, Фамилия \exists должность, \exists зарплата
((фамилия, должность, зарплата) \wedge должность =
'менеджер' \wedge зарплата > 2500)}

Другие языки

Дополнительные категории реляционных языков:

- на основе преобразований;
- графические языки;
- языки четвертого поколения.

Языки на основе преобразований -

класс непроцедурных языков

Используют отношения для преобразования исходных данных к требуемому виду

(примеры: SQUARE , SEQUEL и его версии, SQL).

Другие языки

Графические языки - *рисунок или другое графическое отображение структуры отношения.*

Пользователь создает некий образец желаемого результата, и система возвращает затребованные данные в указанном формате (пример: QBE).

Языки четвертого поколения:

- создают полностью готовое и соответствующее требованиям заказчика прикладное приложение с помощью ограниченного набора команд;
 - предоставляют дружественную по отношению к пользователю среду разработки.
-