
Краткий справочник по Visual FoxPro

Version 1.0

Оглавление

Оглавление	2
Массивы.....	4
Классы, объекты, формы.....	6
ТИПЫ ДАННЫХ – строковый функции	9
ТИПЫ ДАННЫХ – функции преобразования данных.....	9
ТИПЫ ДАННЫХ – функции для дат и временем	12
ТИПЫ ДАННЫХ – цифровые функции.....	16
База данных – манипулирование данными	19
База данных – работа с полями	26
База данных - индексы	28
База данных – работа с записями.....	31
База данных – отношения.....	35
База данных – работа с таблицами	36
Visual FoxPro окружение - среда	43
Visual FoxPro окружение - работа с файлами	51
Распределенный доступ к данным - доступ	54
Распределенный доступ к данным – секретность и шифрование	56
Распределенный доступ к данным – команды определения сессии данных	57
Ввод и вывод – форматирование данных	59
Ввод и вывод – ввод и клавиатуры и мыши.....	62
Ввод и вывод - меню	64
Ввод и вывод - печать	67
Ввод и вывод – отчеты и ярлыки	69
Ввод и вывод - окна	70
Программирование – отладка и управление ошибками	72
Программирование – управление событиями.....	75
Программирование –доступ к файлам на низком уровне.....	76
Программирование – управление переменными памяти	77
Программирование – объектно-ориентированное программирование.....	79

Программирование – выполнение программ.....	82
Программирование – структурное программирование	84
Экономические функции	85
Разработка интернациональных приложений	86
Visual FoxPro события.....	87
Visual FoxPro события для элементов редактирования	90
Visual FoxPro события для базовых классов	98
List & ComboBox свойства, события, методы.....	117
ПРИЛОЖЕНИЕ . Базовые классы	119
ПРИЛОЖЕНИЕ. Значения первого байта файла DBF.....	120
ПРИЛОЖЕНИЕ. Visual FoxPro правила именования и сокращения.....	121
ПРИЛОЖЕНИЕ. Visual FoxPro иерархия событий.....	122
ПРИЛОЖЕНИЕ. Часто используемые значения RGB и Color	123

Массивы

Функции	Описание
AClass()	Помещает в массив переменных памяти имя класса объекта, а также имена его родительских классов..
ACopy()	Копируем элементы из одного массива в другой.
ADATABASES()	Помещает в массив переменных памяти имена всех открытых баз данных и описания путей для этих имен.
ADBOBJECTS()	Помещает в массив переменных памяти имена соединений, отношений, таблиц или представлений SQL из текущей базы данных
ADEL()	Удаляет элемент из одномерного массива или удаляет строку или столбец из двумерного массива
ADIR()	Заносит в массив информацию о файлах и возвращает количество файлов.
AELEMENT()	Вычисляет номер элемента массива исходя из индексов этого элемента.
AERROR()	Создает массив переменных памяти, содержащий информацию о самой последней ошибке, связанной с Visual FoxPro, OLE или ODBC.
AFIELDS()	Помещает информацию о структуре текущей таблицы в массив и возвращает количество полей в таблице.
AFONT()	Заносит в массив информацию о доступных шрифтах.
AINS()	Вставляет элемент в одномерный массив или вставляет строку или столбец в двумерный массив
AINSTANCE()	Помещает экземпляры класса в массив переменных памяти и возвращает число занесенных в массив экземпляров.
ALEN()	Возвращает число элементов, строк или столбцов в массиве.
AMEMBERS()	Помещает в массив переменных памяти имена свойств, процедур и объектов-членов заданного объекта
APPEND FROM ARRAY	Добавляет одну запись в выбранную в данный момент таблицу для каждой строки массива и заполняет каждую запись данными из соответствующей строки массива.
APRINTERS()	Помещает в массив переменных памяти имена принтеров, которые в данный момент описаны в Windows.
ASCAN()	Просматривает массив в поисках элемента, данные которого имеют то же значение и тот же тип, что и указанное выражение.
ASELOBJ()	Помещает в массив переменных памяти указатели на выбранные в данный момент элементы управления в активном конструкторе форм.
ASORT()	Сортирует элементы массива в возрастающем или убывающем порядке.
ASUBSCRIPT()	Возвращает индекс по строке или столбцу элемента массива исходя из его номера.
AUSED()	Помещает в массив переменных памяти псевдонимы и рабочие области для таблиц заданного сеанса.
COPY TO ARRAY	Копирует данные в массив из таблицы, выбранной в данный момент.

Функции	Описание
DECLARE	Создает одномерный или двумерный массив.. Идентична с DIMENSION.
DIMENSION	Создает одномерный или двумерный массив.. Идентична с DECLARE.
DISPLAY MEMORY	Отображает текущее содержимое переменных памяти и массивов.
FOR EACH ... ENDFOR	Выполняет набор команд для каждого из элементов массива или коллекции. Конструкция полезна при работе с одномерными массивами, с каждым из которых надо выполнить последовательность одностипных действий. Быстрее чем обычно употребляемая конструкция цикла FOR....ENDFOR.
GATHER	Замещает данные в текущей записи текущей выбранной таблицы содержимым массива, группы переменных памяти или объекта.
PUBLIC	Определяет глобальные переменные памяти или массивы.
RELEASE	Удаляет из памяти переменные или массивы.
REGIONAL	Создает региональные переменные памяти и массивы. Regional массивы и переменные аналогичны массивам и переменным типа
REPLACE FROM ARRAY	Изменяет данные в полях на значения из массива.
RESTORE FROM	Извлекает переменные памяти и массивы переменных памяти, сохраненные в файле переменных памяти или в мемо-поле, и помещает их в память.
RESTORE SCREEN	Восстанавливает основное окно Visual FoxPro или пользовательское окно, сохраненное перед этим в буфере экрана, в переменной памяти или в элементе массива.
SAVE TO	Сохраняет текущие переменные памяти и массивы в файле переменных памяти или в мемо-поле.
SCATTER	Копирует содержимое текущей записи в набор переменных памяти или в массив.
SELECT - SQL	Получение данных из одной или нескольких таблиц в массив или курсор.
STORE	Сохраняет данные в переменной памяти, массиве или в элементе массива.

Классы, объекты, формы

Функции	Описание
AClass()	Помещает в массив переменных памяти имя класса объекта, а также имена его родительских классов.
ADD CLASS	Добавляет определение класса в библиотеку визуальных классов .VCX..
AINSTANCE()	Помещает экземпляры класса в массив переменных памяти и возвращает число занесенных в массив экземпляров
AMEMBERS()	Помещает в массив переменных памяти имена свойств, процедур и объектов-членов заданного объекта.
APPEND GENERAL	Импортирует объект OLE из файла и помещает его в поле типа General.
ASELOBJ()	Помещает в массив переменных памяти указатели на выбранные в данный момент элементы управления в активном конструкторе форм.
CLEAR EVENTS	Останавливает процесс обработки событий, начатый в READ EVENTS. После выполнения CLEAR EVENTS, программа продолжает выполнение со следующей строки после READ EVENTS.
COMPILE FORM	Компилирует один или несколько объектов Form
COMPOBJ()	Сравнивает свойства двух объектов и возвращает значение "истина" (.T.), если их свойства и значения свойств идентичны.
CREATE CLASS	Открывает конструктор классов, предоставляя возможность создать новое определение класса.
CREATE CLASSLIB	Создает новый, пустой .VCX файл библиотеки визуальных классов.
CREATE FORM	Открывает конструктор форм.
CREATE VIEW	Создает файл среды Visual FoxPro. Команда SET VIEW восстанавливает среду, которая создана командой CREATE VIEW. Файлы среды имеют расширение .VUE.
CREATEBINARY()	Преобразует тип данных из Visual FoxPro в бинарную строку, которую можно передать в ActiveX или использовать при OLE..
CREATEOBJECT()	Создает объект на основе определения класса или на основе доступного приложения OLE.
DEFINE CLASS	Создает пользовательский класс или подкласс, определяет свойства, события и методы для них.
DODEFAULT()	В подклассе выполняет событие или метод родителя, которые имеют эти же имена.
DOEVENTS	Выполняет все задержанные события Windows. События Window помещаются в очередь, если свойство AutoYield установлено в false (.F.) , а производится выполнение программного кода. DOEVENTS выполняет все задержанные события и процессы Windows, если в пользовательском коде есть связь с событиями Windows.
GETOBJECT()	Активизирует объект OLE automation и создает ссылку на него.
GETPEM()	Возвращает текущее значение свойства или код программы для мсобытия или метода.
LOADPICTURE()	Создает ссылку на bitmap, icon или Windows meta файл. Многие свойства для элементов ActiveX требуют эту объектную ссылку. Например, ActiveX Outline поддерживает свойство PictureOpen, которое требует объектную ссылку для ее
MESSAGEBOX()	Отображает диалоговое окно, определенное пользователем.

Функции	Описание
MODIFY CLASS	Открывает конструктор классов, давая возможность изменить имеющееся определение класса или создать новое.
MODIFY FORM	Открывает конструктор форм, позволяя модифицировать или создать форму.
OBJTOCLIENT()	Возвращает позицию или размер элемента управления или объекта относительно содержащего его объекта Form.
PEMSTATUS()	Возвращает определенные атрибуты для свойств, событий или методов (например, Read-Only, Changed, Protected, Type, etc...).
READ EVENTS	Начинает процесс обработки событий. Если построены приложения .EXE или .APP , в которых в главной программе выполняется DO FORM, необходимо задать READ EVENTS перед DO FORM для подсказки FoxPro необходимости обработки процесса событий. При работе через окно команд FoxPro автоматически обрабатывает последовательность событий).
RELEASE CLASSLIB	Закрывает библиотеку .VCX.
REMOVE CLASS	Удаляет определение класса из библиотеки классов .VCX .
RENAME CLASS	Переименовывает определение класса в библиотеку классов.
SAVEPICTURE()	Создает файл картинки (.BMP) на основании объектной ссылки на нее.
SET CLASSLIB	Открывает библиотеку визуальных классов .VCX, содержащую определения классов.
SET CONFIRM	Определяет, можно ли выходить из текстового поля, вводя данные правее его последнего символа. ON Указывает, что вы не можете выходить из текстового поля, вводя данные правее его последнего символа.
SET DATASESSION	Активизирует заданный сеанс формы
SET MESSAGE	Задаст сообщение, отображаемое в основном окне Visual FoxPro или в графической строке состояния, или задает место вывода сообщений, связанных с пользовательскими строками меню и командами меню.
SET NULLDISPLAY	По умолчанию Visual FoxPro выводит значение .NULL. для пустых значений объектов. Использование команды SET NULLDISPLAY позволяет сменить выводимый текст.
SET PALETTE	Определяет, используется ли в Visual FoxPro цветовая палитра, принимаемая по умолчанию.
SET SYSMENU	Делает доступной или недоступной системную строку меню Visual FoxPro в процессе выполнения программы и дает возможность изменить конфигурацию строки меню.
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления., или восстанавливает окружение Visual FoxPro на основании файла, созданного в команде CREATE VIEW.
SYS(1016)	Возвращает объем памяти в байтах, задействованной определенными пользовательскими объектами.
SYS(1269)	Возвращает логическое значение, которое показывает изменилось ли заданное свойство объекта по отношению к значению, принятому по умолчанию.
SYS(1270)	Возвращает ссылку на объект, который находится под курсором мыши или в определенном месте экрана.
SYS(1271)	Возвращает имя файла .SCX, в котором храниться определенный объект.

Функции	Описание
SYS(1272)	Возвращает иерархию объектов для заданного объекта. Функция работает только при выполнении, а не в режиме создания.
SYS(2333)	Если объект ActiveX не работает должным образом при его инициализации, используйте SYS(2333) или SYS(2333, 0) перед инициализацией объекта для отключения интерфейса двойной поддержки. После инициализации объекта используйте SYS(2333, 1) для включения интерфейса двойной поддержки для других элементов, которые будут инициализированы позднее.
SYS(3004)	Возвращает Locale ID, используемый OLE automation и элементами OLE. Locale ID определяет язык на котором производится обмен информацией с элементами ActiveX. По умолчанию в Visual FoxPro Locale ID установлен 1033, English.
SYS(3005)	Устанавливает Locale ID, используемый OLE automation и элементами OLE. SYS(3005) задает глобальный Locale ID (LCID). Locale ID определяет язык, на котором производится обмен информацией при работе с ActiveX controls. По умолчанию в Visual FoxPro Locale ID установлен в 1033, English.
SYS(3006)	Устанавливает Language ID (LangID) и Locale ID (LCID) на основании Language ID и текущего значения Sort ID.
SYS(3053)	Возвращает ODBC environment handle. Если ODBC не загружен, SYS(3053) загружает его и возвращает ODBC environment handle. Environment handle возвращенные SYS(3053) обеспечивает доступ к ODBC через вызовы ODBC API.
SYSMETRIC()	Возвращает размеры экранных элементов операционной системы. SYSMETRIC() позволяет определить размер для меню, окон, элементов управления в Visual FoxPro.
WITH ... ENDWITH	<p>Задаёт несколько свойств для объекта. Позволяет использовать нотацию с точкой (например, .Value) при выполнении множественных манипуляций с объектом, это сокращает объём ввода и повышает производительность. Для примера, можно:</p> <pre> WITH ThisForm.Txt_Total_Owed .Value = 2452.22 .ReadOnly = .T. .FontBold = .T. .FontSize = 10 .Visible = .T. ENDWITH </pre>

ТИПЫ ДАННЫХ – строковый функции

Функции	Описание
\$	Возвращает (.T.) , если строковое выражение содержит другое строковое выражение, в противном -(F.).
ALLTRIM()	Удаляет начальные и конечные пробелы из заданного символьного выражения и возвращает усеченное выражение в виде символьной строки.
ASC()	Возвращает ANSI значение для самого левого символа в символьном выражении.
AT()	Возвращает начальную позицию первого вхождения символьного выражения или тето-поля в другое символьное выражение или тето-поле, по отношению к позиции самого левого символа. Поиск с учетом регистра.
AT_C()	Возвращает начальную позицию первого вхождения символьного выражения или тето-поля в другое символьное выражение или тето-поле, по отношению к позиции самого левого символа. Поиск с учетом регистра. Функция аналогична AT(), но работает и с двухбайтовыми символами.
ATC()	Возвращает начальную позицию первого вхождения символьного выражения или тето-поля в другое символьное выражение или тето-поле, по отношению к позиции самого левого символа. Поиск без учета регистра.
ATCC()	Возвращает начальную позицию первого вхождения символьного выражения или тето-поля в другое символьное выражение или тето-поле, по отношению к позиции самого левого символа. Поиск без учета регистра. Функция аналогична ATC(), но работает и с двухбайтовыми символами.
ATCLINE()	Возвращает номер строки с первым вхождением символьного выражения или тето-поля, независимо от регистра символов в обоих выражениях (то есть прописные или строчные). Номер строки зависит от установок SET MEMOWIDTH.
ATLINE()	Возвращает номер строки с первым вхождением символьного выражения или тето-поля в другое символьное выражение или тето-поле, считая от первой строки. Поиск с учетом регистра. Номер строки зависит от установок SET MEMOWIDTH..
BETWEEN()	Определяет, лежит ли значение некоторого выражения в диапазоне между значениями двух других выражений, имеющих тот же тип данных
BINTOC()	Преобразует целые цифровые значения в бинарную строку. Обычно используется для уменьшения размера индекса, построенного на целых цифровых полях.. Обратное преобразование – CTOBIN().
CHR()	Возвращает символ, который соответствует коду ANSI.
CHRTRAN()	Замещает каждый символ в символьном выражении, совпадающий с некоторым символом второго символьного выражения, на соответствующий символ третьего символьного выражения.
CHRTRANC()	Замещает каждый символ в символьном выражении, совпадающий с некоторым символом второго символьного выражения, на соответствующий символ третьего символьного выражения. CHRTRANC() для замены однобайтового символа двухбайтным и наоборот.
CTOBIN()	Преобразует бинарное символьное выражение в целое цифровое. Обратное к BINTOC().
CTOD()	Преобразует символьное выражение в выражение типа дата
DIFFERENCE()	Возвращает целое число от 0 до 4, представляющее относительное фонетическое различие между двумя символьными выражениями.

Функции	Описание
DTOC()	Возвращает дату в символьном виде исходя из выражения типа Date или DateTime.
EMPTY()	Определяет, является ли значение выражения пустым.
EVALUATE()	Вычисляет символьное выражение и возвращает результат.
INLIST()	Определяет, совпадает ли некоторое выражение с каким-либо другим выражением из заданной группы выражений.
ISALPHA()	Определяет, является ли самый левый символ в символьном выражении буквенным.
ISBLANK()	Определяет, является ли данное выражение пустым. ISBLANK() отличается от EMPTY() и ISNULL(). Для примера, EMPTY() возвращает (.T.) если символьное выражение содержит любые комбинации пустых значений, пробелов, табуляций, созвратов каретки или концов линий; ISBLANK() возвращает (.T.), если выражение содержит только пустые строки или пробелы.
ISDIGIT()	Определяет, является ли самый левый символ заданного символьного выражения цифрой (от 0 до 9).
ISLEADBYTE()	Возвращает (.T.), если первый байт в первом байте символьного выражения является лидирующим байтом двухбайтного символа..
ISLOWER()	Определяет, является ли самый левый символ заданного символьного выражения строчным буквенным символом.
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
ISUPPER()	Определяет, является ли первый символ данного символьного выражения прописным буквенным символом.
LEFT()	Возвращает из символьного выражения заданное число символов, начиная с самого левого.
LEFTC()	Возвращает из символьного выражения заданное число символов, начиная с самого левого. <u>Для двухбайтных символов.</u>
LEN()	Возвращает количество символов в символьном выражении.
LENC()	Возвращает количество символов в символьном выражении. <u>Для двухбайтных символов</u>
LIKE()	Определяет, совпадает ли одно символьное выражение с другим символьным выражением.
LIKEC()	Определяет, совпадает ли одно символьное выражение с другим символьным выражением. Символы могут быть как одно-, так и двухбайтные.
LOWER()	Возвращает заданное символьное выражение, переписанное строчными буквами.
LTRIM()	Возвращает заданное символьное выражение, удалив из него начальные пробелы.
MAX()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наибольшее значение.
MEMLINES()	Возвращает число строк в мемо-поле или переменной. Количество линий зависит от установок в SET MEMOWIDTH.
MLINE()	Возвращает указанную строку мемо-поля в виде символьной строки.. Количество линий зависит от установок в SET MEMOWIDTH.
MIN()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наименьшее значение.

Функции	Описание
NORMALIZE()	Преобразует символьное выражение, заданное пользователем, к виду, который допускает сравнение со значениями, возвращаемыми функциями Visual FoxPro.
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
OCCURS()	Возвращает число вхождений некоторого символьного выражения в другое символьное выражение.
PADC()	Возвращают символьную строку, которая состоит из заданного выражения, дополненного пробелами или определенными символами до указанной длины слева и справа.
PADL()	Возвращают символьную строку, которая состоит из заданного выражения, дополненного пробелами или определенными символами до указанной длины слева.
PADR()	Возвращают символьную строку, которая состоит из заданного выражения, дополненного пробелами или определенными символами до указанной длины справа.
PROPER()	Возвращает из символьного выражения строку, переписанную прописными буквами, как имя собственное.
RAT()	Возвращает числовую позицию последнего вхождения символьного выражения или мемо-поля в другое символьное выражение или мемо-поле. Поиск с учетом регистра.
RATC()	Возвращает числовую позицию последнего вхождения символьного выражения или мемо-поля в другое символьное выражение или мемо-поле. Поиск с учетом регистра. Символы могут быть как одно-, так и двухбайтными.
RATLINE()	Возвращает номер строки с последним вхождением символьного выражения в другое символьное выражение или в мемо-поле, считая от последней строки.
REPLICATE()	Возвращает символьную строку, которая содержит заданное символьное выражение заданное количество раз.
RIGHT()	Возвращает заданное число символов с правого конца символьной строки.
RIGHTC()	Возвращает заданное число символов с правого конца символьной строки. Строка может быть из двухбайтных символов.
RTRIM()	Возвращает символьную строку, которая получается в результате удаления из символьного выражения конечных пробелов.
SET MEMOWIDTH	Задаёт ширину отображаемых на экране мемо-полей и символьных выражений в диапазоне от 8 до 256 столбцов. Установки SET MEMOWIDTH оказывают влияние на функции ATCLINE(), ATLINE(), MEMLINE(), и MLINE() .
SET TEXTMERGE	Включает или отключает вычисление полей, переменных памяти, элементов массива, функций или выражений, заключённых в разделители слияния, и позволяет задать вывод процедуры слияния текста.
SOUNDEX()	Возвращает фонетическое представление заданного символьного выражения.
SPACE()	Возвращает символьную строку, составленную из заданного числа пробелов.
STR()	Возвращает символьный эквивалент заданного числового выражения.
STRCONV()	Преобразует символьное выражение в различные формы: одно-, двухбайтные...
STRTRAN()	Просматривает символьное выражение или мемо-поле в поисках другого символьного выражения или мемо-поля, а затем заменяет каждое найденное выражение третьим символьным выражением или мемо-полем.

Функции	Описание
STUFF()	Возвращает символьную строку, созданную заменой заданного числа символов в символьном выражении другим символьным выражением.
STUFFC()	Возвращает символьную строку, созданную заменой заданного числа символов в символьном выражении другим символьным выражением. Символы могут быть двухбайтные.
SUBSTR()	Возвращает символьную строку из данного символьного выражения или мемо-поля.
SUBSTRC()	Возвращает символьную строку из данного символьного выражения или мемо-поля. Символы могут быть двухбайтные.
SYS(10)	Преобразует номер дня по юлианскому календарю / SYS(1)/ в символьную строку.
SYS(11)	Преобразует выражение даты или символьную строку формата даты в номер дня по юлианскому календарю.
SYS(15)	Преобразует вторую символьную строку, используя первую символьную строку.
SYS(2007)	Возвращает значение контрольной суммы символьного выражения.
TEXT ... ENDTEXT	Выводит строки текста, результаты выражений и функций, а также содержимое переменных памяти.
TRANSFORM()	Возвращает символьную строку, полученную из символьного или числового выражения; строка имеет формат, определенный кодами PICTURE или FUNCTION, которые применяются в команде @ ... SAY.
TRIM()	Возвращает заданное символьное выражение, из которого удалены все концевые пробелы.
TTOC()	Возвращает символьное значение выражения типа DateTime.
TXTWIDTH()	Возвращает длину символьного выражения исходя из средней ширины символа для данного шрифта.
TYPE()	Вычисляет символьное выражение и возвращает тип данных, которому принадлежит содержимое этого выражения.
UPPER()	Возвращает заданное символьное выражение, переведенное в верхний регистр.
VAL()	Возвращает числовое значение символьного выражения, состоящего из цифр.

ТИПЫ ДАННЫХ – функции преобразования данных

Функции	Описание
BINTOC()	Преобразует целые цифровые значения в бинарную строку. Обычно используется для уменьшения размера индекса, построенного на целых цифровых полях.. Обратное преобразование – CTOBIN().
CHR()	Возвращает символ, который соответствует коду ANSI
CPCONVERT()	Преобразует символьные поля, мето-поля или символьные выражения в другую кодовую страницу.
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
CTOBIN()	Преобразует бинарное символьное выражение в целое цифровое. Обратное к BINTOC().
CTOD()	Преобразует символьное выражение в выражение даты.
DMY()	Возвращает символьное выражение в формате "день-месяц-год" (например 31 May 1995) по заданному выражению даты или типа DateTime.
DTOC()	Возвращает дату в символьном виде исходя из выражения типа Date или DateTime.
MTON()	Возвращает цифровое значение в денежном формате.
NTOM()	Возвращает денежный формат (четыре десятичных знака) в цифровое представление.
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
PROPER()	Возвращает из символьного выражения строку, переписанную прописными буквами, как имя собственное.
STR()	Возвращает символьный эквивалент заданного числового выражения.
SYS(10)	Преобразует номер дня по юлианскому календарю / SYS(1)/ в символьную строку
SYS(11)	Преобразует выражение даты или символьную строку формата даты в номер дня по юлианскому календарю.
SYS(15)	Преобразует вторую символьную строку, используя первую символьную строку.
TEXT ... ENDTEXT	Выводит строки текста, результаты выражений и функций, а также содержимое переменных памяти.
TRANSFORM()	Возвращает символьную строку, полученную из символьного или числового выражения; строка имеет формат, определенный кодами PICTURE или FUNCTION, которые применяются в команде @ ... SAY.
TTOC()	Возвращает символьное значение выражения типа DateTime.
TTOD()	Возвращает выражение типа DateTime в Date.
VAL()	Возвращает числовое значение символьного выражения, состоящего из цифр.

ТИПЫ ДАННЫХ – функции для дат и временем

Функции	Описание
BETWEEN()	Определяет, лежит ли значение некоторого выражения в диапазоне между значениями двух других выражений, имеющих тот же тип данных
CDOW()	Возвращает день недели из заданного выражения типа Date или DateTime.
CMONTH()	Возвращает название месяца из данного выражения типа Date или DateTime.
CTOD()	Преобразует символьное выражение в выражение даты.
CTOT()	Возвращает значение типа DateTime по заданному символьному выражению.
DATE()	Возвращает текущую системную дату, которая контролируется операционной системой.
DATETIME()	Возвращает текущую дату и время в формате значения заданных командами SET DATE, SET MARK, SET CENTURY, SET HOURS и SET SECONDS.
DAY()	Возвращает номер дня месяца для данного выражения даты или типа DateTime.
DMY()	Возвращает символьное выражение в формате "день-месяц-год" (например 31 May 1995) по заданному выражению даты или типа DateTime.
DOW()	Возвращает номер дня недели исходя из выражения даты или типа DateTime.
DTOC()	Возвращает дату в символьном виде исходя из выражения типа Date или DateTime.
DTOS()	Возвращает дату в виде символьной строки формата уууymmdd по заданному выражению типа Date или DateTime.
DTOT()	Возвращает значение типа DateTime, извлекаемое из выражения типа Date.
EMPTY()	Определяет, является ли значение выражения пустым, например {}.
FDATE()	Возвращает дату последней модификации файла.
FTIME()	Возвращает время последней модификации файла.
GOMONTH()	Возвращает дату, отстоящую в ту или другую сторону на указанное количество месяцев от даты, определяемой данным выражения типа Date или DateTime.
HOUR()	Возвращает часть выражения типа DateTime, показывающую часы.
INLIST()	Определяет, совпадает ли некоторое выражение с каким-либо другим выражением из заданной группы выражений. С учетом установок SET EXACT.
ISBLANK()	Определяет, является ли данное выражение пустым. ISBLANK() отличается от EMPTY() и ISNULL(). Для примера, EMPTY() возвращает (.T.) если символьное выражение содержит любые комбинации пустых значений, пробелов, табуляций, созвратов каретки или концов линий; ISBLANK() возвращает (.T.), если выражение содержит только пустые строки или пробелы
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
MAX()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наибольшее значение.
MDY()	Возвращает заданное выражение даты или типа DateTime в формате "месяц-день-год", где название месяца написано полностью.

Функции	Описание
MIN()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наименьшее значение.
MINUTE()	Возвращает часть выражения типа DateTime, показывающую минуты.
MONTH()	Возвращает номер месяца для заданного выражения даты или типа DateTime.
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
SEC()	Возвращает ту часть выражения типа DateTime, которая показывает секунды
SECONDS()	Возвращает количество секунд, прошедших после полуночи с точностью до 1 миллисекунды.
SET CENTURY	Определяет, должен ли Visual FoxPro показывать номер века в выражениях даты.
SET DATE	Задаёт формат отображаемых выражений типа даты и DateTime.
SET FLOW	Задаёт первый день недели.
SET FWEK	Задаёт требования, предъявляемые к первой неделе года.
SET HOURS	Устанавливает для системных часов 12-часовой или 24-часовой формат времени.
SET MARK TO	Задаёт разделитель, используемый при отображении дат.
SET SECONDS	Определяет, указываются ли секунды в значении типа DateTime.
SET SYSFORMATS	Определяет, должны ли системные установки Visual FoxPro обновляться вместе с текущими системными установками Windows.
SYS(1)	Возвращает текущую системную дату в виде символьной строки, определяющей номер дня по юлианскому календарю.
SYS(2)	Возвращает число секунд, прошедших после полуночи.
SYS(10)	Преобразует номер дня по юлианскому календарю / SYS(1)/ в символьную строку
SYS(11)	Преобразует выражение даты или символьную строку формата даты в номер дня по юлианскому календарю.
TIME()	Возвращает текущее системное время в 24-часовом формате в виде строки из восьми символов (hh:mm:ss).
TTOC()	Возвращает символьное значение выражения типа DateTime.
TTOD()	Возвращает выражение типа DateTime в Date.
TYPE()	Вычисляет символьное выражение и возвращает тип данных, которому принадлежит содержимое этого выражения.
WEEK()	Возвращает число, представляющее номер недели в году, извлекая его из выражения типа Date или DateTime.
YEAR()	Возвращает год исходя из заданного выражения типа Date или DateTime.

ТИПЫ ДАННЫХ – цифровые функции

Функции	Описание
% Operator	Возвращает остаток от деления одного числового выражения на другое.
ABS()	Возвращает абсолютное значение заданного числового выражения.
ACOS()	Возвращает арккосинус заданного числового выражения.
ASIN()	Возвращает арксинус заданного числового выражения.
ATAN()	Возвращает арктангенс заданного числового выражения.
ATN2()	Возвращает арктангенс заданного значения в любом из четырех квадрантов.
BETWEEN()	Определяет, лежит ли значение некоторого выражения в диапазоне между значениями двух других выражений, имеющих тот же тип данных
BINTOC()	Преобразует целые цифровые значения в бинарную строку. Обычно используется для уменьшения размера индекса, построенного на целых цифровых полях.. Обратное преобразование – CTOBIN().
BITAND()	Возвращает результат побитовой операции И (AND), выполняемой над двумя числовыми значениями.
BITCLEAR()	Сбрасывает (устанавливает равным 0) заданный бит числового значения и возвращает результирующее значение.
BITLSHIFT()	Возвращает результат сдвига битов числового значения влево на заданное количество разрядов.
BITNOT()	Возвращает результат побитовой операции НЕ (NOT), примененной к числовому значению.
BITOR()	Возвращает результат побитовой операции включающего ИЛИ (OR), выполненной над двумя числовыми значениями.
BITRSHIFT()	Возвращает результат операции сдвига битов числового значения вправо на заданное число разрядов.
BITSET()	Устанавливает бит числового значения равным 1 и возвращает результат.
BITTEST()	Возвращает значение "истина" (.T.), если заданный бит числового значения равен 1; в противном случае возвращается значение "ложь" (.F.).
BITXOR()	Возвращает результат побитовой операции исключающего ИЛИ (XOR), выполненной над двумя числовыми значениями.
CALCULATE	Выполняет финансовые и статистические операции над полями в таблице или над выражениями, включающими поля.
CEILING()	Возвращает ближайшее целое число, которое больше или равно заданному числовому выражению.
COS()	Возвращает косинус числового выражения.
CTOBIN()	Преобразует бинарное символьное выражение в целое цифровое. Обратное к BINTOC().
DTOR()	Преобразует градусы в радианы.
EMPTY()	Определяет, что цифровое выражение имеет значение ноль.
EXP()	Возвращает значение функции e^x , где x - заданное числовое выражение.

Функции	Описание
FLOOR()	Возвращает ближайшее целое число, которое меньше или равно заданному числовому выражению.
FV()	Возвращает величину будущих денежных инвестиций.
INLIST()	Определяет, совпадает ли некоторое выражение с каким-либо другим выражением из заданной группы выражений. С учетом установок SET EXACT.
INT()	Вычисляет значение числового выражения и возвращает его целую часть.
ISBLANK()	Определяет, является ли данное выражение пустым. ISBLANK() отличается от EMPTY() и ISNULL(). Для примера, EMPTY() возвращает (.T.) если символьное выражение содержит любые комбинации пустых значений, пробелов, табуляций, созвратов каретки или концов линий; ISBLANK() возвращает (.T.), если выражение содержит только пустые строки или пробелы
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
LOG()	Возвращает натуральный логарифм (по основанию e) заданного числового выражения.
LOG10()	Возвращает десятичный логарифм (по основанию 10) заданного числового выражения.
MAX()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наибольшее значение.
MIN()	Вычисляет несколько заданных выражений и возвращает выражение, имеющее наименьшее значение.
MOD()	Делит одно числовое выражение на другое и возвращает остаток.
MTON()	Возвращает цифровое значение в денежном формате.
NTOM()	Возвращает денежный формат (четыре десятичных знака) в цифровое представление.
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
PAYMENT()	Возвращает размер периодических платежей для ссуды с фиксированным процентом.
PI()	Числовое отношение длины окружности к ее диаметру, число пи.
PV()	Возвращает текущую величину инвестированных средств
RAND()	Возвращает случайное число в диапазоне от 0 до 1.
ROUND()	Возвращает числовое выражение, округленное до заданного количества десятичных знаков.
RTOD()	Преобразует радианы в градусы.
SET DECIMALS	Задаёт число десятичных знаков, отображаемых в числовых выражениях.
SET FIXED	Определяет, является ли фиксированным количество десятичных знаков в изображении числовых данных.
SET HEADINGS	Определяет, следует ли отображать для полей заголовки столбцов и включать информацию о файле, когда выдается команда TYPE, AVERAGE, CALCULATE, DISPLAY, LIST и SUM.
SET POINT	Определяет символ, используемый в качестве десятичного разделителя при изображении числовых выражений и выражений типа Currency.
SET SEPARATOR	Задаёт символ, который отделяет друг от друга группы из трех цифр, стоящие слева от десятичной точки.

Функции	Описание
SIGN()	Возвращает числовое значение 1, -1 или 0 в зависимости от значения заданного числового выражения - положительно оно, отрицательно или равно нулю.
SIN()	Возвращает синус угла.
SQRT()	Возвращает квадратный корень из заданного числового выражения.
TAN()	Эта тригонометрическая функция возвращает тангенс угла.
TYPE()	Вычисляет символьное выражение и возвращает тип данных, которому принадлежит содержимое этого выражения.
VAL()	Возвращает числовое значение символьного выражения, состоящего из цифр.

База данных – манипулирование данными

Функции	Описание
ADATABASES()	Помещает в массив переменных памяти имена всех открытых баз данных и описания путей для этих имен.
ADD TABLE	Добавляет свободную таблицу в текущую базу данных. Таблица, добавленная в базу данных, не является свободной.
ALIAS()	Возвращает псевдоним текущей или указанной рабочей области.
APPEND	Добавляет одну или несколько записей в конец таблицы.
APPEND FROM	Добавляет записи в текущую таблицу из другого файла.
APPEND FROM ARRAY	Добавляет одну запись в выбранную в данный момент таблицу для каждой строки массива и заполняет каждую запись данными из соответствующей строки массива.
APPEND PROCEDURES	Добавляет процедуры, хранящиеся в текстовом файле, к хранимым процедурам текущей базы данных.
AUSED()	Помещает в массив переменных памяти псевдонимы и рабочие области для таблиц заданного сеанса.
BEGIN TRANSACTION	Начинает транзакцию. Транзакции поддерживаются только для таблиц базы данных .
BROWSE	Открывает окно просмотра и выводит записи из текущей или указанной таблицы.
CLOSE ALL	Закрывает все открытые базы данных, таблицы, индексы во всех рабочих областях и выбирает рабочую область 1. Закрывает все открытые файлы, созданные по FCREATE() или открытые FOPEN(). Функции. CLOSE ALL не закрывает файл, открытый в SET PRINT.
CLOSE DATABASES	Закрывает текущую базу данных и ее таблицы. Если нет текущей базы данных, все открытые свободные таблицы, индексы, форматные файлы во всех рабочих областях закрываются, и выбирается рабочая область 1.
CLOSE TABLES	Закрывает все таблицы во всех открытых базах данных, закрывает все свободные таблицы во всех рабочих областях, если нет открытых баз данных.
COMPILE DATABASE	Компилирует хранимые процедуры в базе данных.
COPY PROCEDURES	Копирует хранимые процедуры текущей базы данных в текстовый файл.
COPY STRUCTURE	Создает новую, пустую таблицу с той же структурой, которую имеет выбранная в данный момент таблица
COPY STRUCTURE EXTENDED	Создает новую таблицу с полями, содержащими описание структуры выбранной в данный момент таблицы
COPY TO	Экспортирует данные из определенной таблицы в другую таблицу или в файлы других форматов, отличных от DBF. другие. Подобна команде EXPORT.
COUNT	Считает записи таблицы, удовлетворяющие определенному условию.
CPCONVERT()	Преобразует символьные поля, мемо-поля или символьные выражения в другую кодовую страницу.

Функции	Описание
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
CREATE CONNECTION	Создает именованное соединение и сохраняет его в текущей базе данных.
CREATE CURSOR - SQL	Создает временную таблицу.
CREATE DATABASE	Создает базу данных и открывает ее.
CREATE FROM	Создает таблицу на основании файла, созданного командой COPY STRUCTURE EXTENDED.
CREATE QUERY	Открывает конструктор запросов.
CREATE SQL VIEW	Выводит конструктор представлений, давая возможность создать представление SQL.
CREATE TABLE	Создает таблицу с описаниями полей.
CREATE TRIGGER	Создает для таблицы триггер Delete (удаления), Insert (добавления) или Update (обновления).
CREATE VIEW	Создает файл среды Visual FoxPro. (таблицы, отношения, ресурсный файл и пр.)
CURSORGETPROP()	Возвращает текущие установки свойств таблицы или курсора Visual FoxPro.
CURSORSETPROP()	Задаёт установки свойств для таблицы или курсора Visual FoxPro.
CURVAL()	Возвращает значение поля непосредственно с диска или из удаленного источника. При использовании с OLDVAL() дает возможность контроля за работой в многопользовательской среде для проверки изменения данных другим пользователем.
CREATEOFFLINE()	Переводит существующий обзор в offline.
DBC()	Возвращает имя текущей базы данных и путь к ней.
DBF()	Возвращает имя таблицы, открытой в заданной рабочей области, или имя таблицы из псевдонима таблицы.
DBGETPROP()	Возвращает свойство текущей базы данных, или свойство поля, именованного соединения, таблицы или представления текущей базы данных.
DBSETPROP()	Устанавливает свойство для текущей базы данных или поля, именованного соединения, таблицы или представления текущей базы данных.
DBUSED()	Возвращает значение "истина" (.T.), если указанная база данных открыта.
DELETE	Помечает записи на удаление.
DELETE – SQL	Помечает записи на удаление. В отличие от DELETE, DELETE - SQL использует блокировку записей при пометке множества записей для удаления в многопользовательском режиме. Это уменьшает проблемы при работе многих пользователей, но снижает производительность. Для максимальной производительности файлы надо открывать в монопольном режиме или блокировать командой FLOCK().
DELETE CONNECTION	Удаляет именованное соединение из текущей базы данных.
DELETE DATABASE	Удаляет базу данных с диска.

Функции	Описание
DELETE TRIGGER	Удаляет из базы данных триггер Delete (удаления), Insert (добавления) или Update (обновления) для таблицы.
DELETE VIEW	Удаляет представление SQL из текущей базы данных.
DISPLAY CONNECTIONS	Отображает информацию об именованных соединениях в текущей базе данных.
DISPLAY DATABASE	Отображает информацию о текущей базе данных или о полях, именованных соединениях, таблицах или представлениях текущей базы данных.
DISPLAY PROCEDURES	Отображает имена хранимых процедур в текущей базе данных.
DISPLAY TABLES	Отображает все таблицы, содержащиеся в текущей базе данных, и информацию о них.
DISPLAY VIEWS	Отображает информацию о представлениях SQL в текущей базе данных, указывая, на каких таблицах они базируются - на локальных или на удаленных.
DROP TABLE	Удаляет таблицу из текущей базы данных и с диска.
DROP VIEW	Удаляет SQL view из текущей базы данных. DROP VIEW идентично DELETE VIEW; DROP VIEW является стандартной командой ANSI SQL для удаления SQL view.
DROPOFFLINE()	Отменяет все изменения, сделанные в офф-лайн-обзоре и возвращает обзор в соединение он-лайн.
END TRANSACTION	Завершает текущую транзакцию, которые внесены в таблицы, мемо-файлы таблиц и файлы индексов, включенные в транзакцию, и завершает ее.
EXPORT	Копирует данные из таблицы Visual FoxPro в файл заданного формата.
FOUND()	Возвращает значение "истина" (.T.) в случае успешного выполнения команды CONTINUE, FIND, LOCATE или SEEK.
FREE TABLE	Удаляет из таблицы ссылку на базу данных.
GETFLDSTATE()	Возвращает числовое значение, показывающее, редактировалось ли поле в таблице или курсоре, добавлялись ли записи, изменился ли для текущей записи статус удаления.
GETNEXTMODIFIED()	Возвращает номер следующей модифицированной записи в буферизованном курсоре.
INDBC()	Возвращает значение "истина" (.T.), если заданный объект базы данных принадлежит текущей базе данных; в противном случае возвращается "ложь" (.F.).
ISEXCLUSIVE()	Возвращает значение "истина" (.T.), если таблица или база данных открыта для монопольного использования; в противном случае возвращается "ложь" (.F.).
ISFLOCKED()	Возвращает значение "истина" (.T.), если определенная таблица заблокирована, в противном случае возвращается "ложь" (.F.). ISFLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
ISRLOCKED()	Возвращает значение "истина" (.T.), если запись заблокирована, в противном случае возвращается "ложь" (.F.). ISRLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
KEYMATCH()	Просматривает тег индекса или файл индекса в поисках ключа индекса. Возвращает «истину», если найден индексный тэг.

Функции	Описание
LIST CONNECTIONS	Непрерывным потоком отображает информацию об именованных соединениях текущей базы данных.
LIST DATABASE	Непрерывным потоком отображает информацию о текущей базе данных.
LIST PROCEDURES	Непрерывным потоком отображает имена хранимых процедур в текущей базе данных.
LIST TABLES	Непрерывным потоком отображает список всех таблиц текущей базы данных и информацию об этих таблицах.
LIST VIEWS	Непрерывным потоком отображает информацию о представлениях SQL в текущей базе данных.
LUPDATE()	Возвращает дату последнего обновления таблицы, открытой в другой рабочей области.
MODIFY CONNECTION	Отображает конструктор соединений, давая возможность интерактивно изменить имеющееся именованное соединение, хранящееся в текущей базе данных.
MODIFY DATABASE	Открывает конструктор баз данных, давая возможность в диалоговом режиме модифицировать текущую базу данных.
MODIFY PROCEDURE	Открывает текстовый редактор Visual FoxPro, давая возможность создать новые хранимые процедуры для текущей базы данных или изменить существующие хранимые процедуры текущей базы данных.
MODIFY QUERY	Открывает конструктор запросов, в котором можно модифицировать или создать запрос.
MODIFY VIEW	Отображает конструктор представлений, давая возможность модифицировать имеющееся представление SQL.
OLDVAL()	Возвращает первоначальное значение поля, которое было модифицировано, но не обновлялось. Используется совместно с CURVAL().
OPEN DATABASE	Открывает базу данных.
PACK	Окончательно удаляет все записи, помеченные на удаление в текущей таблице, и сокращает размер мето-файла, связанного с данной таблицей.
PACK DATABASE	Убирает из текущей базы данных записи, помеченные на удаление.
REFRESH()	Обновляет данные в обновляемом представлении SQL.
RELATION()	Возвращает указанное реляционное выражение для таблицы, открытой в заданной рабочей области.
REMOVE TABLE	Удаляет таблицу из текущей базы данных. При удалении таблицы из базы данных, она становится свободной таблицей и может быть добавлена в другую базу данных. Таблицы добавляются командой ADD TABLE. При выполнении REMOVE TABLE все первичные индексы, значения по умолчанию, правила проверки, связанные с таблицей, удаляются.
RENAME CONNECTION	Переименовывает именованное соединение текущей базы данных.
RENAME TABLE	Переименовывает таблицу текущей базы данных.
RENAME VIEW	Переименовывает представление текущей SQL базы данных.
REQUERY()	Повторно извлекает данные для представления SQL.

Функции	Описание
ROLLBACK	Отменяет все изменения, внесенные в ходе текущей транзакции. состояние таблиц, табличных мето-файлов и файлов индексов, в котором они находились в момент начала транзакции.
SEEK	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск.
SEEK()	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск. Возвращает логическое значение «истина» .Т., если запись найдена..
SELECT	Активизирует заданную рабочую область.
SELECT()	Возвращает номер рабочей области, выбранной в данный момент, или наибольший среди номеров неиспользуемых рабочих областей. текущей рабочей области.
SELECT - SQL	Стандартный запрос SQL для получения данных из одной или нескольких таблиц. Обычно используется в обзорах или при получении информации для определенного набора критериев.
SET AUTOSAVE	Определяет, должна ли система Visual FoxPro сбрасывать буферы данных на диск, когда вы выходите из команды READ или возвращаетесь в окно команд.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мето-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET CPDIALOG	Определяет, будет ли при открытии таблицы отображаться диалоговое окно Code Page.
SET DATABASE	Задаёт текущую базу данных.
SET DATASESSION	Активизирует заданный сеанс формы.
SET DELETED	Определяет, будет ли Visual FoxPro обрабатывать записи, помеченные на удаление, и будут ли они доступны для использования в других командах.
SET EXCLUSIVE	Определяет, в каком режиме Visual FoxPro открывает файлы таблиц в сети - для монопольного пользования или для совместного.
SET FIELDS	Определяет поля в таблице, к которым разрешен доступ.
SET FILTER	Задаёт условие, соблюдение которого делает записи текущей таблицы доступными.
SET FULLPATH	Определяет, должны ли функции CDX(), DBF(), MDX() и NDX() возвращать путь вместе с именем файла.
SET INDEX	Открывает один или несколько файлов индексов для текущей таблицы.
SET KEY	Задаёт способ доступа к записям некоторого диапазона исходя из их ключей индекса.

Функции	Описание
SET LOCK	Разрешает или запрещает автоматическую блокировку файлов для определенных команд.
SET MULTLOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.
SET NEAR	Определяет, где располагается указатель записи после неудачного выполнения поиска записи командой FIND, SEEK или SEEK().
SET ORDER	Задаёт управляющий файл или тег индекса для данной таблицы.
SET NULL	Определяет, каким образом поддерживаются значения NULL командами ALTER TABLE, CREATE TABLE и INSERT.
SET OPTIMIZE	Включает или отключает оптимизацию по технологии Rushmore.
SET REFRESH	Определяет, нужно ли (и как часто) обновлять окно просмотра, отражая в нем изменения, внесенные в записи другими пользователями сети.
SET RELATION	Устанавливает отношение между двумя открытыми таблицами.
SET RELATION OFF	Разрывает установленное отношение между родительской таблицей в текущей рабочей области и связанной с ней дочерней таблицей.
SET REPROCESS	Указывает, сколько раз или как долго Visual FoxPro может повторять попытки блокировать файл или запись после неудачной попытки блокировки.
SET SKIP	Создает отношение один-ко-многим между таблицами.
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления.
SETFLDSTATE()	Присваивает значение состояния поля или статуса удаления полю или записи в таблице или курсоре.
SQLCANCEL()	Запрашивает отмену выполняемого оператора SQL.
SQLCOLUMNS()	Сохраняет в курсоре Visual FoxPro список имен столбцов заданной таблицы источника данных и информацию об этих столбцах.
SQLCOMMIT()	Фиксирует транзакцию.
SQLCONNECT()	Устанавливает соединение с источником данных.
SQLDISCONNECT()	Разрывает соединение с источником данных.
SQLEXEC()	Посылает оператор SQL в источник данных, где этот оператор подвергается обработке.
SQLGETPROP()	Возвращает текущие или стандартные установки для активного соединения.
SQLMORERESULTS()	Копирует еще одно результирующее множество в курсор Visual FoxPro, если такие результирующие множества еще есть.
SQLROLLBACK()	Аннулирует все изменения, внесенные в ходе текущей транзакции.
SQLSETPROP()	Задаёт установки для активного соединения.
SQLSTRINGCONNECT()	Устанавливает соединение с источником данных с использованием строки соединения.
SQLTABLES()	Сохраняет в курсоре Visual FoxPro имена таблиц источника данных.
SYS(2012)	Возвращает размер блока мемо-поля, установленный для данной таблицы.

Функции	Описание
SYS(2029)	Возвращает значение, соответствующее типу таблицы. (например, FoxPro, Visual FoxPro, Dbase IV, etc...).
SYS(3051)	Задаёт интервал времени (в миллисекундах), в течение которого Visual FoxPro ждёт, прежде чем попытается снова заблокировать запись, таблицу, мемо-файл или файл индекса после предыдущей неудачной попытки. Используйте SET REPROCESS.
SYS(3052)	Определяет, будет ли Visual FoxPro использовать установку SET REPROCESS при попытке заблокировать файл индекса или мемо-файл.
SYS(3054)	Определяет, будет ли выводиться уровень оптимизации Rushmore для запроса.
TABLEREVERT()	Сбрасывает изменения, внесенные в буферизованную запись или в буферизованную таблицу, или в курсор, и восстанавливает содержимое по данным OLDVAL() для удаленных курсоров или по текущим значениям на диске для локальных таблиц и курсоров.
TABLEUPDATE()	Фиксирует изменения, внесенные в буферизованную запись либо в буферизованную таблицу или курсор.
TARGET()	Возвращает псевдоним таблицы, с которой устанавливается отношение согласно предложению INTO команды SET RELATION.
TOTAL	Вычисляет итоговые суммы по числовым полям в текущей таблице.
_TRIGGERLEVEL	Содержит доступное только для чтения числовое значение, показывающее уровень вложенности текущей процедуры триггера
TXNLEVEL()	Возвращает числовое значение, указывающее уровень текущей транзакции.
UNLOCK	Снимает блокировку записи, блокировку группы записей или блокировку файла с данной таблицы либо снимает блокировки всех записей и файлов для всех открытых таблиц.
UPDATE - SQL	Обновляет записи в таблице актуальными значениями.
VALIDATE DATABASE	Обеспечивает корректное размещение таблиц и индексов в текущей базе данных.

База данных – работа с полями

Функции	Описание
AFIELDS()	Помещает информацию о структуре текущей таблицы в массив и возвращает количество полей в таблице.
APPEND GENERAL	Импортирует объект OLE из файла и помещает его в поле типа General.
APPEND MEMO	Копирует содержимое текстового файла в мемо-поле.
BLANK	Стирает информацию во всех полях текущей записи, если не задано никаких аргументов.
CALCULATE	Выполняет финансовые и статистические операции над полями в таблице или над выражениями, включающими поля.
COPY TO ARRAY	Копирует данные в массив из таблицы, выбранной в данный момент.
CURVAL()	Возвращает значение поля непосредственно с диска или из удаленного источника. При использовании с OLDVAL() дает возможность контроля за работой в многопользовательской среде для проверки изменения данных другим пользователем.
FCOUNT()	Возвращает количество полей в таблице.
FIELD()	Возвращает имя поля таблицы, указанного номером.
FLDLIST()	Возвращает поля и выражения вычисляемых полей для списка полей, заданного командой SET FIELDS.
FSIZE()	Возвращает величину заданного поля или файла в байтах.
GATHER	Замещает данные в текущей записи текущей выбранной таблицы содержимым массива, группы переменных памяти или объекта.
ISBLANK()	Определяет, является ли данное выражение пустым. ISBLANK() отличается от EMPTY() и ISNULL(). Для примера, EMPTY() возвращает (.T.) если символьное выражение содержит любые комбинации пустых значений, пробелов, табуляций, созвратов каретки или концов линий; ISBLANK() возвращает (.T.), если выражение содержит только пустые строки или пробелы
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
MEMLINES()	Возвращает число строк в мемо-поле. Зависит от установок SET MEMOWIDTH.
MLINE()	Возвращает указанную строку мемо-поля в виде символьной строки. Зависит от установок SET MEMOWIDTH.
MODIFY GENERAL	Открывает окна редактирования для полей типа General из текущей записи.
MODIFY MEMO	Открывает окно редактирования для полей типа мемо из текущей записи.
NDX()	Возвращает имя открытого файла индекса (.IDX) для текущей или заданной таблицы.
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
OLDVAL()	Возвращает первоначальное значение поля, которое было модифицировано, но не обновлялось. Используется совместно с CURVAL().

Функции	Описание
RATLINE()	Возвращает номер строки с последним вхождением символьного выражения в другое символьное выражение или в мемо-поле, считая от последней строки.
SCATTER	Копирует содержимое текущей записи в набор переменных памяти или в массив.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мемо-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET NOCPTRANS	Запрещает преобразовывать в другую кодовую страницу выбранные поля открытой таблицы.
SET TEXTMERGE	Включает или отключает вычисление полей, переменных памяти, элементов массива, функций или выражений, заключённых в разделители слияния, и позволяет задать вывод процедуры слияния текста.
SET TEXTMERGE DELIMITERS	Задаёт разделители слияния текста.
TOTAL	Вычисляет итоговые суммы по числовым полям в текущей таблице.

База данных - индексы

Функции	Описание
BINTOC()	Преобразует целые цифровые значения в бинарную строку. Обычно используется для уменьшения размера индекса, построенного на целых цифровых полях.. Обратное преобразование – CTOBIN().
BOF()	Определяет, установлен ли указатель записи в начале таблицы.
CANDIDATE()	Возвращает значение "истина" (.T.), если тег индекса является тегом индекса-кандидата; в противном случае возвращает "ложь" (.F.).
CDX()	Возвращает имя открытого файла составного индекса позиции индекса (.CDX)
CONTINUE	Продолжает работу предыдущей команды LOCATE. команда LOCATE нашла нужную запись, и инициирует перемещает указатель записи к следующей записи, для которой значение логического выражения, заданного в предыдущей команде LOCATE, есть "истина" (.T.).
COPY INDEXES	Создает теги составного индекса на основе файлов простых индексов .IDX.
COPY TAG	Создает файл простого индекса (.IDX) на основе тега из файла составного индекса.
CPCONVERT()	Преобразует символьные поля, мемо-поля или символьные выражения в другую кодовую страницу.
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
CTOBIN()	Преобразует бинарное символьное выражение в целое цифровое. Обратна к BINTOC().
DESCENDING()	Возвращает логическое значение, указывающее, было создании тега индекса или в командах USE, SET INDEX или SET ORDER с параметром DESCENDING .
DELETE TAG	Удаляет тег или теги из файла составного индекса (.CDX).
EOF()	Определяет, установлен ли указатель записи после последней записи в текущей или заданной таблице.
FOR()	Возвращает выражение индексного фильтра для открытого файла простого индекса .IDX или для тега индекса.
FOUND()	Возвращает значение "истина" (.T.) в случае успешного выполнения команды CONTINUE, FIND, LOCATE или SEEK.
IDXCOLLATE()	Возвращает последовательность сортировки для индекса или тега индекса.
INDEX	Создает файл индекса, позволяющий отображать и обрабатывать табличные записи в логическом порядке.
KEY()	Возвращает выражение ключа индекса для тега индекса или файла индекса.
KEYMATCH()	Просматривает тег индекса или файл индекса в поисках ключа индекса. Возвращает «истину», если найден индексный тэг.
LOCATE	Последовательно просматривает таблицу в поисках первой записи, совпадающей с заданным логическим выражением.
LOOKUP()	Ищет в таблице первую запись, в которой значение поля совпадает с заданным выражением.

Функции	Описание
MDX()	Возвращает имя открытого файла составного индекса .CDX, соответствующего заданному номеру позиции индекса. MDX() аналогична CDX().
NDX()	Возвращает имя открытого файла индекса (.IDX) для текущей или заданной таблицы.
ORDER()	Возвращает имя управляющего файла индекса или тега для текущей или заданной таблицы.
PACK	Окончательно удаляет все записи, помеченные на удаление в текущей таблице, и сокращает размер мето-файла, связанного с данной таблицей.
PACK DATABASE	Убирает из текущей базы данных записи, помеченные на удаление.
PRIMARY()	Возвращает значение "истина" (.T.), если тег индекса является тегом первичного индекса; в противном случае возвращает "ложь" (.F.).
REINDEX	Перестраивает открытые файлы индексов.
SEEK	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск.
SEEK()	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск. Возвращает логическое значение «истина» .T., если запись найдена..
SELECT	Активизирует заданную рабочую область.
SELECT()	Возвращает номер рабочей области, выбранной в данный момент, или наибольший среди номеров неиспользуемых рабочих областей. текущей рабочей области.
SELECT - SQL	Стандартный запрос SQL для получения данных из одной или нескольких таблиц. Обычно используется в обзорах или при получении информации для определенного набора критериев.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET FILTER	Задаёт условие, соблюдение которого делает записи текущей таблицы доступными.
SET FULLPATH	Определяет, должны ли функции CDX(), DBF(), MDX() и NDX() возвращать путь вместе с именем файла.
SET INDEX	Открывает один или несколько файлов индексов для текущей таблицы.
SET KEY	Задаёт способ доступа к записям некоторого диапазона исходя из их ключей индекса.
SET NEAR	Определяет, где располагается указатель записи после неудачного выполнения поиска записи командой FIND, SEEK или SEEK().
SET ORDER	Задаёт управляющий файл или тег индекса для данной таблицы.
SET UNIQUE	Определяет, поддерживает ли файл индекса записи с повторяющимися значениями ключа индекса.
SYS(14)	Возвращает выражение индекса для открытого файла простого индекса .IDX или индексные выражения для тегов в файлах составного индекса .CDX. Команда эквивалентна KEY().

Функции	Описание
SYS(21)	Возвращает, в качестве символьной строки, номер позиции индекса для главного управляющего тега составного индекса .CDX или главного управляющего файла индекса .IDX в рабочей области, выбранной в данный момент.
SYS(22)	Возвращает имя главного управляющего тега составного индекса .CDX или главного управляющего файла индекса .IDX для данной таблицы.
SYS(2021)	Возвращает выражение фильтра для открытого файла простого индекса (.IDX) или выражения фильтра для тегов в файлах составного индекса (.CDX).
SYS(3054)	Определяет, будет ли выводиться уровень оптимизации Rushmore для запроса.
TAG()	Возвращает имя тега из открытого файла многоключевого составного индекса .CDX или имя открытого файла простого индекса .IDX.
TAGCOUNT()	Возвращает число тегов файлов составных индексов .CDX и открытых файлов простых индексов .IDX.
TAGNO()	Возвращает индексную позицию для тегов файла составного индекса .CDX и для открытых файлов простых индексов .IDX.
UNIQUE()	Возвращает значение "истина" (.T.), если заданный тег индекса или файл индекса был создан с параметром UNIQUE или конструкцией SET UNIQUE ON; в противном - "ложь" (.F.).

База данных – работа с записями

Функции	Описание
APPEND	Добавляет одну или несколько записей в конец таблицы.
APPEND FROM	Добавляет записи в текущую таблицу из другого файла.
APPEND FROM ARRAY	Добавляет одну запись в выбранную в данный момент таблицу для каждой строки массива и заполняет каждую запись данными из соответствующей строки массива.
APPEND MEMO	Копирует содержимое текстового файла в мемо-поле.
BEGIN TRANSACTION	Начинает транзакцию. Транзакции поддерживаются только для таблиц базы данных .
BLANK	Стирает информацию во всех полях текущей записи, если не задано никаких аргументов.
BOF()	Определяет, установлен ли указатель записи в начале таблицы.
BROWSE	Открывает окно просмотра и выводит записи из текущей или указанной таблицы.
CALCULATE	Выполняет финансовые и статистические операции над полями в таблице или над выражениями, включающими поля.
CONTINUE	Продолжает работу предыдущей команды LOCATE. команда LOCATE нашла нужную запись, и иницирует перемещает указатель записи к следующей записи, для которой значение логического выражения, заданного в предыдущей команде LOCATE, есть "истина" (.T.).
COPY MEMO	Копирует содержимое заданного мемо-поля из текущей записи в текстовый файл.
COPY TO	Экспортирует данные из определенной таблицы в другую таблицу или в файлы других форматов, отличных от DBF. другие. Подобна команде EXPORT.
COPY TO ARRAY	Копирует данные в массив из таблицы, выбранной в данный момент.
COUNT	Считает записи таблицы, удовлетворяющие определенному условию.
CPCONVERT()	Преобразует символьные поля, мемо-поля или символьные выражения в другую кодовую страницу.
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
CURVAL()	Возвращает значение поля непосредственно с диска или из удаленного источника. При использовании с OLDVAL() дает возможность контроля за работой в многопользовательской среде для проверки изменения данных другим пользователем.
DELETE	Помечает записи на удаление.
DELETE - SQL	Помечает записи на удаление. В отличие от DELETE, DELETE - SQL использует блокировку записей при попытке удаления множества записей в таблицах, открытых для совместного доступа.
DELETED()	Возвращает логическое значение, которое показывает, помечена ли текущая запись на удаление.
END TRANSACTION	Завершает текущую транзакцию. которые внесены в таблицы, мемо-файлы таблиц и файлы индексов, включенные в транзакцию, и завершает ее.

Функции	Описание
EOF()	Определяет, установлен ли указатель записи после последней записи в текущей или заданной таблице.
EXPORT	Копирует данные из таблицы Visual FoxPro в файл заданного формата.
FCOUNT()	Возвращает количество полей в таблице.
FILTER()	Возвращает выражение фильтра таблицы, заданное в рабочей области.
FOUND()	Возвращает значение "истина" (.T.) в случае успешного выполнения команды CONTINUE, FIND, LOCATE или SEEK.
GATHER	Замещает данные в текущей записи текущей выбранной таблицы содержимым массива, группы переменных памяти или объекта.
GETFLDSTATE()	Возвращает числовое значение, показывающее, редактировалось ли поле в таблице или курсоре, добавлялись ли записи, изменился ли для текущей записи статус удаления.
GETNEXT MODIFIED()	Возвращает номер следующей модифицированной записи в буферизованном курсоре.
GO/GOTO	Перемещает указатель записи к записи с заданным номером.
INSERT	Вставляет в текущую таблицу новую запись.. Используйте APPEND или INSERT - SQL .
INSERT - SQL	Добавляет запись в конец таблицы, содержащей заданные значения полей.
ISBLANK()	Определяет, является ли данное выражение пустым. ISBLANK() отличается от EMPTY() и ISNULL(). Для примера, EMPTY() возвращает (.T.) если символьное выражение содержит любые комбинации пустых значений, пробелов, табуляций, созвратов каретки или концов линий; ISBLANK() возвращает (.T.), если выражение содержит только пустые строки или пробелы
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
ISRLOCKED()	Возвращает значение "истина" (.T.), если запись заблокирована, в противном случае возвращается "ложь" (.F.). ISRLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
KEYMATCH()	Просматривает тег индекса или файл индекса в поисках ключа индекса. Возвращает «истину», если найден индексный тэг.
LOCATE	Последовательно просматривает таблицу в поисках первой записи, совпадающей с заданным логическим выражением.
LOCK()	Пытается заблокировать одну или несколько записей в таблице.
LOOKUP()	Ищет в таблице первую запись, в которой значение поля совпадает с заданным выражением.
MEMLINES()	Возвращает число строк в мемо-поле. Зависит от установок SET MEMOWIDTH.
MLINE()	Возвращает указанную строку мемо-поля в виде символьной строки. Зависит от установок SET MEMOWIDTH.
MODIFY GENERAL	Открывает окна редактирования для полей типа General из текущей записи.
MODIFY MEMO	Открывает окно редактирования для полей типа мемо из текущей записи.

Функции	Описание
OLDVAL()	Возвращает первоначальное значение поля, которое было модифицировано, но не обновлялось. Используется совместно с CURVAL().
ORDER()	Возвращает имя управляющего файла индекса или тега для текущей или заданной таблицы.
PACK	Окончательно удаляет все записи, помеченные на удаление в текущей таблице, и сокращает размер мето-файла, связанного с данной таблицей.
PACK DATABASE	Убирает из текущей базы данных записи, помеченные на удаление.
RATLINE()	Возвращает номер строки с последним вхождением символьного выражения в другое символьное выражение или в мето-поле, считая от последней строки.
RECALL	Снимает отметку с записей выбранной таблицы, помеченных на удаление.
RECCOUNT()	Возвращает число записей в текущей или заданной таблице.
RECNO()	Возвращает номер текущей записи в текущей или заданной таблице.
RECSIZE()	Возвращает размер (ширину) записи таблицы.
REFRESH()	Обновляет данные в обновляемом представлении SQL.
REPLACE	Обновляет записи в таблице.
REPLACE FROM ARRAY	Обновляет содержимое полей значениями из массива переменных памяти.
REQUERY()	Повторно извлекает данные для представления SQL.
RLOCK()	Пытается заблокировать одну или несколько записей таблицы. RLOCK() аналогична LOCK().
ROLLBACK	Отменяет все изменения, внесенные в ходе текущей транзакции. состояние таблиц, табличных мето-файлов и файлов индексов, в котором они находились в момент начала транзакции.
SCAN ... ENDSCAN	Перемещает указатель записи по таблице, выбранной в данный момент, и для каждой записи, удовлетворяющей указанным условиям, выполняет определенный блок команд.
SCATTER	Копирует содержимое текущей записи в набор переменных памяти или в массив.
SEEK	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск.
SEEK()	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск. Возвращает логическое значение «истина» .T., если запись найдена..
SELECT - SQL	Стандартный запрос SQL для получения данных из одной или нескольких таблиц. Обычно используется в обзорах или при получении информации для определенного набора критериев.
SET AUTOSAVE	Определяет, должна ли система Visual FoxPro сбрасывать буферы данных на диск, когда вы выходите из команды READ или возвращаетесь в окно команд.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мето-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.

Функции	Описание
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET DELETED	Определяет, будет ли Visual FoxPro обрабатывать записи, помеченные на удаление, и будут ли они доступны для использования в других командах.
SET FIELDS	Определяет поля в таблице, к которым разрешен доступ.
SET FILTER	Задаёт условие, соблюдение которого делает записи текущей таблицы доступными.
SET KEY	Задаёт способ доступа к записям некоторого диапазона исходя из их ключей индекса.
SET LOCK	Разрешает или запрещает автоматическую блокировку файлов для определенных команд.
SET MULTILOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.
SET NEAR	Определяет, где располагается указатель записи после неудачного выполнения поиска записи командой FIND, SEEK или SEEK().
SET ORDER	Задаёт управляющий файл или тег индекса для данной таблицы.
SET ODOMETER	Задаёт интервал, с которым команды, обрабатывающие записи, сообщают значения счетчика записей.
SET OPTIMIZE	Включает или отключает оптимизацию по технологии Rushmore.
SET REFRESH	Определяет, нужно ли (и как часто) обновлять окно просмотра, отражая в нем изменения, внесенные в записи другими пользователями сети.
SET RELATION	Устанавливает отношение между двумя открытыми таблицами.
SET RELATION OFF	Разрывает установленное отношение между родительской таблицей в текущей рабочей области и связанной с ней дочерней таблицей.
SET REPROCESS	Указывает, сколько раз или как долго Visual FoxPro может повторять попытки блокировать файл или запись после неудачной попытки блокировки.
SET SKIP	Создаёт отношение один-ко-многим между таблицами.
SKIP	Перемещает указатель записи вперед или назад по таблице.
SYS(3051)	Задаёт интервал времени (в миллисекундах), в течение которого Visual FoxPro ждёт, прежде чем попытается снова блокировать запись, таблицу, мемо-файл или файл индекса после предыдущей неудачной попытки. Используйте SET REPROCESS.
SYS(3052)	Определяет, будет ли Visual FoxPro использовать установку SET REPROCESS при попытке блокировать файл индекса или мемо-файл.
SYS(3054)	Определяет, будет ли выводиться уровень оптимизации Rushmore для запроса.
TOTAL	Вычисляет итоговые суммы по числовым полям в текущей таблице.
TXNLEVEL()	Возвращает числовое значение, указывающее уровень текущей транзакции.
UNLOCK	Снимает блокировку записи, блокировку группы записей или блокировку файла с данной таблицы либо снимает блокировки всех записей и файлов для всех открытых таблиц.
UPDATE - SQL	Обновляет записи в таблице актуальными значениями.
ZAP	Удаляет из таблицы все записи, оставляя лишь структуру таблицы. При использовании ZAP не работает триггер Delete.

База данных – отношения

Функции	Описание
RELATION()	Возвращает указанное реляционное выражение для таблицы, открытой в заданной рабочей области.
SELECT - SQL	Стандартный запрос SQL для получения данных из одной или нескольких таблиц. Обычно используется в обзорах или при получении информации для определенного набора критериев.
SET RELATION	Устанавливает отношение между двумя открытыми таблицами.
SET RELATION OFF	Разрывает установленное отношение между родительской таблицей в текущей рабочей области и связанной с ней дочерней таблицей.
SET SKIP	Создает отношение один-ко-многим между таблицами.
TARGET()	Возвращает псевдоним таблицы, с которой устанавливается отношение согласно предложению INTO команды SET RELATION.

База данных – работа с таблицами

Функции	Описание
ADD TABLE	Добавляет свободную таблицу в текущую базу данных. Таблица, добавленная в базу данных, не является свободной.
AFIELDS()	Помещает информацию о структуре текущей таблицы в массив и возвращает количество полей в таблице.
ALIAS()	Возвращает псевдоним текущей или указанной рабочей области.
ALTER TABLE - SQL	Программно изменяется структура таблицы.
APPEND	Добавляет одну или несколько записей в конец таблицы.
APPEND FROM	Добавляет записи в текущую таблицу из другого файла.
APPEND FROM ARRAY	Добавляет одну запись в выбранную в данный момент таблицу для каждой строки массива и заполняет каждую запись данными из соответствующей строки массива.
AUSED()	Помещает в массив переменных памяти псевдонимы и рабочие области для таблиц заданного сеанса.
AVERAGE	Вычисляет арифметическое среднее числовых выражений или полей.
BEGIN TRANSACTION	Начинает транзакцию. Транзакции поддерживаются только для таблиц базы данных .
BOF()	Определяет, установлен ли указатель записи в начале таблицы.
BROWSE	Открывает окно просмотра и выводит записи из текущей или указанной таблицы.
CHANGE	Вывод полей для редактирования. Работает как EDIT.
CLOSE ALL	Закрывает все открытые базы данных, таблицы, индексы во всех рабочих областях и выбирает рабочую область 1. Закрывает все открытые файлы, созданные по FCREATE() или открытые FOPEN(). Функции. CLOSE ALL не закрывает файл, открытый в SET PRINT.
CLOSE DATABASES	Закрывает текущую базу данных и ее таблицы. Если нет текущей базы данных, все открытые свободные таблицы, индексы, форматные файлы во всех рабочих областях закрываются, и выбирается рабочая область 1.
CLOSE TABLES	Закрывает все таблицы во всех открытых базах данных, закрывает все свободные таблицы во всех рабочих областях, если нет открытых баз данных.
COPY MEMO	Копирует содержимое заданного мемо-поля из текущей записи в текстовый файл.
COPY STRUCTURE	Создает новую, пустую таблицу с той же структурой, которую имеет выбранная в данный момент таблица
COPY STRUCTURE EXTENDED	Создает новую таблицу с полями, содержащими описание структуры выбранной в данный момент таблицы
COPY TO	Экспортирует данные из определенной таблицы в другую таблицу или в файлы других форматов, отличных от DBF. другие. Подобна команде EXPORT.
COPY TO ARRAY	Копирует данные в массив из таблицы, выбранной в данный момент.
COUNT	Считает записи таблицы, удовлетворяющие определенному условию.

Функции	Описание
CPCONVERT()	Преобразует символьные поля, мемо-поля или символьные выражения в другую кодовую страницу.
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
CREATE	Создает таблицу новую таблицу Visual FoxPro.
CREATE CURSOR - SQL	Создает временную таблицу.
CREATE FROM	Создает таблицу на основании файла, созданного командой COPY STRUCTURE EXTENDED.
CREATE QUERY	Открывает конструктор запросов.
CREATE SQL VIEW	Выводит конструктор представлений, давая возможность создать представление SQL.
CREATE TABLE	Создает таблицу с описаниями полей.
CREATE TRIGGER	Создает для таблицы триггер Delete (удаления), Insert (добавления) или Update (обновления).
CREATE VIEW	Создает файл среды Visual FoxPro. Команда SET VIEW восстанавливает среду, которая создана командой CREATE VIEW. Файлы среды имеют расширение .VUE.
CURSORGETPROP()	Возвращает текущие установки свойств таблицы или курсора Visual FoxPro.
CURSORSETPROP()	Задаёт установки свойств для таблицы или курсора Visual FoxPro.
CURVAL()	Возвращает значение поля непосредственно с диска или из удаленного источника. При использовании с OLDVAL() дает возможность контроля за работой в многопользовательской среде для проверки изменения данных другим пользователем.
DBF()	Возвращает имя таблицы, открытой в заданной рабочей области, или имя таблицы из псевдонима таблицы.
DELETE	Помечает записи на удаление.
DELETE - SQL	Помечает записи на удаление. В отличие от DELETE, DELETE - SQL использует блокировку записей при попытке удаления множества записей в таблицах, открытых для совместного доступа.
DELETED()	Возвращает логическое значение, которое показывает, помечена ли текущая запись на удаление.
DISPLAY STRUCTURE	Отображает структуру файла таблицы.
DISPLAY TABLES	Отображает все таблицы, содержащиеся в текущей базе данных, и информацию о них.
DROP TABLE	Удаляет таблицу из текущей базы данных и с диска.
EDIT	Выводит поля для редактирования.
END TRANSACTION	Завершает текущую транзакцию. которые внесены в таблицы, мемо-файлы таблиц и файлы индексов, включенные в транзакцию, и завершает ее.

Функции	Описание
EOF()	Определяет, установлен ли указатель записи после последней записи в текущей или заданной таблице.
EXPORT	Копирует данные из таблицы Visual FoxPro в файл заданного формата.
FCOUNT()	Возвращает количество полей в таблице.
FILTER()	Возвращает выражение фильтра таблицы, заданное в рабочей области.
FLOCK()	Пытается заблокировать текущую или заданную таблицу.
FLUSH	Немедленно сохраняет на диске изменения в таблице и индексе.
FOUND()	Возвращает значение "истина" (.T.) в случае успешного выполнения команды CONTINUE, FIND, LOCATE или SEEK.
FREE TABLE	Удаляет из таблицы ссылку на базу данных.
GATHER	Замещает данные в текущей записи текущей выбранной таблицы содержимым массива, группы переменных памяти или объекта.
GETFLDSTATE()	Возвращает числовое значение, показывающее, редактировалось ли поле в таблице или курсоре, добавлялись ли записи, изменился ли для текущей записи статус удаления.
GETNEXTMODIFIED()	Возвращает номер следующей модифицированной записи в буферизованном курсоре.
HEADER()	Возвращает число байтов в заголовке файла текущей или заданной таблицы.
IMPORT	Импортирует данные из файла внешнего формата, создавая новую таблицу Visual FoxPro.
INSERT - SQL	Добавляет запись в конец таблицы, содержащей заданные значения полей.
ISEXCLUSIVE()	Возвращает значение "истина" (.T.), если таблица или база данных открыта для монопольного использования; в противном случае возвращается "ложь" (.F.).
ISFLOCKED()	Возвращает значение "истина" (.T.), если определенная таблица заблокирована, в противном случае возвращается "ложь" (.F.). ISFLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
ISREADONLY()	Определяет, открыта ли таблица с атрибутом "только для чтения".
ISRLOCKED()	Возвращает значение "истина" (.T.), если запись заблокирована, в противном случае возвращается "ложь" (.F.). ISRLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
KEY()	Возвращает выражение ключа индекса для тега индекса или файла индекса.
KEYMATCH()	Просматривает тег индекса или файл индекса в поисках ключа индекса. Возвращает «истину», если найден индексный тэг.
LIST TABLES	Непрерывным потоком отображает список всех таблиц текущей базы данных и информацию об этих таблицах.
LOCK()	Пытается заблокировать одну или несколько записей в таблице.
LUPDATE()	Возвращает дату последнего обновления таблицы, открытой в другой рабочей области.

Функции	Описание
MEMLINES()	Возвращает число строк в мемо-поле. Зависит от установок SET MEMOWIDTH.
MLINE()	Возвращает указанную строку мемо-поля в виде символьной строки. Зависит от установок SET MEMOWIDTH.
MODIFY GENERAL	Открывает окна редактирования для полей типа General из текущей записи.
MODIFY MEMO	Открывает окно редактирования для полей типа мемо из текущей записи.
MODIFY QUERY	Открывает конструктор запросов, в котором можно модифицировать или создать запрос.
MODIFY STRUCTURE	Отображает диалоговое окно Конструктор таблиц, в котором можно модифицировать структуру таблицы.
OLDVAL()	Возвращает первоначальное значение поля, которое было модифицировано, но не обновлялось. Используется совместно с CURVAL().
ORDER()	Возвращает имя управляющего файла индекса или тега для текущей или заданной таблицы.
PACK	Окончательно удаляет все записи, помеченные на удаление в текущей таблице, и сокращает размер мемо-файла, связанного с данной таблицей.
PACK DATABASE	Убирает из текущей базы данных записи, помеченные на удаление.
RATLINE()	Возвращает номер строки с последним вхождением символьного выражения в другое символьное выражение или в мемо-поле, считая от последней строки.
RECALL	Снимает отметку с записей выбранной таблицы, помеченных на удаление.
RECCOUNT()	Возвращает число записей в текущей или заданной таблице.
RECNO()	Возвращает номер текущей записи в текущей или заданной таблице.
RECSIZE()	Возвращает размер (ширину) записи таблицы.
REFRESH()	Обновляет данные в обновляемом представлении SQL.
RELATION()	Возвращает указанное реляционное выражение для таблицы, открытой в заданной рабочей области.
REINDEX	Перестраивает открытые файлы индексов.
REMOVE TABLE	Удаляет таблицу из текущей базы данных. При удалении таблицы из базы данных, она становится свободной таблицей и может быть добавлена в другую базу данных. Таблицы добавляются командой ADD TABLE. При выполнении REMOVE TABLE все первичные индексы, значения по умолчанию, правила проверки, связанные с таблицей, удаляются.
REPLACE	Обновляет записи в таблице.
REPLACE FROM ARRAY	Обновляет содержимое полей значениями из массива переменных памяти.
RENAME TABLE	Переименовывает таблицу текущей базы данных.
REQUERY()	Повторно извлекает данные для представления SQL.
RLOCK()	Пытается заблокировать одну или несколько записей таблицы. RLOCK() идентична LOCK().

Функции	Описание
ROLLBACK	Отменяет все изменения, внесенные в ходе текущей транзакции. состояние таблиц, табличных мето-файлов и файлов индексов, в котором они находились в момент начала транзакции.
SCAN ... ENDSCAN	Перемещает указатель записи по таблице, выбранной в данный момент, и для каждой записи, удовлетворяющей указанным условиям, выполняет определенный блок команд.
SCATTER	Копирует содержимое текущей записи в набор переменных памяти или в массив.
SEEK	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск.
SEEK()	Просматривает индексированную таблицу в поисках первого вхождения записи, ключ индекса которой возвращает логическое значение, указывающее, насколько успешным был поиск. Возвращает логическое значение «истина» .T., если запись найдена..
SELECT	Активизирует заданную рабочую область.
SELECT()	Возвращает номер рабочей области, выбранной в данный момент, или наибольший среди номеров неиспользуемых рабочих областей. текущей рабочей области.
SELECT - SQL	Стандартный запрос SQL для получения данных из одной или нескольких таблиц. Обычно используется в обзорах или при получении информации для определенного набора критериев.
SET ANSI	Определяет, каким образом оператор = в командах Visual FoxPro SQL будет сравнивать строки, имеющие разную длину.
SET AUTOSAVE	Определяет, должна ли система Visual FoxPro сбрасывать буферы данных на диск, когда вы выходите из команды READ или возвращаетесь в окно команд.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мето-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET CPDIALOG	Определяет, будет ли при открытии таблицы отображаться диалоговое окно Code Page.
SET DATABASE	Задаёт текущую базу данных.
SET DATASESSION	Активизирует заданный сеанс формы.
SET DELETED	Определяет, будет ли Visual FoxPro обрабатывать записи, помеченные на удаление, и будут ли они доступны для использования в других командах.
SET EXCLUSIVE	Определяет, в каком режиме Visual FoxPro открывает файлы таблиц в сети - для монопольного пользования или для совместного.
SET FIELDS	Определяет поля в таблице, к которым разрешен доступ.
SET FILTER	Задаёт условие, соблюдение которого делает записи текущей таблицы доступными.
SET FULLPATH	Определяет, должны ли функции CDX(), DBF(), MDX() и NDX() возвращать путь вместе с именем файла.

Функции	Описание
SET INDEX	Открывает один или несколько файлов индексов для текущей таблицы.
SET KEY	Задаёт способ доступа к записям некоторого диапазона исходя из их ключей индекса.
SET LOCK	Разрешает или запрещает автоматическую блокировку файлов для определенных команд.
SET MULTILOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.
SET NEAR	Определяет, где располагается указатель записи после неудачного выполнения поиска записи командой FIND, SEEK или SEEK().
SET ORDER	Задаёт управляющий файл или тег индекса для данной таблицы.
SET NULL	Определяет, каким образом поддерживаются значения NULL командами ALTER TABLE, CREATE TABLE и INSERT.
SET OLEOBJECT	Указывает, должна ли система Visual FoxPro просматривать реестр Windows Registry, когда объект найти не удастся.
SET OPTIMIZE	Включает или отключает оптимизацию по технологии Rushmore.
SET REFRESH	Определяет, нужно ли (и как часто) обновлять окно просмотра, отражая в нем изменения, внесенные в записи другими пользователями сети.
SET RELATION	Устанавливает отношение между двумя открытыми таблицами.
SET RELATION OFF	Разрывает установленное отношение между родительской таблицей в текущей рабочей области и связанной с ней дочерней таблицей.
SET REPROCESS	Указывает, сколько раз или как долго Visual FoxPro может повторять попытки блокировать файл или запись после неудачной попытки блокировки.
SET SAFETY	Определяет, будет ли Visual FoxPro отображать диалоговое окно перед тем, как вести запись поверх существующего файла, а также будут ли вычисляться правила таблиц или полей, значения по умолчанию и сообщения об ошибках при внесении изменений в конструкторе таблиц или командой ALTER TABLE.
SET SKIP	Создает отношение один-ко-многим между таблицами.
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления.
SORT	Сортирует записи в выбранной в данный момент таблице и выводит отсортированные записи в новую таблицу.
SUM	Суммирует все или только заданные числовые поля в таблице, выбранной в данный момент.
SYS(14)	Возвращает выражение индекса для открытого файла простого индекса .IDX или индексные выражения для тегов в файлах составного индекса .CDX. Команда эквивалентна KEY().
SYS(2012)	Возвращает размер блока мемо-поля, установленный для данной таблицы.
SYS(2029)	Возвращает значение, соответствующее типу таблицы. (например, FoxPro, Visual FoxPro, Dbase IV, etc...).
SYS(3051)	Задаёт интервал времени (в миллисекундах), в течение которого Visual FoxPro ждет, прежде чем попытается снова блокировать запись, таблицу, мемо-файл или файл индекса после предыдущей неудачной попытки. Используйте SET REPROCESS.

Функции	Описание
SYS(3052)	Определяет, будет ли Visual FoxPro использовать установку SET REPROCESS при попытке заблокировать файл индекса или мето-файл.
SYS(3054)	Определяет, будет ли выводиться уровень оптимизации Rushmore для запроса.
TABLEREVERT()	Сбрасывает изменения, внесенные в буферизованную запись или в буферизованную таблицу, или в курсор, и восстанавливает содержимое по данным OLDVAL() для удаленных курсоров или по текущим значениям на диске для локальных таблиц и курсоров.
TABLEUPDATE()	Фиксирует изменения, внесенные в буферизованную запись либо в буферизованную таблицу или курсор.
TARGET()	Возвращает псевдоним таблицы, с которой устанавливается отношение согласно предложению INTO команды SET RELATION.
TOTAL	Вычисляет итоговые суммы по числовым полям в текущей таблице.
TXNLEVEL()	Возвращает числовое значение, указывающее уровень текущей транзакции.
UNLOCK	Снимает блокировку записи, блокировку группы записей или блокировку файла с данной таблицы либо снимает блокировки всех записей и файлов для всех открытых таблиц.
UPDATE	Обновляет таблицу, открытую в рабочей области, выбранной в данный момент, внося в нее данные из другой таблицы. Используйте лучше UPDATE - SQL.
UPDATE - SQL	Обновляет записи в таблице актуальными значениями.
USE	Открывает таблицу и связанные с ней индексы или открывает обзор SQL.
USED()	Определяет, открыта ли таблица в заданной рабочей области или используется ли заданный алиас в какой-нибудь рабочей области.
ZAP	Удаляет из таблицы все записи, оставляя лишь структуру таблицы. При использовании ZAP не работает триггер Delete.

Visual FoxPro окружение - среда

Функции	Описание
ADATABASES()	Помещает в массив переменных памяти имена всех открытых баз данных и описания путей для этих имен.
ADIR()	Заносит в массив информацию о файлах и возвращает количество файлов.
AFONT()	Заносит в массив информацию о доступных шрифтах
APRINTERS()	Помещает в массив переменных памяти имена принтеров, которые в данный момент описаны в Windows.
ASSIST	Запускает программу, заданную системной переменной <code>_ASSIST</code> .
AUSED()	Помещает в массив переменных памяти псевдонимы и рабочие области для таблиц заданного сеанса.
_BROWSER	Содержит имя приложения просмотра классов.
_BUILDER	Содержит имя приложения-конструктора Visual FoxPro
CAPSLOCK()	Возвращает значение текущего режима CAPS LOCK или производит включение или выключение этого режима.
CD CHDIR	Заменяет каталог Visual FoxPro, используемый по умолчанию, заданным каталогом.
CLEAR RESOURCES	Удаляет из памяти кешированные картинки, фонты, сурсоры или иконки. Если не задано имя файла, все названные ресурсы удаляются из памяти.
COL()	Возвращает текущую позицию курсора по столбцам.
CREATE COLOR SET	Создает цветовой набор на основе текущих установок цвета.
CREATE VIEW	Создает файл среды Visual FoxPro. Команда SET VIEW восстанавливает среду, которая создана командой CREATE VIEW. Файлы среды имеют расширение .VUE.
CURDIR()	Возвращает текущий каталог.
DBC()	Возвращает имя текущей базы данных и путь к ней.
DBF()	Возвращает имя таблицы, открытой в заданной рабочей области, или имя таблицы из псевдонима таблицы.
DIR or DIRECTORY	Отображает информацию о файлах некоторого каталога.
DIRECTORY()	Возвращает “истину” (.T.), если заданный каталог есть на диске.
DISKSPACE()	Возвращает объем свободного пространства в байтах на диске, используемом по умолчанию.
DISPLAY DLLS	Отображает информацию о DLL-функциях 32-разрядной версии Windows, зарегистрированных в Visual FoxPro с помощью команды DECLARE - DLL.
DISPLAY STATUS	Отображает состояние среды Visual FoxPro.
FKLABEL()	Возвращает имя функциональной клавиши (F1, F2, F3 ...) по ее номеру.

Функции	Описание
FKMAX()	Возвращает число программируемых функциональных клавиш или комбинаций с их участием на данной клавиатуре.
FONTMETRIC()	Возвращает атрибуты шрифтов, установленных в операционной системе на данный момент.
GETPICT()	Открывает диалоговое окно Open и возвращает имя выбранного файла рисунка.
GETPRINTER()	Отображает диалоговое окно Print Setup (Windows) и возвращает имя выбранного принтера.
GETCOLOR()	Отображает диалоговое окно Color (Windows) и возвращает номер выбранного цвета.
GETCP()	Запрашивает кодовую страницу, отображая на экране диалоговое окно Code Page, а затем возвращает номер выбранной кодовой страницы.
GETDIR()	Отображает диалоговое окно Select Directory, в котором можно выбрать нужный каталог.
GETENV()	Возвращает содержимое заданной переменной среды выполнения MS-DOS.
GETFILE()	Отображает диалоговое окно Open и возвращает имя выбранного файла.
GETFONT()	Отображает диалоговое окно Font и возвращает имя выбранного шрифта.
HELP	Открывает окно справки.
HOME()	Возвращает имя каталога, из которого производился запуск Visual FoxPro.
IMESTATUS()	Переключает Окно IME (Редактор Ввода Методов) ил возвращает его статус.
INKEY()	Возвращает число, соответствующее первому нажатию клавиши или кнопки мыши в буфере клавиатуры.
INSMODE()	Возвращает текущий режим вставки или включает или выключает этот режим.
ISCOLOR()	Определяет, может ли компьютер формировать цветное изображение.
ISMOUSE()	Возвращает значение "истина" (.T.), если аппаратное обеспечение мыши установлено.
LIST DLLS	Отображает непрерывным потоком информацию о DLL-функциях 32-разрядной версии Windows, зарегистрированных в Visual FoxPro с помощью команды DECLARE - DLL.
LIST STATUS	Отображает непрерывным потоком информацию таблицу и/или состояние среды.
LOCFILE()	Ищет файл на диске и возвращает имя файла вместе с его путем.
MD MKDIR	Создает новый каталог.
MEMORY()	Возвращает объем памяти, доступной для выполнения внешней программы.
MESSAGE()	Возвращает текущее сообщение об ошибке в виде символьной строки или содержимое строки программы, вызвавшей ошибку.
NUMLOCK()	Возвращает текущий режим Num Lock или включает или выключает этот режим.
OBJTOCLIENT()	Возвращает позицию или размер элемента управления или объекта относительно содержащего его объекта Form.
ON()	Возвращает команду, назначенную любой из следующих ON APLABOUT, ON ERROR, ON ESCAPE, ON KEY, ON KEY LABEL, ON MACHELP, ON PAGE или ON READERROR.

Функции	Описание
OS ()	Возвращает название и номер версии операционной системы, под управлением которой работает Visual FoxPro.
PROGRAM()	Возвращает имя выполняемой в данный момент программы, или имя программы, выполнявшейся в момент возникновения ошибки. Команда идентичка SYS(16), но имеет больше возможностей.
PUTFILE()	Открывает диалоговое окно Save As и возвращает задаваемое имя файла.
QUIT	Завершает текущий сеанс Visual FoxPro и возвращает управление в операционную систему.
RD RMDIR	Удаляет каталог с диска.
READKEY()	Возвращает значение, которое соответствует клавише, нажатой для выхода из некоторых команд редактирования, или значение, указывающее, как закончилась последняя команда READ.
RELEASE LIBRARY	Удаляет одну библиотеку внешних программ API из памяти.
RELEASE PROCEDURE	Закрывает файл процедуры, открытый с помощью команды SET PROCEDURE.
RESTORE MACROS	Восстанавливает в памяти макрокоманды клавиатуры, сохраненные в файле макрокоманд клавиатуры или в мето-поле.
RESTORE SCREEN	Восстанавливает основное окно Visual FoxPro или пользовательское окно, сохраненное перед этим в буфере экрана, в переменной памяти или в элементе массива.
RESTORE WINDOW	Восстанавливает в памяти описания окон и статусы окон, сохраненные в файле окон или в мето-поле.
RGB()	Возвращает одно значение цвета из набора, включающего красный, зеленый и синий компоненты.
RGBSCHEME()	Возвращает цветовую пару RGB или список цветовых пар RGB для заданной цветовой схемы.
ROW()	Возвращает текущую позицию курсора по строкам.
SAVE MACROS	Сохраняет набор макрокоманд клавиатуры в файле макрокоманд клавиатуры или в мето-поле
SAVE SCREEN	Сохраняет изображение основного окна Visual FoxPro или активного пользовательского окна в буфере экрана, переменной памяти или элементе массива.
SAVEPICTURE()	Создает файл картинки (.BMP) на основании объектой ссылки на нее.
SCHEME()	Возвращает цветовую пару или список цветовых пар из заданной цветовой схемы.
SCOLS()	Возвращает число столбцов, предусмотренных для основного окна Visual FoxPro.
_SCREEN	Задаёт свойства и методы основного окна Visual FoxPro.
SET()	Возвращается текущая установка заданной команды SET в виде символьной или числовой строки.
SET ANSI	Определяет, каким образом оператор = в командах Visual FoxPro SQL будет сравнивать строки, имеющие разную длину.
SET AUTOSAVE	Определяет, должна ли система Visual FoxPro сбрасывать буферы данных на диск, когда вы выходите из команды READ или возвращаетесь в окно команд.

Функции	Описание
SET BELL	Включает или выключает звуковой сигнал компьютера, а также устанавливает атрибуты сигнала.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мето-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.
SET CENTURY	Определяет, должен ли Visual FoxPro показывать номер века в выражениях даты.
SET CLASSLIB	Открывает библиотеку визуальных классов .VCX, содержащую определения классов.
SET CLEAR	Определяет, следует ли очищать основное окно Visual FoxPro при выдаче команд SET FORMAT.
SET CLOCK	Определяет, должен ли Visual FoxPro отображать системные часы, и задает для них позицию в основном окне Visual FoxPro.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET COLOR OF	Задаёт цвета для пользовательских меню и окон.
SET COLOR OF SCHEME	Задаёт цвета в цветовой схеме или копирует одну цветовую схему в другую.
SET COLOR SET	Загружает ранее определенный цветовой набор.
SET COMPATIBLE	Контролирует совместимость с FoxBASE+ и другими языками Xbase.
SET CONSOLE	Разрешает или запрещает вывод из программы в основное окно Visual FoxPro или в активное окно, определенное пользователем.
SET CPCOMPILE	Задаёт кодовую страницу для компилируемых программ.
SET CPDIALOG	Определяет, будет ли при открытии таблицы отображаться диалоговое окно Code Page.
SET CURRENCY	Определяет символ денежной единицы и задает его позицию в отображаемых выражениях типа Numeric, Currency, Float и Double.
SET CURSOR	Определяет, отображается ли курсор в то время, когда Visual FoxPro ожидает ввода.
SET DATE	Задаёт формат отображаемых выражений типа даты и DateTime.
SET DEBUG	Делает окна отладки и трассировки доступными или недоступными из системы меню Visual FoxPro.
SET DEBUGOUT	Направляет отладочный вывод в файл. Отладочный вывод формируется командами ASSERT, DEBUGOUT и событиями, которые определены в SET EVENTLIST или в диалоге Трассировки событий (Event Tracking).
SET DECIMALS	Задаёт число десятичных знаков, отображаемых в числовых выражениях.
SET DEFAULT	Задаёт диск или каталог, используемые по умолчанию.
SET DELETED	Определяет, будет ли Visual FoxPro обрабатывать записи, помеченные на удаление, и будут ли они доступны для использования в других командах.
SET DEVELOPMENT	Предписывает Visual FoxPro сравнивать при запуске программы дату и время создания программы с датой и временем создания соответствующего скомпилированного объектного файла.

Функции	Описание
SET DEVICE	Направляет вывод команды @ ... SAY на экран, на принтер или в файл.
SET DISPLAY	Предоставляет возможность изменять текущий режим дисплея на мониторах, поддерживающих различные режимы. Осталось от FoxPro for DOS, не используется в Windows.
SET ESCAPE	Определяет, должно ли нажатие клавиши Esc прерывать выполнение программ и команд.
SET EXACT	Задаёт правила, согласно которым Visual FoxPro сравнивает две строки разной длины.
SET EXCLUSIVE	Определяет, в каком режиме Visual FoxPro открывает файлы таблиц в сети - для монопольного пользования или для совместного.
SET FDOW	Задаёт первый день недели.
SET FIXED	Определяет, является ли фиксированным количество десятичных знаков в изображении числовых данных.
SET FUNCTION	Назначает выражение (макрокоманду клавиатуры) функциональной клавише или комбинации клавиш.
SET FWEEK	Задаёт требования, предъявляемые к первой неделе года.
SET HELP	Включает или отключает оперативную справку Visual FoxPro или задаёт файл справки.
SET HELPFILTER	Предписывает Visual FoxPro отображать в окне справки подмножество тем справки в стиле .DBF.
SET HOURS	Устанавливает для системных часов 12-часовой или 24-часовой формат времени.
SET INTENSITY	Определяет, использует ли Visual FoxPro расширенный цветовой атрибут для отображения полей.
SET KEYCOMP	Контролирует навигацию в Visual FoxPro, осуществляемую с помощью клавиатуры.
SET LIBRARY	Открывает файл внешней библиотеки API (интерфейса прикладного программирования).
SET MACKEY	Задаёт клавишу или комбинацию клавиш, при нажатии которых отображается диалоговое окно Macro Key Definition.
SET MACHelp	Манипулирует элементом Microsoft FoxPro Help меню Help.
SET MARK TO	Задаёт разделитель, используемый при отображении дат.
SET MEMOWIDTH	Задаёт ширину отображаемых на экране мемо-полей и символьных выражений. SET MEMOWIDTH влияет на функции ATCLINE(), ATLINE(), MEMLINE() и MLINE().
SET MESSAGE	Задаёт сообщение, отображаемое в основном окне Visual FoxPro или в графической строке состояния, или задаёт место вывода сообщений, связанных с пользовательскими строками меню и командами меню.
SET MOUSE	Включает или отключает мышь и управляет ее чувствительностью.
SET NOTIFY	Разрешает или отменяет отображение некоторых системных сообщений.
SET NULLDISPLAY	По умолчанию Visual FoxPro выводит значение .NULL. для пустых значений объектов. Использование команды SET NULLDISPLAY позволяет сменить выводимый текст.
SET ODOMETER	Задаёт интервал, с которым команды, обрабатывающие записи, сообщают значения счетчика записей.

Функции	Описание
SET PALETTE	Определяет, используется ли в Visual FoxPro цветовая палитра, принимаемая по умолчанию.
SET PATH	Задаёт путь для поиска файлов.
SET POINT	Определяет символ, используемый в качестве десятичного разделителя при изображении числовых выражений и выражений типа Currency.
SET PRINTER	Разрешает или запрещает вывод на принтер, или направляет вывод в файл, порт или на сетевой принтер.
SET PROCEDURE	Открывает файл процедур. (.PRG или .FXP файл, содержащий процедуры и функции)
SET RESOURCE	Обновляет или задаёт файл ресурсов.
SET SAFETY	Определяет, будет ли Visual FoxPro отображать диалоговое окно перед тем, как вести запись поверх существующего файла, а также будут ли вычисляться правила таблиц или полей, значения по умолчанию и сообщения об ошибках при внесении изменений в конструкторе таблиц или командой ALTER TABLE.
SET SCOREBOARD	Задаёт, где Visual FoxPro отображает состояние клавиш Num Lock, Caps Lock и Insert.
SET SECONDS	Определяет, указываются ли секунды в значении типа DateTime.
SET SEPARATOR	Задаёт символ, который отделяет друг от друга группы из трех цифр, стоящие слева от десятичной точки.
SET STATUS BAR	Отображает или удаляет с экрана графическую строку состояния.
SET STICKY	Определяет, что выбранные меню должны оставаться развернутыми до тех пор, пока вы не выберете элемент меню, нажмете клавишу Esc или щелкнете вне меню.
SET SYSFORMATS	Определяет, должны ли системные установки Visual FoxPro обновляться вместе с текущими системными установками Windows.
SET SYSMENU	Делает доступной или недоступной системную строку меню Visual FoxPro в процессе выполнения программы и дает возможность изменить конфигурацию строки меню.
SET TALK	Определяет, будет ли Visual FoxPro отображать результаты команд.
SET TEXTMERGE	Включает или отключает вычисление полей, переменных памяти, элементов массива, функций или выражений, заключенных в разделители слияния, и позволяет задать вывод процедуры слияния текста.
SET TEXTMERGE DELIMITERS	Задаёт разделители слияния текста.
SET TOPIC	Задаёт тему (или темы) справки, которую следует открыть при активизации системы справки Visual FoxPro
SET TOPIC ID	Задаёт тему справки, которая откроется при активизации системы справки Visual FoxPro. Раздел справки определяется идентификатором, который представляется в разделе MAP файла проекта помощи.
SET TYPEAHEAD	Задаёт максимальное число символов, которые можно сохранить в буфере клавиатуры.
SET UDFPARMS	Задаёт в Visual FoxPro способ передачи параметров в пользовательскую функцию (UDF) - по значению или по ссылке.

Функции	Описание
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления.
SET XCMDFILE	аает файл ресурсов Macintosh для внешних команд (XCMD) и внешних функций (XFCN).
SYS(0)	Возвращает сетевую информацию о машине в случае, когда Visual FoxPro используется в сетевой среде.
SYS(3)	Возвращает допустимое имя файла, которое можно использовать при создании временных файлов.. Используйте SUBSTR(SYS(2015), 3, 10) для создания уникального имени файла.
SYS(5)	Возвращает диск, используемый в данный момент по умолчанию в Visual FoxPro. (например. C:, D:, E:, etc...) или том (Windows NT только).
SYS(6)	Возвращает текущее печатающее устройство. Как в установках SET PRINTER.
SYS(7)	Возвращает имя текущего файла формата. Как в установках SET FORMAT.
SYS(9)	Возвращает серийный номер данной версии Visual FoxPro.
SYS(12)	Возвращает объем памяти в доступной для выполнения внешней программы области, лежащей ниже отметки 640K.
SYS(16)	Возвращает имя файла выполняющейся программы. Если SYS(16) возвращает путь с именем программы, то PROGRAM() возвращает имя программы.
SYS(17)	Возвращает тип используемого центрального процессора (CPU).
SYS(18)	Возвращает набранное прописными буквами имя переменной памяти, элемента массива или поля, использованное при создании текущего элемента управления. Используйте свойство Active Control .
SYS(24)	Возвращает границу памяти EMS, установленную в файле конфигурации FoxPro для MS-DOS.
SYS(100)	Возвращает текущую установку SET CONSOLE.. Используйте SET("CONSOLE").
SYS(101)	Возвращает текущую установку SET DEVICE.. Используйте SET("DEVICE").
SYS(102)	Возвращает текущую установку SET PRINTER. Используйте SET("PRINTER").
SYS(103)	Возвращает текущую установку SET TALK. Используйте SET("TALK").
SYS(1001)	Возвращает общий виртуальной объем памяти, доступной программе управления памятью Visual FoxPro., которая примерно в пять раз превосходит размер доступной физической памяти.
SYS(1016)	Возвращает объем памяти, задействованной определенными пользовательскими объектами.
SYS(1023)	Включает режим диагностики системы справки, давая возможность перехватывать параметры HelpContextID, передаваемые в систему справки Visual FoxPro.
SYS(1024)	Отключает режим диагностики системы справки, включенный функцией SYS(1023).
SYS(2002)	Включает или выключает курсор на экране. Используйте SET CURSOR.
SYS(2003)	Возвращает имя текущего каталога на используемом по умолчанию диске.
SYS(2004)	Возвращает имя каталога, из которого запускается Visual FoxPro.
SYS(2005)	Возвращает имя текущего файла ресурсов Visual FoxPro.

Функции	Описание
SYS(2006)	Возвращает тип используемой графической карты и монитора.
SYS(2008)	Задаёт форму курсора в режиме вставки и в режиме замены.
SYS(2009)	Переключает форму курсора между режимом вставки и замены.
SYS(2010)	Возвращает установку FILES из файла CONFIG.SYS. <i>Бесполезно в Visual FoxPro.</i>
SYS(2015)	Возвращает уникальное 10-символьное имя процедуры, которое состоит из символа подчеркивания и некоторой комбинации букв и цифр.
SYS(2016)	Возвращает имя окна, использованное в последней команде SHOW GETS WINDOW. Используйте метод Refresh.
SYS(2017)	В предыдущих версиях FoxPro данная функция очищала основное окно FoxPro и отображала экран запуска FoxPro.
SYS(2019)	Возвращает имя и местоположение файла конфигурации Visual FoxPro.
SYS(2020)	Возвращает общий размер (в байтах) диска, используемого по умолчанию.
SYS(2022)	Возвращает размер кластера (блока) для заданного диска в байтах.
SYS(2023)	Возвращает диск и каталог, где Visual FoxPro хранит временные файлы.
SYS(2027)	Возвращает путь MS-DOS в системе записи, принятой для путей Macintosh.
SYS(3004)	Возвращает идентификатор Locale ID, используемый объектами автоматизации OLE и элементами управления OLE.
SYS(3005)	Устанавливает идентификатор Locale ID, используемый объектами автоматизации OLE и элементами управления OLE.
SYS(3006)	Устанавливает идентификаторы Language ID и Locale ID.
SYS(3050)	Устанавливает размер приоритетного или фонового буфера в памяти.
SYS(3053)	Возвращает ODBC environment handle. Если ODBC не загружен, SYS(3053) загружает его и возвращает ODBC environment handle. Environment handle, возвращенные SYS(3053) обеспечивает доступ к ODBC через вызовы ODBC API.
SYSMETRIC()	Возвращает размеры экранных элементов операционной системы. SYSMETRIC() позволяет определить размер для меню, окон, элементов управления в Visual FoxPro.
TXTWIDTH()	Возвращает длину символьного выражения исходя из средней ширины символа для данного шрифта.
VERSION()	Возвращает символьную строку, содержащую номер используемой версии Visual FoxPro.
WFONT()	Возвращает имя, размер или стиль текущего шрифта для окна в Visual FoxPro
_WIZARD	Содержит имя программы поставщика (мастера) Visual FoxPro.

Visual FoxPro окружение - работа с файлами

Функции	Описание
ADIR()	Заносит в массив информацию о файлах и возвращает количество файлов.
ASSIST	Запускает программу, заданную системной переменной <code>_ASSIST</code> .
CD CHDIR	Заменяет каталог Visual FoxPro, используемый по умолчанию, заданным каталогом.
CLOSE ALL	Закрывает все открытые базы данных, таблицы, индексы во всех рабочих областях и выбирает рабочую область 1. Закрывает все открытые файлы, созданные по <code>FCREATE()</code> или открытые <code>FOPEN()</code> . Функции. <code>CLOSE ALL</code> не закрывает файл, открытый в <code>SET PRINT</code> .
CLOSE ALTERNATE	Закрывает альтернативный файл, открытый командой <code>SET ALTERNATE</code> .
CLOSE DATABASES	Закрывает текущую базу данных и ее таблицы. Если нет текущей базы данных, все открытые свободные таблицы, индексы, форматные файлы во всех рабочих областях закрываются, и выбирается рабочая область 1.
CLOSE FORMAT	Закрывает различные типы файлов.
CLOSE INDEXES	Закрывает различные типы файлов.
CLOSE PROCEDURE	Закрывает различные типы файлов.
COPY FILE	Создает копию файла любого типа.
CURDIR()	Возвращает текущий каталог.
DELETE FILE	Удаляет файл с диска
DIR or DIRECTORY	Отображает информацию о файлах некоторого каталога.
DIRECTORY()	Возвращает "истину" (.T.), если заданный каталог есть на диске.
DISPLAY FILES	Отображает информацию о файлах.
DISKSPACE()	Возвращает объем свободного пространства в байтах на диске, используемом по умолчанию.
ERASE	Стирает файл с диска.
FDATE()	Возвращает дату последней модификации файла
FILE()	Возвращает значение "истина" (.T.), если заданный файл найден на диске. Используйте <code>EXISTS()</code> .
FILER	Открывает программу обслуживания файлов Filer.
FLUSH	Немедленно сохраняет на диске изменения в таблице и индексе.
FSIZE()	Возвращает величину заданного поля или файла в байтах.
FTIME()	Возвращает время последней модификации файла.
FULLPATH()	Возвращает путь к заданному файлу или путь относительно другого файла. Используйте <code>EXISTS()</code> .

Функции	Описание
GETDIR()	Отображает диалоговое окно Select Directory, в котором можно выбрать нужный каталог.
GETFILE()	Отображает диалоговое окно Open и возвращает имя выбранного файла.
GETPICT()	Открывает диалоговое окно Open и возвращает имя выбранного файла рисунка.
ISNULL()	Возвращает значение "истина" (.T.), если результатом вычисления выражения является значение NULL; в противном случае возвращает "ложь" (.F.).
LIST FILES	Выводит сплошным потоком информацию о файлах в текущем каталоге, которые соответствуют маске.
LOCFILE()	Ищет файл на диске и возвращает имя файла вместе с его путем.
LUPDATE()	Возвращает дату последнего обновления таблицы, открытой в другой рабочей области.
MD MKDIR	Создает новый каталог.
MODIFY COMMAND	Открывает окно редактирования для изменения или создания программного файла..
MODIFY FILE	Открывает окно редактирования для изменения или создания текстового файла..
NVL()	Возвращает отличное от NULL значение одного из двух выражений.
OPEN DATABASE	Открывает базу данных.
PUTFILE()	Открывает диалоговое окно Save As и возвращает задаваемое имя файла.
RD RMDIR	Удаляет каталог с диска.
RENAME	Заменяет имя файла новым именем.
SAVEPICTURE()	Создает файл картинки (.BMP) на основании объектной ссылки на нее.
SET ALTERNATE	Направляет в текстовый файл экранный или печатный вывод, созданный командами ?, ??, DISPLAY или LIST.
SET DEFAULT	Задаёт диск или каталог, используемые по умолчанию.
SET FULLPATH	Определяет, должны ли функции CDX(), DBF(), MDX() и NDX() возвращать путь вместе с именем файла.
SET PATH	Задаёт путь для поиска файлов.
SET SAFETY	Определяет, будет ли Visual FoxPro отображать диалоговое окно перед тем, как вести запись поверх существующего файла, а также будут ли вычисляться правила таблиц или полей, значения по умолчанию и сообщения об ошибках при внесении изменений в конструкторе таблиц или командой ALTER TABLE.
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления.
SET VOLUME	Maps MS-DOS drive designators (A:, B:, C:, and so on) to volumes or folders in FoxPro for Macintosh.
SYS(3)	Возвращает допустимое имя файла, которое можно использовать при создании временных файлов.. Используйте SUBSTR(SYS(2015), 3, 10) для создания уникального имени файла.
SYS(5)	Возвращает диск, используемый в данный момент по умолчанию в Visual FoxPro. (например. C:, D:, E:, etc...) или том (Windows NT только).

Функции	Описание
SYS(2000)	Возвращает имя первого файла, соответствующего заданному шаблону имен файлов.
SYS(2014)	Возвращает кратчайший путь для заданного файла относительно текущего или указанного каталога.
SYS(2015)	Возвращает уникальное 10-символьное имя процедуры, которое состоит из символа подчеркивания и некоторой комбинации букв и цифр.
SYS(2020)	Возвращает общий размер (в байтах) диска, используемого по умолчанию.
SYS(2022)	Возвращает размер кластера (блока) для заданного диска в байтах.
SYS(2023)	Возвращает диск и каталог, где Visual FoxPro хранит временные файлы.
SYS(2027)	Возвращает путь MS-DOS в системе записи, принятой для путей Macintosh.
TYPE	Отображает содержимое файла.

Распределенный доступ к данным - доступ

Функции	Описание
BEGIN TRANSACTION	Начинает транзакцию. Транзакции поддерживаются только для таблиц базы данных .
CURSORGETPROP()	Возвращает текущие установки свойств таблицы или курсора Visual FoxPro.
CURSORSETPROP()	Задаёт установки свойств для таблицы или курсора Visual FoxPro.
CURVAL()	Возвращает значение поля непосредственно с диска или из удаленного источника. При использовании с OLDVAL() даёт возможность контроля за работой в многопользовательской среде для проверки изменения данных другим пользователем.
DBGETPROP()	Возвращает свойство текущей базы данных, или свойство поля, именованного соединения, таблицы или представления текущей базы данных.
DBSETPROP()	Устанавливает свойство для текущей базы данных или поля, именованного соединения, таблицы или представления текущей базы данных.
DELETE - SQL	Помечает записи на удаление. В отличие от DELETE, DELETE - SQL использует блокировку записей при попытке удаления множества записей в таблицах, открытых для совместного доступа.
DROPOFFLINE()	Отменяет все изменения, сделанные в офф-лайн-обзоре и возвращает обзор в соединение он-лайн.
END TRANSACTION	Завершает текущую транзакцию, которые внесены в таблицы, мемо-файлы таблиц и файлы индексов, включенные в транзакцию, и завершает ее.
FLOCK()	Пытается заблокировать текущую или заданную таблицу.
ISFLOCKED()	Возвращает значение "истина" (.T.), если определенная таблица заблокирована, в противном случае возвращается "ложь" (.F.). ISFLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
ISRLOCKED()	Возвращает значение "истина" (.T.), если запись заблокирована, в противном случае возвращается "ложь" (.F.). ISRLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
LOCK()	Пытается заблокировать одну или несколько записей в таблице.
OLDVAL()	Возвращает первоначальное значение поля, которое было модифицировано, но не обновлялось. Используется совместно с CURVAL().
RLOCK()	Пытается заблокировать одну или несколько записей таблицы. RLOCK() аналогична LOCK().
SET EXCLUSIVE	Определяет, в каком режиме Visual FoxPro открывает файлы таблиц в сети - для монопольного пользования или для совместного.
SET FIELDS	Определяет поля в таблице, к которым разрешен доступ.
SET LOCK	Разрешает или запрещает автоматическую блокировку файлов для определенных команд.
SET MULTLOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.

Функции	Описание
SET REFRESH	Определяет, нужно ли (и как часто) обновлять окно просмотра, отражая в нем изменения, внесенные в записи другими пользователями сети.
SET REPROCESS	Указывает, сколько раз или как долго Visual FoxPro может повторять попытки блокировать файл или запись после неудачной попытки блокировки.
SYS(3051)	Задаёт интервал времени (в миллисекундах), в течение которого Visual FoxPro ждёт, прежде чем попытается снова блокировать запись, таблицу, мемо-файл или файл индекса после предыдущей неудачной попытки. Используйте SET REPROCESS.
SYS(3052)	Определяет, будет ли Visual FoxPro использовать установку SET REPROCESS при попытке блокировать файл индекса или мемо-файл.
TABLEREVERT()	Сбрасывает изменения, внесенные в буферизованную запись или в буферизованную таблицу, или в курсор, и восстанавливает содержимое по данным OLDVAL() для удаленных курсоров или по текущим значениям на диске для локальных таблиц и курсоров.
TABLEUPDATE()	Фиксирует изменения, внесенные в буферизованную запись либо в буферизованную таблицу или курсор.
TXNLEVEL()	Возвращает числовое значение, указывающее уровень текущей транзакции.
UNLOCK	Снимает блокировку записи, блокировку группы записей или блокировку файла с данной таблицы либо снимает блокировки всех записей и файлов для всех открытых таблиц.
USE	Открывает таблицу и связанные с ней индексы или открывает обзор SQL.
USED()	Определяет, открыта ли таблица в заданной рабочей области или используется ли заданный алиас в какой-нибудь рабочей области.

Распределенный доступ к данным – секретность и шифрование

Функции	Описание
SYS(0)	Возвращает сетевую информацию о машине в случае, когда Visual FoxPro используется в сетевой среде.
SYS(2007)	Возвращает значение контрольной суммы символьного выражения.

Распределенный доступ к данным – команды определения сессии данных

Функции	Описание
SET ANSI	Определяет, каким образом оператор = в командах Visual FoxPro SQL будет сравнивать строки, имеющие разную длину.
SET AUTOSAVE	Определяет, должна ли система Visual FoxPro сбрасывать буферы данных на диск, когда вы выходите из команды READ или возвращаетесь в окно команд.
SET BLOCKSIZE	Определяет, каким образом Visual FoxPro распределяет дисковое пространство для хранения мемо-полей.
SET CARRY	Определяет, должен ли Visual FoxPro переносить данные из текущей записи в новую запись, создаваемую командой INSERT, APPEND или BROWSE.
SET CENTURY	Определяет, должен ли Visual FoxPro показывать номер века в выражениях даты.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET CONFIRM	Определяет, можно ли выходить из текстового поля, вводя данные правее его последнего символа.
SET CURRENCY	Определяет символ денежной единицы и задаёт его позицию в отображаемых выражениях типа Numeric, Currency, Float и Double.
SET DATABASE	Задаёт текущую базу данных.
SET DATE	Задаёт формат отображаемых выражений типа даты и DateTime.
SET DECIMALS	Задаёт число десятичных знаков, отображаемых в числовых выражениях.
SET DELETED	Определяет, будет ли Visual FoxPro обрабатывать записи, помеченные на удаление, и будут ли они доступны для использования в других командах.
SET DELIMITERS	Определяет, имеются ли разделители в текстовых полях.
SET EXACT	Задаёт правила, согласно которым Visual FoxPro сравнивает две строки разной длины.
SET EXCLUSIVE	Определяет, в каком режиме Visual FoxPro открывает файлы таблиц в сети - для монопольного пользования или для совместного.
SET FIELDS	Определяет поля в таблице, к которым разрешен доступ.
SET FIXED	Определяет, является ли фиксированным количество десятичных знаков в изображении числовых данных.
SET LOCK	Разрешает или запрещает автоматическую блокировку файлов для определенных команд.
SET MULTILOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.
SET MARK TO	Задаёт разделитель, используемый при отображении дат.
SET MEMOWIDTH	Задаёт ширину отображаемых на экране мемо-полей и символьных выражений. SET MEMOWIDTH влияет на функции ATCLINE(), ATLINE(), MEMLINE() и MLINE().
SET MULTILOCKS	Определяет, можно ли с помощью функций LOCK() или RLOCK() блокировать сразу несколько записей.

Функции	Описание
SET NEAR	Определяет, где располагается указатель записи после неудачного выполнения поиска записи командой FIND, SEEK или SEEK().
SET NULL	Определяет, каким образом поддерживаются значения NULL командами ALTER TABLE, CREATE TABLE и INSERT.
SET POINT	Определяет символ, используемый в качестве десятичного разделителя при изображении числовых выражений и выражений типа Currency.
SET REFRESH	Определяет, нужно ли (и как часто) обновлять окно просмотра, отражая в нем изменения, внесенные в записи другими пользователями сети.
SET REPROCESS	Указывает, сколько раз или как долго Visual FoxPro может повторять попытки блокировать файл или запись после неудачной попытки блокировки.
SET SAFETY	Определяет, будет ли Visual FoxPro отображать диалоговое окно перед тем, как вести запись поверх существующего файла, а также будут ли вычисляться правила таблиц или полей, значения по умолчанию и сообщения об ошибках при внесении изменений в конструкторе таблиц или командой ALTER TABLE.
SET SEPARATOR	Задаёт символ, который отделяет друг от друга группы из трех цифр, стоящие слева от десятичной точки.
SET TALK	Определяет, будет ли Visual FoxPro отображать результаты команд.
SET UNIQUE	Определяет, поддерживает ли файл индекса записи с повторяющимися значениями ключа индекса.
SYS(3051)	Задаёт интервал времени (в миллисекундах), в течение которого Visual FoxPro ждёт, прежде чем попытается снова блокировать запись, таблицу, мемо-файл или файл индекса после предыдущей неудачной попытки. Используйте SET REPROCESS.
SYS(3052)	Определяет, будет ли Visual FoxPro использовать установку SET REPROCESS при попытке блокировать файл индекса или мемо-файл.

Ввод и вывод – форматирование данных

Функции	Описание
\ \	Вывод строк текста.
? ??	Вычисляет выражения и выводит результаты.
@ ... BOX	Рисует прямоугольник по заданным координатам. Для Visual FoxPro используйте элемент Shape.
@ ... CLEAR	Очищает часть основного окна Visual FoxPro или окна, определенного пользователем.
@ ... EDIT	Создает область редактирования символьного типа переменных, элементов массива, полей или мемо полей. Для Visual FoxPro используйте элемент Editbox.
@ ... FILL	Изменяет цвета текста, размещенного внутри некоторой области на экране.
@ ... GET	Создает поле редактирования. Для Visual FoxPro используйте элемент TextBox.
@ ... GET	Создает списка. Для Visual FoxPro используйте элемент Combobox.
@ ... GET	Создает клавиши. Для Visual FoxPro используйте элемент Commandbutton.
@ ... GET	Создает списка. Для Visual FoxPro используйте элемент Listbox.
@ ... GET	Создает клавиши выбора. Для Visual FoxPro используйте элемент OptionGroup.
@ ... GET	Создает спинеры. Для Visual FoxPro используйте элемент Spinner.
@ ... SAY	Выводит данные на экран или на печать в заданную позицию, выраженную номерами строки и столбца. Для Visual FoxPro используйте элемент Label для вывода текста и Textbox – для вывода содержимого полей или переменных памяти.
@ ... TO	Рисует прямоугольник, окружность или эллипс. Для Visual FoxPro используйте элемент Shape.
ACTIVATE SCREEN	Посылает весь последующий вывод в основное окно Visual FoxPro вместо активного окна, определенного пользователем.
AFONT()	Заносит в массив информацию о доступных шрифтах
APPEND	Добавляет одну или несколько записей в конец таблицы.
BROWSE	Открывает окно просмотра и выводит записи из текущей или указанной таблицы.
CHANGE	Вывод полей для редактирования. Работает как EDIT.
CLEAR	Освобождает заданные элементы в памяти.
CLEAR READ	Освобождает заданные элементы из памяти.
CLOSE FORMAT	Закрывает различные типы файлов.
COL()	Возвращает текущую позицию курсора по столбцам.
CREATE SCREEN	Открывает конструктор форм.
DISPLAY	Отображает информацию о текущей таблице в основном окне Visual FoxPro или в пользовательском окне.
EDIT	Выводит поля для редактирования.

Функции	Описание
FUNCTION	Идентифицирует начало определения пользовательской функции.
INKEY()	Возвращает число, соответствующее первому нажатию клавиши или кнопки мыши в буфере клавиатуры.
LIST	Непрерывным потоком отображает информацию о таблице или о среде окружения..
MESSAGEBOX()	Отображает диалоговое окно, определенное пользователем.
MODIFY SCREEN	Открывает Конструктор форм, в котором можно модифицировать или создать форму.
OBJNUM()	Возвращает номер объекта, соответствующий элементу управления в команде @ ... GET. Используйте свойство TabIndex.
OBJTOCLIENT()	Возвращает позицию или размер элемента управления или объекта относительно содержащего его объекта Form.
OBJVAR()	Возвращает имя переменной памяти, элемента массива или поля, связанного с элементом управления команды @ ... GET. Используйте свойство Name
RDLEVEL()	Возвращает уровень текущей команды READ. Используйте Конструктор форм вместо READ.
READ	Активизирует элементы. Используйте Конструктор форм.
READKEY()	Возвращает значение, которое соответствует клавише, нажатой для выхода из некоторых команд редактирования, или значение, указывающее, как закончилась последняя команда READ.
RGB()	Возвращает одно значение цвета из набора, включающего красный, зеленый и синий компоненты.
RGBSCHEME()	Возвращает цветовую пару RGB или список цветовых пар RGB для заданной цветовой схемы.
ROW()	Возвращает текущую позицию курсора по строкам.
SAVE SCREEN	Сохраняет изображение основного окна Visual FoxPro или активного пользовательского окна в буфере экрана, переменной памяти или элементе массива.
SCHEME()	Возвращает цветовую пару или список цветовых пар из заданной цветовой схемы.
SCOLS()	Возвращает число столбцов, предусмотренных для основного окна Visual FoxPro.
SCROLL	Прокручивает содержимое области в основном окне Visual FoxPro или пользовательском окне вверх, вниз, влево или вправо.
SET BLINK	Определяет, можно ли задавать атрибут мерцания или атрибут высокой яркости.
SET CLEAR	Определяет, следует ли очищать основное окно Visual FoxPro при выдаче команд SET FORMAT.
SET COLOR OF	Задаст цвета для пользовательских меню и окон.
SET COLOR OF SCHEME	Задаст цвета в цветовой схеме или копирует одну цветовую схему в другую.
SET COLOR SET	Загружает ранее определенный цветовой набор.
SET COLOR TO	Задаст цвета пользовательских меню и окон. Используйте SET COLOR OF SCHEME.
SET DECIMALS	Задаст число десятичных знаков, отображаемых в числовых выражениях.

Функции	Описание
SET FIXED	Определяет, является ли фиксированным количество десятичных знаков в изображении числовых данных.
SET FORMAT	Открывает файл формата, предназначенный для использования в командах APPEND, CHANGE, EDIT и INSERT.
SET HEADINGS	Определяет, следует ли отображать для полей заголовки столбцов и включать информацию о файле, когда выдается команда TYPE, AVERAGE, CALCULATE, DISPLAY, LIST и SUM.
SET MARK TO	Задаёт разделитель, используемый при отображении дат.
SET MEMOWIDTH	Задаёт ширину отображаемых на экране мемо-полей и символьных выражений. SET MEMOWIDTH влияет на функции ATCLINE(), ATLINE(), MEMLINE() и MLINE().
SET MESSAGE	Задаёт сообщение, отображаемое в основном окне Visual FoxPro или в графической строке состояния, или задаёт место вывода сообщений, связанных с пользовательскими строками меню и командами меню.
SET READBORDER	Определяет, нужно ли окружать обрамлением текстовые поля, создаваемые командой @ ... GET.
SET SPACE	Определяет, будет ли отображаться пробел между отдельными полями и выражениями при использовании команды ? или ?? .
SET TEXTMERGE	Включает или отключает вычисление полей, переменных памяти, элементов массива, функций или выражений, заключённых в разделители слияния, и позволяет задать вывод процедуры слияния текста.
SET TEXTMERGE DELIMITERS	Задаёт разделители слияния текста.
SHOW GET	Заново отображает элемент управления, назначенный заданной переменной памяти, элементу массива или полю.
SHOW GETS	Заново отображает элемент управления, назначенный заданной переменной памяти, элементу массива или полю.
SHOW OBJECT	Заново отображает заданный элемент управления.
SROWS()	Данная функция возвращает число строк, имеющихся в основном окне Visual FoxPro.
SYS(7)	Возвращает имя текущего файла формата. Как в установках SET FORMAT.
TEXT ... ENDTEXT	Выводит строки текста, результаты выражений и функций, а также содержимое переменных памяти.
TRANSFORM()	Возвращает символьную строку, полученную из символьного или числового выражения; строка имеет формат, определённый кодами PICTURE или FUNCTION, которые применяются в команде @ ... SAY.
TXTWIDTH()	Возвращает длину символьного выражения исходя из средней ширины символа для данного шрифта.
UPDATED()	Возвращает значение "истина" (.T.), если в ходе выполнения текущей команды READ какие-либо данные были изменены в интерактивном режиме. Используйте события InteractiveChange или ProgrammaticChange.

Ввод и вывод – ввод и клавиатуры и мыши

Функции	Описание
CAPSLOCK()	Возвращает значение текущего режима CAPS LOCK или производит включение или выключение этого режима.
CHRSAW()	Определяет наличие символа в буфере клавиатуры.
CLEAR MACROS	Освобождает заданные переменные из памяти.
CLEAR TYPEAHEAD	Освобождает заданные переменные из памяти.
INKEY()	Возвращает число, соответствующее первому нажатию клавиши или кнопки мыши в буфере клавиатуры.
INSMODE()	Возвращает текущий режим вставки или включает или выключает этот режим.
KEYBOARD	Помещает заданное символьное выражение в буфер клавиатуры.
LASTKEY()	Возвращает целое число, соответствующее последней нажатой клавише.
MCOL()	Возвращает позицию указателя мыши по столбцам в основном окне Visual FoxPro или в окне, определенном пользователем.
MDOWN()	Определяет, была ли нажата кнопка мыши и возвращает соответственно значение "истина" (.T.) или "ложь" (.F.).
MOUSE	Имитирует щелканье мышью (включая двойное) и перемещение мыши (в том числе при нажатой кнопке).
MROW()	Возвращает позицию указателя мыши по строкам в основном окне Visual FoxPro или в окне, определенном пользователем.
MWINDOW()	Возвращает имя окна, в котором позиционируется указатель мыши.
NUMLOCK()	Возвращает текущий режим Num Lock или включает или выключает этот режим.
ON ESCAPE	Задаёт команду, которая выполнится, если во время работы программы или команды нажать клавишу Esc.
ON KEY	Задаёт команду, которая будет выполнена при нажатии некоторой клавиши или комбинации клавиш во время работы команды READ.
ON KEY LABEL	Задаёт команду, которая будет выполнена, если нажать некоторую клавишу или комбинацию клавиш или щелкнуть кнопкой мыши.
PLAY MACRO	Выполняет макрокоманду клавиатуры.
POP KEY	Восстанавливает назначения команды ON KEY LABEL, занесенные в стек командой PUSH KEY.
PUSH KEY	Заносит все текущие установки команд ON KEY LABEL в стек памяти.
RESTORE MACROS	Восстанавливает в памяти макрокоманды клавиатуры, сохраненные в файле макрокоманд клавиатуры или в мето-поле.
SAVE MACROS	Сохраняет набор макрокоманд клавиатуры в файле макрокоманд клавиатуры или в мето-поле.
SET CONFIRM	Определяет, можно ли выходить из текстового поля, вводя данные правее его последнего символа.

Функции	Описание
SET CURSOR	Определяет, отображается ли курсор в то время, когда Visual FoxPro ожидает ввода.
SET FUNCTION	Назначает выражение (макрокоманду клавиатуры) функциональной клавише или комбинации клавиш.
SET KEYCOMP	Контролирует навигацию в Visual FoxPro, осуществляемую с помощью клавиатуры.
SET MACKEY	Задаёт клавишу или комбинацию клавиш, при нажатии которых отображается диалоговое окно Macro Key Definition.
SET TYPEAHEAD	Задаёт максимальное число символов, которые можно сохранить в буфере клавиатуры.
SYS(2002)	Включает или выключает курсор на экране. Используйте SET CURSOR.
SYS(2008)	Задаёт форму курсора в режиме вставки и в режиме замены.
SYS(2009)	Переключает форму курсора между режимом вставки и замены.

Ввод и вывод - меню

Функции	Описание
<code>\ \</code>	Вывод строк текста.
ACTIVATE MENU	Отображает и активизирует строку меню.
ACTIVATE POPUP	Отображает выпадающее меню на экране и активизирует его.
BAR()	Возвращает номер последнего выбранного элемента меню, определенного с помощью команды DEFINE POPUP , или номер элемента меню Visual FoxPro.
BARPROMPT()	Возвращает текст элемента меню.
CLEAR MENUS	Освобождает заданные элементы из памяти.
CLEAR POPUPS	Освобождает заданные элементы из памяти.
CNTBAR()	Возвращает количество элементов в меню, определенном пользователем, или в системном меню Visual FoxPro.
CNTPAD()	Возвращает количество заголовков меню в строке меню, определенной пользователем, или в системной строке меню Visual FoxPro.
CREATE MENU	Открывает конструктор меню в Visual FoxPro и FoxPro для Windows и построитель меню в FoxPro для Macintosh.
DEACTIVATE MENU	Деактивирует строку меню, определенную пользователем, и удаляет ее с экрана, оставляя при этом определение строки меню в памяти.
DEACTIVATE POPUP	Деактивирует меню, созданное командой DEFINE POPUP .
DEFINE BAR	Создает пункт меню для меню, созданного командой DEFINE POPUP .
DEFINE MENU	Создает строку меню.
DEFINE PAD	Создает заголовок меню (pad) в меню, определенном пользователем или в системном меню Visual FoxPro.
DEFINE POPUP	Создает выпадающее меню.
GETBAR()	Возвращает номер элемента в меню, определенном с помощью команды DEFINE POPUP , или в системном меню Visual FoxPro.
GETPAD()	Возвращает имя меню для заданной позиции строки меню.
HIDE MENU	Скрывает одну или несколько активных пользовательских строк меню.
HIDE POPUP	Скрывает одно или несколько активных меню, созданных с помощью команды DEFINE POPUP .
MENU()	Возвращает имя активной строки меню в виде символьной строки из прописных букв.
MODIFY MENU	Открывает конструктор меню, в котором можно модифицировать или создать систему меню.
MOVE POPUP	Перемещает меню, определенное пользователем с помощью команды DEFINE POPUP , или системное меню FoxPro для MS-DOS (такое как <code>_MFILE</code> , <code>MEDIT</code> или <code>_MDATA</code>) на новое место.

Функции	Описание
MRKBAR()	Определяет, помечен ли элемент пользовательского или системного меню Visual FoxPro.
MRKPAD()	Определяет, помечен ли заголовок меню в строке меню, определенной пользователем, или в системной строке меню Visual FoxPro.
ON BAR	Задаёт меню или строку меню, которые активизируются при выборе определенного элемента из меню.
ON EXIT BAR	Выполняет команду при выходе из заданного элемента меню.
ON EXIT MENU	Выполняет команду при выходе из любого заголовка меню в заданной строке меню.
ON EXIT PAD	Выполняет команду при выходе из заданного заголовка меню.
ON EXIT POPUP	Выполняет команду при выходе из любого элемента заданного меню.
ON PAD	Задаёт меню или строку меню, активизируемые при выборе данного заголовка меню.
ON SELECTION BAR	Задаёт команду, которая выполняется при выборе конкретного элемента меню.
ON SELECTION MENU	Задаёт команду, которая выполняется при выборе любого заголовка меню в строке меню.
ON SELECTION PAD	Задаёт команду, которая выполняется при выборе конкретного заголовка меню в строке меню.
ON SELECTION POPUP	Задаёт команду, которая выполняется при выборе любого элемента из заданного меню или из любого меню.
PAD()	Возвращает самый последний заголовок меню, выбранный из строки меню, в виде строки прописных символов.
POP MENU	Восстанавливает заданное определение строки меню, которое было занесено в стек командой PUSH MENU.
POP POPUP	Восстанавливает заданное определение меню, которое было занесено в стек командой PUSH POPUP.
POPUP()	Возвращает символьную строку, содержащую имя активного меню, или логическое значение, которое указывает, определено ли заданное меню.
PRMBAR()	Возвращает текст элемента меню.
PRMPAD()	Возвращает текст заголовка меню.
PROMPT()	Возвращает текст заголовка меню, выбранного из строки меню, или текст элемента меню, выбранного из меню.
PUSH MENU	Помещает определение строки меню в стек памяти, содержащий определения строк меню.
PUSH POPUP	Помещает определение меню в стек памяти, в котором хранятся определения меню.
RELEASE BAR	Удаляет из памяти заданный элемент некоторого меню или все его элементы.
RELEASE MENUS	Удаляет из памяти строки меню, определенные пользователем.
RELEASE PAD	Удаляет из памяти конкретный заголовок меню или все заголовки меню.
RELEASE POPUPS	Удаляет из памяти конкретное меню или все меню.

Функции	Описание
SET BORDER	Задаёт оформление для текстовых полей, созданных по команде @ ... TO, меню, созданных командой DEFINE POPUP, и окон, созданных по команде DEFINE WINDOW.
SET MARK OF	Определяет символ для пометки заголовка меню, выводит или удаляет символ.
SET SKIP OF	Делает доступным или недоступным меню, строку меню
SIZE POPUP	Изменяет размеры меню, созданного командой DEFINE POPUP.
SKPBAR()	Определяет состояние элемента меню (доступен или недоступен), заданное с помощью команды SET SKIP OF.
SKPPAD()	Определяет состояние заголовка меню (доступен или недоступен), заданное с помощью команды SET SKIP OF
SHOW MENU	Отображает одну или несколько определенных пользователем строк меню, не активизируя их.
SHOW POPUP	Отображает одно или несколько меню, определенных с помощью команды DEFINE POPUP, не активизируя их.
SYS(1500)	Программно активизирует пункт системного меню Visual FoxPro.
SYS(2013)	Возвращает символьную строку, ограниченную пробелом, которая содержит внутренние имена системных меню Visual FoxPro.

Ввод и вывод - печать

Функции	Описание
_ASCII COLS	Задаёт число столбцов в текстовом файле, созданном с помощью команды REPORT ... TO FILE ASCII.
_ASCII ROWS	Задаёт число строк в текстовом файле, созданном с помощью команды REPORT ... TO FILE ASCII
\ \	Вывод строк текста.
? ??	Вычисляет выражения и выводит результаты.
???	Направляет вывод непосредственно на принтер.
@ ... SAY	Выводит данные на экран или на печать в заданную позицию, выраженную номерами строки и столбца. Для Visual FoxPro используйте элемент Label для вывода текста и Textbox – для вывода содержимого полей или переменных памяти.
APRINTERS()	Помещает в массив переменных памяти имена принтеров, которые в данный момент описаны в Windows.
CLEAR RESOURCES	Удаляет из памяти кешированные картинки, шрифты, курсоры или иконки. Если не задано имя файла, все названные ресурсы удаляются из памяти.
CLOSE ALTERNATE	Закрывает альтернативный файл, открытый командой SET ALTERNATE.
DEFINE BOX	Рисует прямоугольник вокруг напечатанного текста. Используйте редактор отчетов.
DISPLAY	Отображает информацию о текущей таблице в основном окне Visual FoxPro или в пользовательском окне.
EJECT	Посылает на принтер код перевода страницы. перевода страницы, если значение системной переменной памяти _PADVANCE равно FORMFEED.
EJECT PAGE	Передаёт на принтер код условного перевода страницы. поточного вывода.
GETPRINTER()	Отображает диалоговое окно Print Setup (Windows) и возвращает имя выбранного принтера.
LIST	Непрерывным потоком отображает информацию о таблице или переменных среды.
ON PAGE	Задаёт команду, которая выполнится, когда печать дойдёт до строки отчёта с заданным номером или когда будет выдана команда EJECT PAGE.
PCOL()	Возвращает текущую позицию по столбцам печатающей головки принтера.
PRINTJOB ... ENDPRINTJOB	Активизирует установки для системных переменных памяти, связанных с заданием печати.
PRINTSTATUS()	Возвращает значение "истина" (.T.), если принтер или печатающее устройство находится в режиме ONLINE, и значение "ложь" (.F.) в противном случае. Аналогично SYS(13), но SYS(13) возвращает READY вместо (.T.) или OFFLINE вместо (.F.). В сети, если очередь принтера принимает задания, статус всегда TRUE независимо от того, находится принтер в offline или online.
PROW()	Возвращает текущий номер строки для печатающей головки принтера.
PRTINFO()	Возвращает действующую в данный момент установку принтера.

Функции	Описание
SET ALTERNATE	Направляет в текстовый файл экранный или печатный вывод, созданный командами ?, ??, DISPLAY или LIST.
SET MARGIN	Задаёт левое поле печати, определяя вид всего вывода, направляемого на принтер. Не имеет смысла при использовании редактора отчётов или выполнении отчётов.
SET PDSETUP	Загружает набор установок драйвера принтера или удаляет текущий набор установок драйвера принтера.
SET PRINTER	Разрешает или запрещает вывод на принтер, или направляет вывод в файл, порт или на сетевой принтер.
SYS(6)	Возвращает текущее печатающее устройство. Как в установках SET PRINTER.
SYS(13)	Возвращает статус принтера.. Используйте PRINTSTATUS().
SYS(102)	Возвращает текущую установку SET PRINTER. Используйте SET("PRINTER").
SYS(1037)	Отображает диалоговое окно Page Setup.
TYPE	Отображает содержимое файла.

Ввод и вывод – отчеты и ярлыки

Функции	Описание
_ASCII COLS	Задаёт число столбцов в текстовом файле, созданном с помощью команды REPORT ... TO FILE ASCII.
_ASCII ROWS	Задаёт число строк в текстовом файле, созданном с помощью команды REPORT ... TO FILE ASCII
\\ \\	Вывод строк текста.
CREATE LABEL	Задаёт имя файла этикетки.
CREATE REPORT	Создаёт отчет согласно описанию.
LABEL	Выводит этикетку для файла согласно описанию этикеток.
MODIFY LABEL	Открывает конструктор этикеток, в котором можно создавать или модифицировать этикетку.
MODIFY REPORT	Открывает Конструктор отчетов, в котором можно модифицировать или создать отчет.
REPORT	Отображает или печатает отчет под управлением описания отчета, созданного в MODIFY REPORT или CREATE REPORT.

Ввод и вывод - окна

Функции	Описание
ACTIVATE WINDOW	Отображает и активизирует одно или несколько окон, определенных пользователем, или системных окон Visual FoxPro.
CLEAR WINDOWS	Освобождает определенные элементы из памяти.
CLOSE MEMO	Закрывает одно или несколько окон редактирования мемо-полей.
CREATE SCREEN	Открывает конструктор форм.
DEACTIVATE WINDOW	Деактивизирует окна, определенные пользователем, или системные окна Visual FoxPro и удаляет их с экрана, но не из памяти.
DEFINE WINDOW	Создает окно и определяет его атрибуты.
HIDE WINDOW	Скрывает активное окно, определенное пользователем, или системное окно Visual FoxPro.
MODIFY SCREEN	Открывает Конструктор форм, в котором можно модифицировать или создать форму.
MODIFY WINDOW	Изменяет определенное пользователем окно или главное окно Visual FoxPro.
MOVE WINDOW	Перемещает на новое место окно, определенное пользователем командой DEFINE WINDOW, или системное окно Visual FoxPro (такое как окно команд или окно просмотра).
MWINDOW()	Возвращает имя окна, в котором позиционируется указатель мыши.
RELEASE WINDOWS	Удаляет из памяти пользовательские окна или системные окна Visual FoxPro.
RESTORE WINDOW	Восстанавливает в памяти описания окон и статусы окон, сохраненные в файле окон или в мемо-поле.
SAVE WINDOWS	Сохраняет заданные описания окон или все описания в файле окон или в мемо-поле.
SCROLL	Прокручивает содержимое области в основном окне Visual FoxPro или пользовательском окне вверх, вниз, влево или вправо.
SET BORDER	Задаёт обрамление для текстовых полей, созданных по команде @ ... TO, меню, созданных командой DEFINE POPUP, и окон, созданных по команде DEFINE WINDOW.
SET CONSOLE	Разрешает или запрещает вывод из программы в основное окно Visual FoxPro или в активное окно, определенное пользователем.
SET SHADOWS	Устанавливает или убирает затенение на пользовательских и системных окнах, меню, диалоговых окнах и предупреждениях.
SET WINDOW OF MEMO	Задаёт окно, в котором можно редактировать мемо-поля.
SHOW WINDOW	Отображает одно или несколько пользовательских или системных окон Visual FoxPro, не активизируя их.
SIZE WINDOW	Изменяет размер окна, созданного командой DEFINE WINDOW, или системного окна Visual FoxPro.

Функции	Описание
SYS(2016)	Возвращает имя окна, использованное в последней команде SHOW GETS WINDOW. Используйте метод Refresh.
WBORDER()	Определяет, имеется ли обрамление у активного или заданного окна.
WCHILD()	Возвращает число дочерних окон в родительском окне или имена дочерних окон в порядке их размещения в родительском окне.
WCOLS()	Возвращает число столбцов в активном или заданном окне.
WEXIST()	Определяет, существует ли заданное пользовательское окно.
WFONT()	Возвращает имя, размер или стиль текущего шрифта для окна в Visual FoxPro
WLAST()	Возвращает имя окна, которое было активно перед текущим окном, или указывает, было ли заданное окно активно перед текущим окном.
WLCOL()	Возвращает координату по столбцам для левого верхнего угла активного или заданного окна.
WLROW()	Возвращает координату по строкам для левого верхнего угла активного или заданного окна.
WMAXIMUM()	Определяет, развернуто ли активное или заданное окно.
WMINIMUM()	Определяет, свернуто ли активное или заданное окно.
WONTOP()	Определяет, находится ли активное или заданное окно впереди всех остальных окон.
WOUTPUT()	Определяет, направлен ли вывод в активное или заданное окно.
WPARENT()	Возвращает имя родительского окна для активного или заданного окна.
WREAD()	Определяет, участвует ли активное или заданное окно в работе текущей команды READ.
WROWS()	Возвращает число строк в активном или заданном окне.
WTITLE()	Возвращает заголовок, назначенный активному или заданному окну.
WVISIBLE()	Определяет, было ли активизировано заданное окно и не является ли оно скрытым.
ZOOM WINDOW	Изменяет размеры и расположение пользовательского окна или системного окна Visual FoxPro.

Программирование – отладка и управление ошибками

Функции	Описание
*	Начало невыполнимого текста в программе.
&&	Указывает на начало неисполняемого комментария внутри строки файла программы.
AERROR()	Создает массив переменных памяти, содержащий информацию о самой последней ошибке, связанной с Visual FoxPro, OLE или ODBC.
ASSERT	Выводит диалоговое окно при вычислении значения в выражении в (.F.). Диалог содержит клавиши Cancel, Debug, Ignore и Ignore All.
CLEAR EVENTS	Останавливает процедуру обработки событий, начатую командой READ EVENTS. После выполнения CLEAR EVENTS программа продолжается с места, следующего за командой READ EVENTS.
CREATE VIEW	Создает файл среды Visual FoxPro. Команда SET VIEW восстанавливает среду, которая создана командой CREATE VIEW. Файлы среды имеют расширение .VUE.
DEBUG	Открывает отладчик Visual FoxPro.
DEBUGOUT	Направляет результаты вычислений в окно Output отладчика.
ERROR	Генерирует ошибку Visual FoxPro
ERROR()	Возвращает номер ошибки, запустившей программу.
EXTERNAL	Предупреждает диспетчер проектов о неопределенной ссылке.
LINENO()	Возвращает номер выполняемой строки программы относительно первой строки основной программы.
LIST CONNECTIONS	Непрерывным потоком отображает информацию об именованных соединениях текущей базы данных.
LIST DATABASE	Непрерывным потоком отображает информацию о текущей базе данных.
LIST DLLS	Отображает непрерывным потоком информацию о DLL-функциях 32-разрядной версии Windows, зарегистрированных в Visual FoxPro с помощью команды DECLARE - DLL.
LIST FILES	Выводит сплошным потоком информацию о файлах в текущем каталоге, которые соответствуют маске.
LIST MEMORY	Выводит сплошным потоком информацию о содержимом текущей памяти и системных переменных.
LIST OBJECTS	Непрерывным потоком отображает информацию об объекте или группе объектов.
LIST PROCEDURES	Непрерывным потоком отображает имена хранимых процедур в текущей базе данных.
LIST STATUS	Отображает непрерывным потоком информацию таблицу и/или состояние среды.
LIST TABLES	Непрерывным потоком отображает список всех таблиц текущей базы данных и информацию об этих таблицах.
LIST VIEWS	Непрерывным потоком отображает информацию о представлениях SQL в текущей базе данных.
MESSAGE()	Возвращает текущее сообщение об ошибке в виде символьной строки или содержимое строки программы, вызвавшей ошибку.

Функции	Описание
MESSAGEBOX()	Отображает диалоговое окно, определенное пользователем.
NOTE	Отмечает начало неисполняемой строки комментария в файле программы.
ON ERROR	Задаёт команду, которая выполняется при возникновении ошибки.
ON READERROR	Задаёт команду, которая выполняется в случае ошибки ввода данных. Используйте метод Error.
ON()	Возвращает команду, назначенную любой из следующих ON APLABOUT, ON ERROR, ON ESCAPE, ON KEY, ON KEY LABEL, ON MACHELP, ON PAGE или ON READERROR.
PROGRAM()	Возвращает имя выполняемой в данный момент программы, или имя программы, выполнявшейся в момент возникновения ошибки. Команда идентична SYS(16), но имеет больше возможностей.
RETRY	Повторно выполняет предыдущую команду.
SET ALTERNATE	Направляет в текстовый файл экранный или печатный вывод, созданный командами ?, ??, DISPLAY или LIST.
SET ASSERT	Определяет, выполняется ли команда ASSERT или игнорируется..
SET COVERAGE	Переключает режим проверки кода или определяет выходной файл, в который выводится информация по анализу выполняемого кода.
SET DEBUG	Делает окна отладки и трассировки доступными или недоступными из системы меню Visual FoxPro.
SET DEBUGOUT	Направляет отладочный вывод в файл. Отладочный вывод формируется командами ASSERT, DEBUGOUT и событиями, которые определены в SET EVENTLIST или в диалоге Трассировки событий (Event Tracking).
SET DEVELOPMENT	Предписывает Visual FoxPro сравнивать при запуске программы дату и время создания программы с датой и временем создания соответствующего скомпилированного объектного файла.
SET ECHO	Открывает окно трассировки для отладки программы.
SET ESCAPE	Определяет, должно ли нажатие клавиши Esc прерывать выполнение программ и команд.
SET EVENTLIST	Определяет список слежения событий в Debug Output Window или в выходной файл, определенный в SET EVENTTRACKING.
SET EVENTTRACKING	Переключает режим слежения событий или определяет в какой файл выводится информация о слежении событий.
SET LOGERRORS	Определяет, посылает ли Visual FoxPro сообщения об ошибках компиляции в текстовый файл.
SET NOTIFY	Разрешает или отменяет отображение некоторых системных сообщений.
SET STEP	Открывает окно трассировки и приостанавливает выполнение программы для проведения отладки
SET TALK	Определяет, будет ли Visual FoxPro отображать результаты команд.
SET TRBETWEEN	Включает или отключает трассировку, выполняемую в окне трассировки в промежутке между точками останова.
SET VIEW	Открывает или закрывает окно представления или восстанавливает среду Visual FoxPro из файла представления.

Функции	Описание
SYS(0)	Возвращает сетевую информацию о машине в случае, когда Visual FoxPro используется в сетевой среде.
SYS(16)	Возвращает имя файла выполняющейся программы. Если SYS(16) возвращает путь с именем программы, то PROGRAM() возвращает имя програмы.
SYS(18)	Возвращает набранное прописными буквами имя переменной памяти, элемента массива или поля, использованное при создании текущего элемента управления. Используйте свойство Active Control .
SYS(103)	Возвращает текущую установку SET TALK. Используйте SET(“TALK”).
SYS(2018)	Возвращает параметр последнего сообщения об ошибке.
TYPE()	Вычисляет символьное выражение и возвращает тип данных, которому принадлежит содержимое этого выражения.
VARREAD()	Возвращает имя переменной памяти, элемента массива или поля (прописными буквами), использованного для создания текущего элемента управления. Используйте свойства ControlSource или Name.
WAIT	Выдает сообщение и приостанавливает работу Visual FoxPro до тех пор, пока не будет нажата какая-либо клавиша или кнопка мыши.

Программирование – управление событиями

Функции	Описание
DOEVENTS	Выполняет все отложенные события Windows. События помежаются в очередь, если установлено свойство AutoYield в значение “ложь” (.F.) , а программа выполняется. DOEVENTS выполняет все отложенные события Windows и выполняет пользовательский код, с ними связанный.
ON APLABOUT	Задаёт команду, которая выполняется, когда пользователь выбирает команду About в меню Apple.
ON ERROR	Задаёт команду, которая выполняется при возникновении ошибки.
ON ESCAPE	Задаёт команду, которая выполнится, если во время работы программы или команды нажать клавишу Esc.
ON KEY	Задаёт команду, которая будет выполнена при нажатии некоторой клавиши или комбинации клавиш во время работы команды READ.
ON KEY LABEL	Задаёт команду, которая будет выполнена, если нажать некоторую клавишу или комбинацию клавиш или щёлкнуть кнопкой мыши.
ON MACHelp	Задаёт команду, которая будет выполнена, если пользователь выберет в меню Help элемент Microsoft FoxPro Help.
ON PAD	Задаёт меню или строку меню, активизируемые при выборе данного заголовка меню.
ON PAGE	Задаёт команду, которая выполнится, когда печать дойдёт до строки отчёта с заданным номером или когда будет выдана команда EJECT PAGE.
ON SELECTION BAR	Задаёт команду, которая выполняется при выборе конкретного элемента меню.
ON SELECTION MENU	Задаёт команду, которая выполняется при выборе любого заголовка меню в строке меню.
ON SELECTION PAD	Задаёт команду, которая выполняется при выборе конкретного заголовка меню в строке меню.
ON SELECTION POPUP	Задаёт команду, которая выполняется при выборе любого элемента из заданного меню или из любого меню.
ON SHUTDOWN	Specifies a command that executes when you try to exit Visual FoxPro, or Microsoft Windows.
ON ()	Возвращает команду, назначенную любой из следующих ON APLABOUT, ON ERROR, ON ESCAPE, ON KEY, ON KEY LABEL, ON MACHelp, ON PAGE или ON READERROR.
POP KEY	Восстанавливает назначения команды ON KEY LABEL, занесённые в стек командой PUSH KEY.
PUSH KEY	Заносит все текущие установки команд ON KEY LABEL в стек памяти.
READ EVENTS	Начинает процесс обработки событий. Если построены приложения .EXE или .APP , в которых в главной программе выполняется DO FORM, необходимо задать READ EVENTS перед DO FORM для подсказки FoxPro необходимости обработки процесса событий. При работе через окно команд FoxPro автоматически обрабатывает последовательность событий).

Программирование –доступ к файлам на низком уровне

Функции	Описание
FCHSIZE()	Изменяет размер файла, открытого с помощью файловой функции низкого уровня.
FCLOSE()	Записывает и закрывает файл или коммуникационный порт, открытые с помощью функций доступа к файлам на низком уровне.
FCREATE()	Создает и открывает файл низкого уровня.
FEOF()	Определяет, установлен ли указатель файла в конце файла.
FERROR()	Возвращает номер, соответствующий самой последней ошибке в файловой функции низкого уровня.
FFLUSH()	Сбрасывает на диск файл, открытый с помощью функции низкого уровня.
FGETS()	Возвращает последовательность байтов из файла или порта связи, открытого с помощью файловой функции низкого уровня до символа возврата каретки.
FOPEN()	Открывает файл или порт связи для использования файловыми функциями низкого уровня.
FPUTS()	Записывает символьную строку с возвратом каретки и переводом строки в файл или порт связи, открытый с помощью файловой функции низкого уровня.
FREAD()	Возвращает заданное число байтов из файла или порта связи, открытого с помощью файловой функции низкого уровня.
FSEEK()	Перемещает указатель файла, открытого с помощью файловой функции низкого уровня.
FWRITE()	Записывает символьную строку в файл или порт связи, открытый с помощью файловой функции низкого уровня.
HEADER()	Возвращает число байтов в заголовке файла текущей или заданной таблицы.

Программирование – управление переменными памяти

Функции	Описание
&	Выполняет макроподстановку.
=	Вычисляет одно или несколько выражений.
ACOPY()	Копирует элементы из одного массива в другой.
ADEL()	Удаляет элемент из одномерного массива или удаляет строку или столбец из двумерного массива.
ADIR()	Заносит в массив информацию о файлах и возвращает количество файлов.
AELEMENT()	Вычисляет номер элемента массива исходя из индексов этого элемента.
AINS()	Вставляет элемент в одномерный массив или вставляет строку или столбец в двумерный массив.
LEN()	Возвращает количество символов в символьном выражении.
ASCAN()	Просматривает массив в поисках элемента, данные которого имеют то же значение и тот же тип, что и указанное выражение.
ASORT()	Сортирует элементы массива в возрастающем или убывающем порядке.
ASUBSCRIPT()	Возвращает строчный или столбцевой индекс элемента массива исходя из его номера.
AVERAGE	Вычисляет арифметическое среднее числовых выражений или полей.
CLEAR ALL	Освобождает определенные элементы из памяти.
CLEAR MEMORY	Освобождает определенные элементы из памяти.
DISPLAY MEMORY	Отображает текущее содержимое переменных памяти и массивов.
GETEXPR	Отображает диалоговое окно построителя выражений, в котором можно создать выражение и сохранить его в переменной памяти или в элементе массива.
LIST MEMORY	Выводит сплошным потоком информацию о содержимом текущей памяти и системных переменных.
LOCAL	Создает локальные переменные памяти и массивы переменных памяти.
LPARAMETERS	Присваивает локальным переменным памяти или массивам переменных значения, переданные из вызывающей программы.
PARAMETERS	Возвращает количество параметров, переданных последней вызванной программой, процедуре или пользовательской функции.
PRIVATE	Скрывает от текущей программы заданные переменные памяти или массивы, которые были определены в вызывающей программе.
PUBLIC	Определяет глобальные переменные памяти или массивы.
REGIONAL	Создает региональные переменные памяти и массивы.
RELEASE	Удаляет переменные памяти и массивы из памяти.

Функции	Описание
RESTORE FROM	Извлекает переменные памяти и массивы переменных памяти, сохраненные в файле переменных памяти или в тето-поле, и помещает их в память.
SAVE TO	Сохраняет текущие переменные памяти и массивы в файле переменных памяти или в тето-поле.
STORE	Сохраняет данные в переменной памяти, массиве или в элементе массива.
SUM	Суммирует все или только заданные числовые поля в таблице, выбранной в данный момент.
WAIT	Выдает сообщение и приостанавливает работу Visual FoxPro до тех пор, пока не будет нажата какая-либо клавиша или кнопка мыши.

Программирование – объектно-ориентированное программирование

Функции	Описание
::	Выполняет метод родительского класса в рамках метода подкласса.
@ ... CLASS	Создает элемент управления или объект, который можно активизировать с помощью команды READ.
_WIZARD	Содержит имя программы постоителя (мастера) Visual FoxPro.
AClass()	Помещает в массив переменных памяти имя класса объекта, а также имена его родительских классов.
ADATABASES()	Помещает в массив переменных памяти имена всех открытых баз данных и описания путей для этих имен.
ADBOBJECTS()	Помещает в массив переменных памяти имена соединений, отношений, таблиц или представлений SQL из текущей базы данных.
ADD CLASS	Добавляет определение класса в библиотеку визуальных классов .VCX.
AINSTANCE()	Помещает экземпляры класса в массив переменных памяти и возвращает число занесенных в массив экземпляров.
AMEMBERS()	Помещает в массив переменных памяти имена свойств, процедур и объектов-членов заданного объекта
APPEND GENERAL	Импортирует объект OLE из файла и помещает его в поле типа General.
ASELOBJ()	Помещает в массив переменных памяти указатели на выбранные в данный момент элементы управления в активном конструкторе форм.
_BUILDER	Содержит имя приложения-конструктора Visual FoxPro
CLEAR EVENTS	Останавливает процедуру обработки событий, начатую командой READ EVENTS. После выполнения CLEAR EVENTS программа продолжается с места, следующего за командой READ EVENTS.
COMPOBJ()	Сравнивает свойства двух объектов и возвращает значение "истина" (.T.), если их свойства и значения свойств идентичны.
CREATE CLASS	Задаст имя определения класса, который предстоит создать.
CREATE CLASSLIB	Создает новый, пустой .VCX файл библиотеки визуальных классов.
CREATE FORM	Открывает конструктор форм.
CREATEBINARY()	Преобразует тип данных из Visual FoxPro в бинарную строку, которую можно передать в ActiveX или использовать при OLE..
CREATEOBJECT()	Создает объект на основе определения класса или на основе доступного приложения OLE.
DEFINE CLASS	Создает пользовательский класс или подкласс, определяет свойства, события и методы для них.
DISPLAY OBJECTS	Отображает информацию об объекте или группе объектов.

Функции	Описание
DODEFAULT()	В подклассе выполняет событий или метод родителя, которые имеют эти же имена.
DOEVENTS	Выполняет все отложенные события Windows. События поочередно выполняются, если установлено свойство AutoYield в значение “ложь” (.F.), а программа выполняется. DOEVENTS выполняет все отложенные события Windows и выполняет пользовательский код, с ними связанный.
GETOBJECT()	Активизирует объект OLE automation и создает ссылку на него.
GETPEM()	Возвращает текущее значение свойства или программный код для события или метода.
LIST OBJECTS	Непрерывным потоком отображает информацию об объекте или группе объектов.
LOADPICTURE()	Создает ссылку на bitmap, icon или Windows meta файл. Многие свойства для элементов ActiveX требуют эту объектную ссылку. Например, ActiveX Outline поддерживает свойство PictureOpen, которое требует объектную ссылку для ее
MODIFY CLASS	Открывает конструктор классов, давая возможность изменить имеющееся определение класса или создать новое.
OBJTOCLIENT()	Возвращает позицию или размер элемента управления или объекта относительно содержащего его объекта Form.
PEMSTATUS()	Возвращает определенные атрибуты для свойств, событий или методов (например, Read-Only, Changed, Protected, Type, etc...).
READ EVENTS	Начинает процесс обработки событий. Если построены приложения .EXE или .APP, в которых в главной программе выполняется DO FORM, необходимо задать READ EVENTS перед DO FORM для подсказки FoxPro необходимости обработки процесса событий. При работе через окно команд FoxPro автоматически обрабатывает последовательность событий).
RELEASE CLASSLIB	Закрывает библиотеку .VCX.
REMOVE CLASS	Удаляет определение класса из библиотеки визуальных классов .VCX.
RENAME CLASS	Переименовывает определение класса в библиотеку классов.
_SCREEN	Задаёт свойства и методы основного окна Visual FoxPro.
SAVEPICTURE()	Создаёт файл картинки (.BMP) на основании объектной ссылки на нее.
SET CLASSLIB	Открывает библиотеку визуальных классов .VCX, содержащую определения классов.
SET MESSAGE	Задаёт сообщение, отображаемое в основном окне Visual FoxPro или в графической строке состояния, или задаёт место вывода сообщений, связанных с пользовательскими строками меню и командами меню.
SET NULLDISPLAY	По умолчанию Visual FoxPro выводит значение .NULL. для пустых значений объектов. Использование команды SET NULLDISPLAY позволяет сменить выводимый текст.
SET OLEOBJECT	Указывает, должна ли система Visual FoxPro просматривать реестр Windows Registry, когда объект найти не удастся.
SET PALETTE	Определяет, используется ли в Visual FoxPro цветовая палитра, принимаемая по умолчанию.
SET SYSMENU	Делает доступной или недоступной системную строку меню Visual FoxPro в процессе выполнения программы и даёт возможность изменить конфигурацию строки меню.

Функции	Описание
SYS(1016)	Возвращает объем памяти, задействованной определенными пользовательскими объектами.
SYS(1269)	Возвращает логическое значение, которое показывает изменилось ли заданное свойство объекта по отношению к значению, принятому по умолчанию.
SYS(1270)	Возвращает ссылку на объект, который находится под курсором мыши или в определенном месте экрана.
SYS(1271)	Возвращает имя файла .SCX, в котором храниться определенный объект.
SYS(1272)	Возвращает иерархию объектов для заданного объекта. Функция работает только при выполнении, а не в режиме создания.
SYS(2333)	Если объект ActiveX не работает должным образом при его инициализации, используйте SYS(2333) или SYS(2333, 0) перед инициализацией объекта для отключения интерфейса двойной поддержки. После инициализации объекта используйте SYS(2333, 1) для включения интерфейса двойной поддержки для других элементов, которые будут инициализированы позднее.
SYS(3004)	Возвращает идентификатор Locale ID, используемый объектами автоматизации OLE и элементами управления OLE.
SYS(3005)	Устанавливает идентификатор Locale ID, используемый объектами автоматизации OLE и элементами управления OLE.
SYS(3006)	Устанавливает идентификаторы Language ID и Locale ID.
SYS(3053)	Возвращает дескриптор среды ODBC.
SYSMETRIC()	Возвращает размеры экранных элементов операционной системы.
WITH ... ENDWITH	<p>Задаёт несколько свойств для объекта. Позволяет использовать нотацию с точкой (например, .Value) при выполнении множественных манипуляций с объектом, это сокращает объём ввода и повышает производительность. Для примера, можно:</p> <pre> WITH ThisForm.Txt_Total_Owed .Value = 2452.22 .ReadOnly = .T. .FontBold = .T. .FontSize = 10 .Visible = .T. ENDWITH </pre>

Программирование – выполнение программ

Функции	Описание
BUILD APP	Создает файл приложения с расширением имени .APP, используя информацию из файла проекта.
BUILD DLL	Создает библиотеку типа (DLL) с расширением . , используя информацию из файла проекта.
BUILD EXE	Создает на основе проекта исполняемый файл.
BUILD PROJECT	Создает и строит файл проекта.
CALL	Выполняет двоичный файл, внешнюю команду или внешнюю функцию, помещенную в память с помощью команды LOAD. Используйте SET LIBRARY.
CANCEL	Прекращает выполнение текущего файла программы Visual FoxPro.
CLEAR PROGRAM	Освобождает определенные элементы из памяти.
CLOSE PROCEDURE	Закрывает различные типы файлов.
COMPILE	Компилирует один или несколько исходных файлов и для каждого из них создает объектный файл.
CREATE PROJECT	Открывает диспетчер проектов, в котором можно создать проект.
#DEFINE ... #UNDEF	Создает и освобождает константы этапа компиляции.
DO	Выполняет программу или процедуру Visual FoxPro.
DO FORM	Выполняет скомпилированную форму или набор форм, созданных с помощью конструктора форм.
FUNCTION	Идентифицирует начало определения пользовательской функции.
#IF ... #ENDIF	Условно включает код на этапе компиляции.
#INCLUDE	Указывает, что препроцессор Visual FoxPro должен рассматривать содержимое заданного файла заголовка как включенное в программу Visual FoxPro.
LOAD	Помещает в память двоичный файл, внешнюю команду или внешнюю функцию. Используйте SET LIBRARY
LPARAMETERS	Присваивает локальным переменным памяти или массивам переменных значения, переданные из вызывающей программы.
MESSAGEBOX()	Отображает диалоговое окно, определенное пользователем.
MODIFY PROJECT	Открывает Диспетчер проектов, в котором можно модифицировать или создать файл проекта.
PARAMETERS	Возвращает количество параметров, переданных последней вызванной программой, процедуре или пользовательской функции.
PARAMETERS()	Возвращает количество параметров, переданных последней вызванной программой, процедуре или пользовательской функции.

Функции	Описание
PCOUNT()	Возвращает количество переданных параметров в программу, процедуру или функцию. Аналогична PARAMETERS(). Используйте PARAMETERS() как более информативную.
PROCEDURE	Идентифицирует начало процедуры в файле программы.
RELEASE LIBRARY	Удаляет одну библиотеку внешних программ API из памяти.
RELEASE MODULE	Удаляет из памяти один двоичный файл, внешнюю команду или внешнюю функцию.
RELEASE PROCEDURE	Закрывает файл процедуры, открытый с помощью команды SET PROCEDURE.
RESUME	Продолжает выполнение приостановленной программы.
RETURN	Возвращает управление вызывающей программе.
RUN !	Выполняет команды и программы внешней операционной системы.
SET ESCAPE	Определяет, должно ли нажатие клавиши Esc прерывать выполнение программ и команд.
SET LIBRARY	Открывает файл внешней библиотеки API (интерфейса прикладного программирования).
SET LOGERRORS	Определяет, посылает ли Visual FoxPro сообщения об ошибках компиляции в текстовый файл.
SET PROCEDURE	Открывает файл процедур. (.PRG или .FXP файл, содержащий процедуры и функции)
SET SYSMENU	Делает доступной или недоступной системную строку меню Visual FoxPro в процессе выполнения программы и дает возможность изменить конфигурацию строки меню.
SET UDFPARMS	Задаёт в Visual FoxPro способ передачи параметров в пользовательскую функцию (UDF) - по значению или по ссылке.
SET VOLUME	Maps MS-DOS drive designators (A:, B:, C:, and so on) to volumes or folders in FoxPro for Macintosh.
SET XCMDFILE	аёт файл ресурсов Macintosh для внешних команд (XCMD) и внешних функций (XFCN).
SUSPEND	Делает паузу в выполнении программы и возвращает в интерактивную среду Visual FoxPro.
SYS(2015)	Возвращает уникальное 10-символьное имя процедуры, которое состоит из символа подчеркивания и некоторой комбинации букв и цифр.
WAIT	Выдает сообщение и приостанавливает работу Visual FoxPro до тех пор, пока не будет нажата какая-либо клавиша или кнопка мыши.

Программирование – структурное программирование

Функции	Описание
DO CASE ... ENDCASE	Выполняет первым тот набор команд, которые удовлетворяют заданным условиям.
DO WHILE ... ENDDO	Выполняет группу команд, входящих в условный цикл.
EXIT	Осуществляет выход из цикла DO WHILE, FOR или SCAN.
FOR ... ENDFOR	Выполняет группу команд заданное число раз.
FOR EACH ... ENDFOR	Выполняет набор команд для каждого из элементов массива или коллекции. Конструкция полезна при работе с одномерными массивами, с каждым из которых надо выполнить последовательность однотипных действий. Быстрее чем обычно употребляемая конструкция цикла FOR....ENDFOR.
#IF ... #ENDIF	Условно включает код на этапе компиляции.
#IFDEF #IFNDEF ... #ENDIF	Выполняет условное включение группы команд в код на этапе компиляции, если определена константа этапа компиляции.
IF ... ENDIF	Осуществляет условное выполнение группы команд исходя из значения логического выражения
IIF()	Возвращает одно из двух значений, в зависимости от значения данного логического выражения.
SCAN ... ENDSCAN	Перемещает указатель записи по таблице, выбранной в данный момент, и для каждой записи, удовлетворяющей указанным условиям, выполняет определенный блок команд.

Экономические функции

Функции	Описание
CALCULATE	Выполняет финансовые и статистические операции над полями в таблице или над выражениями, включающими поля.
FV()	Возвращает величину будущих денежных инвестиций.
PAYMENT()	Возвращает размер периодических платежей для ссуды с фиксированным процентом.
PV()	Возвращает текущую величину инвестированных средств

Разработка интернациональных приложений

Функции	Описание
CPCONVERT()	Преобразует символьные поля, метео-поля или символьные выражения в другую кодовую страницу.
CPCURRENT()	Возвращает номер кодовой страницы из файла конфигурации Visual FoxPro (если она там установлена) или номер текущей кодовой страницы операционной системы.
CPDBF()	Возвращает кодовую страницу, которой была помечена данная открытая таблица.
IDXCOLLATE()	Возвращает последовательность сортировки для индекса или тега индекса.
ISFLOCKED()	Возвращает значение "истина" (.T.), если определенная таблица заблокирована, в противном случае возвращается "ложь" (.F.). ISFLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
ISRLOCKED()	Возвращает значение "истина" (.T.), если запись заблокирована, в противном случае возвращается "ложь" (.F.). ISRLOCKED() аналогична SYS(2011), но возвращает логическое значение, которое не требует локализации для интернациональных приложений.
SET COLLATE	Задаёт последовательность сортировки для символьных полей в последующих операциях индексирования и сортировки.
SET CPCOMPILE	Задаёт кодовую страницу для компилируемых программ.
SET CPDIALOG	Определяет, будет ли при открытии таблицы отображаться диалоговое окно Code Page.
SET FDOW	Задаёт первый день недели.
SET F WEEK	Задаёт требования, предъявляемые к первой неделе года.
SET NOCPTRANS	Запрещает преобразовывать в другую кодовую страницу выбранные поля открытой таблицы.
SET POINT	Определяет символ, используемый в качестве десятичного разделителя при изображении числовых выражений и выражений типа Currency.
SET SEPARATOR	Задаёт символ, который отделяет друг от друга группы из трех цифр, стоящие слева от десятичной точки.
SET SYSFORMATS	Определяет, должны ли системные установки Visual FoxPro обновляться вместе с текущими системными установками Windows.

Visual FoxPro события

События	Выполняет метод
Activate	Имеет место, когда становится активным объект FormSet, Form или Page или когда отображается объект Toolbar
AfterCloseTables	Имеет место после освобождения таблиц или представлений, заданных в среде данных для формы, набора форм или отчета.
AfterDock	Имеет место после того, как объект ToolBar фиксируется.
AfterRowColChange	Имеет место, когда пользователь переходит в другую строку или другой столбец сетки, после того как новая ячейка получит фокус, и после того как в новой строке или новом столбце произойдет событие When для данного объекта.
BeforeDock	Имеет место перед фиксацией объекта ToolBar.
BeforeOpenTables	Имеет место непосредственно перед открытием таблиц и представлений, связанных со средой данных набора форм, формы или отчета.
BeforeRowColChange	Имеет место перед тем, как пользователь изменяет активную строку или столбец, прежде чем новая ячейка получит фокус.
Click	Имеет место в следующих случаях: кнопку мыши при курсоре, указывающем на элемент управления, или изменил значение какого-либо элемента управления, или щелкнул пустую область формы.
DblClick	Имеет место, когда пользователь быстро дважды нажимает и тут же отпускает левую (первую) кнопку мыши.
Deactivate	Имеет место, когда объект-контейнер (например форма) перестает быть активным, потому что ни один из содержащихся в нем объектов не находится в фокусе.
Deleted	Имеет место, когда пользователь удаляет пометку записи на удаление или когда выдается команда DELETE.
Destroy	Имеет место, когда освобождается объект.
DoCmd	Выполняет команду Visual FoxPro, например, при Visual FoxPro application automation server.
DownClick	Имеет место, когда пользователь нажимает в элементе управления кнопку со стрелкой вниз. (Combo Box, List Box или Spinner).
DragDrop	Имеет место при завершении операции перетаскивания с помощью мыши.
DragOver	Имеет место, когда элемент управления, перетаскиваемый с помощью мыши, накрывает объект назначения.
DropDown	Имеет место, когда после нажатия кнопки со стрелкой в элементе управления ComboBox или ListBox должен появиться список. <i>Используйте событие для обработки выбранного элемента перед его выводом.</i>
Error	Имеет место при возникновении в методе ошибки этапа выполнения.
ErrorMessage	Имеет место, когда событие Valid возвращает значение "ложь" (.F.), и обеспечивает выдачу сообщения об ошибке.
GotFocus	Имеет место, когда объект получает фокус - в результате действий со стороны пользователя или в результате выполнения кода.
Init	Имеет место при создании объекта.

События	Выполняет метод
InteractiveChange	Имеет место, когда пользователь изменяет значение элемента управления с помощью клавиатуры или мыши.. Это событие наступает после каждого символа .
KeyPress	Имеет место, когда пользователь нажимает и отпускает клавишу.
Load	Имеет место непосредственно перед созданием объекта. Если метотод вернул .F. , объект не создается.
LostFocus	Имеет место, когда объект теряет фокус.
Message	Когда элемент теряет фокус, отображает сообщение в строке состояния внизу экрана.
MiddleClick	Имеет место, когда пользователь нажал среднюю кнопку мыши на трехкнопочной мыши.
MouseDown	Имеет место, когда пользователь нажимает кнопку мыши.
MouseMove	Имеет место, когда пользователь перемещает указатель мыши по объекту.
MouseUp	Имеет место, когда пользователь отпускает кнопку мыши.
MouseWheel	Имеет место, когда пользователь вращает колесико на мыши, которая его имеет.
Moved	Имеет место, когда объект перемещается на новую позицию или когда установки свойства Top или Left объекта-контейнера изменяются программным способом
Paint	Имеет место, когда перерисовывается форма или панель инструментов.
ProgrammaticChange	Имеет место, когда в коде изменяется значение элемента управления.
QueryUnload	Имеет место перед выгрузкой объекта Form.
RangeHigh	Имеет место, когда элемент управления Spinner или TextBox теряет фокус и когда элемент управления ComboBox или ListBox получает фокус.
RangeLow	Имеет место, когда элемент управления Spinner или TextBox теряет фокус или когда элемент управления ComboBox или ListBox получает фокус.
Resize	Имеет место при изменении размеров объекта.
RightClick	Имеет место, когда пользователь нажимает и затем отпускает правую кнопку мыши и при этом указатель мыши находится над элементом управления.
Scrolled	Имеет место в элементе управления Grid при щелчке горизонтальной или вертикальной полосы прокрутки или при перемещении бегунка.
Timer	Имеет место, когда истекает интервал времени, заданный свойством Interval.
UIEnable	Имеет место для всех объектов, содержащихся внутри объекта Page, каждый раз, когда этот объект Page активизируется или деактивизируется.
UnDock	Имеет место, когда объект ToolBar сдвигается мышью с позиции, в которой он был фиксирован.
Unload	Имеет место при освобождении объекта.
UpClick	Имеет место, когда пользователь нажимает в элементе управления кнопку прокрутки со стрелкой вверх.

События	Выполняет метод
Valid	Имеет место перед тем, как элемент управления теряет фокус. Если метод вернул значение .F. , Visual FoxPro не позволит покинуть фокус с данного элемента.
When	Имеет место перед тем, как элемент управления получает фокус. Если метод вернул.F., Visual FoxPro не даст возможности получить фокус элементу.

Visual FoxPro события для элементов редактирования

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
AddItem	M	Добавляет новый элемент списка в элемент управления поле ввода с раскрывающимся списком или поле ввода со списком, предоставляя при этом возможность задать индекс элемента.		+	+					
AddListItem	M	Добавляет новый элемент списка в элемент управления поле ввода с раскрывающимся списком или поле ввода со списком, давая возможность задать идентификатор элемента.		+	+					
Alignment	C	Задаёт вид выравнивания текста, связанного с данным элементом управления.	+		+		+	+	+	+
AllowTabs	C	Определяет, разрешены ли табуляторы в элементе управления поле редактирования.								+
AutoSize	C	Определяет, нужно ли автоматически изменять размеры элемента управления, чтобы в нем можно было разместить его содержимое.	+			+	+			
BackColor	C	Задают цвет фона, используемые при отображении текста и графики в объекте.	+		+		+	+	+	+
BackStyle	C	Задаёт фон объекта - прозрачный или непрозрачный.	+				+		+	+
BaseClass	C	Задаёт имя базового класса Visual FoxPro, на основе которого создан указанный объект.	+	+	+	+	+	+	+	+
BorderColor	C	Задаёт цвет обрамления объекта.		+	+			+	+	+
BorderStyle	C	Задаёт стиль обрамления объекта.			+			+	+	+
BoundColumn	C	Определяет, какой столбец многостолбцового элемента управления ListBox или ComboBox привязан к свойству Value этого элемента управления.		+	+					
BoundTo	C	Определяет как задается свойство Value для combobox или listbox свойствами List или ListIndex.		+	+					
Cancel	C	Определяет, является ли элемент управления CommandButton кнопкой Cancel; это означает, что если пользователь нажмет клавишу ESC, произойдет событие Click для кнопки Cancel.				+				
Caption	C	Задаёт текст, отображаемый в названии объекта.	+			+	+			
Century	C	Определяет выводится ли век в дате для элемента textbox.							+	
Class	C	Возвращает имя класса, на базе которого создан объект.	+	+	+	+	+	+	+	+
ClassLibrary	C	Задаёт имя файла библиотеки пользовательских классов, которая содержит класс данного объекта.	+	+	+	+	+	+	+	+
Clear	M	Очищает содержимое элемента управления ComboBox или ListBox.		+	+					
Click	M	Имеет место в следующих случаях: - Пользователь нажал и отпустил левую (главную) кнопку мыши при курсоре, указывающем на элемент управления, или изменил значение какого-либо элемента управления, или щелкнул пустую область формы.	+	+	+	+	+	+	+	+
ColorScheme	C	Задаёт цветовую схему, используемую в элементах	+	+	+	+	+	+	+	+

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
		управления.								
ColorSource	C	Определяет, как устанавливаются цвета элемента управления.	+	+	+	+	+	+	+	+
ColumnCount	C	Задаёт количество объектов Column в элементе управления Grid, ComboBox или ListBox.		+	+					
ColumnLines	C	Показывает или скрывает линии между столбцами.		+	+					
ColumnWidths	C	Задаёт ширину столбцов для элемента управления ComboBox или ListBox.		+	+					
Comment	C	Сохраняет информацию об объекте.	+	+	+	+	+	+	+	+
ControlSource	C	Задаёт источник данных, к которому привязывается объект.	+	+	+		+	+	+	+
DateFormat	C	Определяет формат значений типа Date и DateTime при выводе в элементе textbox.							+	
DateMark	C	Определяет разделитель для значений типа Date и DateTime в элементе textbox.							+	
DblClick	M	Имеет место, когда пользователь быстро дважды нажимает и тут же отпускает левую (первую) кнопку мыши.	+	+	+		+	+	+	+
Default	C	Определяет, какой элемент управления CommandButton реагирует на нажатие клавиши ENTER, если в активном объекте Form имеется несколько командных кнопок.				+				
Destroy	M	Имеет место, когда освобождается объект.	+	+	+	+	+	+	+	+
DisabledBackColor	C	Задают цвет фона для недоступного элемента управления.	+	+	+		+	+	+	+
DisabledBackColor	C	Задают цвет фона для недоступных элементов списка в элементе управления ComboBox или ListBox.		+	+					
DisabledForeColor	C	Задают основной цвет символов для недоступного элемента управления.	+	+	+	+	+	+	+	+
DisabledItemForeColor	C	Задают основной цвет символов для недоступных элементов списка в элементе управления ComboBox или ListBox.		+	+					
DisabledPicture	C	Задаёт рисунок, который нужно отобразить для недоступного элемента управления.	+			+	+			
DisplayValue	C	Задаёт содержимое первого столбца выбранного элемента списка в элементе управления ListBox или ComboBox.		+	+					
DownClick	M	Имеет место, когда пользователь нажимает в элементе управления кнопку со стрелкой вниз.			+			+		
DownPicture	C	Задаёт рисунок, отображаемый при выборе элемента управления.	+			+	+			
Drag	M	Начинает, заканчивает или отменяет операцию перетаскивания с помощью мыши.	+	+	+	+	+	+	+	+
DragDrop	M	Имеет место при завершении операции перетаскивания с помощью мыши.	+	+	+	+	+	+	+	+
DragIcon	C	Задаёт значок, отображаемый в качестве курсора во время операции перетаскивания с помощью мыши.	+	+	+	+	+	+	+	+
DragMode	C	Задаёт режим для операции перетаскивания с помощью мыши - явный или автоматический.	+	+	+	+	+	+	+	+

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
DragOver	M	Имеет место, когда элемент управления, перетаскиваемый с помощью мыши, накрывает объект назначения.	+	+	+	+	+	+	+	+
DropDown	M	Имеет место, когда после нажатия кнопки со стрелкой в элементе управления ComboBox должен появиться список.			+					
Enabled	C	Определяет, может ли объект реагировать на события, генерируемые пользователем.	+	+	+	+	+	+	+	+
Error	M	Имеет место при возникновении в методе ошибки этапа выполнения.	+	+	+	+	+	+	+	+
ErrorMessage	M	Имеет место, когда событие Valid возвращает значение "ложь" .	+	+	+	+	+	+	+	+
FirstElement	C	Задаёт первый элемент массива, который должен быть отображен в элементе управления ComboBox или ListBox.		+	+					
FontBold	C	Задают для текста стиль Bold (полужирный).	+	+	+	+	+	+	+	+
FontCondense	C	Определяет стиль текста как конденсент.	+	+	+	+	+	+	+	+
FontExtend	C	Определяет стиль текста как расширенный.	+	+	+	+	+	+	+	+
FontItalic	C	Задают для текста стиль Italic (курсив).	+	+	+	+	+	+	+	+
FontName	C	Задаёт имя шрифта, используемого для отображения текста.	+	+	+	+	+	+	+	+
FontOutline	C	Определяет, обводится ли контуром текст, связанный с элементом управления.	+	+	+	+	+	+	+	+
FontShadow	C	Определяет, затеняется ли текст, связанный с элементом управления.	+	+	+	+	+	+	+	+
FontSize	C	Задаёт размер шрифта для текста, отображаемого вместе с объектом.	+	+	+	+	+	+	+	+
FontStrikethru	C	Задают для текста стиль Strikethru (перечёркнутый) .	+	+	+	+	+	+	+	+
FontUnderline	C	Задают для текста стиль Underline (подчёркнутый).	+	+	+	+	+	+	+	+
ForeColor	C	Задают основной цвет символов для недоступного элемента управления.	+		+	+	+	+	+	+
Format	C	Задаёт входное и выходное форматирование свойства Value элемента управления.			+			+	+	+
GotFocus	M	Имеет место, когда объект получает фокус - в результате действий со стороны пользователя или в результате выполнения кода.	+	+	+	+	+	+	+	+
Height	C	Задаёт ширину объекта на экране.	+	+	+	+	+	+	+	+
HelpContextID	C	Задаёт идентификатор контекста для темы в файле справки, который требуется для обеспечения контекстно-зависимой справки по данному объекту.	+	+	+	+	+	+	+	+
HideSelection	C	Определяет, остается ли текст выделенным, когда элемент управления перестаёт быть в фокусе.			+			+	+	+
Hours	C	Определяет выводить ли дату в формате 12 или 24 часов для значений типа DateTime.							+	
IMEMode	C	Определяет установки окна редактора ввода методов Input Method Editor (IME) для отдельного элемента управления.			+				+	+

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
IncrementalSearch	C	Определяет, поддерживает ли данный элемент управления пошаговый поиск при вводе с клавиатуры.		+	+			+		
IndexToItemID	M	Возвращает индекс элемента с указанным идентификатором.		+	+					
Init	M	Имеет место при создании объекта.	+	+	+	+	+	+	+	+
InputMask	C	Определяет, каким образом данные вводятся и отображаются в элементе управления.			+			+	+	
IntegralHeight	C	Определяет что высота элемента типа EditBox или ListBox автоматически выравнивается для вывода последнего элемента.		+					+	+
InteractiveChange	M	Имеет место, когда пользователь изменяет значение элемента управления с помощью клавиатуры или мыши.	+	+	+			+	+	+
ItemBackColor	C	Задают цвет фона, используемые в тексте элементов списков ComboBox и ListBox.		+	+					
ItemData	C	Задаёт ссылку по индексу на одномерный массив, длина которого совпадает с установкой свойства List элемента управления ComboBox или ListBox.		+	+					
ItemForeColor	C	Задают основной цвет символов, используемые в тексте элементов списков ComboBox и ListBox.		+	+					
ItemIDData	C	Создаёт ссылку по уникальному идентификатору на одномерный массив, длина которого совпадает с установкой свойства List элемента управления ComboBox или ListBox.		+	+					
ItemIDToIndex	M	Возвращает индекс позиции элемента в списке данного элемента управления.		+	+					
ItemTips	C	Определяет выводить ли подсказку tip для элементов, которые выводятся в combobox или listbox.		+	+					
KeyboardHighValue	C	Задают максимальное значения, которые можно ввести в элементе управления текстового окна Spinner с клавиатуры.						+		
KeyboardLowValue	C	Задают минимальное значения, которые можно ввести в элементе управления текстового окна Spinner с клавиатуры.						+		
KeyPress	M	Имеет место, когда пользователь нажимает и отпускает клавишу.	+	+	+	+	+	+	+	+
Left	C	Для элементов управления задает положение левого края объекта относительно его родительского объекта.	+	+	+	+	+	+	+	+
List	C	Массив символьных строк, используемый для доступа к элементам списка ComboBox или ListBox.		+	+					
ListCount	C	Содержит число элементов в списке ComboBox или ListBox.		+	+					
ListIndex	C	Задаёт номер индекса выбранного элемента списка ComboBox или ListBox.		+	+					
ListItem	C	Массив символьных строк, используемый для доступа к элементам списка ComboBox или ListBox по идентификаторам элементов.		+	+					
ListItemID	C	Задаёт уникальный идентификатор для выбранного		+	+					

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
		элемента списка в данном элементе управления.								
LostFocus	M	Имеет место, когда объект теряет фокус.	+	+	+	+	+	+	+	+
Margin	C	Задаёт ширину поля в текстовой части элемента управления.			+			+	+	+
MaxLength	C	Задаёт максимальную длину строки текста (в символах), которую можно ввести в элементе управления EditBox.							+	+
MemoWindow	C	Задаёт имя пользовательского окна, которое должно использоваться, если источником данных для элемента управления TextBox оказывается мемо-поле.							+	
Message	M	Отображает сообщение в строке состояния внизу экрана.	+	+	+	+	+	+	+	+
MiddleClick	M	Имеет место, когда нажата средняя клавиша мыши на трехкнопочной мыши.	+	+	+	+	+	+	+	+
MouseDown	M	Имеет место, когда пользователь нажимает кнопку мыши.	+	+	+	+	+	+	+	+
MouseIcon	C	Определяет вид указателя мыши, который выводится при помещении мыши над объектом.	+	+	+	+	+	+	+	+
MouseMove	M	Имеет место, когда пользователь перемещает указатель мыши по объекту.	+	+	+	+	+	+	+	+
MousePointer	C	Задаёт форму указателя мыши, которую он принимает, когда на этапе выполнения мышь находится над той или иной частью объекта.	+	+	+	+	+	+	+	+
MouseUp	M	Имеет место, когда пользователь отпускает кнопку мыши.	+	+	+	+	+	+	+	+
MouseWheel	M	Имеет место, когда пользователь вращает колесо мыши, если оно на ней есть.	+	+	+	+	+	+	+	+
Move	M	Перемещает объект.	+	+	+	+	+	+	+	+
MoverBars	C	Определяет, отображаются ли в элементе управления ListBox линейки прокрутки.		+						
MultiSelect	C	Определяет, может ли пользователь выделять несколько элементов списка ListBox и как это делается.		+						
Name	C	Задаёт имя объекта, на который имеется ссылка в коде.	+	+	+	+	+	+	+	+
NewIndex	C	Задаёт индекс последнего элемента, добавленного в элемент управления ComboBox или ListBox.		+	+					
NewItemID	C	Задаёт идентификатор самого последнего элемента, добавленного в элемент управления ComboBox или ListBox.		+	+					
NullDisplay	C	Определяет текст, который выводится если значение установлено в NULL.		+	+			+	+	+
NumberOfElements	C	Задаёт количество элементов в массиве, которые используются для заполнения списка в элементе управления ComboBox или ListBox.		+	+					
OpenWindow	C	Определяет, должно ли автоматически открываться окно, когда элемент управления TextBox, привязанный к мемо-полю, получает фокус.							+	
Parent	C	Определяет ссылку на объект-контейнер элемента	+	+	+	+	+	+	+	+

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
		управления.								
ParentClass	C	Возвращает имя класса, на котором базируется класс объекта.	+	+	+	+	+	+	+	+
PasswordCharacter	C	Определяет, что отображается в элементе управления текстового поля - символы, введенные пользователем, или метки-заполнители; определяет символ, используемый в качестве метки-заполнителя.							+	+
Picture	C	Задаёт растровый файл (.	+	+	+	+	+			
ProgrammaticChange	M	Имеет место, когда в коде изменяется значение элемента управления.	+	+	+			+	+	+
RangeHigh	M	Имеет место, когда элемент управления Spinner или TextBox теряет фокус и когда элемент управления ComboBox или ListBox получает фокус.		+	+			+	+	
RangeLow	M	Имеет место, когда элемент управления Spinner или TextBox теряет фокус или когда элемент управления ComboBox или ListBox получает фокус.		+	+			+	+	
ReadExpression	M	Возвращает выражение, введенное в качестве значения свойства в окне свойств.	+	+	+	+	+	+	+	+
ReadMethod	M	Возвращает текст заданного метода.	+	+	+	+	+	+	+	+
ReadOnly	C	Определяет, может ли пользователь редактировать элемент управления или разрешены ли обновления таблицы или представления, связанных с объектом Cursor.			+			+	+	+
Refresh	M	[[FormSet.	+	+	+	+	+	+	+	+
RemoveItem	M	Удаляет элемент из списка элемента управления ComboBox или ListBox.		+	+					
RemoveListItem	M	Удаляет элемент из списка элемента управления ComboBox или ListBox.		+	+					
Requery	M	Повторно опрашивает источник строк, к которому привязан элемент управления ListBox или ComboBox.		+	+					
ResetToDefault	M	Восстанавливает значения свойств, событий и методов в значение по умолчанию.	+	+	+	+	+	+	+	+
RightClick	M	Имеет место, когда пользователь нажимает и затем отпускает правую (вторичную) кнопку мыши и при этом указатель мыши находится над элементом управления.	+	+	+	+	+	+	+	+
RightToLeft	C	Определяет последовательность чтения справа на лево. Игнорируется в версиях не для Ближнего Востока.	+	+	+	+	+	+	+	+
RowSource	C	Задаёт источник значений для элемента управления ComboBox или ListBox.		+	+					
RowSourceType	C	Задаёт тип источника значений для элемента управления.		+	+					
SaveAsClass	M	Description]) Сохраняет экземпляр объекта в качестве определения класса в библиотеке классов.	+	+	+	+	+	+	+	+
ScrollBars	C	Задаёт тип полос прокрутки, имеющихся в элементе управления.								+
Seconds	C	Определяет, что секунды выводятся в поле типа DateTime для элемента textbox.							+	
Selected	C	Определяет, выбран ли элемент списка в элементе		+	+					

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
		управления ComboBox or ListBox control.								
SelectedBack Color	C	Задают цвет фона для выделенного текста.			+			+	+	+
SelectedFore Color	C	Задают основной цвет символов для выделенного текста.			+			+	+	+
SelectedID	C	Определяет, выбран ли элемент списка в элементе управления ComboBox или ListBox.		+	+					
SelectedItem BackColor	C	Задают цвет фона для выделенного элемента списка ComboBox или ListBox.		+	+					
SelectedItemForeColor	C	Задают основной цвет символов для выделенного элемента списка ComboBox или ListBox.		+	+					
SelectOnEntry	C	Определяет, будет ли выделена ячейка объекта Column, если пользователь щелкнет ее мышью или перейдет в нее с помощью клавиши TAB.			+			+	+	+
SelLength	C	Возвращает количество символов, которые пользователь выделил в области ввода текста данного элемента управления, или задает количество символов, которые следует выделить.			+			+	+	+
SelStart	C	Возвращает начальную точку фрагмента текста, выделенного пользователем в области ввода текста данного элемента управления, или указывает позицию курсора при отсутствии выделенного текста.			+			+	+	+
SelText	C	Возвращает текст, выделенный пользователем в области ввода текста данного элемента управления; если выделенных символов нет, возвращает пустую строку ("").			+			+	+	+
SetFocus	M	Устанавливает фокус в элементе управления.	+	+	+	+	+	+	+	+
ShowWhatsThis	M	Выводит подсказку типа Whats This Help определенную для объекта свойством WhatsThisHelpID.	+	+	+	+	+	+	+	+
Sorted	C	Определяет, должны ли элементы списка ComboBox или ListBox автоматически сортироваться в алфавитном порядке.		+	+					
SpecialEffect	C	Задаёт специфические опции формата для элемента управления.	+	+	+	+	+	+	+	+
SpinnerHigh Value	C	Задают максимальное, которое можно ввести в элементе управления Spinner, нажимая кнопки со стрелками.						+		
SpinnerLow Value	C	Задают минимальные значения, которое можно ввести в элементе управления Spinner, нажимая кнопки со стрелками.						+		
StatusBarText	C	Задаёт текст, который отображается в строке состояния, когда элемент управления получает фокус.	+	+	+	+	+	+	+	+
StrictDateEntry	C	Определяет, что дата может быть выведена в специфическом формате.							+	
Style	C	Задаёт стиль элемента управления.	+		+	+	+		+	
TabIndex	C	Задаёт порядок табуляции для элементов управления объекта Page и для объектов Form внутри FormSet.	+	+	+	+	+	+	+	+
TabStop	C	Определяет, можно ли с помощью клавиши TAB	+	+	+	+	+	+	+	+

Название		Назначение	CheckBox	ListBox	ComboBox	CommandButton	OptionButton	Spinne	TextBox	EditBox
		перемещать фокус в объект.								
Tag	C	Хранит всю вспомогательную информацию, необходимую для программы.	+	+	+	+	+	+	+	+
TerminateRe ad	C	Определяет, деактивируется ли объект Form или FormSet, если щелкнуть данный элемент управления.	+	+	+	+	+	+	+	+
Text	C	Содержит текст для элемента			+			+	+	+
ToolTipText	C	Задаёт текст, который появляется в виде подсказки для данного элемента управления.	+	+	+	+	+	+	+	+
Top	C	Для элементов управления задаёт расстояние между верхним краем элемента управления и верхним краем его родительского объекта.	+	+	+	+	+	+	+	+
TopIndex	C	Задаёт элемент, который должен стоять вверху списка.		+	+					
TopItemID	C	Задаёт идентификатор элемента, стоящего вверху списка..		+	+					
UIEnable	M	Имеет место для всех объектов, содержащихся внутри объекта Page, каждый раз, когда этот объект Page активизируется или деактивизируется.	+	+	+	+	+	+	+	+
UpClick	M	Имеет место, когда пользователь нажимает в элементе управления кнопку прокрутки со стрелкой вверх.			+			+		
Valid	M	Имеет место перед тем, как элемент управления теряет фокус.	+	+	+	+	+	+	+	+
Value	C	Задаёт текущее состояние элемента управления.	+	+	+		+	+	+	+
Visible	C	Определяет, является объект видимым или скрытым.	+	+	+	+	+	+	+	+
WhatsThisHelpID	C	Определяет идентификатор помощи Help topic context ID для Whats This Help	+	+	+	+	+	+	+	+
When	M	Имеет место перед тем, как элемент управления получает фокус.	+	+	+	+	+	+	+	+
Width	C	Содержит значения ширины для данного элемента управления..	+	+	+	+	+	+	+	+
WordWrap	C	Определяет, по вертикали или по горизонтали расширяется элемент управления Label, чтобы вместить текст, заданный его свойством Caption.				+				
WriteExpression	M	Записывает выражение в свойство.	+	+	+	+	+	+	+	+
ZOrder	M	Помещает заданный объект Form или элемент управления впереди или позади в z-порядке в пределах графического уровня объекта.	+	+	+	+	+	+	+	+

Visual FoxPro события для базовых классов

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
Activate	M	Имеет место, когда становится активным объект FormSet, Form или Page или когда отображается объект Toolbar.							+	+							+						+
ActivateCell	M	Активизирует ячейку в элементе управления сетка.									+												
ActiveColumn	C	Возвращает столбец, содержащий активную ячейку элемента управления сетка.									+												
ActiveControl	C	Указывает активный элемент управления данного объекта.		+	+				+								+						+
ActiveForm	C	Указывает активный объект Form в объекте FormSet или SCREEN.							+	+													
ActivePage	C	Возвращает номер активного объекта Page в составе объекта PageFrame.																+					
ActiveRow	C	Задаёт строку, содержащую активную ячейку элемента управления сетка.									+												
AddColumn	M	Добавляет объект Column в элемент управления сетка.									+												
AddObject	M	Добавляет объект в контейнер на этапе выполнения.	+	+			+	+	+	+	+					+	+	+					+
AfterCloseTables	M	Имеет место после освобождения таблиц или представлений, заданных в среде данных для формы, набора форм или отчета.						+															
AfterDock	M	Имеет место после того, как объект ToolBar фиксируется.																					+
AfterRowColChange	M	Имеет место, когда пользователь переходит в другую строку или другой столбец сетки, после того как новая ячейка получит фокус, и после того как в новой строке или новом столбце произойдет событие When для данного объекта.									+												
Alias	C	Задаёт псевдоним, используемый всеми таблицами и представлениями, связанными с объектом Cursor.				+																	
Alignment	C	Задаёт вид выравнивания текста, связанного с данным элементом управления.										+		+									
AllowAddNew	C	Определяет, можно ли добавлять записи в таблицу в grid.									+												

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
AllowHeaderSizing	C	Определяет, можно ли изменять размер заголовка столбца для grid во время работы.									+												
AllowRowSizing	C	Определяет, можно ли изменять высоту строки для записи в grid во время работы.									+												
AlwaysOnTop	C	Запрещает другим окнам заслонять окно объекта Form.							+														
AutoCenter	C	Определяет, следует ли автоматически центрировать объект Form в основном окне Visual FoxPro, когда он впервые появляется на экране.							+														
AutoCloseTables	C	Определяет, должны ли таблицы или представления, задаваемые средой данных, закрываться при освобождении формы, набора форм или отчета.						+															
AutoOpenTables	C	Определяет, должны ли таблицы или представления, связанные со средой данных набора форм, формы или отчета, загружаться автоматически.						+															
AutoRelease	C	Определяет, следует ли освобождать объект FormSet после того, как освобожден последний объект Form, входящий в FormSet.								+													
AutoSize	C	Определяет, нужно ли автоматически изменять размеры элемента управления, чтобы в нем можно было разместить его содержимое.	+											+		+							
BackColor	C	Задает цвет фона, используемые при отображении текста и графики в объекте.	+	+	+				+		+	+		+		+	+				+		+
BackStyle	C	Задает фон объекта - прозрачный или непрозрачный.	+	+	+								+	+		+	+				+		
BaseClass	C	Задает имя базового класса Visual FoxPro, на основе которого создан указанный объект.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
BeforeDock	M	Имеет место перед фиксацией объекта ToolBar.																					+
BeforeOpenTables	M	Имеет место непосредственно перед открытием таблиц и представлений, связанных со средой данных набора форм, формы или отчета.						+															

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
BeforeRowColChange	M	Имеет место перед тем, как пользователь изменяет активную строку или столбец, прежде чем новая ячейка получит фокус.									+												
BorderColor	C	Задаёт цвет обрамления объекта.	+	+	+								+		+	+		+			+		
BorderStyle	C	Задаёт стиль обрамления объекта.	+						+				+	+	+	+					+		
BorderWidth	C	Задаёт ширину обрамления элемента управления.		+	+										+			+			+		
Bound	C	Определяет, привязывается ли элемент управления в объекте Column к источнику объекта.																					
Box	M	Рисует прямоугольник в объекте Form.							+														
BufferModeOverride	C	Определяет переписывать ли свойство BufferMode для уровня form или formset.				+			+	+													
ButtonCount	C	Задаёт количество кнопок в элементе управления CommandGroup или OptionGroup.	+													+							
Buttons	C	- Или - Массив для доступа ко всем кнопкам группы.	+													+							
Caption	C	Задаёт текст, отображаемый в названии объекта.							+			+		+			+						+
ChildAlias	C	Задаёт псевдоним дочерней таблицы.																	+				
ChildOrder	C	Задаёт тег индекса для источника записи элемента управления Grid.									+								+				
Circle	M	Рисует окружность или эллипс в объекте Form.							+														
Class	C	Возвращает имя класса, на базе которого создан объект.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
ClassLibrary	C	Задаёт имя файла библиотеки пользовательских классов, которая содержит класс данного объекта.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Click	M	Пользователь нажал и отпустил левую (главную) кнопку мыши при курсоре, указывающем на элемент управления, или изменил значение какого-либо элемента управления, или щёлкнул пустую область формы.	+	+	+				+		+	+	+	+	+	+	+	+			+		+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
ClipControls	C	Определяет, должны ли методы графики, используемые для событий Paint, перерисовывать весь объект целиком или только вновь открывшиеся части.							+														
Closable	C	Определяет, можно ли закрыть объект Form, дважды щелкнув кнопку управляющего меню или выбрав в этом меню команду Close.							+														
CloseTables	M	Приводит к закрытию таблиц и представлений, связанных со средой данных.						+															
Cls	M	Удаляет графику и текст из объекта Form.							+														
ColorScheme	C	Задаёт цветовую схему, используемую в элементах управления.												+							+		
ColorSource	C	Определяет, как устанавливаются цвета элемента управления.	+	+	+				+				+	+	+	+	+	+			+		+
ColumnCount	C	Задаёт количество объектов Column в элементе управления Grid, ComboBox или ListBox.									+												
ColumnOrder	C	Задаёт относительный порядок объектов Column в элементе управления Grid.																					
Columns	C	Массив для доступа к отдельным объектам Column в элементе управления Grid по номерам объектов Column.									+												
Comment	C	Сохраняет информацию об объекте.	+	+	+		+		+	+	+		+	+	+	+	+	+			+	+	+
ControlBox	C	Определяет, будет ли кнопка управляющего меню появляться в левом верхнем углу объекта Form или ToolBar на этапе выполнения.							+														+
ControlCount	C	Задаёт количество элементов управления в объекте-контейнере.		+	+		+		+								+						+
Controls	C	Массив для доступа к элементам управления в объекте-контейнере.		+	+		+		+								+						+
ControlSource	C	Задаёт источник данных, к которому привязывается объект.	+													+							
CurrentControl	C	Указывает элемент управления в объекте Column, который используется для отображения значений активной ячейки.																					

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
CurrentX	C	Задаёт горизонтальную (X) координату для следующего метода рисования.							+														
CurrentY	C	Задаёт вертикальную (Y) координату для следующего метода рисования.							+														
CursorSource	C	Задаёт имя таблицы или представления, связанного с объектом Cursor.				+																	
Curvature	C	Задаёт кривизну углов элемента управления Shape.																			+		
Database	C	Задаёт путь к базе данных, которая содержит таблицу или представление, связанное с объектом Cursor.				+																	
DataSession	C	Определяет, может ли объект Form, FormSet или Toolbar выполняться в собственном сеансе и иметь отдельную среду данных.							+	+													+
DataSessionID	C	Возвращает идентификатор сеанса, соответствующий локальному сеансу для объекта FormSet, Form или Toolbar.							+	+													+
DblClick	M	Имеет место, когда пользователь быстро дважды нажимает и тут же отпускает левую (первую) кнопку мыши.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
Deactivate	M	Имеет место, когда объект-контейнер (например форма) перестаёт быть активным, потому что ни один из содержащихся в нём объектов не находится в фокусе.							+	+							+						+
DefOLELCID	C	Определяет значение по умолчанию для OLE Locale ID в форме или главном окне Visual FoxPro.							+														
DeleteColumn	M	Удаляет объект Column из элемента управления Grid.									+												
Deleted	M	Имеет место, когда пользователь удаляет пометку записи на удаление или когда выдается команда DELETE.									+												
DeleteMark	C	Определяет, будет ли в элементе управления Grid отображаться столбец маркеров удаления.									+												
Desktop	C	Определяет, содержится ли объект Form в главном окне Visual FoxPro.							+														
Destroy	M	Имеет место, когда освобождается объект.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+		+	+	+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
DisabledBackColor	C	Задает цвет фона для недоступного элемента управления.												+									
DisabledForeColor	C	Задает основной цвет символов для недоступного элемента управления.												+									
Dock	M	Паркует объект ToolBar у границы основного окна Visual FoxPro.																					+
Docked	C	Содержит логическое значение, указывающее, запаркован ли пользовательский объект ToolBar.																					+
DockPosition	C	Задает позицию, в которой паркуется пользовательский объект ToolBar.																					+
DoScroll	M	Прокручивает элемент управления Grid, моделируя работу пользователя с полосами прокрутки.									+												
Drag	M	Начинает, заканчивает или отменяет операцию перетаскивания с помощью мыши.	+	+	+						+		+	+	+	+	+	+			+		
DragDrop	M	Имеет место при завершении операции перетаскивания с помощью мыши.	+	+	+				+		+		+	+	+	+	+	+			+		+
DragIcon	C	Задает значок, отображаемый в качестве курсора во время операции перетаскивания с помощью мыши.	+	+	+						+		+	+	+	+	+	+			+		
DragMode	C	Задает режим для операции перетаскивания с помощью мыши - явный или автоматический.	+	+	+						+		+	+	+	+	+	+			+		
DragOver	M	Имеет место, когда элемент управления, перетаскиваемый с помощью мыши, накрывает объект назначения.	+	+	+				+		+		+	+	+	+	+	+			+		+
Draw	M	Рисует объект Form.		+	+				+														
DrawMode	C	Совместно с другими цветовыми свойствами определяет, в каком виде элемент управления Shape или Line отображается на экране.							+						+						+		
DrawStyle	C	Задает стиль линии, который следует использовать при рисовании с помощью методов графики.							+														
DrawWidth	C	Задает ширину линии в выводе методов графики.							+														

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
DynamicAlign	C	Определяет выравнивание текста и элементов управления в объекте Column.																					
DynamicBackColor	C	Задает цвет фона , используемые при отображении текста и графики в объекте Column.																					
DynamicCurrentControl	C	Определяет какой объект содержится в объекте Column для вывода значения активной ячейки.																					
DynamicFontBold	C	Задает для текста стиль Bold (полужирный)																					
DynamicFontItalic	C	Задает для текста стиль Italic (курсив).																					
DynamicFontName	C	Задает имя шрифта, используемого для отображения текста в объекте Column.																					
DynamicFontOutline	C	Определяет, обводится ли контуром текст, связанный с элементом управления.																					
DynamicFontShadow	C	Определяет, затеняется ли текст, связанный с элементом управления.																					
DynamicFontSize	C	Задает размер шрифта для текста, отображаемого в объекте Column.																					
DynamicFontStrikeThru	C	Задает для текста стиль Strikethru (перечеркнутый) .																					
DynamicFontUnderline	C	Задает для текста стиль Underline (подчеркнутый).																					
DynamicForeColor	C	Задает основной цвет символов, используемые при отображении текста и графики в объекте Column.																					
DynamicInputMask	C	Определяет, каким образом данные вводятся и отображаются в элементе управления.																					
Enabled	C	Определяет, может ли объект реагировать на события, генерируемые пользователем.	+	+	+				+		+		+	+	+	+	+	+		+	+	+	+
Error	M	Имеет место при возникновении в методе ошибки этапа выполнения.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+		+	+	+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
ErrorMessage	M	Имеет место, когда событие Valid возвращает значение "ложь" (.).	+													+							
Exclusive	C	Определяет, открыта ли в монопольном режиме таблица, связанная с объектом Cursor.				+																	
FillColor	C	Задает цвет, который используется для окраски фигур, нарисованных в объекте с помощью программ графики.						+												+			
FillStyle	C	Задает узор, используемый для окраски объектов Shape и фигур, созданных с помощью методов графики Circle и Box.						+												+			
Filter	C	Исключает из рассмотрения записи, которые не удовлетворяют критерию, определенному данным выражением.				+																	
FontBold	C	Задают для текста стиль Bold (полужирный).							+		+	+		+			+						
FontCondense	C	Определяет стиль текста как конденсент.							+		+	+		+			+						
FontExtend	C	Определяет стиль текста как расширенный.							+		+	+		+			+						
FontItalic	C	Задают для текста стиль Italic (курсив).							+		+	+		+			+						
FontName	C	Задает имя шрифта, используемого для отображения текста.							+		+	+		+			+						
FontOutline	C	Определяет, обводится ли контуром текст, связанный с элементом управления.							+		+	+		+			+						
FontShadow	C	Определяет, затеняется ли текст, связанный с элементом управления.							+		+	+		+			+						
FontSize	C	Задает размер шрифта для текста, отображаемого вместе с объектом.							+		+	+		+			+						
FontStrikethru	C	Задают для текста стиль Strikethru (перечеркнутый) .							+		+	+		+			+						
FontUnderline	C	Задают для текста стиль Underline (подчеркнутый).							+		+	+		+			+						
ForeColor	C	Задает основной цвет символов, используемые при отображении текста и графики в объекте.		+	+				+		+	+		+			+						+
Format	C	Задает входное и выходное форматирование свойства Value элемента управления.																					

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
FormCount	C	Содержит количество объектов Form в составе FormSet.								+													
Forms	C	Массив для доступа к отдельным объектам Form в составе FormSet.								+													
GotFocus	M	Имеет место, когда объект получает фокус - в результате действий со стороны пользователя или в результате выполнения кода.		+	+				+														
GridLineColor	C	Задаёт цвет линий, разделяющих ячейки в элементе управления Grid.									+												
GridLines	C	Определяет, отображаются ли в элементе управления Grid горизонтальные и вертикальные линии.									+												
GridLineWidth	C	Задаёт в пикселях толщину линий, разделяющих ячейки в элементе управления Grid.									+												
HalfHeightCaption	C	Определяет, какая высота у названия объекта Form - половинная или нормальная.							+														
HeaderHeight	C	Задаёт высоту заголовков столбцов в элементе управления сетка.									+												
Height	C	Задаёт ширину объекта на экране.	+	+	+		+		+		+		+	+	+	+		+			+	+	+
HelpContextID	C	Задаёт идентификатор контекста для темы в файле справки, который требуется для обеспечения контекстно-зависимой справки по данному объекту.	+	+	+		+		+		+		+	+	+	+	+	+			+		+
Hide	M	Скрывает объект Form, FormSet или ToolBar, устанавливая для его свойства Visible значение "ложь".							+	+													+
Highlight	C	Определяет, выделяется ли ячейка элемента управления Grid, получившая фокус.									+												
HighlightRow	C	Определяет, выделяется ли текущая строка и ячейка в Grid.									+												
Icon	C	Задаёт значок, который отображается на этапе выполнения для объекта Form при его свертывании.							+														
Init	M	Имеет место при создании объекта.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+		+	+	+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
InitialSelectedAlias	C	Определяет текущий алиас для объекта Cursor при загрузке объекта data environment .						+															
InputMask	C	Определяет, каким образом данные вводятся и отображаются в элементе управления.																					
InteractiveChange	M	Имеет место, когда пользователь изменяет значение элемента управления с помощью клавиатуры или мыши.	+													+							
Interval	C	Задаёт интервал времени в миллисекундах между вызовами события Timer элемента управления Timer.																				+	
KeyPress	M	Имеет место, когда пользователь нажимает и отпускает клавишу.							+														
KeyPreview	C	Определяет, перехватываются ли события KeyPress элементов управления событием KeyPress содержащего их объекта Form.							+														+
Left	C	Для элементов управления задаёт положение левого края объекта относительно его родительского объекта.	+	+	+		+		+		+		+	+	+	+		+			+	+	+
LeftColumn	C	Содержит номер самого левого столбца, отображаемого в элементе управления Grid.									+												
Line	M	Рисует линию в объекте Form.							+														
LineSlant	C	Задаёт направление наклона линии - из левого верхнего угла в правый нижний или из левого нижнего в правый верхний.													+								
LinkMaster	C	Задаёт родительскую таблицу, которая связана с дочерней таблицей, отображенной в элементе управления Grid.									+												
Load	M	Имеет место непосредственно перед созданием объекта.							+	+													
LockScreen	C	Определяет, как изменяются установки свойств объекта Form и содержащихся в нем объектов - в пакетном режиме или индивидуально.							+														+
LostFocus	M	Имеет место, когда объект теряет фокус.		+	+				+														
MacDesktop	C	Определяет, располагается ли форма на главном окне Visual FoxPro.							+														
MaxButton	C	Определяет, есть ли в объекте Form кнопка Maximize.							+														

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
MaxHeight	C	Определяет максимальную высоту, которую можно установить для объекта Form.							+														
MaxLeft	C	Задаёт максимально допустимое расстояние от развернутого объекта Form до левого края основного окна FoxPro.							+														
MaxTop	C	Задаёт максимально допустимое расстояние развернутого объекта Form до верхнего края основного окна FoxPro.							+														
MaxWidth	C	Определяет максимальную ширину, которую можно установить для объекта Form.							+														
MDIForm	C	Указывает, совместим ли объект Form со стандартом MDI (Multiple Document Interface).							+														
Message	M	Отображает сообщение в строке состояния внизу экрана.	+													+							
MiddleClick	M	Имеет место, когда нажата средняя клавиша мыши на трехкнопочной мыши.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
MinButton	C	Определяет, есть ли в объекте Form кнопка Minimize.							+														
MinHeight	C	Задаёт минимальную высоту, которую можно установить для объекта Form.							+														
MinWidth	C	Задаёт минимальную ширину, которую можно установить для объекта Form.							+														
MouseDown	M	Имеет место, когда пользователь нажимает кнопку мыши.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
MouseIcon	C	Определяет вид указателя мыши, который выводится при помещении мыши над объектом.	+	+	+				+		+		+	+	+	+	+	+			+		+
MouseMove	M	Имеет место, когда пользователь перемещает указатель мыши по объекту.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
MousePointer	C	Задаёт форму указателя мыши, которую он принимает, когда на этапе выполнения мышь находится над той или иной частью объекта.	+	+	+				+		+		+	+	+	+	+	+			+		+
MouseUp	M	Имеет место, когда пользователь отпускает кнопку мыши.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
MouseWheel	M	Имеет место, когда пользователь вращает колесо мыши, если оно на ней есть.	+	+	+				+		+	+	+	+	+	+	+	+			+		+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
Movable	C	Определяет, может ли пользователь перемещать данный объект на этапе выполнения.							+														+
Moved	M	Имеет место, когда объект перемещается на новую позицию или когда установки свойства Top или Left объекта-контейнера изменяются программным способом.	+	+	+				+		+		+	+	+	+		+			+		+
Name	C	Задаёт имя объекта, на который имеется ссылка в коде.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
NoDataOnLoad	C	Вызывает активизацию связанного с объектом Cursor представления, не загружая в него данные.				+																	
Objects	C	Массив для доступа к объекту Application.		+	+		+		+	+							+						+
OneToMany	C	При прохождении по записям родительской таблицы задаёт, должен ли указатель записи оставаться на одной и той же родительской записи, пока указатель записи в дочерней таблице не пройдет по всем связанным записям.																	+				
OpenTables	M	Открывает программным путем таблицы и представления, связанные со средой данных.						+															
OpenViews	C	Определяет тип view, связанного с объектом data environment для formset, form или report при автоматическом открытии.						+															
Order	C	Задаёт управляющий тег индекса для объекта Cursor.				+																	
PageCount	C	Задаёт число страниц, содержащихся в объекте PageFrame.																+					
PageHeight	C	Задаёт высоту страницы, которая может содержать элементы управления.																+					
PageOrder	C	Задаёт относительный порядок страниц в объекте PageFrame.															+						
Pages	C	Массив, используемый для доступа к страницам в объекте PageFrame.																+					
PageWidth	C	Задаёт ширину страницы.																+					
Paint	M	Имеет место, когда перерисовывается форма или панель инструментов.							+														+
Panel	C	Задаёт активную панель в элементе управления Grid.									+												

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
PanelLink	C	Определяет, будут ли связаны левая и правая панели элемента управления Grid после его разбиения.									+												
Parent	C	Определяет ссылку на объект-контейнер элемента управления.	+	+	+	+	+	+	+		+	+	+	+	+	+	+	+	+	+	+	+	+
ParentAlias	C	Задаёт псевдоним родительской таблицы.																	+				
ParentClass	C	Возвращает имя класса, на котором базируется класс объекта.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Partition	C	Определяет, разбивается ли элемент управления Grid на две панели и задаёт линию разбиения относительно левого края Grid.									+												
Picture	C	Задаёт растровый файл.		+	+		+		+				+				+						
Point	M	Возвращает цвет в модели RGB (красный-зеленый-синий) для заданной точки формы.							+														
Print	M	Выдаёт символьную строку в объекте Form.							+														
Programmatic Change	M	Имеет место, когда в коде изменяется значение элемента управления.	+													+							
PSet	M	Устанавливает для точки объекта Form или основного окна Visual FoxPro основной цвет символов.							+														
QueryUnload	M	Имеет место перед выгрузкой объекта Form.							+														
ReadActivate	M	Имеет место, когда становится активным новый объект Form в составе объекта FormSet.								+													
ReadCycle	C	Определяет, переходит ли фокус из последнего объекта в составе FormSet в первый объект этого контейнера.								+													
ReadDeactivate	M	Имеет место, когда деактивируется новый объект Form в составе объекта FormSet.								+													
ReadExpression	M	Возвращает выражение, введенное в качестве значения свойства в окне свойств.	+	+	+		+		+	+	+	+	+	+	+	+	+	+			+	+	+
ReadLock	C	Определяет, должны ли блокироваться все записи, на которые имеются ссылки в каких-либо объектах Form в составе FormSet.								+													
ReadMethod	M	Возвращает текст заданного метода.	+	+	+		+		+	+	+	+	+	+	+	+	+	+			+	+	+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
ReadMouse	C	Определяет, можно ли в объектах Form, принадлежащих данному объекту FormSet, переходить от одного элемента управления к другому с помощью мыши.								+													
ReadOnly	C	Определяет, может ли пользователь редактировать элемент управления или разрешены ли обновления таблицы или представления, связанных с объектом Cursor.				+					+												
ReadSave	C	Определяет, можно ли с помощью команды READ вновь активизировать объект.								+													
ReadShow	M	Событие ReadShow имеет место, когда выдается команда SHOW GETS в активном объекте FormSet и когда активизируется объект FormSet.								+													
ReadTimeout	C	Определяет, как долго остается активным объект FormSet, если пользователь ничего не вводит.								+													
ReadValid	M	Имеет место сразу после деактивизации объекта FormSet.								+													
ReadWhen	M	Имеет место после загрузки объекта FormSet.								+													
RecordMark	C	Определяет, отображается ли в элементе управления Grid столбец отметки записей.									+												
RecordSource	C	Задаёт источник данных, к которому привязан элемент управления Grid.									+												
RecordSourceType	C	Определяет, как открывается таблица, заполняющая элемент управления Grid.									+												
Refresh	M	Перерисовывает объект Form или элемент управления и обновляет все значения.	+	+	+				+	+	+	+				+	+	+					+
RelationalExpr	C	Задаёт выражение на основе полей родительской таблицы, связанное с индексом в дочерней таблице и объединяющее эти две таблицы.									+								+				
RelativeColumn	C	Задаёт активный столбец в видимой части элемента управления Grid.									+												
RelativeRow	C	Задаёт активную строку в видимой части элемента управления Grid.									+												

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
Release	M	Освобождает объект FormSet или Form из памяти.							+	+													
ReleaseType	C	Возвращает целое число, которое определяет, как освобождается объект Form.							+														
RemoveObject	M	Удаляет заданный объект из объекта-контейнера на этапе выполнения.	+	+			+	+	+	+	+					+	+	+					+
ResetToDefault	M	Восстанавливает значения свойств, событий и методов в значение по умолчанию.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Resizable	C	Определяет, может ли пользователь на этапе выполнения изменять размеры объекта Column.																					
Resize	M	Имеет место при изменении размеров объекта.		+	+				+		+							+					+
RightClick	M	Имеет место, когда пользователь нажимает и затем отпускает правую (вторичную) кнопку мыши и при этом указатель мыши находится над элементом управления.	+	+	+				+		+	+	+	+	+	+	+	+			+		+
RightToLeft	C	Определяет последовательность чтения справа на лево. Игнорируется в версиях не для Ближнего Востока.							+		+			+				+					
RowHeight	C	Задаёт высоту строк в элементе управления Grid.									+												
SaveAsClass	M	Сохраняет экземпляр объекта в качестве определения класса в библиотеке классов.	+	+	+		+		+	+	+		+	+	+	+		+		+	+	+	+
ScaleMode	C	Задаёт единицу измерения в системе координат объекта при использовании методов графики или при позиционировании элементов управления.							+														+
ScrollBars	C	Задаёт тип полос прокрутки, имеющихся в элементе управления.									+												
Scrolled	M	Имеет место в элементе управления Grid при щелчке горизонтальной или вертикальной полосы прокрутки или при перемещении бегунка.									+												
SelectOnEntry	C	Определяет, будет ли выделена ячейка объекта Column, если пользователь щелкнет ее мышью или перейдет в нее с помощью клавиши TAB.																					

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
SetAll	M	Присваивает установку некоторого свойства всем элементам управления данного объекта Container или определенному классу таких элементов управления.	+	+					+	+	+					+	+	+					+
SetFocus	M	Устанавливает фокус в элементе управления.		+	+						+						+						
ShowTips	C	Определяет, следует ли показывать подсказки для элементов управления заданного объекта Form или ToolBar.							+	+													+
ShowWhatsThis	M	Выводит подсказку типа Whats This Help определенную для объекта свойством WhatsThisHelpID.	+	+	+		+		+		+		+	+	+	+	+	+			+		+
ShowWindow	C	Определяет является ли форма или панель управления формой top-level или дочерней формой.							+														+
Sizable	C	Определяет, можно ли изменять размеры объекта.																					+
SizeBox	C	Определяет, имеет ли форма элемент в верхнем левом углу для изменения размеров формы (size box).							+														
Sparse	C	Определяет, затрагивает свойство CurrentControl все ячейки объекта Column или только активную ячейку.																					
SpecialEffect	C	Задаёт специфические опции формата для элемента управления.	+	+	+											+		+			+		
SplitBar	C	Определяет, имеет ли объект Grid разделитель на две части (split bar).									+												
StatusBarText	C	Задаёт текст, который отображается в строке состояния, когда элемент управления получает фокус.									+												
Stretch	C	Определяет, как изображение изменяется в размерах, чтобы поместиться внутри элемента управления Image.											+										
TabIndex	C	Задаёт порядок табуляции для элементов управления объекта Page и для объектов Form внутри FormSet.	+	+	+				+		+			+		+		+					
TabStop	C	Определяет, можно ли с помощью клавиши TAB перемещать фокус в объект.		+	+				+		+							+					
TabStretch	C	Определяет порядок действий в том случае, когда вкладки не умещаются в элементе управления PageFrame.																+					

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
TabStyle	C	Определяет, являются ли закладки выровненными или нет. (justified или non-justified).																+					
Tag	C	Хранит всю вспомогательную информацию, необходимую для программы.	+						+		+		+	+	+	+					+	+	
TerminateRead	C	Определяет, деактивизируется ли объект Form или FormSet, если щелкнуть данный элемент управления.	+													+							
TextHeight	M	Возвращает высоту текстовой строки, предполагая, что она отображается текущим шрифтом.							+														
TextWidth	M	Возвращает ширину текстовой строки, предполагая, что она отображается текущим шрифтом.							+														
Timer	M	Имеет место, когда истекает интервал времени, заданный свойством Interval.																				+	
ToolTipText	C	Задаёт текст, который появляется в виде подсказки для данного элемента управления.									+						+				+		
Top	C	Для элементов управления задает расстояние между верхним краем элемента управления и верхним краем его родительского объекта.	+	+	+		+		+		+		+	+	+	+		+			+	+	+
UIEnable	M	Имеет место для всех объектов, содержащихся внутри объекта Page, каждый раз, когда этот объект Page активизируется или деактивизируется.	+	+	+						+		+	+	+	+		+			+		
UnDock	M	Имеет место, когда объект ToolBar сдвигается мышью с позиции, в которой он был фиксирован.																					+
Unload	M	Имеет место при освобождении объекта.							+	+													
Valid	M	Имеет место перед тем, как элемент управления теряет фокус.	+								+					+							
Value	C	Задаёт текущее состояние элемента управления.	+								+					+							
View	C	Задаёт тип просмотра для элемента управления Grid.									+												
Visible	C	Определяет, является объект видимым или скрытым.	+	+	+				+	+	+		+	+	+	+		+			+		+

Название		Назначение	Column	CommandGroup	Container	Control	Cursor	Custom	DataEnvironment	Form	FormSet	Grid	Header	Image	Label	Line	OptionGroup	Page	PageFrame	Relation	Shape	Timer	Toolbar
WhatsThisButton	C	Определяет, имеет ли форма клавишу "What's This" (вопросительный знак) в своем заголовке.							+														
WhatsThisHelpID	C	Определяет идентификатор помощи Help topic context ID для Whats This Help	+	+	+		+		+		+		+	+	+	+	+	+			+		+
WhatsThisMode	M	Выводит указатель в виде символа вопроса " Whats This Help" и дает возможность работать в режиме Whats This Help.							+														
When	M	Имеет место перед тем, как элемент управления получает фокус.	+								+					+							
Width	C	Задаёт ширину объекта.	+	+	+		+		+		+		+	+	+	+		+			+	+	+
WindowList	C	Задаёт список объектов Form, которые могут принять участие в работе команды READ текущего объекта Form.								+													
WindowState	C	Определяет, свернуто или развернуто окно объекта Form на этапе выполнения.							+														
WindowType	C	Определяет, как объект FormSet или Form ведет себя, когда он показывается или выполняется по команде DO FORM.							+	+													
WordWrap	C	Определяет, по вертикали или по горизонтали расширяется элемент управления Label, чтобы вместить текст, заданный его свойством Caption.												+									
WriteExpression	M	Записывает выражение в свойство.	+	+	+		+		+	+	+	+	+	+	+	+	+	+			+	+	+
ZoomBox	C	Определяет, имеет ли форма клавишу zoom (сворачивание/разворачивание).							+														
ZOrder	M	Помещает заданный объект Form или элемент управления впереди или позади в z-порядке в пределах графического уровня объекта. Помещает элемент управления, содержащийся в объекте ToolBar, впереди или позади в массиве элементов управления, определяющем порядок, в котором элементы управления появляются на ToolBar.	+	+	+				+		+		+	+	+	+	+	+		+	+		+

List & ComboBox свойства, события, методы

Свойства, методы и события <u>List Boxes</u>	Описание
AddItem	Добавляет новый элемент списка в элемент управления поле ввода с раскрывающимся списком или поле ввода со списком, предоставляя при этом возможность задать индекс элемента. RowSource должен быть 0-None или 1-Value (используется индекс элемента).
AddListItem	Добавляет новый элемент списка в элемент управления поле ввода с раскрывающимся списком или поле ввода со списком, давая возможность задать идентификатор элемента. RowSource должен быть 0-None или 1-Value (uses the item's ID).
BoundColumn	Определяет, какой столбец многостолбцового элемента управления привязан к свойству Value этого элемента управления.
BoundTo	Определяет, связан ли список с цифровыми данными или нет.
Click Event	Имеет место в следующих случаях: кнопку мыши при курсоре, указывающем на элемент управления, или изменил значение какого-либо элемента управления.
ColumnCount	Задаёт количество столбцов.
ColumnWidths	Задаёт ширину столбцов.
ControlSource	Задаёт источник данных, к которому привязывается объект.
DbfClick Event	Имеет место, когда пользователь быстро дважды нажимает и тут же отпускает левую (первую) кнопку мыши.
DisplayValue	Задаёт содержимое первого столбца выбранного элемента списка.
KeyPress Event	Имеет место, когда пользователь нажимает и отпускает клавишу.
IncrementalSearch	Определяет, поддерживает ли данный элемент управления последовательный поиск при вводе с клавиатуры.
List	Массив символьных строк, используемый для доступа.
ListCount	Содержит число элементов в списке.
ListIndex	Задаёт номер индекса выбранного элемента списка.
ListItem	Массив символьных строк, используемый для доступа к элементам списка по item's ID.
ListItemID	Задаёт уникальный идентификатор ID для выбранного элемента списка в данном элементе управления.
NewIndex	Задаёт индекс последнего элемента, добавленного в элемент управления.
NewItemID	Задаёт идентификатор самого последнего элемента.
MultiSelect	Определяет, может ли пользователь выделять несколько элементов списка.
Requery	Повторно опрашивает источник строк, к которому привязан элемент управления.
RemoveItem	Удаляет элемент из списка элемента управления.
RemoveListItem	Удаляет элемент из списка элемента управления по ID.

Свойства, методы и события <u>List Boxes</u>	Описание
RowSource	Задаёт источник значений.
Selected	Определяет, выбран ли элемент списка.
SelectedID	Определяет, выбран ли элемент списка (uses the item's ID).
Sorted	Определяет, должны ли элементы списка автоматически сортироваться в алфавитном порядке.

ПРИЛОЖЕНИЕ . Базовые классы

Имя класса	Является контейнером?
Check Box	
Column	Да
Combo Box	
Command Button	
Command Group	Да
Container	Да
Control	
Custom	Может быть
Edit Box	
Form	Да
FormSet	Да
Grid	Да
Header	
Image	
Label	
Line	
List Box	
OleBoundControl	
OleContainerControl	
OptionGroup	Да
Page	Да
PageFrame	Да
Separator	
Shape	
Spinner	
Text Box	
Timer	
ToolBar	Да

ПРИЛОЖЕНИЕ. Значения первого байта файла DBF

Значение (Dec)	Тип используемого
2	FoxBASE
3	FoxPro, FoxBASE+, dBASE III PLUS, dBASE IV (без мемо)
48	Visual FoxPro
67	dBASE IV SQL файлы (без мемо)
99	dBASE IV SQL system file (без мемо)
131	FoxBASE+, dBASE III PLUS (с мемо)
139	dBASE IV (с мемо)
203	dBASE IV ASQL table file (с мемо)
245	FoxPro 2.x (или более ранних версий) (с мемо)
251	FoxBASE

ПРИЛОЖЕНИЕ. Visual FoxPro правила именования и сокращения

Ключевое слово	Ссылки
Parent	Определяет ссылку на объект-контейнер элемента управления.
This	Ссылка на текущий объект для события, процедуры, свойства.
ThisForm	Определяет ссылку на форму, которая содержит текущий объект.
ThisFormSet	Определяет ссылку на набор форм, который содержит текущий объект.

ПРИЛОЖЕНИЕ. Visual FoxPro иерархия событий**0) Событие BeforeOpenTable для Data Environment****1) Событие Load Formy**

Событие Load наступает самым первым перед всеми остальными событиями формы.

2) Инициализация Data Environment

Инициализация объекта "Data Environment" осуществляется после загрузки формы, но перед ее инициализацией и перед инициализацией любого из объектов управления формы. Это необходимо для того, чтобы была возможность связывания данных из таблиц с элементами управления в форме.

2) Инициализация элементов управления формы

Перед инициализацией формы инициализируются все элементы управления, которые в ней содержатся. Если один из элементов в процессе своей инициализации вернул значение FALSE (.F.) из своего события **INIT()**, оставшиеся элементы управления и сама форма не инициализируются, а вызывается событие "Destroy".

3) Инициализация формы

Наступает инициализация формы. В ходе этого события можно использовать данные из таблиц. Можно устанавливать указатель записи таблиц, открытых в "Data Environment" или "Load Event" формы. Если событие **INIT()** вернуло значение FALSE (.F.) форма не активизируется и вызывается событие "Destroy".

5) События WHEN() и "Focus для элементов управления формы"

После завершения активизации формы выполняется событие **WHEN()** для первого элемента управления в форме. Если оно удовлетворяется, т.е. в нем нет кода или оно вернуло значение TRUE (.T.), форма передает фокус этому элементу. Если событие **WHEN()** для первого элемента не выполнилось, т.е. код возврата FALSE (.F.), Visual FoxPro проверяет событие **WHEN()** для следующего элемента управления. Если ни один из элементов не получил фокус, сама форма не может получить фокус.

6) Другие события при выполнении формы

После того как элемент управления получил фокус, могут наступить другие события, например, "KeyPress" или события обработки мыши..

Событие **VALID()** вызывается перед событием **LostFocus()**. Только когда **VALID()** вернет .T. или есть код, который позволит пользователю покинуть поле, наступает событие **LostFocus()**.

Клик мыши вызывает события **MouseDown()** и **MouseUp()** перед самым событием **Click()**. И затем только после этих трех событий наступает событие **Valid()** (Если клик пришелся вне поля, на котором только что было управление).

7) Заккрытие формы

Когда нажата клавиша для закрытия формы, наступает последовательность событий, подобная той, что была при открытии формы :

QueryUnload() событие.

Form's Destroy() событие.

Destroy() событие для каждого элемента.

Form's событие Unload(), если все события Destroy() завершились успешно.

Наконец, DataEnvironment закрывает все таблицы и файлы.

AfterCloseTables() событие, перед событием

Destroy() для DataEnvironment

ПРИЛОЖЕНИЕ. Часто используемые значения RGB и Color

Цвет	Значение RGB	Значение Color
Black	0,0,0	0
Blue	0,0,255	16711680
Cyan	0,255,255	16776960
Green	0,255,0	65280
Grey	192,192,192	12632256
Magenta	255,0,255	16711935
Red	255,0,0	255
White	255,255,255	16777215
Yellow	255,255,0	65535
Dark Blue	0,0,128	8388608
Dark Cyan	0,128,128	8421376
Dark Green	0,128,0	32768
Dark Grey	128,128,128	8421504
Dark Magenta	128,0,128	8388736
Dark Red	128,0,0	128
Dark Yellow	128,128,0	32896

●●●● Ну, вот и конец ☺ ●●●●