

Введение в язык SQL

Схема взаимодействия клиента и СУБД



Язык SQL является стандартным средством взаимодействия с БД через реляционную СУБД, промежуточным звеном между БД и пользователем (или прикладной программой).

Особенности языка SQL

- Язык SQL не процедурный (декларативный)
- Один оператор SQL обрабатывает одновременно весь набор элементов
- Использует трехзначную логику (3VL)

Происхождение названия языка SQL

SQL - structured query language
язык структурированных запросов

Где живут наши клиенты?

```
SELECT город, регион FROM клиенты
```

Декларативный и процедурный языки программирования

SELECT

город, регион

FROM

КЛИЕНТЫ

```
Sub GetCustomerNames()  
    Dim db As Database  
    Dim rs As Recordset  
    Dim i As Integer  
    Dim address() As String  
    Set db = CurrentDb()  
    Set rs = db.OpenRecordset("КЛИЕНТЫ")  
    rs.MoveLast  
    ReDim address (rs.RecordCount - 1, 1)  
    With rs  
        .MoveFirst  
        i = 0  
        Do Until .EOF  
            address (i,0) = ![город]  
            address (i,1) = ![регион]  
            i = i + 1  
            .MoveNext  
        Loop  
    End With  
    rs.Close  
    db.Close  
End Sub
```

Формы использования SQL: встроенный и интерактивный

Способы встраивания SQL в прикладные программы

1) *Внедренные SQL-запросы*

Отдельные SQL-запросы внедряются прямо в исходный текст программы и смешиваются с операторами базового языка. Этот подход позволяет создавать программы, обращающиеся непосредственно к БД. Специальные программы-предкомпиляторы преобразуют исходный текст с целью замены SQL-запросов соответствующими вызовами подпрограмм СУБД, затем текст компилируется и собирается обычным способом.

2) *Использование прикладного интерфейса программирования* (Application Programming Interface – API)

Работа с БД реализуется через предоставляемый набор функций. Вызывая эти функции, программа передает СУБД операторы SQL и получает обратно результаты запросов. В этом случае не требуется специализированный препроцессор.

Пример включения операторов SQL в исходный код приложения, написанного на процедурном языке:

```
Sub GetAuthorNames2()  
    Dim db As Database  
    Dim rs As Recordset  
    Set db = CurrentDb()  
    Set rs = db.OpenRecordset("SELECT имя, фамилия FROM авторы;")  
    ' Далее программа обрабатывает результаты  
    ' запроса, находящиеся в переменной rs.  
    rs.Close  
    db.Close  
End Sub
```

SQL используется для реализации всех функций СУБД:

- **организация данных** (определяет и изменяет структуры представления данных и устанавливает отношения между элементами БД); □
- **чтение данных** (предоставляет пользователю или приложению возможность читать из БД содержащуюся в ней информацию и пользоваться ею); □
- **обработка данных** (дает возможность изменять базу данных, т.е. добавлять новые данные, удалять или обновлять уже имеющиеся); □
- **управление доступом** (позволяет задавать необходимые возможности пользователя по чтению и изменению данных, а также защищать их от несанкционированного доступа); □
- **совместное использование данных** (координирует совместное использование данных пользователями, работающими параллельно, чтобы они не мешали друг другу); □
- **целостность данных** (обеспечивает целостность БД, защищая ее от разрушения из-за несогласованных изменений или отказа системы).

SQL – это неотъемлемая часть СУБД, инструмент, с помощью которого программируется взаимодействие клиентского приложения с реляционной БД.

Категории операторов языка SQL

- Язык определения данных
(Data Definition Language или DDL)
 - создание объектов баз данных
CREATE SCHEMA, CREATE TABLE, CREATE VIEW,
CREATE DOMAIN и др..
- Язык манипулирования данными
(Data Manipulation Language или DML)
 - запись, изменение и поиск данных.
SELECT, INSERT, UPDATE и DELETE.
- Операторы управления данными
(Data Control Statements).
 - определение прав пользователей на операции с объектами базы данных. (Часто они рассматриваются как часть DDL.)
GRANT, REVOKE.

Дополнительные категории операторов языка SQL

- операторы для определения ограничений целостности и триггеров;
- средства определения представлений БД;
- средства управления транзакциями.

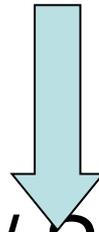
Примеры реляционных СУБД

- IBM DB2
- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- SQLite
- Microsoft Access

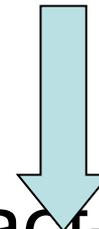
Расширенные языки некоторых СУБД

СУБД:

Microsoft Oracle
SQL Server



PL/SQL



Transact-SQL

Языки СУБД:

История версий стандарта языка SQL

SQL-86 (1986/1987 гг.);

SQL-89 (1989 г.; фактически SQL-86 с малыми добавками);

SQL-92 (1992 г.; синонимы: **SQL2**, SQL92);

SQL:1999 (синонимы: **SQL3**, SQL-99;

главные дополнения — объектные возможности языка, хранимые процедуры, поддержка регулярных выражений, рекурсивных запросов, поддержка триггеров, базовые процедурные расширения, неscalaрные типы данных и некоторые объектно-ориентированные возможности):

SQL:2003 (Введены расширения для работы с XML-данными, оконные функции, применяемые для работы с OLAP-базами данных, генераторы последовательностей и основанные на них типы данных).

С этого момента очередные версии стандарта выходят не в полном объеме, как прежде, а в виде "дополнительных частей"

SQL:2006 (развивает расширения, касающиеся XML, возникшие в SQL:2003)

SQL:2008 (Улучшены возможности оконных функций; устранены некоторые неоднозначности стандарта SQL:2003; введен ряд частных дополнений, как, например, TRUNCATE TABLE или выдача первых N записей)

На основе SQL-92 построен ГОСТ Р ИСО/МЭК 9075-93.

Разделы стандарта SQL2

- Раздел 1: **"Область применения"**;
- Раздел 2: **"Нормативные ссылки"**
 - перечень дополнительных стандартов;
- Раздел 3: **"Определения, обозначения и соглашения"**;
- Раздел 4: **"Понятия"**
 - базовые понятия, используемые при определении языка;
- Раздел 5: **"Лексические элементы"**;
- Раздел 6: **"Скалярные выражения"**
 - определение элементов языка, производящих скалярные значения;
- Раздел 7: **"Выражения запросов"**
 - определение элементов языка, производящих строки и таблицы данных;
- Раздел 8: **"Предикаты"**;

Разделы стандарта SQL2 (продолжение)

- Раздел 9: **"Правила присваивания данных"**
 - правила, определяющие порядок выборки данных из базы данных или сохранения данных в базе данных, и правила формирования операций над множествами;
- Раздел 10: **"Дополнительные общие элементы"**
 - дополнительные общие языковые элементы, используемые в разных частях языка;
- Раздел 11: **"Определение схемы и манипулирование схемой"**
 - определение средств создания схемы и управления схемой;
- Раздел 12: **"Модуль"**
 - определение модулей и процедур;
- Раздел 13: **"Манипулирование данными"**
 - определение операторов манипулирования данными;
- Раздел 14: **"Управление транзакциями";**
- Раздел 15: **"Управление подключениями";**
- Раздел 16: **"Управление сессиями";**

Разделы стандарта SQL2 (продолжение)

- Раздел 17: **"Динамический SQL"**
 - определение средств, предназначенных для динамического выполнения операторов SQL;
- Раздел 18: **"Управление диагностикой"**
- Раздел 19: **"Встроенный SQL"**
 - определение синтаксиса встраивания SQL в некоторые стандартные языки программирования;
- Раздел 20: **"Прямой вызов SQL"**
 - определение подмножества SQL/92, которое может использоваться в режиме прямого вызова;
- Раздел 21: **"Информационная схема и схема определений"**
 - определение представляемых таблиц, содержащих информацию о схеме;
- Раздел 22: **"Коды статуса"**
 - определение значений, идентифицирующих статус завершения SQL-операторов и механизмов, посредством которых эти значения возвращаются;
- Раздел 23: **"Соответствие"**
 - определение критериев соответствия языку SQL/92;

Приложения к стандарту SQL2 (продолжение)

- Приложение А: **"Уровни языка SQL"** - перечисление правил, определяющих подмножества вводного SQL (EntrySQL) и промежуточного SQL (IntermediateSQL) стандарта SQL/92;
- Приложение В: **"Элементы, определяемые в реализации"** - перечисление средств, для которых в стандарте установлено, что синтаксис, или смысл, или результат воздействия на базу данных частично или полностью определяется в реализации, и описание определяющей информации, которая должна быть обеспечена в реализации для каждого случая;
- Приложение С: **"Элементы, зависимые от реализации"** - перечисление средств, для которых в стандарте явно установлено, что их смысл или результат воздействия на базу данных зависит от реализации;
- Приложение D: **"Неодобряемые свойства"** - перечисление свойств, которые разработчики стандарта не хотели бы видеть в его следующей версии;
- Приложение Е: **"Несовместимости с SQL/89"** - перечисление несовместимостей между SQL/92 и SQL/89;
- Приложение F: **"Поддержка и интерпретация SQL"** - разъясняются интерпретации SQL и коррективы, которые были внесены после принятия SQL/89.

Свойства реляционной структуры данных

В реляционной БД данные представлены в виде совокупности нескольких таблиц

Все данные в реляционной БД содержатся в таблицах. Таблицы представляют собой совокупность каких-либо сведений об объектах, явлениях, процессах реального мира.

Таблицы содержат:

Строки. Каждая строка (или запись) представляет собой совокупность атрибутов (свойств) конкретного экземпляра объекта.

Столбцы. Каждый столбец (поле) представляет собой атрибут или совокупность атрибутов.

Поле строки является минимальным элементом таблицы. Каждый столбец в таблице имеет определенное имя, тип данных и размер.

Свойства таблицы

- Является структурным элементом, который содержит данные
- Имеет имя, уникальное внутри БД
- Содержит данные, соответствующие одному объектному типу (типу сущностей)
- Содержит не менее одного столбца
- Может не содержать ни одной строки
- Содержит на пересечении столбца и строки *значение*
- Каждый столбец имеет имя и набор свойств



Пример таблицы из базы данных

Народные песни

Название песни	Жанр	Шифр сборника	Номер в сборнике
Мы с подружкой бойкие	Танец	Ч-61	20
Я сидел на бережку	Лирика	Пч	8
Пишет милый письмецо	Свадебные песни	Ч-56	20
Не ходи да не ходи	Частушки	Ч-56	65
Балалаечка-струна	Эпос	Ч-56	73
Распрекрасненькое	Детство	Лг-58	13
Поехал князь Михайло	Причитания	БС-2	19

Связь таблиц

Таблицы связаны с помощью первичных и внешних ключей

Песня

Название песни	Жанр	Номер в сборнике	Шифр сборника
Мы с подружкой бойкие	Ч	С.20	Ч-61
Я сидел на бережку	Т	8.	Пч
Пишет милый письмецо	Ч	С.20	Ч-56
Не ходи да не ходи	Ч	С.65	Ч-56
Балалаечка-струна	Ч	С.73	Ч-56
Распрекрасненькое	Л	13.	Лг-58
Летал голубь при долине	Л	22.	Лг-58
Поехал князь Михайло	Э	19.	БС-2

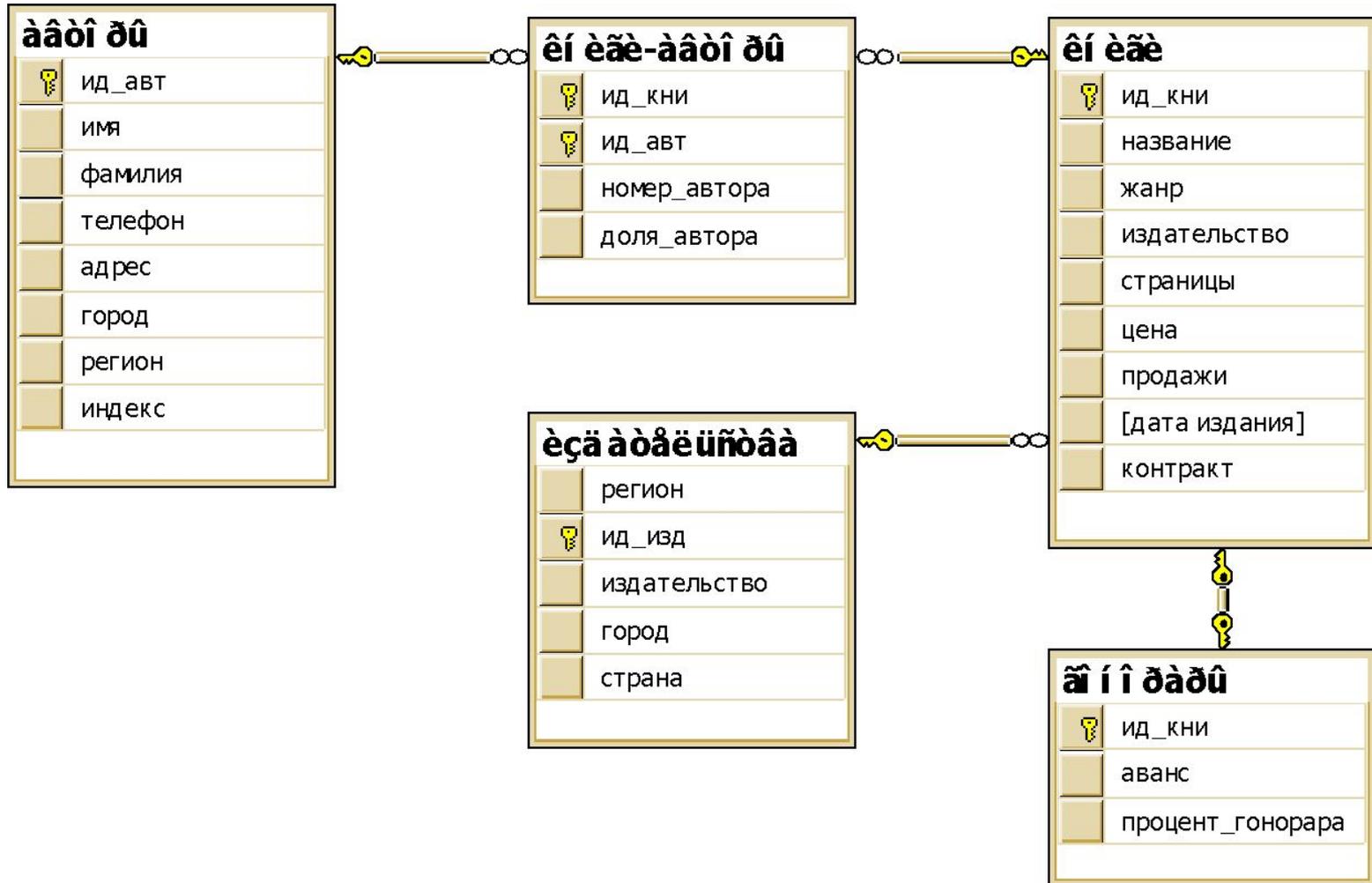
Сборник

Шифр библио	Библиография
Пс-58	Псковские песни /Сост. Н.Л.Котикова. –Пск
Ч-61	Русские частушки, страдания, припевки /Со
Пч	Песни Печоры /Сост. Н.П.Колпакова, Ф.В.С
Ч-56	Русские частушки /Сост. Н.Л.Котикова. –б.о
Лг-58	Рубцов Ф. Народные песни Ленинградской
БС-2	Былины Севера. Т.2. Прионежье, Пинега и П

Указатель жанров

Шифр жанра	Название жанра
Т	Танец
Л	Лирика
С	Свадебные песни
Ч	Частушки
Э	Эпос
Д	Детство
К	Календарно-обрядовые песни
П	Причитания

Схема БД книги_ру



Соответствие терминов

Модель данных	SQL	Файловая система
Отношение	Таблица	Файл
Атрибут	Столбец	Поле
Кортеж	Строка	Запись

Свойства атрибута (столбца)

- Столбец представляет атрибут (свойство) сущности
- Для каждого столбца определен свой домен
- Столбец содержит однотипные данные
- Порядок следования столбцов не имеет значения

Свойства кортежа (строки)

- Строка содержит свойства (атрибуты) одного экземпляра сущности
- Порядок следования строк не имеет значения
- Нет одинаковых строк (хотя бы в одном столбце значения различаются)
- Каждая строка однозначно определяется *первичным ключом*

Свойства значения

- Все значения атомарны (неделимы)
- Значение может отсутствовать; отсутствие значения обозначается NULL

Типы данных в SQL

Таблица КНИГИ

ид_кни	Название	Жанр	Изда- тель- ство	Стра- ницы	Цена	Продажи	Дата издания
K01	1977!	история	И01	107	21,99	566	01.08.2000
K02	История Руси	история	И03	14	19,95	9566	01.04.1998
K03	Домашний компьютер	компьютер	И02	1226	39,95	25667	01.09.2000
K04	Хозяйка Четырех Стихий	фантастика	И01	510	12,99	13001	31.05.1999
K05	Общая психология	психология	И01	201	6,95	201440	01.01.2001
K06	Биография американца	биография	И01	473	19,95	11320	31.07.2000
K07	Нарочно не придумаешь	детектив	И03	333	23,95	1500200	01.10.1999
K08	Кенгуру	детские	И01	86	10	4095	01.06.2001
K09	Зеркало зла	детские	И01	22	13,95	5000	31.05.2002
K10	Парк грез	фантастика	И01	NULL	NULL	NULL	NULL
K11	Деловая психология	психология	И01	826	7,99	94123	30.11.2000
K12	Родник жемчужин	поэзия	И01	507	12,99	100001	31.08.2000
K13	История крестовых походов	история	И03	802	29,99	10467	31.05.1999
K14	200 Years of German Humor	история	И05	14	19,95	9566	01.04.1998

Типы данных в SQL

У каждого столбца, локальной переменной, выражения и параметра есть определенный тип данных.

Тип данных — атрибут, определяющий, какого рода данные могут храниться в объекте:

- целые числа,
- символы,
- данные денежного типа,
- метки времени и даты,
- двоичные строки
- ...

Категории типов данных

Тип данных	Хранит данные
Character string	Строки символов
Bit string	Строки битов
Exact numeric	Целые и действительные числа с плавающей десятичной точкой
Approximate numeric	Числа с плавающей точкой
Datetime	Значения даты и времени
Interval	Интервалы даты и времени

Примеры литералов

Числовые: 40 , 12.34

Символьные: '40' , 'история'

Даты и времени : '2002-05-10' , '09:45:00'
'2000.31.08 0:00:00'

Строковые типы данных

Тип	Описание
CHARACTER	Строка фиксированной длины
CHARACTER VARYING	Строка переменной длины
NATIONAL CHARACTER	Совпадает с типом CHARACTER, только в столбце этого типа можно хранить лишь стандартизованные многобайтовые или двухбайтовые знаки (Unicode)
NATIONAL CHARACTER VARYING	Совпадает с типом CHARACTER VARYING, только в столбце этого типа можно хранить лишь стандартизованные многобайтовые или двухбайтовые знаки (Unicode)

Строковые типы данных в некоторых СУБД

- Microsoft Access - text, memo;
- Microsoft SQL Server - varchar, text, nchar, nvarchar, ntext;
- Oracle - char, varchar, varchar2, nchar, nvarchar, nvarchar2;
- MySQL - char, varchar, nchar, nvarchar, text, tinytext, mediumtext, longtext;

Битовые типы данных

Тип	Описание	Пример
BIT	Строка фиксированной длины	B'01001011'
BIT VARYING	Строка переменной длины	B'0101'

Битовые типы данных в некоторых СУБД

- Microsoft Access - yes, no, binary, OLE object;
- Microsoft SQL Server - binary, varbinary, image;
- Oracle - raw, long raw, blob, bfile;
- MySQL - blob, tinyblob, mediumblob, longblob;

Точные числовые типы данных

Тип	Описание
NUMERIC	произвольное рациональное число, хранимое в столбце, который определяется как NUMERIC(точность [, масштаб])
DECIMAL	Аналогичен типу NUMERIC
INTEGER	произвольное целое число
SMALLINT	Аналогичен типу INTEGER

Точные числовые типы данных в некоторых СУБД

- **Microsoft Access** - decimal, integer, byte, long integer;
- **Microsoft SQL Server** - numeric, decimal, integer, smallint, bigint, tinyint ;
- **Oracle** - numeric, decimal, integer, smallint, number ;
- **MySQL** - numeric, decimal, integer, smallint, bigint, mediumint, tinyint ;

Действительные числовые типы данных

Тип	Описание
FLOAT	произвольное рациональное приближение действительного числа с плавающей точкой FLOAT(precision)
REAL	совпадает с типом FLOAT; точность автоматически определяет СУБД
DOUBLE PRECISION	совпадает с типом FLOAT; точность автоматически определяет СУБД

Действительные числовые типы данных в некоторых СУБД

- Microsoft Access - single, double ;
- Microsoft SQL Server - float, real ;
- Oracle - float, real, double precision, number ;
- MySQL - float, real, double precision ;

Календарные типы данных

Тип данных	Описание
DATE	yyyy-mm-dd
TIME	hh:mm:ss
TIMESTAMP	yyyy-mm-dd hh:mm:ss
TIME WITH TIME ZONE	эквивалентен типу time, его формат включает дополнительное поле TIME_ZONE_OFFSET
TIMESTAMP WITH TIME ZONE	эквивалентен типу TIMESTAMP, его формат содержит дополнительное поле TIME_ZONE_OFFSET

Поля типа datetime и их допустимые значения

Поле	Интервал допустимых значений
YEAR	от 0001 до 9999
MONTH	от 01 до 12
DAI	от 01 до 31
HOUR	от 00 до 23
MINUTE	от 00 до 59
SECOND	от 00 до 61.999...
TIME_ZONE_OFFSET	от -12:59 до +13:00

Календарные типы данных в некоторых СУБД

- Microsoft Access - date/time ;
- Microsoft SQL Server - datetime, smalldatetime ;
- Oracle - date, timestamp ;
- MySQL - date, time, datetime, timestamp ;

Интервальные типы данных

Тип	Описание	Допустимые типы столбцов
Year-month	некое значение в годах, или в месяцах, или в годах и месяцах	INTERVAL YEAR, INTERVAL YEAR(точность), INTERVAL MONTH, INTERVAL MONTH(точность), INTERVAL YEAR TO MONTH, INTERVAL YEAR(точность) TO MONTH
Day-time	значения, выраженные в некой комбинации дней, часов, минут и секунд	INTERVAL MINUTE, INTERVAL DAY(точность), INTERVAL DAY TO HOUR, INTERVAL DAY(точность) TO SECOND, INTERVAL MINUTE(точность) TO SECOND (дробная точность)

Интервальные типы данных в некоторых СУБД

- Microsoft Access - не поддерживает ;
- Microsoft SQL Server - не поддерживает ;
- Oracle - Interval ;
- MySQL - не поддерживает ;

Скалярные выражения

- числовые
- строковые
- даты и времени
(в т.ч. интервальные)

Таблица КНИГИ

ид_кни	Название	Жанр	Изда- тель- ство	Стра- ницы	Цена	Продажи	Дата издания
K01	1977!	история	И01	107	21,99	566	01.08.2000
K02	История Руси	история	И03	14	19,95	9566	01.04.1998
K03	Домашний компьютер	компьютер	И02	1226	39,95	25667	01.09.2000
K04	Хозяйка Четырех Стихий	фантастика	И01	510	12,99	13001	31.05.1999
K05	Общая психология	психология	И01	201	6,95	201440	01.01.2001
K06	Биография американца	биография	И01	473	19,95	11320	31.07.2000
K07	Нарочно не придумаешь	детектив	И03	333	23,95	1500200	01.10.1999
K08	Кенгуру	детские	И01	86	10	4095	01.06.2001
K09	Зеркало зла	детские	И01	22	13,95	5000	31.05.2002
K10	Парк грез	фантастика	И01	NULL	NULL	NULL	NULL
K11	Деловая психология	психология	И01	826	7,99	94123	30.11.2000
K12	Родник жемчужин	поэзия	И01	507	12,99	100001	31.08.2000
K13	История крестовых походов	история	И03	802	29,99	10467	31.05.1999
K14	200 Years of German Humor	история	И05	14	19,95	9566	01.04.1998

Типы выражений

Выражение		Пример
Case	(Выбор)	CASE WHEN n <> 0 THEN x/n ELSE 0 END
Cast	(Преобразование типов)	CAST(дата_издания AS CHARACTER)
Datetime	(Дата или время)	время_старта + '01:30'
Interval	(Интервал между датами или между отметками времени суток)	INTERVAL '7' DAY*2
Numeric	(Числовой)	(продажи*цена) / 12
String	(Строковый)	'Dear ' фамилия '!'

Категории типов данных (Transact-SQL)

Точные числа

Приблизительные числа

Дата и время

Символьные строки

Символьные строки в Юникоде

Двоичные данные

Прочие типы данных

Типы данных больших значений:
varchar(max) и **nvarchar(max)**

Типы данных больших объектов: **text**,
ntext, **image**, **varbinary(max)** и **xml**

Типы данных (Transact-SQL)

Точные числа

bigint

bit

decimal

int

money

numeric

smallint

smallmoney

tinyint

Приблизительные числа

float

Real

Дата и время

date

datetime2

datetime

datetimeoffset

smalldatetime

Time

Типы данных (Transact-SQL)

Символьные строки

[char](#);

[text](#)

[varchar](#)

Символьные строки в Юникоде

[nchar](#)

[ntext](#)

[nvarchar](#)

Двоичные данные

[binary](#)

[image](#)

[varbinary](#)

Прочие типы данных

[uniqueidentifier](#)

[hierarchyid](#)

[sql_variant](#)

[Типы пространственной геометрии](#)

[Типы пространственной географии](#)

[rowversion](#)

[cursor](#)

[xml](#)

[table](#)

Тип данных выражения (Transact-SQL)

При объединении одним оператором двух выражений с разными типами данных, параметрами сортировки, точностями, масштабами или длинами, результат определяется следующим образом:

- Тип данных результата определяется применением правил очередности типов данных к входным выражениям.

[Приоритет типов данных \(Transact-SQL\)](#)

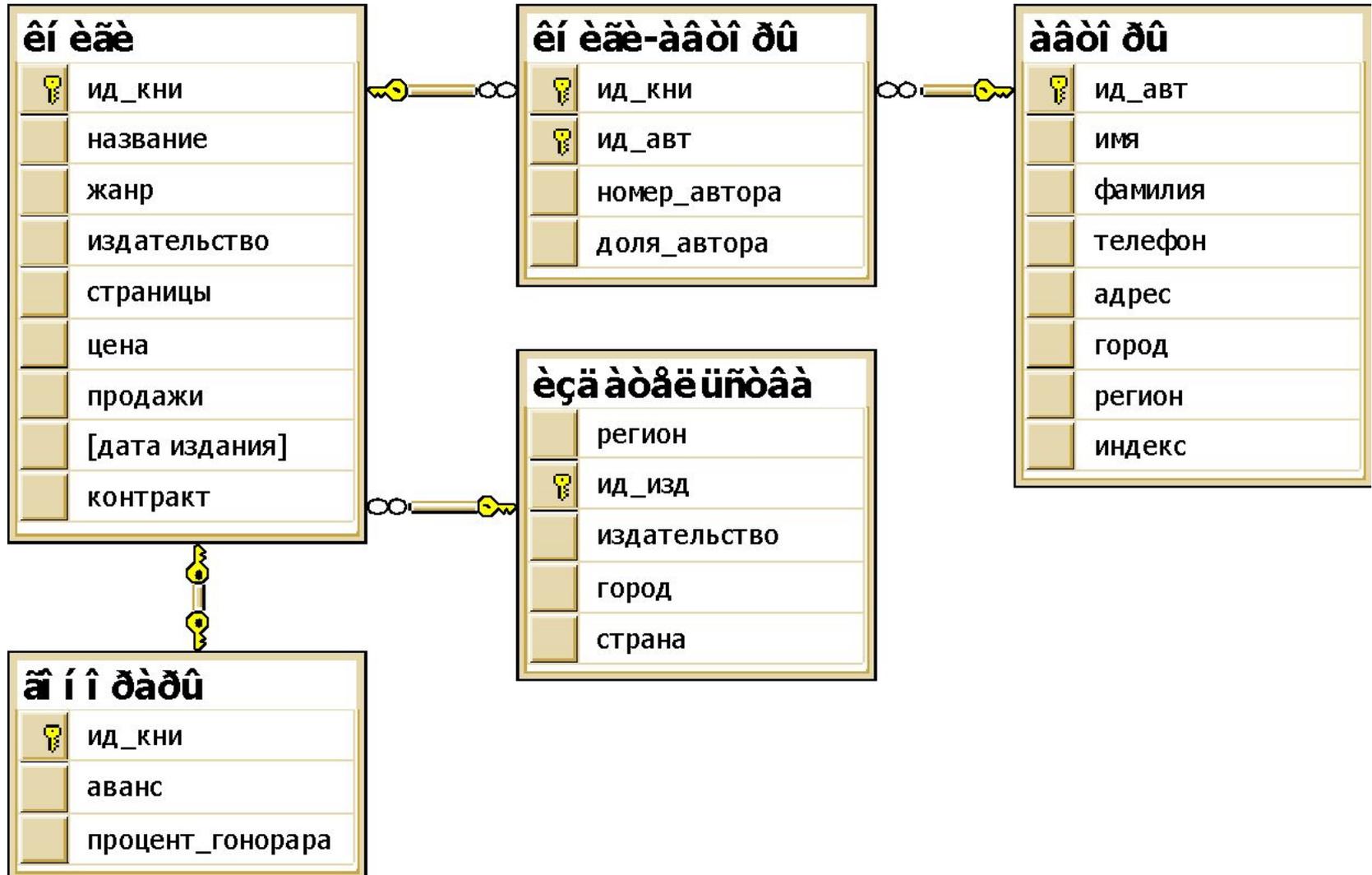
- Параметры сортировки результата определяются правилами очередности параметров сортировки, если тип данных результата — **char**, **varchar**, **text**, **nchar**, **nvarchar** или **ntext**.

[Очередность параметров сортировки \(Transact-SQL\)](#)

- Точность, масштаб и длина результата зависят от точности, масштаба и длины входных выражений

[Точность, масштаб и длина \(Transact-SQL\)](#)

Схема БД книги_ру



Пример команды SQL с комментарием

– – Показать авторов, проживающих в Москве

```
SELECT фамилия, имя  
FROM авторы  
WHERE регион = 'Москва'  
ORDER BY фамилия ;
```

Результат выполнения запроса

фамилия	имя
Савельева	Наталья
Щербинин	Владимир

Пример команды SQL с комментарием

-- Показать авторов, проживающих в Москве

SELECT фамилия, имя

FROM авторы

WHERE регион = 'Москва'

ORDER BY фамилия ;

комментарий

предложения

ключевые
слова

имена
переменных

завершающая
точка с запятой

Форма записи команды SQL

select ид_авт

, фамилия

FROM

авторы WHERE регион = 'Санкт-Петербург' order

by фамилия

;

SELECT ид_авт, фамилия

FROM авторы

WHERE регион = 'Санкт-Петербург'

ORDER BY фамилия

;

Соответствие терминов

Модель данных	SQL	Файловая система
Отношение	Таблица	Файл
Атрибут	Столбец	Поле
Кортеж	Строка	Запись