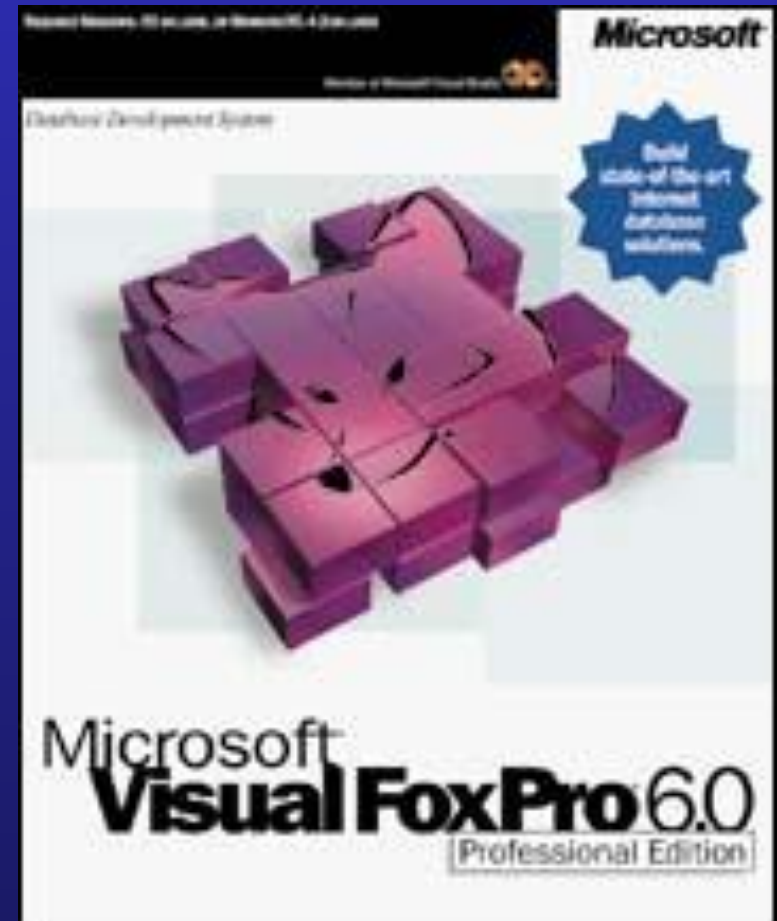


База данных Visual FoxPro 6.0

(основные понятия
и инструментальные средства среды разработки)



Что

- SQL (Structured Query Language) - язык структурированных запросов: является универсальным языком для создания, модификаций и управления данными в реляционных базах данных.

- Историческая справка:

- предложен в 1970 И.Ф. Коддом (E.F.Codd) в исследовательской лаборатории IBM
- начало 80-х Oracle Relational Technology создаёт СУБД на основе SQL
- в 1989 уже существует более 25 SQL-подобных СУБД
- в 1983 выходит ANSI-стандарт (*American National Standards Institute*) SQL, а в 1992 расширенный стандарт (SQL2 или SQL-92). Все промышленные версии стараются поддерживать ANSI-SQL, но не обеспечивая его полностью, однако, каждая из них имеет свои расширения, стремясь покрыть требуемые функциональные возможности.

Общие требования к реляционной СУБД

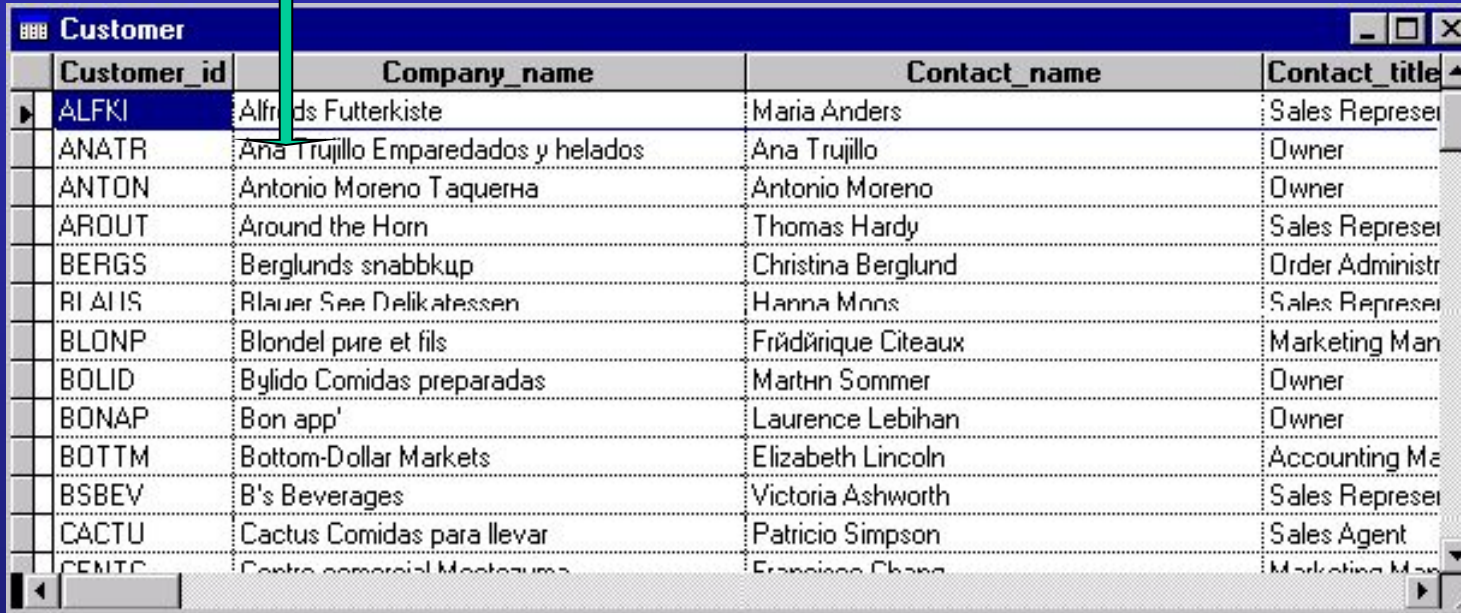
- Вся информация представлена в виде таблиц.
- Поддерживать логическую структуру данных независимо от их физического представления.
- Использовать язык высокого уровня для работы со структурой данных, получения данных и их изменения.
- Обеспечивать теоретико-множественные операции над данными, такие как объединение, пересечение и дополнение.
- Поддерживать виртуальные таблицы, как альтернативный способ получения данных.
- Позволять различать неопределённые (пропущенные) данные.
- Обеспечивать механизмы для
 - поддержки реляционной целостности,
 - авторизованного доступа,
 - транзакционности изменений,
 - восстановления данных.

Данные в реляционной СУБД

Вся информация представлена в виде таблиц:

Отношение (relation), таблица (table), файл (file)

Ячейка (cell), адрес (address), значение атрибута (attribute value)



Customer_id	Company_name	Contact_name	Contact_title
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Represent
ANATR	Ana Trujillo Emparedados y helados	Ana Trujillo	Owner
ANTON	Antonio Moreno Taqueria	Antonio Moreno	Owner
AROUT	Around the Horn	Thomas Hardy	Sales Represent
BERGS	Berglunds snabbkup	Christina Berglund	Order Administr
RIADIS	Riauer See Delikatessen	Hanna Moos	Sales Represent
BLOMP	Blondel pure et fins	Frdrique Citeaux	Marketing Man
BOLID	Bolido Comidas preparadas	Martin Sommer	Owner
BONAP	Bon app'	Laurence Lebihan	Owner
BOTTM	Bottom-Dollar Markets	Elizabeth Lincoln	Accounting Ma
BSBEV	B's Beverages	Victoria Ashworth	Sales Represent
CACTU	Cactus Comidas para llevar	Patricio Simpson	Sales Agent
CEMTC	Centro comercial Mactrums	Francisco Chang	Marketing Man

Кортеж (tuple), строка (row), запись (record)

Атрибут (attribute), столбец (column), поле (field)

Структура СУБД

- В реляционных СУБД обычно существует два типа таблиц:
 - *пользовательские таблицы (user tables)*
 - *системные таблицы (system tables)*.

В VFP системные таблицы в виде одной единственной таблицы-файла с расширением dbc. Обычно также системные таблицы объединяются в системную базу данных (в MS SQL Server например, это база *Master*)

- Системные таблицы (известные также под названием *системный каталог [system catalog]*) содержат описания баз(ы) данных. Доступ к этим таблицам обычно открыт и осуществляется либо через системные процедуры-функции либо непосредственно.
- **Независимость** данных в реляционных СУБД обеспечивается на двух уровнях:
 - **Физическая независимость** - означает независимость от способа физического хранения данных.
 - **Логическая независимость** - означает, что правильное функционирование не зависит от изменений, вносимых в структуру пользовательских данных.

Язык программирования

Единый язык высокого уровня, позволяющий:

- описывать данные (data definition)
- манипулировать данными (data manipulation)
- вести администрирование данных (data administration)

Любая операция осуществляется посредством:

- операторов (statement)
- команд (command)

Манипулирование с данными различается на

- выборку данных (data retrieval) или запросы к данным (data queries)
 - модификация данных (data modification)

Язык SQL является непроцедурным языком (nonprocedural language)

База данных

- Является источником данных приложения
- Свою информацию хранит в файлах с расширениями:
 - *.dbc - основные данные
 - *.dct - *мета* поля
 - *.dscx - индексы

Структура файла:

Name	Type	Width
objectid	Integer	4
parentid	Integer	4
objecttype	Character	10
objectname	Character	128
property	Memo (binary)	4
code	Memo (binary)	4
riinfo	Character	6
user	Memo	4

- уникальный код объекта
- значение кода родительского объекта
- типа объекта
- название объекта
- свойства объекта
- откомпилированный код
- тип контроля целостности
- информация пользователя

Объекты базы данных

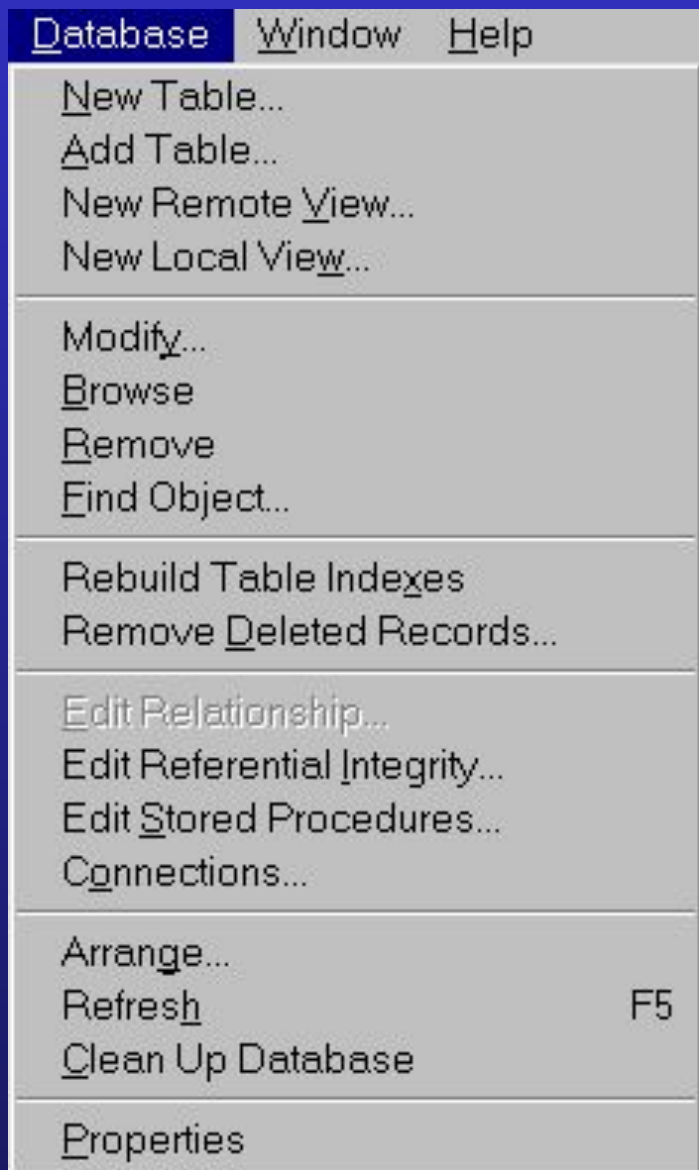


- Таблицы
- локальные представления
- удалённые представления
- соединения с вн. источниками
- сохранённые процедуры

Каждый из объектов имеет ряд своих свойств, значения которых хранятся в базе данных.

Кроме того, таблицы хранятся в файлах с расширениями: *.dbf, *.fpt, *.cdx

Работа с объектами базы данных



- создать таблицу
- добавить таблицу
- создать удалённое представление
- создать локальное представление
- вызвать среду редактирования текущего объекта
- просмотр таблицы значений объекта
- удалить объект из базы данных
- найти объект базы данных
- обновить индексы
- удалить записи, отмеченные на удаление
- редактировать межтабличные отношения
- редактирование признаков ссылочной целостности
- редактирование сохранённых процедур
- редактирование соединений с внешними источниками
- переустановить расположение объектов
- обновить данные об объектах базы данных
- выполнить команду РАСК для базы данных
- вызвать диалог редактирования свойств базы данных

Работа с объектами базы данных

База данных:

- Expand All
- Collapse All
- Find Object...

раскрыть все
свернуть все
найти объект

Таблица:

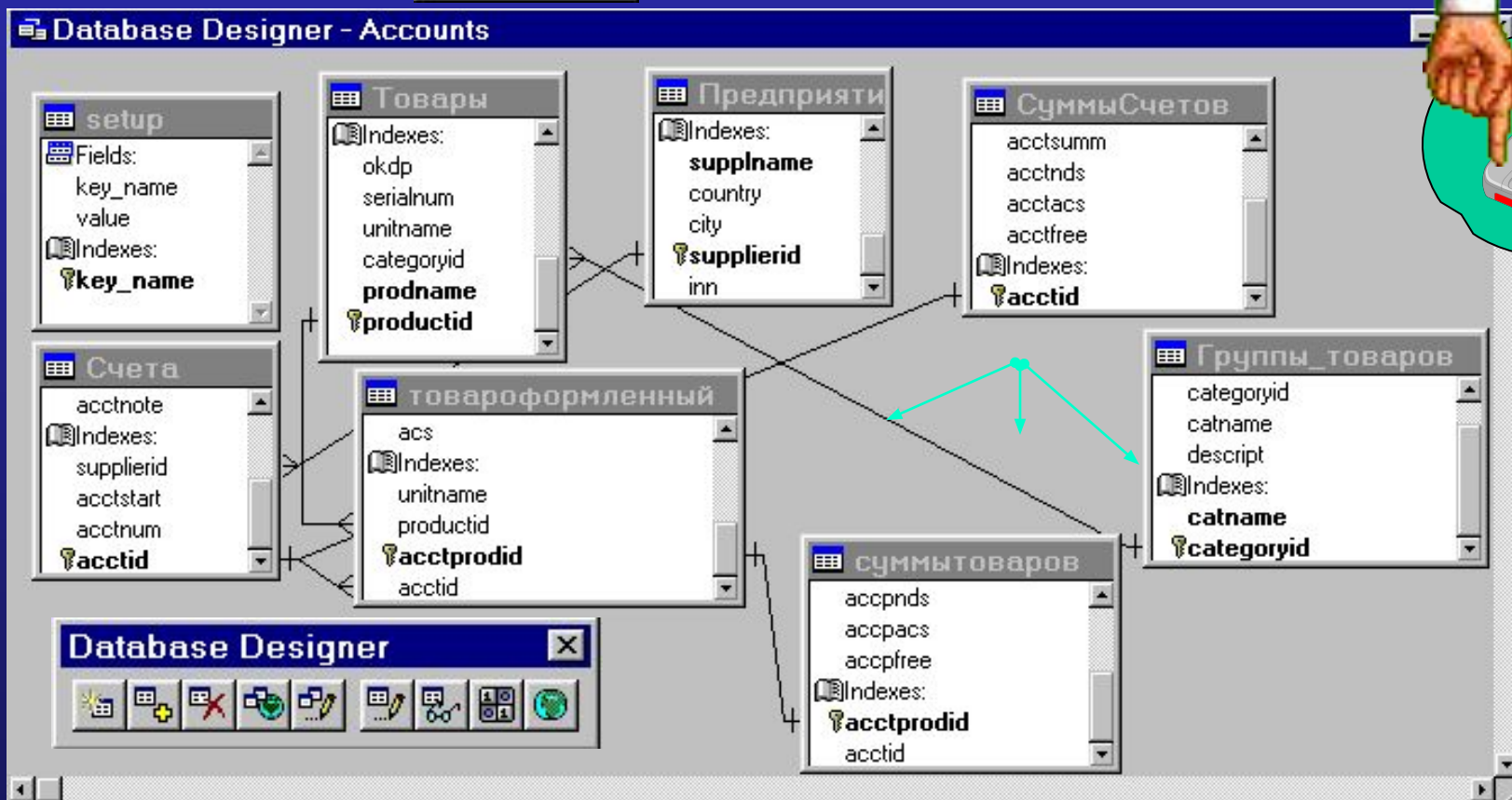
- Browse
- Delete
- Collapse
- Modify...
- Help...

список
удалить
свернуть
редактировать
помощь

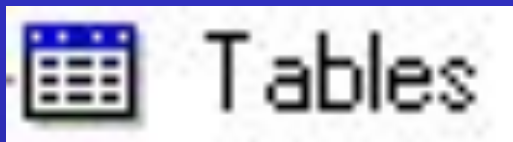
Отношение:

- Remove Relationship
- Edit Relationship...
- Edit Referential Integrity...
- Help...

удалить
редактировать
целостность
помощь



Список свойств таблицы



- *Name* - название таблицы
- *DeleteTrigger* - триггер (храняемая процедура особого типа, которую пользователь не вызывает непосредственно, а исполнение которой обусловлено действием по модификации данных удаления записи из таблицы)
- *InsertTrigger* - триггер добавления записи в таблице
- *UpdateTrigger* - триггер изменения записи в таблице
- *Path* - месторасположение таблицы
- *PrimaryKey* - первичный ключ таблицы
- *RuleExpression* - предикат контроля значений полей уровня записи
- *RuleText* - сообщение об ошибке значений полей
- *Comment* - комментарий таблицы

Свойства таблицы

- *Name* - alias таблицы
- *Record validation* - функция контроля данных полей уровня записи:
 - *Rule* - функция-предикат (это утверждение, истинность которого зависит от значения переменных, входящих в него).
 - *Message* - строка-сообщение об ошибке
- *Triggers* - функции-предикаты, вызываемые при событиях: *добавления (Insert)*, *изменения (Update)*, *удаления (Delete)* данных записи. У буферизованных таблиц при вызове функции

Table Designer - products.dbf

Fields | Indexes | **Table**

Name: Товары

Database: d:\myapp\wfp\accounts\data\accounts.dbc

Statistics

Table file: d:\myapp\wfp\accounts\data\products.dbf

Records: 3 Fields: 9 Length: 145

Record validation

Rule: rulechk_товары()

Message: rulemsg_товары()

Triggers

Insert trigger:

Update trigger: _ri_update_товар

Delete trigger: _ri_delete_товар

Table Comment:

Список товаров/услуг

Это название отображается в *Project Manager* и используется как *alias* при открытии таблицы

Это свойство редактируется также из *Project Manager* см. *Description*

Эти функции автоматически обновляются при переустановках в *Referential Integrity*

Все перечисленные свойства сохраняются в базу данных *Visual FoxPro*, и следовательно недоступны для свободных (*free*) таблиц.

Свойства поля таблицы



- *Name* - название поля
- *Type* - тип поля
- *Size* - размер поля
- *Format* - формат поля
- *InputMask* - шаблон поля
- *Caption* - заголовок поля
- *RuleExpression* - предикат контроля значения поля
- *RuleText* - сообщение об ошибке значения поля
- *DefaultValue* - значение поля по умолчанию
- *DisplayLibrary* - библиотека элементов управления
- *DisplayClassLibrary* - название класса поля
- *Comment* - комментарий поля

Поля таблицы

Type - типы полей:

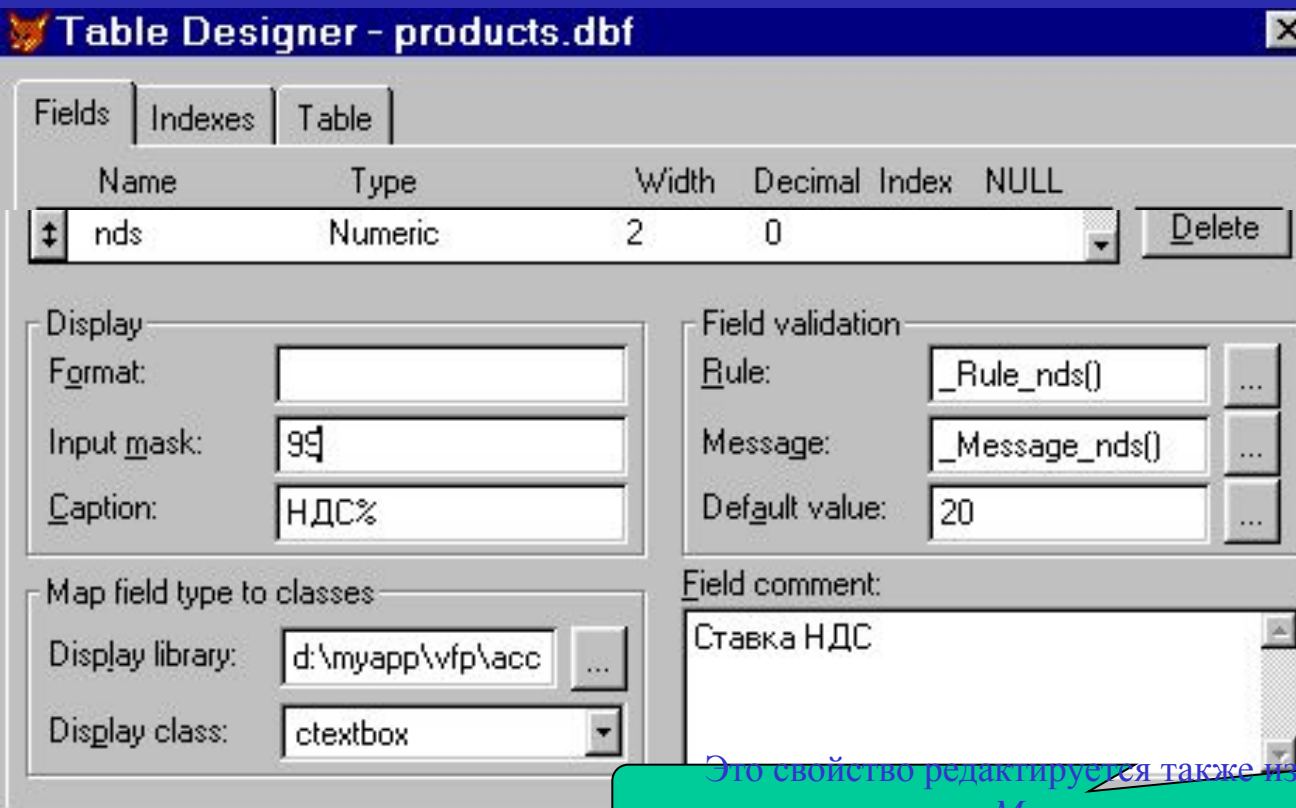
- Character
- Currency
- Numeric
- Float
- Date
- DateTime
- Double
- Integer
- Logical
- Memo
- General
- Character (binary)
- Memo (binary)

The screenshot shows the 'Table Designer' window for 'products.dbf'. It features a 'Fields' tab with a table of field definitions. A green callout box highlights the 'Index' column for the 'serialnum' field, with the text: 'Используя эту кнопку, Вы можете изменить порядок полей'. Below the table are sections for 'Display' (Format, Input mask, Caption), 'Field validation' (Rule, Message, Default value), 'Map field type to classes' (Display library, Display class), and 'Field comment'.

Name	Type	Width	Decimal	Index	NULL
productid	Character	10		↑	
prodname	Character	40		↑	
categoryid	Character	10		↑	
serialnum	Character			↑	
unitname	Character			↑	
unitprice	Numeric	15	2		

- *Name* - название поля (до 128 символов в *.dbc)
- *Width* - общая длина
- *Decimal* - количество знаков после точки
- *Index* - автоматически добавляет *Regular* индекс в *CDX*-файл *
- *NULL* - признак допустимости *NULL* значения поля

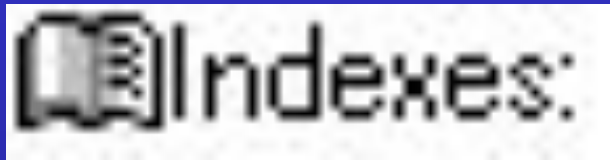
Поля таблицы



Это свойство редактируется также из Project Manager см. Description

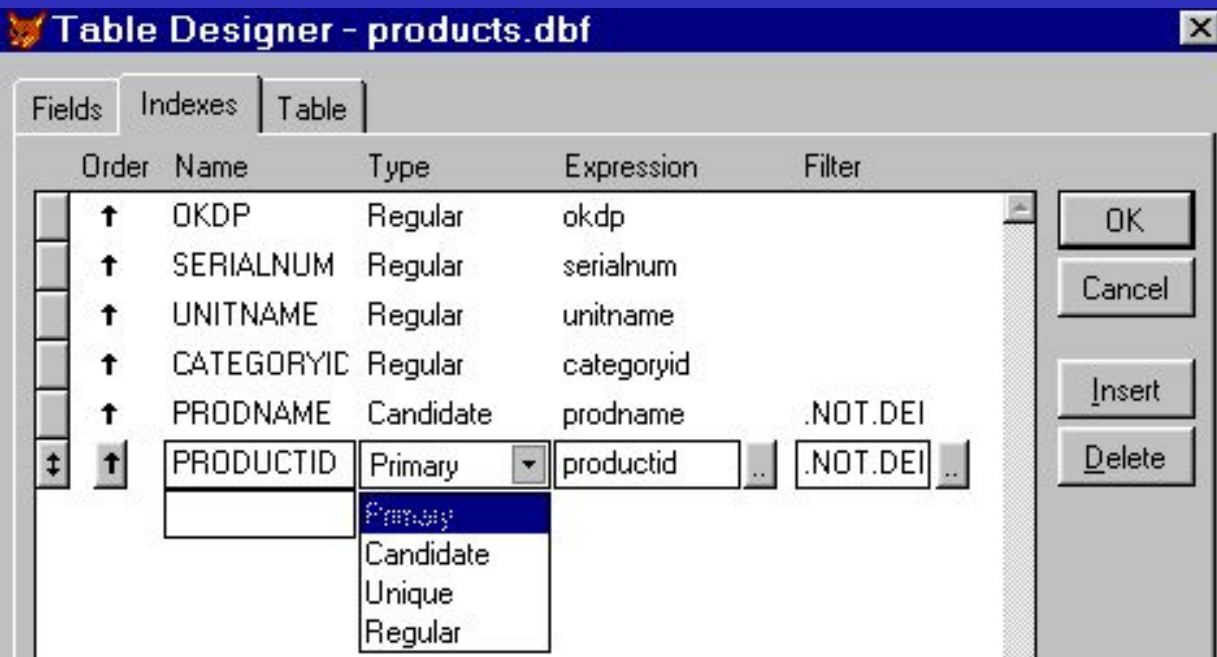
- *Display* - обеспечивает форматирование для операции ввода /вывода:
 - *Format* - задаёт размер и стиль отображения для *Browse, Form* и *Report*
 - *Input mask* - обеспечивает шаблон ввода данных
 - *Caption* - задаёт заголовок столбца/названия поля для *Browse, Form* и *Report*

Свойства индекса таблицы



- *Name* - название индекса
- *Order* - направление сортировки индекса
- *Type* - тип индекса
- *Expression* - определяющее выражение индекса
- *Filter* - фильтрующее выражение индекса

Индексы таблицы

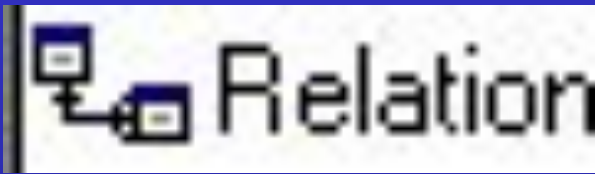


Используя эту кнопку,
Вы можете изменить порядок индексов

- *Order* - определяет порядок сортировки данных поля
- *Name* - название индекса (до 10 символов)
- *Type* - определяет тип индекса
- *Expression* - «определяющее» выражение для значений индекса
- *Filter* - «фильтрующее» выражение значений индекса

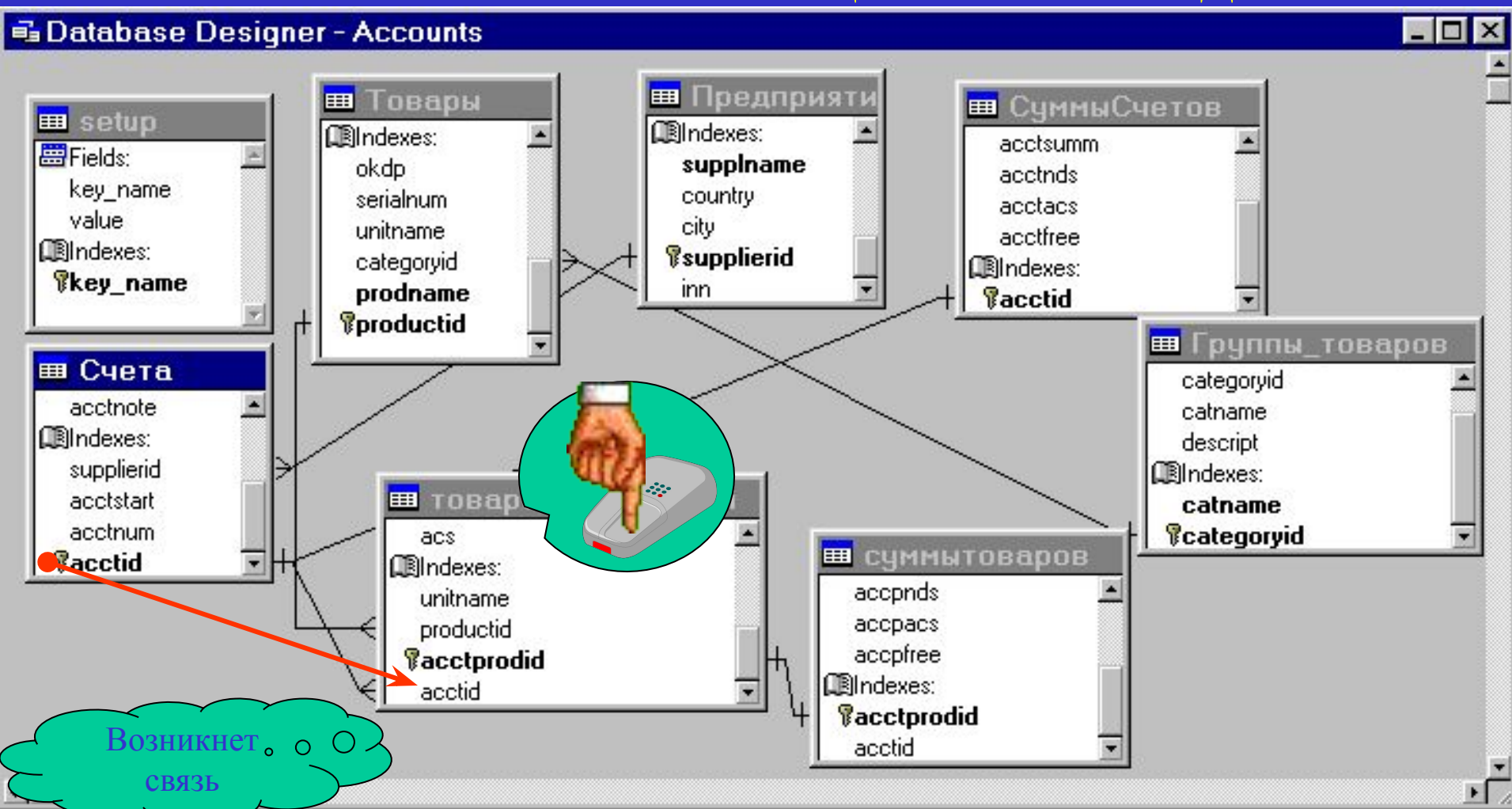
- *Primary* - не допускает повторение значений, один на таблицу.
- *Candidate* - тоже, что и Primary, но может быть несколько в таблице.
- *Unique* - допускает повторения, но сохраняет только одно значение.
- *Regular* - не контролирует значения.

Свойства отношения



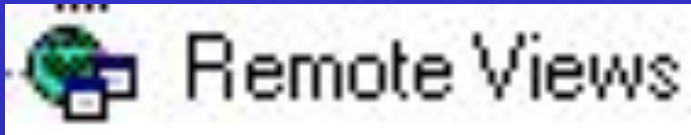
- *Application* - ссылка на объект приложения, содержащий данный
- *ChildAlias* - название подчинённой таблицы
- *ChildOrder* - индекс подчинённой таблицы
- *Comment* - комментарий
- *Name* - название
- *OneToMany* - признак один-ко-многим
- *ParentAlias* - название родительской таблицы
- *RelationalExpr* - реляционное выражение родительской таблицы
- *Tag* - дополнительная информация

Обеспечение ссылочной целостности данных



- Щёлкнув левой клавишей мыши на первичном ключе родительской таблицы и не отпуская клавиши, переместите указатель мыши на индекс подчинённой таблицы.
- Названия индексов:
 - в родительской - *первичный (primary)* [только один в таблице]
 - в подчинённой - *внешний (foreign)* [может быть несколько в одной таблице, и соответствует числу родительских таблиц]

Список свойств представления (view)



- *Name* - название представления
- *BatchUpdateCount* - количество update-операторов
- *CompareMemo* - включать ли мемо в WHERE
- *ConnectName* - имя используемого соединения
- *FetchAsNeeded* - использовать ли выборку данных
- *FetchMemo* - выбирать ли мемо поля
- *FetchSize* - количество записей в выборке
- *MaxRecords* - максимальное количество требуемых записей
- *Offline* - является ли отсоединённым представлением
- *ParameterList* - список параметров и их типов
- *Prepared* - повторно используемый ли запрос (допустимо ли REQUERY())
- *RuleExpression* - выражение контроля уровня записи
- *RuleText* - сообщение об ошибке контроля уровня записи
- *SendUpdates* - требуется ли посылка изменений источнику
- *ShareConnection* - разделяемое ли соединение используется
- *SourceType* - тип представления
- *SQL* - SQL-оператор
- *Tables* - список используемых таблиц
- *UpdateType* - способ выполнения изменений
- *UseMemoSize* - минимальная длина мемо-полей
- *WhereType* - способ формирования WHERE
- *Comment* - комментарий таблицы

Представление-
виртуальная
(логическая)
таблица,
представляющая
собой
поименованный
запрос (синоним к
запросу), который
будет подставлен
как подзапрос при
использовании
представления.

Представления (view) среда редактирования

Условия межтабличных объединений
Вы можете определять используя
левую клавишу мыши

The screenshot shows a software interface for creating database views. At the top, a data table displays transaction records. Below it, the 'View Designer' window is open, showing three tables: 'Предприятия', 'Счета', and 'Суммысчетов'. The 'Join' tab is selected, showing two inner joins between the tables. A dropdown menu is open, listing join types: <None>, Inner Join, Left Outer Join, Right Outer Join, and Full Join. The 'Fields' tab is also visible, showing a table with columns for Type, Field Name, Not, Criteria, Value, and Logical.

№ счёта	Дата сч.	Поставщик	ИНН	Сумма	Сумма общ.	Сумм
1	24.06.98	ТОО ИВС	(нет)	4,00	4,40	0,40
2	24.06.98	Рога и копыта	(нет)	1502,00	1802,20	300,20

Type	Field Name	Not	Criteria	Value	Logical
↔ Inner Joir	Предприятия.supplie	=		Счета.supplierid	
↔ Inner Joir	Счета.acctid	=		Суммысчетов.acc	
		=			

Позволяет создать /изменить *SQL запрос* к данным, в результате выполнения которого, получается *Vfp cursor*

- В верхней части диалога отображается схема используемых таблиц и межтабличных связей
- В нижней, на закладках: *Fields, Join, Filter, Order By, Group By*, - обеспечивается возможность изменения соответствующих частей *SQL-оператора*
- На закладке *Update Criteria* Вы можете настроить режимы взаимодействия с источником данных
- В *Miscellaneous* Вы можете дополнительно ограничить число записей (или их %), а также исключить дублирование данных

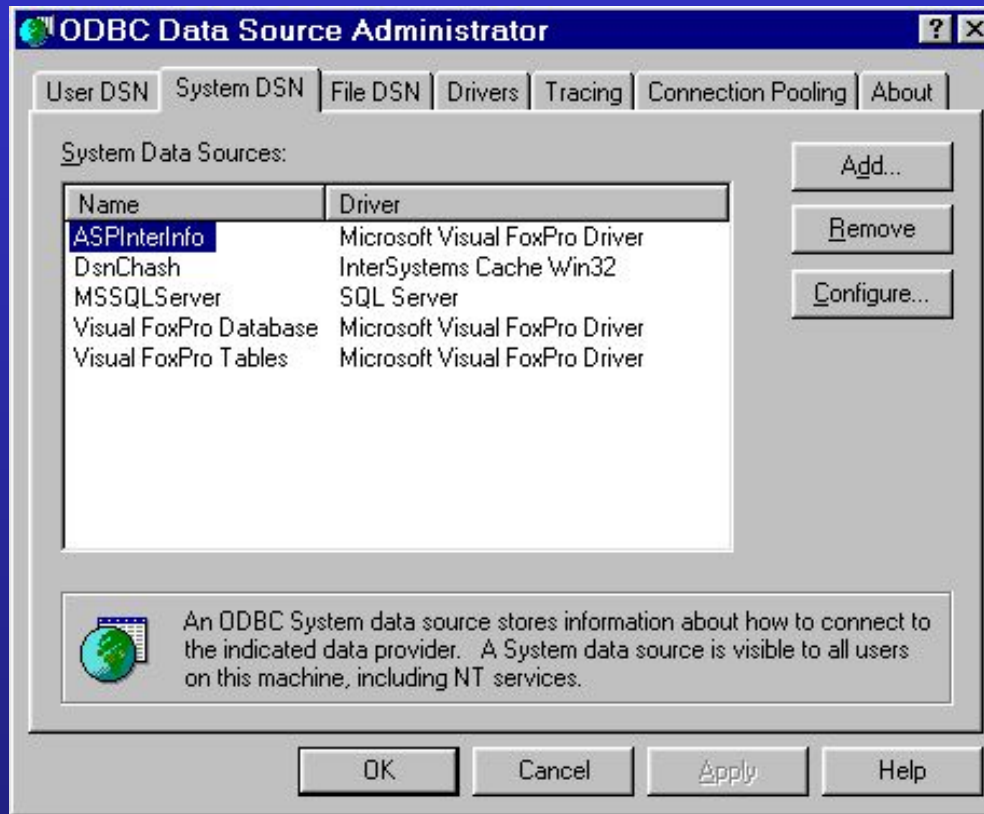
Список свойств соединения



Connections

- *Name* - название соединения
- *Asynchronous* - допустим ли *Asynchronous* режим соединения
- *BatchMode* - допустим ли *Batch* режим соединения
- *ConnectionString* - строка параметров соединения
- *ConnectTimeout* - *Timeout* соединения
- *Database* - *Database* соединения
- *DataSource* - *DSN* соединения
- *DispLogin* - режим диалога ввода пароля ODBC
- *DispWarnings* - допустим ли вывод сообщений
- *IdleTimeout* - время ожидания закрытия
- *PacketSize* - размер сетевого пакета
- *PassWord* - *PWD* соединения
- *QueryTimeout* - время ожидания результата
- *Transactions* - транзакционный режим
- *UserId* - *UID* соединения
- *WaitTime* - время задержки контроля завершения запроса
- *Comment* - комментарий соединения

Соединения с внешними источниками данных



- *Внешние источники данных* должны быть специфицированы в *ODBC Data Source Administrator*.
- В *Visual FoxPro* базе данных может быть использовано несколько таких источников, в качестве соединений к внешним данным, редактируемых из диалога *Connections*.
- *Внешние представления (Remote view)* имеют имя соединения в качестве свойства, параметры которого прозрачно используются для получения соединения с внешним источником данных.

Параметры соединения

Connection Designer - Connect1

Specify data source

Data source, userid, password Connection string

Verify Connection...
New Data Source...

Data source: ASPInterInfo Userid: Michael Password: ***** Database: InetInfo

Display ODBC login prompts

When login information is not specified Always Never

Data processing

Asynchronous execution
 Display warnings
 Batch processing
 Automatic transactions

Packet size: 4096

Timeout intervals

Connection (sec): 15 Idle (min): 0
Query (sec): 0 Wait time (ms): 100

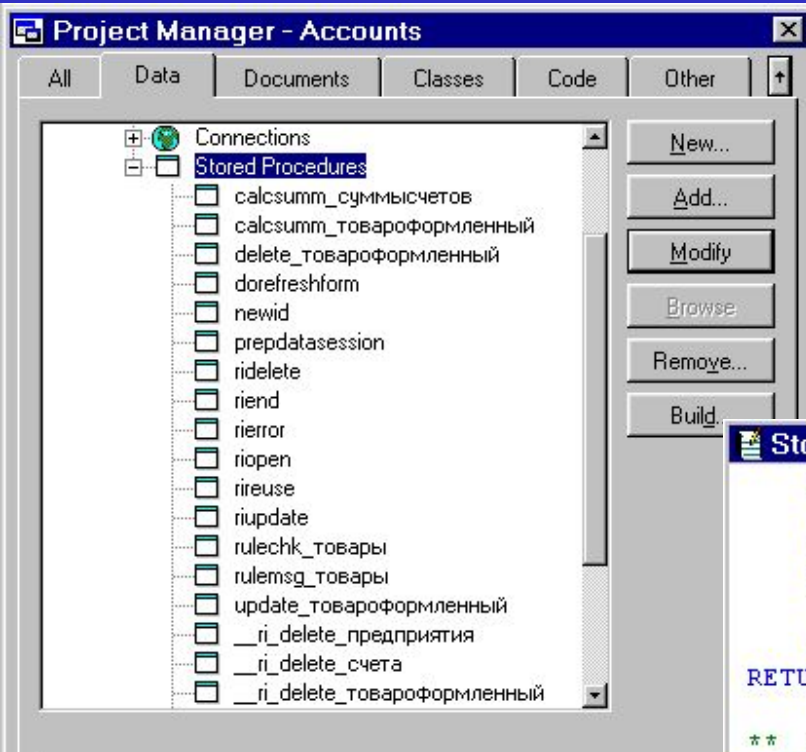
OK Cancel

Этот диалог может быть вызван из *Project Manager*

Нажатие кнопки ОК приводит к диалогу сохранения параметров соединения под некоторым именем в базу данных Visual FoxPro, которое в дальнейшем используется *Внешними представлениями (Remote view)* в качестве соединения. Список соединений базы данных редактируется из диалога *Connections*.

- *Specify data source* - определяет внешний источник данных
 - возможны два варианта:
 - *Data source, userid, password* - повторяют параметры источника, зарегистрированного в *ODBC Administrator*.
 - *Connection string* - задаёт строку, содержащую перечисление параметров и их значений, используя ';' в качестве разделителя.
 - *Verify Connection...* - может быть использована для проверки работоспособности соединения.
 - *New Data Source...* - вызывает диалог *ODBC Administrator*
- *Display ODBC login prompt* - определяет должен ли вызываться диалог *ввода пароля* при установке соединения: *Когда необходимо, Всегда, Никогда*.
- Оставшаяся часть диалога позволяет задать ряд параметров ODBC соединения, которые могут отличаться от задаваемых по умолчанию

Сохранённые процедуры



- Список всех сохранённых процедур удобно представлен в *Project Manager*.
- Язык написания - это подмножество языка *Visual FoxPro*, ориентированное на работу с данными.
- Если Вы используете *контроль ссылочной целостности*, генерируемый средствами *Visual FoxPro*, не пытайтесь редактировать полученный код вручную.

The image shows a screenshot of the 'Stored Procedures for Accounts' window. It displays the source code for the 'RIDELETE' procedure. The code includes comments and SQL statements for locking, deleting, and unlocking records.

```
IF lcOldTalk = 'ON'
    SET TALK ON
ENDIF
SELECT (lnOldArea)

RETURN lbRetVal
RETURN

**__RI_HEADER!@ Do NOT REMOVE or MODIFY this line!!!! @!__RI_HEADER**
procedure RIDELETE
local llRetVal
llRetVal=.t.
IF (ISRLOCKED() and !deleted()) OR !RLOCK()
    llRetVal=.F.
ELSE
    IF !deleted()
        DELETE
        IF CURSORGETPROP('BUFFERING') > 1
            =TABLEUPDATE()
        ENDIF
        llRetVal=pnerror=0
    ENDIF not already deleted
ENDIF
UNLOCK RECORD (RECNO())
RETURN llRetVal
```

- Если у Вас возникла необходимость сделать что-нибудь с данными, не задумываясь оформляйте это в виде сохранённой процедуры, никак не следует делать это методом конкретного экземпляра формы.
- При обращении к сохранённым процедурам контролируйте активность базы данных

Схема объектов базы данных

