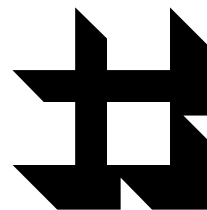


ГЛАВА 13



Отчеты

Введение в систему отчетов

Отчет — это форматированное представление данных, выводимое на экран, в файл или на принтер.

Отчеты — такая же неотъемлемая часть работы с СУБД, как базы данных и формы для ввода данных. Разработчики предоставили программистам средства для эффективной разработки компонентов приложения, отвечающих за форматированное визуальное представление данных. Конечно, можно вывести данные для визуального просмотра простой командой `List`, но подавляющее большинство программистов все же используют мощную систему подготовки отчетов.

В девятой версии Visual FoxPro система отчетов подверглась серьезным изменениям.

Новшеств довольно много. В ней появились многополосные отчеты, защита объектов в Конструкторе отчетов, точное позиционирование объектов, улучшенные меню и т. д. Самое главное улучшение — отчет стал объектом. Новый класс `ReportListener` обеспечивает доступ к событиям генерации и визуализации отчетов во время исполнения программы. Поддерживается вывод отчетов в разных форматах, в том числе в XML, HTML, а также в виде изображений. Мощные средства перехвата событий в макете приложения оснащены настраиваемыми конструкторами (рис. 13.1). Ко всем плюсам усовершенствованной системы отчетов добавляется еще один — обратная совместимость с уже существующими отчетами предыдущих версий пакета Visual FoxPro.

Отчеты могут быть созданы табличной и свободной формы (иногда их еще называют многоколоночным и многостраничным). Табличный отчет представляет собой напечатанную таблицу, каждая запись которой является строкой отчета. Если вы желаете распечатать этикетки, карточки, чеки и другие документы, в которых каждый реквизит должен находиться в строго определенном месте документа, то нужно создавать отчет в свободной форме.

Если вы пока не очень опытны в создании отчетов, набросайте отчет на бумаге и покажите своему заказчику. Возможно, вы сэкономите себе немало сил и времени, если согласуете с ним отчеты заранее.

Отчет подразделяется на области. Области отчета дают вам возможность определить, где и когда отображать данные на странице отчета и на этикетке. Предположим, что вы создасте список физических лиц, упорядоченный по фамилиям. Вы можете использовать области отчета, чтобы систематизировать информацию в отчете. Ниже приведен перечень информации, которую вы можете разместить в различных областях отчета:

- ◆ наименование списка, используя область заголовка (**title band**), который появляется один раз в начале отчета (рис. 13.2);
- ◆ общее количество человек в списке, используя область итогов (**summary band**), которая расположена в конце отчета;
- ◆ название отчета, номер страницы и текущую дату, используя верхний или нижний колонтитулы страницы (**page header** или **page footer bands**), которые расположены в начале или в конце каждой страницы соответственно;
- ◆ линия, разделяющая фамилии, начинающиеся с каждой буквы алфавита, чтобы облегчить поиск, используя верхний и нижний колонтитулы группы (**group header** и **footer bands**), которые появляются каждый раз, когда меняется первая буква фамилии.

Вы можете включать во все области отчета объекты, например, объект `Field` и вычисляемые выражения, чтобы отобразить данные.

При создании отчеты уже содержат в себе несколько областей по умолчанию; однако вы можете добавить и другие области в отчет или этикетку. В табл. 13.1 приведены все доступные области отчета и их примерный порядок следования на раскладке страницы.

Таблица 13.1. Области отчета

Область отчета	Описание
Title (Заголовок)	Отображает информацию о заголовке отчета или его титульном листе. Необязательная область может быть размещена на отдельной странице в начале отчета или на первой странице, непосредственно перед областью верхнего колонтитула. Не может быть длиннее одной страницы
Page Header (Верхний колонтитул страницы)	Отображает информацию, которая представлена один раз в начале страницы. Обязательная область на всех страницах, кроме случая, когда области Заголовок и Итоги (Title и Summary) на отдельных страницах. Не может быть длиннее одной страницы
Column Header (Верхний колонтитул колонки)	Отображает информацию или данные, которые представлены один раз для совокупности колонок. Добавляется автоматически для каждой колонки, созданной пользователем. Не появляется, если отчет имеет только одну колонку

Group Header (Верхний колонтитул группы)	Отображает информацию или данные, которые представлены один раз на группу, предшествует группе. Добавляется автоматически для каждой группы данных, созданной пользователем
--	---

Таблица 13.1 (окончание)

Область отчета	Описание
Detail Header (Верхний колонтитул детализации)	Отображает информацию или данные, которые представлены один раз на запись данных, предшествует записи. Добавляется пользователем
Detail (Детализация)	Одна область детализации обязательна в раскладке отчета. Пользователь может по желанию добавить дополнительную область детализации. Отображает данные записи. Может быть длиннее одной страницы
Detail Footer (Нижний колонтитул детализации)	Необязательная область, доступна для каждой области детализации в раскладке. Суммарная информация для данной записи
Group Footer (Нижний колонтитул группы)	Добавляется автоматически для каждой группы, созданной пользователем, и содержит вычисляемые данные для группы данных
Column Footer (Нижний колонтитул колонки)	Добавляется автоматически для каждой колонки, созданной пользователем. Не появляется, если отчет имеет только одну колонку. Содержит суммарную информацию и итоги для колонки
Page Footer (Нижний колонтитул страницы)	Обязательная область на всех страницах, за исключением случая, когда области Заглавие и Итоги (Title и Summary) на отдельных страницах
Summary (Итоги)	Необязательная область, может быть размещена на отдельной странице в конце отчета или непосредственно перед нижним колонтитулом страницы на последнем листе отчета. Содержит итоги всего отчета

Максимальное число вложенных групп данных — 74, и максимальное количество областей детализации — 20. Создание области детализации не уменьшает доступного количества групп данных, и создание групп данных не ограничивает количество областей детализации, которые вы можете добавить в ваш отчет.

Visual FoxPro предоставляет в ваше распоряжение несколько инструментов для создания отчетов:

- ◆ непосредственный вывод на принтер;
- ◆ стандартный отчет (Quick Report), который позволяет создать отчет, в котором поля таблицы будут являться колонками отчета;
- ◆ мастер отчетов (Report Wizard), который позволяет создать отчет. Выбрав его необходимые характеристики — группировку данных, нужные поля таблиц, сортировку данных;

- ◆ Конструктор отчетов, который позволяет программисту создавать отчет самостоятельно;
- ◆ класс ReportListener.

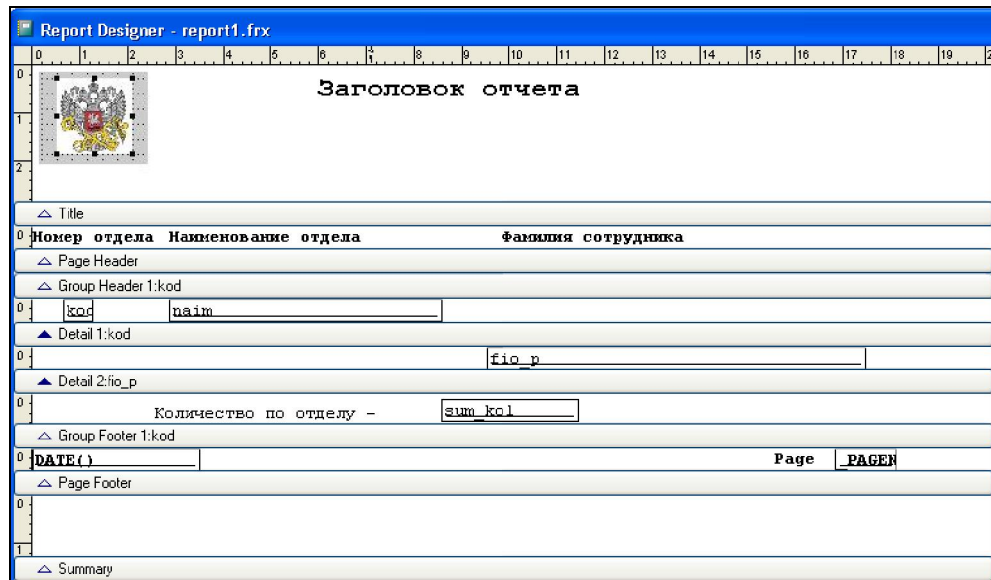


Рис. 13.1. Конструктор отчетов

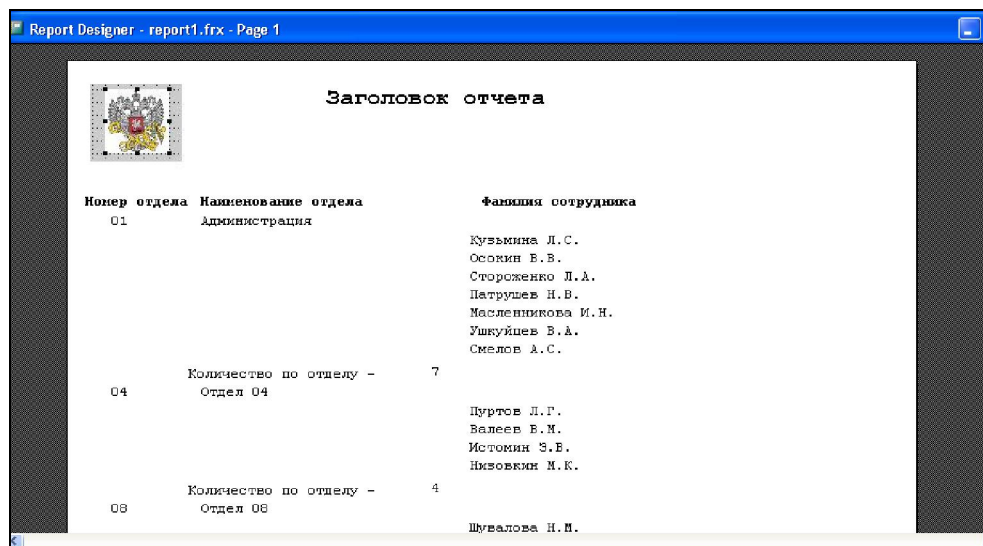


Рис. 13.2. Пример отчета

Непосредственный вывод на принтер

Ряд команд Visual FoxPro позволяют вывести информацию непосредственно на принтер. Конечно, то, что получится, это не отчет, но информацию о содержимом таблицы получить можно. В частности, в этих командах можно указать поля таблицы, которые необходимо вывести на печать.

Команда `List` выводит на экран или принтер содержимое открытой таблицы.

Синтаксис:

```
LIST [FIELDS FieldList] [Scope] [FOR lExpression1]
    [WHILE lExpression2] [OFF] [NOCONSOLE] [NOOPTIMIZE]
    [TO PRINTER [PROMPT] | TO FILE FileName [ADDITIVE]]
```

Аналогичную функцию выполняет команда

```
Select [FieldList] ... TO RPINTER
```

Стандартный отчет (Quick Report)

Это самый простой способ получить отчет. Назвать его профессионально выполненным вряд ли можно, но для того, чтобы создать отчет "на скорую руку", он годится. Порядок ваших действий должен быть таким:

1. Откройте таблицу, по которой необходимо сделать отчет.

2. В главном меню выберите команду **File | New | Report**, или наберите в командном окне команду `CREATE REPORT`.
3. Выберите в меню **Report | Quick Report**. Откроется диалоговое окно **Quick Report** (рис. 13.3).

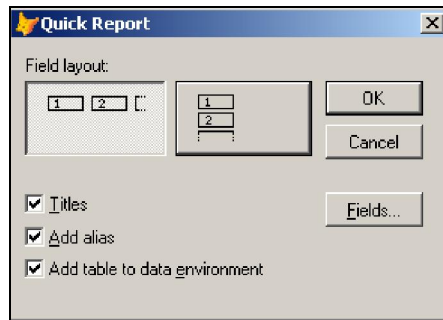


Рис. 13.3. Диалоговое окно **Quick Report**

Кнопки **Field layout** позволяют выбрать внешний вид отчета, будет ли он многоколоночным или многостраничным. В многоколоночном отчете каждая запись будет расположена в отдельной строке (рис. 13.4), в многостраничной — каждое поле в отдельной строке (рис. 13.5).

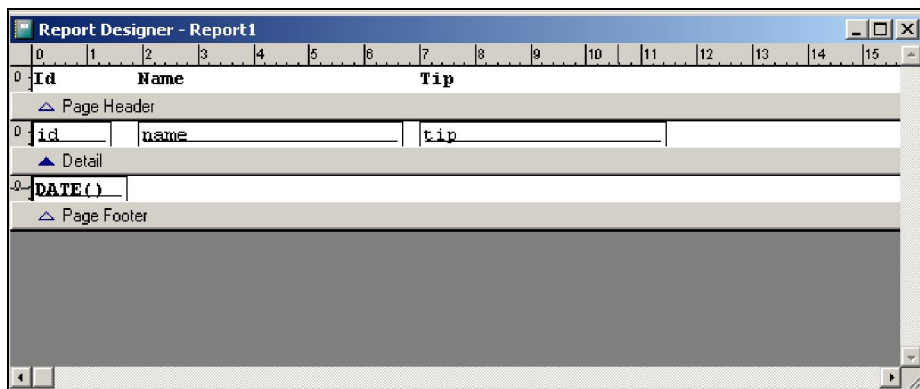


Рис. 13.4. Окно компоновки отчета Многоколоночный отчет

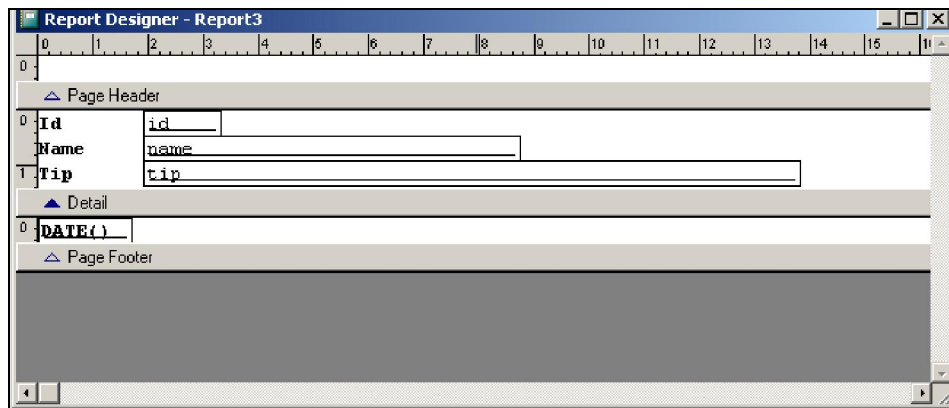



Рис. 13.5. Окно компоновки отчета. Многостраничный отчет

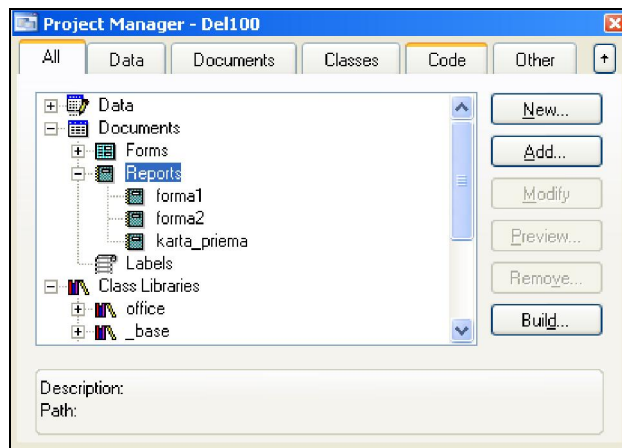
При установке флажка **Titles** в отчете печатаются заголовки колонок. При установке флажка **Add alias** имя таблицы автоматически добавляется к имени поля, т. е. поле выглядит как func.name, например.

Установка флажка **Add table to data environment** добавляет таблицу в среду окружения отчета **Data Environment**. После нажатия кнопки **OK** на экране появляется окно компоновки отчета.

4. Выберите необходимые поля и разместите их в требуемом порядке. При желании можно манипулировать длиной полей, изменять заголовки колонок и т. д. Созданный отчет можно сохранить, либо нажав кнопки **<Ctrl>+<W>**, либо нажав на "крестик" в правом верхнем углу отчета (в этом случае появится запрос на сохранение отчета). Можно также использовать **File | Save** главного меню Visual FoxPro или кнопку  (Save) на панели инструментов.
5. Напечатать отчет можно, выбрав пункт главного меню **Report | Run Report** или нажав правую кнопку мыши, находясь в области отчета.

Мастер отчетов

Мастер отчетов можно вызвать как из проекта, выбрав в дереве проекта ветку Documents, Reports, так и из главного меню Visual FoxPro, выполнив команду **File | New | Report**. Если вы вызываете мастер отчетов из проекта, то увидите на экране следующую картинку (рис. 13.6).

Рис. 13.6. Вкладка **Reports** в Менеджере проектов

Если вы нажмете кнопку **New**, то появится окно, в котором происходит диалог выбора мастера проектов или стандартного отчета (рис. 13.7).

Рис. 13.7. Окно диалога **Wizard Selection**

Мастер отчетов запросит требуемый тип отчета: "One-to-Many Report Wizard", который создает отчет для таблиц с отношением "один-ко-многим", или простой однотабличный отчет Report Wizard. Попробуем создать простой однотабличный отчет. Создание отчета в этом случае происходит в пошаговом режиме.

1. Выбор базы данных и таблицы. Нужно определиться, какие таблицы вам необходимы, какие поля этих таблиц вы будете размещать в отчете (рис. 13.8). Для выполнения шага 1 служит список **Databases and tables**.

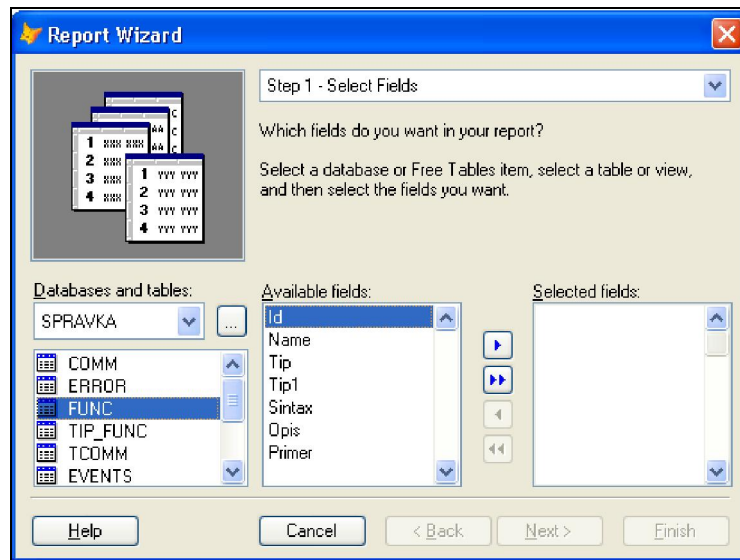


Рис. 13.8. Выбор необходимых таблиц и полей для отчета

Если список таблиц пуст или содержит не те таблицы, которые вам требуются, нажмите кнопку справа от списка баз данных. Откройте необходимые таблицы.

Как только вы выбираете нужную таблицу из перечня, в связанном списке **Available fields** появляются доступные для использования поля. Вы должны выбрать нужные вам поля и перенести их в список **Selected fields**. Используйте для переноса кнопки, расположенные между двумя списками. Когда список требуемых полей сформирован, нажмите кнопку **Next** для перехода к следующему шагу.

Если при создании или модификации таблицы было задано свойство *Caption*, то в отчете в заголовках колонок появится то название, которое вы им дали, в противном случае колонки будут озаглавлены именем поля.

2. Определение полей для группировки отчета.

Если данные необходимо сгруппировать, то это действие нужно выполнить на втором шаге (рис. 13.9). Данные могут быть сгруппированы по типам, отделам, кодам и т. д.

Для выполнения группировки выберите нужное поле из раскрывающегося списка 1. Если требуется сложная группировка, можно использовать списки 2 и 3. При нажатии на кнопку **Grouping options...** откроется окно **Grouping Intervals**, в котором определяются интервалы группировки (рис. 13.10).

Выбрав нужный интервал, вы определите уровень фильтрации ваших данных.

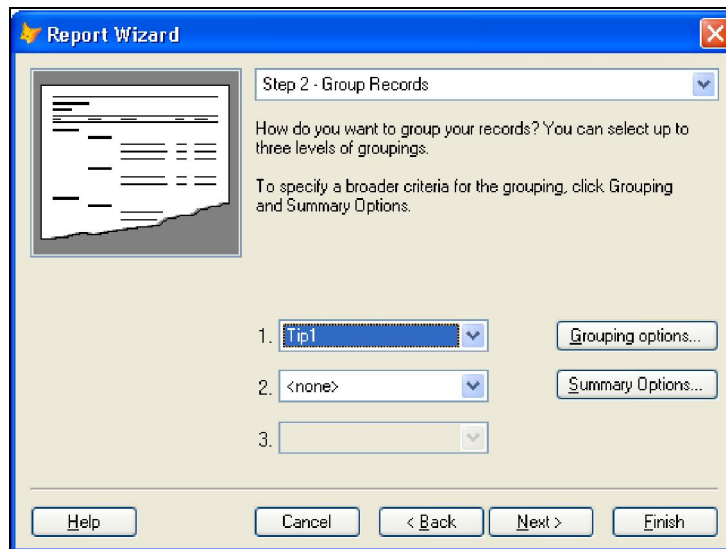


Рис. 13.9. Группировка полей отчета

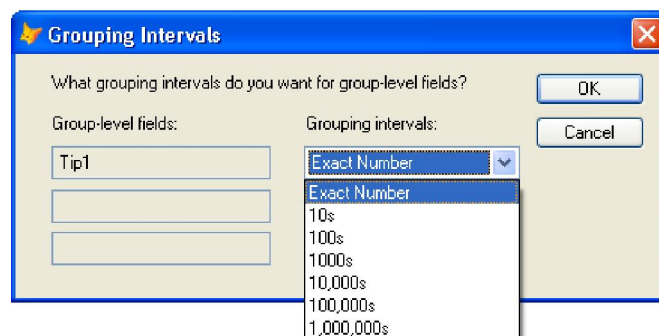


Рис. 13.10. Интервалы группировки

Если нажать кнопку **Summary Options...**, то можно использовать арифметические функции SUM, AVG, COUNT, MIN и MAX, которые выполняются для числовых полей.

Установите флажок в нужном месте, выбрав поле и функцию (табл. 13.2).

Таблица 13.2. Описание функций окна *Summary Options*

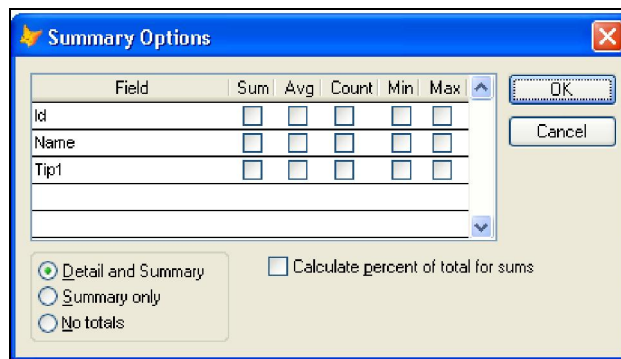
Столбец	Описание
Field	Наименование поля, по которому будет вычисляться итоговое значение
Sum	В отчет помещается итоговая сумма по указанному полю
Avg	В отчет помещается среднее значение поля

Таблица 13.2 (окончание)

Столбец	Описание
Count	В отчет помещается количество записей
Min	В отчет помещается минимальное значение
Max	В отчет помещается максимальное значение по столбцу

В отчете может быть вычислено процентное соотношение промежуточного суммарного значения к общему значению. Для вычисления установите флажок в поле **Calculate percent of total for sums**.

В диалоговом окне **Summary Options** под таблицей находится переключатель, управляющий отображением в отчете областей данных (рис. 13.11, табл. 13.3).

Рис. 13.11. Диалоговое окно **Summary Options**Таблица 13.3. Назначение переключателей суммирования окна **Summary Options**

Опция	Назначение
Detail and Summary (Область данных и итоги)	В отчете будут показаны промежуточные данные и итоговые
Summary only (Только итоги)	В отчете отображается область данных и итоговые значения
No totals (Нет итогов)	В отчете будет показана только область данных

3. Выбор стиля отчета.

Список содержит пять видов стилей. Предварительный просмотр каждого стиля возможен в левом верхнем углу отчета (рис. 13.12).

4. Указываем порядок размещения полей отчета, а также расположение (ориентацию) — книжная или альбомная.

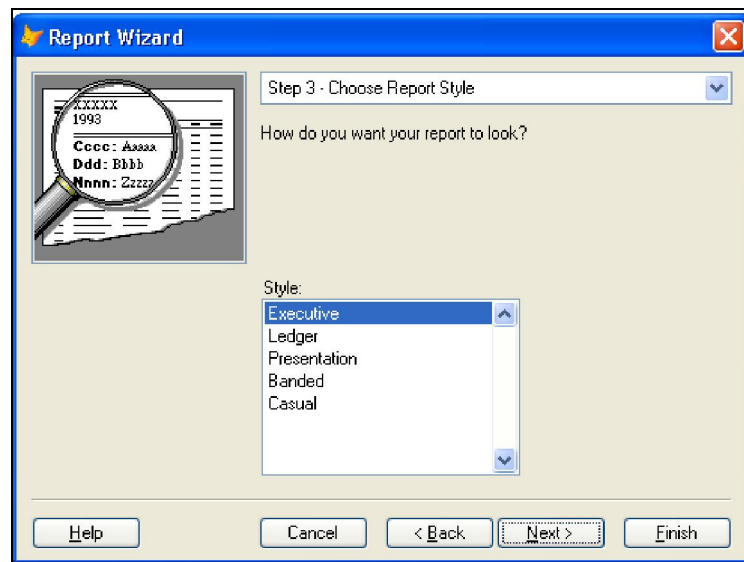


Рис. 13.12. Выбор стиля отчета

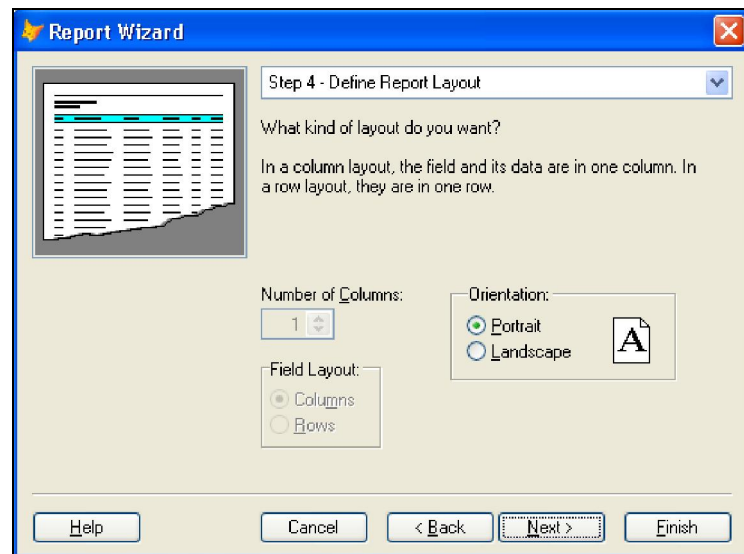


Рис. 13.13. Определение порядка размещения объектов в отчете

5. Задаются поля, по которым требуется упорядочивание данных отчета, т. е. сортировка (рис. 13.13).

Поля, по которым можно производить сортировку, находятся в списке **Available fields or index tag**. Для того чтобы по выбранному полю осуществлялась сортировка, нужно перенести поле из правого списка в левый. Выберите нужное поле слева и нажмите кнопку **Add**. Если поле выбрано ошибочно, можно удалить его из списка с помощью кнопки **Remove**. Переключатели **Ascending** и **Descending** служат для определения порядка сортировки: **Ascending** — по возрастанию и **Descending** — по убыванию (рис. 13.14).

6. Задание заголовка отчета и порядка дальнейшей работы с отчетом (рис. 13.15).

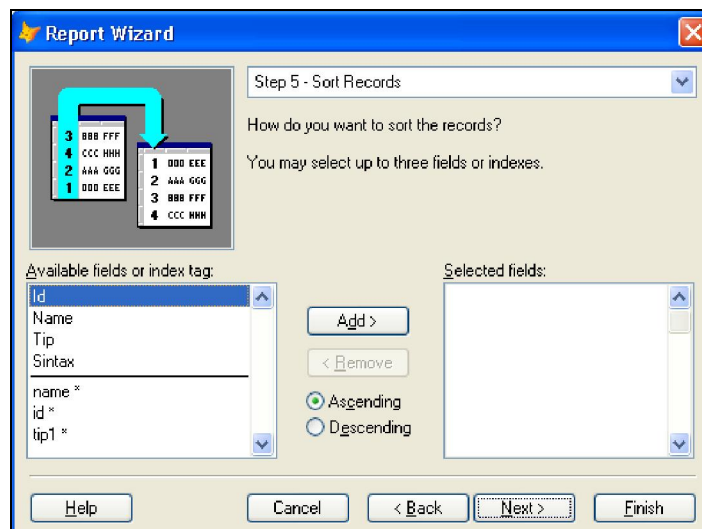


Рис. 13.14. Установка критериев сортировки данных отчета

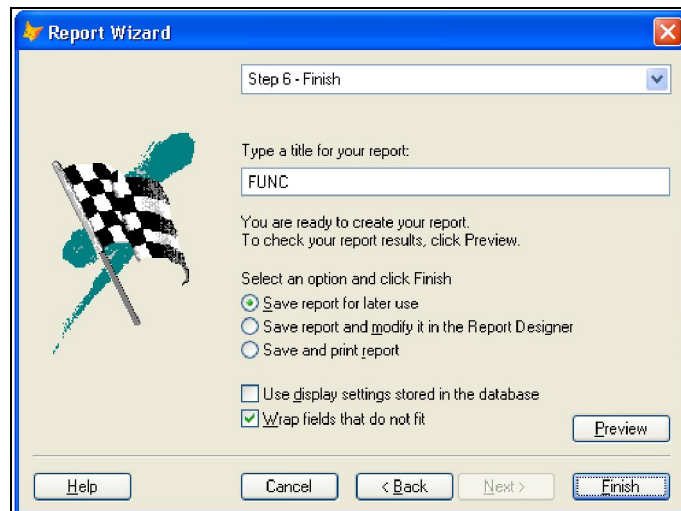


Рис. 13.15. Заключительный этап создания отчета

В нижней части окна находятся 3 переключателя и два флага. Описание их дано соответственно в табл. 13.4 и 13.5.

Таблица 13.4. Опции сохранения отчета

Опция	Описание
Save report for later use (Сохранить отчет для дальнейшего использования)	Сохраняет заданный отчет
Save report and modify it in the Report Designer (Сохранить отчет и модифицировать его в Конструкторе отчетов)	Сохраняет заданный отчет и открывает его в Конструкторе отчетов
Save and print report (Сохранить и напечатать отчет)	Сохраняет отчет и выводит его на печать

Таблица 13.5. Флажки окна заключительного этапа создания отчета

Флажок	Описание
Use display settings stored in the database	Записывает установки отображения для полей, записанных в базе данных
Wrap fields that do not fit	Разрешает перенос в полях, если данные не помещаются

В правом нижнем углу размещена кнопка **Preview** (Предварительный просмотр). Вы можете ей воспользоваться, и, если увиденная картина вам не понравится, мо-

жете нажать кнопку **Back** и вернуться к предыдущим шагам, чтобы скорректировать свои действия (рис. 13.15).

Конструктор отчетов

Это наиболее "серьезный", профессиональный инструмент создания отчетов. Существуют несколько способов вызова Конструктора отчетов:

1. Выполните команду меню **File | New | Report | New**.
2. В окне **Project Manager** выберите вкладку **Documents**, затем выберите **Reports** и **New**. В открывшемся окне выберите **New Report**.
3. В командном окне введите команду `CREATE REPORT` или `MODIFY REPORT`.
4. Используйте отчет, созданный мастером, сохраните его, а затем корректируйте.

После выполнения любого из вышеназванных действий мы получим обычный отчет, подобный созданному в QuickReport. Чтобы сделать отчет более функциональным, нужно настроить его (рис. 13.16).

Если нажать правую кнопку мыши, находясь на макете отчета, то можно увидеть контекстное меню. Выбрав в этом меню пункт **Optional Bands**, вы получите возможность настроить новые зоны отчета — **Title** и **Summary**, отвечающие за заголовок отчета и итоговые данные. Для настройки зон необходимо установить флажки **Title Band** и **Summary Band**. В этом случае в отчете появятся новые зоны (рис. 13.17).

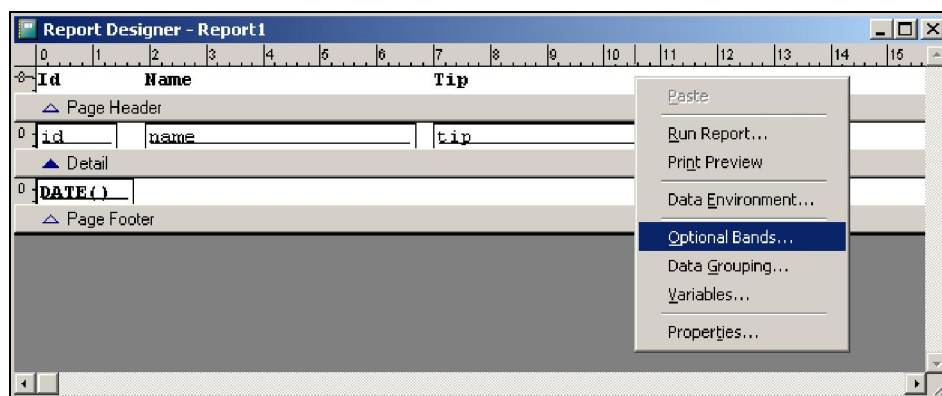


Рис. 13.16. Настройка отчета средствами конструктора отчетов

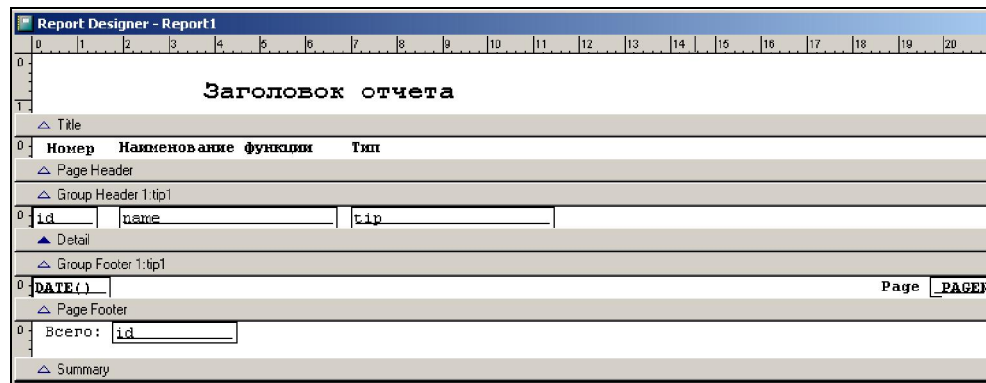


Рис. 13.17. Окно Конструктора отчетов

Полный список зон отчета содержит:

- ◆ **Title** — заголовок отчета;
- ◆ **Page Header** — верхний заголовок каждой страницы;
- ◆ **Group Header** — верхний заголовок каждой группы данных;
- ◆ **Detail** — поля каждой записи;
- ◆ **Group Footer** — нижний заголовок каждой группы;
- ◆ **Page Footer** — нижний заголовок каждой страницы;
- ◆ **Summary** — итоговые данные.

Конструирование отчета включает в себя:

- ◆ определение среды окружения;
- ◆ размещение данных;
- ◆ размещение разного рода графических изображений (линий, прямоугольников, рисунков и т. д.);
- ◆ сохранение отчета.

Среда окружения отчета

Среда окружения для отчета, созданного с помощью мастера, уже определена. Для того чтобы определить среду окружения для нового отчета, следует выбрать в главном меню **View | Data Environment**. В появившемся окне при нажатии правой кнопки мыши появится контекстное меню, в котором нужно выбрать пункт **Add**. Откроется диалоговое окно **Add Table or View** для выбора необходимых таблиц, представлений и других файлов, используемых в отчете.









Панель инструментов Конструктора отчетов

Для работы с отчетом необходимо активизировать панель инструментов Конструктора отчетов (рис. 13.18). Выполните для этого команду **View | Report Controls Toolbar**. Назначение каждой кнопки подсвечивается при наведении на нее указателя мыши.



Рис. 13.18. Панель инструментов Конструктора отчетов

Таблица 13.6. Назначение кнопок панели инструментов *Report Controls Toolbar*

Кнопка	Назначение
 Select Objects	Разрешение выбора любого инструмента
 Label	Ввод текста в отчет
 Table	Размещение поля таблицы в отчете
 Line	Изображение линии
 Rectangle	Изображение прямоугольника
 Rounded Rectangle	Изображение овала
 Picture/OLE Bound Control	Размещение OLE-объекта в отчете
 Button Lock	Закрепление выбранной кнопки

Панель инструментов можно пристыковать к краю окна Visual FoxPro, можно разместить где угодно в пределах экрана — окно мобильно.

Выполнение надписей в отчете

Разрешается выполнять надписи в любой зоне отчета. Выберите на панели инструментов кнопку **Label**, поместите указатель мыши в нужное место отчета, далее выполняйте саму надпись. Для выбора шрифта воспользуйтесь командой главного меню **Format | Font**. Данный пункт меню активизируется, когда вы находитесь в поле надписи **Label** на макете отчета. Если надпись необходимо удалить, нужно выбрать ее и нажать клавишу <Delete> на клавиатуре.

Двойной щелчок мыши на поле надписи откроет диалоговое окно **Label Properties**, в котором можно настроить параметры надписи. Можно управлять размещением текста в отчет с помощью кнопок этого окна:

- ◆ **Float** — перемещает текст в зависимости от окружающих его полей;
- ◆ **Fix relative to top of band** — фиксирует текст в зависимости от верхней границы полосы отчета;
- ◆ **Fix relative to bottom of band** — фиксирует текст относительно нижней границы полосы отчета.

В поле ввода **Comment** можно расположить неотображаемые в отчете комментарии к тексту. Кнопка **Print When** открывает диалоговое окно для определения условий печати текста: удалять ли пустые строки в отчете и признак перехода на следующую страницу.

Размещение табличных полей в отчете

Поля таблицы размещаются в зоне отчета, обозначенной как **Detail**. Для размещения любого поля в этой зоне необходимо открыть среду окружения (Data Environment), выбрать любое поле и перетащить его мышью на поле **Detail**. Двойной щелчок мыши на выбранном поле открывает окно **Field Properties**, которое позволяет настроить размещение поля в зоне **Detail** и внешний вид при печати поля — размер и тип шрифта, условия печати (рис. 13.19).

Форматирование позволяет вывести символьные поля прописными буквами, выровнять их по центру или по правому/левому краю. Цифровые поля могут быть представлены в формате денежной единицы или в удобном для восприятия виде для нулевых или отрицательных значений. Поля, содержащие дату, могут быть представлены в одном из допустимых форматов даты, которые определяются командой `SET DATE`.

В конце отчета в зоне **Summary** можно вывести некоторые итоговые значения. Для того чтобы указать поле для вычисления, можно открыть среду окружения и перетащить необходимое поле в зону **Summary**. Указать функцию для подсчета итогов можно на вкладке **Calculate**. Список опций **Calculation type** приведен в табл. 13.7.

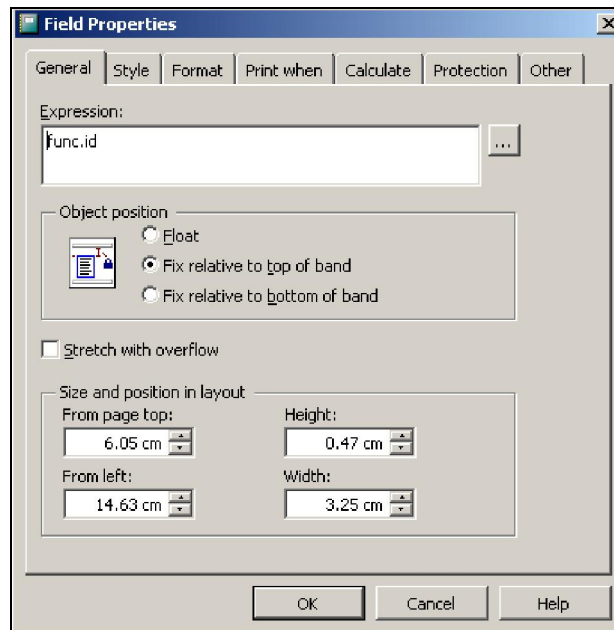


Рис. 13.19. Окно свойств поля Конструктора отчетов

Таблица 13.7. Список опций *Calculation type*

Опция	Назначение
None	Вычисления не выполняются
Count	Подсчитывается количество значений поля
Sum	Вычисление итоговой суммы по полю
Average	Вычисление среднего арифметического значения поля
Lowest	Вычисление минимального значения
Highest	Вычисление максимального значения
Standard deviation	Вычисление девиации
Variance	Вычисление отклонения значений поля от среднего значения в группе

Как правило, в конце каждой страницы отчета печатается дата выполнения отчета. Не забудьте настроить дату в удобном для вас формате и определить достаточный размер окна для вывода даты. Иногда не хватает места в окне, и дата получается нечитабельной, например "01/05/0".

Группировка данных отчета

Что дает нам группировка данных? Это:

- ◆ возможность вычислять итоги внутри группы;
- ◆ возможность печатать каждую группу на новой странице;
- ◆ печать текста в верхних и нижних заголовках, отличающегося для разных групп;
- ◆ возможность нумеровать страницы сначала для каждой группы.

Например, группировкой можно воспользоваться для того, чтобы напечатать список функций Visual FoxPro по выполняемым действиям — символьная, работы с окнами, с файлами и т. д. Для группировки данные должны быть отсортированы или индексированы по тому выражению, по которому создается группировка. Саму группировку можно задать в диалоговом окне **Data Grouping**, которое можно вызвать из главного меню: **Report | Data Grouping**. При каждом изменении значения выражения, указанного для группировки, происходит вычисление итогов, заданных для группы, заполняются зоны **GROUP FOOTER** и **GROUP HEADER**. Как правило, критерий сортировки представляет собой отдельное поле, но это может быть и переменная, функция или выражение.

Переменные в отчете

При работе над созданием отчета можно использовать все переменные, которые доступны к моменту начала выполнения отчета. Кроме этого, в Конструкторе отчетов предусмотрена возможность определить переменные для отчета, например, для хранения некоторых итогов (рис. 13.20). Чтобы получить доступ к созданию переменных отчета, воспользуйтесь командой меню **Report | Variables**.

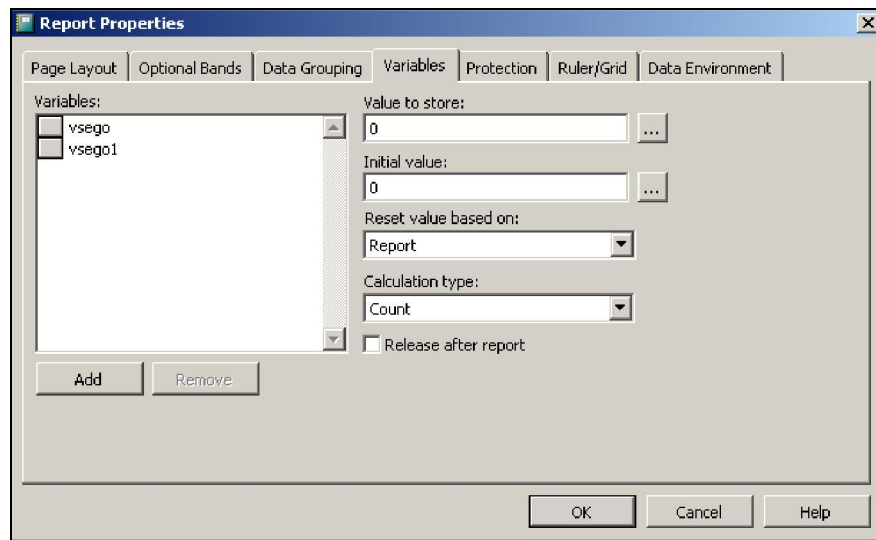


Рис. 13.20. Окно создания переменных отчета

Имя переменной может содержать буквы, цифры, символы подчеркивания и не может начинаться с цифры. По умолчанию вновь созданной переменной присваивается начальное значение, равное 0. Если установить флажок **Release after report**, то переменная будет освобождена из памяти по окончании выполнения отчета.

Переменные могут использоваться в отчете в качестве полей, для этого список переменных выводится в окне **Expression Builder** (Построитель выражений) (рис. 13.21).

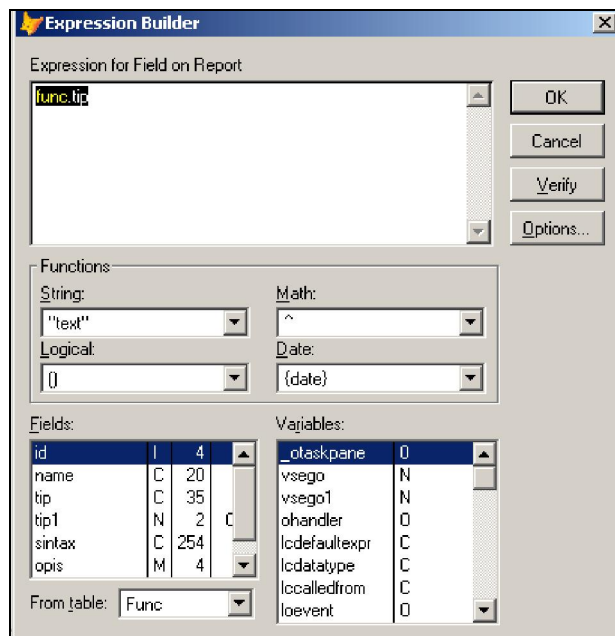


Рис. 13.21. Окно Expression Builder

Не нужно забывать о том, что переменные должны находиться в области видимости, иначе вместо отчета вы получите сообщение об ошибке.

Очень часто в выражение для поля включается не само поле, а некоторая функция для его вычисления. В выражение можно включить и функцию `IIF()`, которая имеет следующий синтаксис:

`IIF(Условное выражение, ВыражениеTRUE, ВыражениеFALSE)`

Таким образом, в отчет может быть включена печать по некоторому логическому условию.

Разметка страницы

Иногда при печати готового отчета возникает конфликт, когда сформированная страница превышает область заполнения страницы, предусмотренную настройкой принтера. К чему это приводит? Иногда обрезаются часть нужного текста, иногда печатаются пустые страницы. Поэтому разметку страницы лучше настроить. Можно это сделать, уменьшив правую и нижнюю границы отчета. Второй способ заключается в том, чтобы разметить страницы. Выполните команду **File | Page Setup**.


В появившемся диалоговом окне определите параметры:

- ◆ **Number** — количество колонок;
- ◆ **Width** — ширина колонок;

- ◆ **Spacing** — расстояние между колонками;
- ◆ **Left Margin** — ширина левого поля.

Кроме того, можно установить другой размер шрифта, и даже сменить стиль шрифта.

OLE-объекты в отчете

В отчеты можно помещать рисунки, эмблемы, изображения. Для этого воспользуйтесь панелью **Report Controls** и выберите на ней кнопку  **Picture/OLE Bound Control**. При этом появляется диалоговое окно определения OLE-объекта (рис. 13.22).

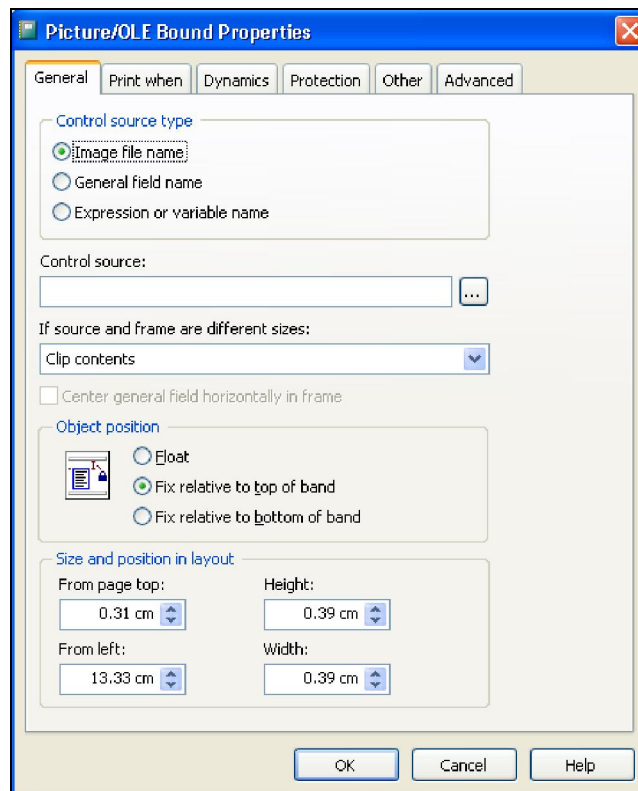


Рис. 13.22. Диалоговое окно определения OLE-объекта

Выбираем необходимое изображение в окне ввода **Control Source**, раздвигаем мышью рамку изображения так, чтобы оно получилось необходимого нам формата, и ваш рисунок уже появился в отчете. Рисунок из файла должен быть в формате BMP, ICO, ANI, JPEG, GIF или CUR.

Можно выбрать в окне **Picture/OLE Bound Properties** опцию **General field name** и указать место расположения рисунка в зоне **Detail**, тогда рисунок появится в каждой записи. Это может быть фотография сотрудника, изображение товара и т. д.

Это же диалоговое окно позволяет управлять размерами рисунка. В комбинированном списке **If source and frame are different sizes** содержится несколько вариантов управления размером рисунка, в случае, когда рисунок превышает выделенную для него область:

- ◆ **Clip picture** — верхний левый угол рисунка совмещается с верхним левым углом рамки, а то, что не поместилось, не отображается;
- ◆ **Scale picture, retain shape** — рисунок сжимается пропорционально размеру рамки;
- ◆ **Scale picture, fill the frame** — рисунок растягивается горизонтально и вертикально пропорционально размерам рамки.

Просмотр и печать отчета

Перед печатью отчет желательно запустить для просмотра. При этом выявляется ряд неточностей или ошибок, которые лучше исправить на этапе просмотра. Просмотр можно запустить из основного меню, либо **Report | Print Preview**, либо **File | Print Preview**. Команда **Print Preview** содержит также контекстное меню отчета. Отчет можно сохранить командой **File | Save**. Впрочем, некоторым разработчикам не нравится окно предварительного просмотра отчета, и они разрабатывают собственные. В Интернете можно найти ряд альтернативных решений.

Многоколоночный отчет

В многостраничном отчете, как вы уже видели, данные располагаются так, как изображено на рис. 13.3, т. е. каждое поле — в отдельной строке. При таком расположении, как правило, правая часть страницы остается пустой, а это нерационально. Для того чтобы отчет расположить, например, в две колонки, нужно использовать настройку **Columns Number**, которая располагается в окне **Report Property**. Вызвать это диалоговое окно можно либо выполнив команду главного меню **File | Page Setup**, либо нажав правую кнопку мыши в поле отчета. Порядок действий должен быть примерно такой:

1. Создаем новый отчет.
2. В **Data Environment** подключаем нужную таблицу (для примера подключим таблицу `c:\Program Files\Microsoft Visual FoxPro\Sample\Data\Customer`, содержащую адреса клиентов).
3. С помощью **Quick Report** размещаем поля адреса на макете отчета.

4. Вызываем окно **Report Property**, как это описано выше (например, через **File | Page Setup**).
5. Устанавливаем в окне **Columns Number = 2**. При этом автоматически изменяется ширина колонки **Width** (рис. 13.23).

Посмотрим на созданный отчет. Выглядит он теперь таким образом, как на рис. 13.24.

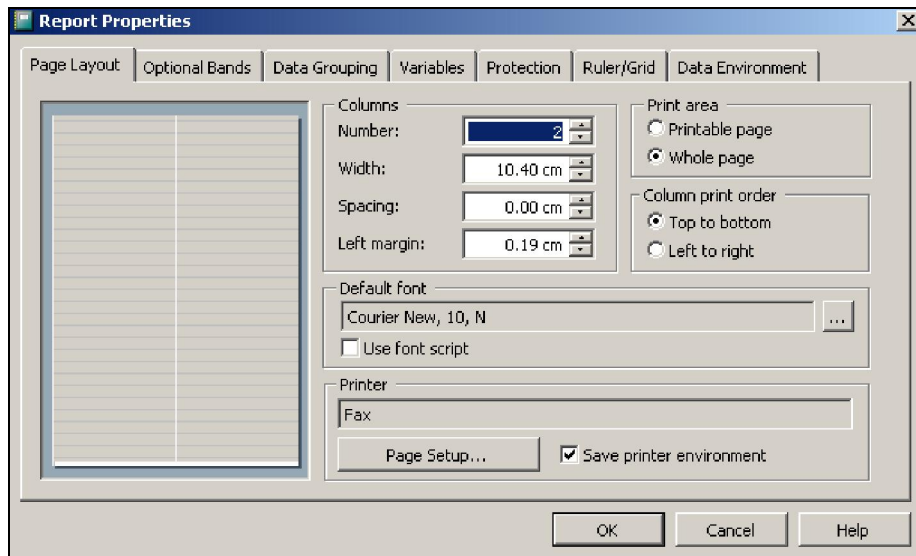


Рис. 13.23. Установка двух колонок в поле отчета

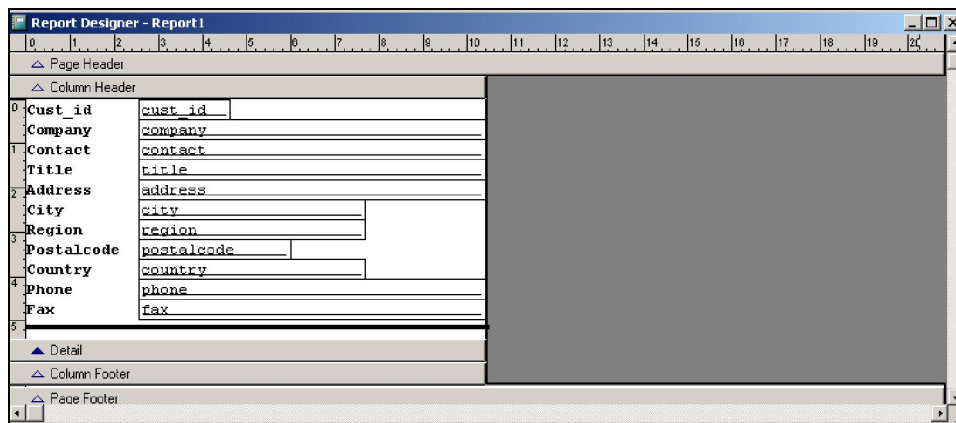


Рис. 13.24. Многостраничный отчет

Посмотрим на созданный отчет, запустив его в режиме предварительного просмотра (рис. 13.25).

6. Сохраним отчет командой меню **File | Save**.

Report Designer - Report1 - Page 1			
Cust_id	ALFKI	Cust_id	BLAUS
Company	Alfreds Futterkiste	Company	Blauer See Delikatessen
Contact	Maria Anders	Contact	Hanna Moos
Title	Sales Representative	Title	Sales Representative
Address	Obere Str. 57	Address	Forsterstr. 57
City	Berlin	City	Mannheim
Region		Region	
Postalcode	12209	Postalcode	68306
Country	Germany	Country	Germany
Phone	030-0074321	Phone	0621-08460
Fax	030-0076545	Fax	0621-08924
Cust_id	ANATR	Cust_id	BLONP
Company	Ana Trujillo Emparedados y helados	Company	Blondel pure et fils
Contact	Ana Trujillo	Contact	Friedrique Citeaux
Title	Owner	Title	Marketing Manager
Address	Avda. de la Constituciyn 2222	Address	24, place Klüber
City	México D.F.	City	Strasbourg
Region		Region	
Postalcode	05021	Postalcode	67000
Country	Mexico	Country	France
Phone	(5) 555-4729	Phone	88.60.15.31
Fax	(5) 555-3745	Fax	88.60.15.32

Рис. 13.25. Многостраничный отчет в окне предварительного просмотра

Создание этикеток

Этикетка — это простой отчет. В этикетках не применяется группировка данных и верхний и нижний колонтитулы. Поля этикеток отображаются без отступов и пробелов.

Для создания этикеток применяется мастер этикеток **Label Wizard** или конструктор этикеток **Label Designer** (рис. 13.26). При их использовании создаются файлы этикеток, имеющие расширение lbx или lbt. Для создания этикеток применяются те же команды, что и в отчетах, но команда **Report** заменяется на **Label**, например:

CREATE LABEL — для создания этикетки;

LABEL FORM <имя файла этикетки .LBX> **TO PRINTER** — для печати.

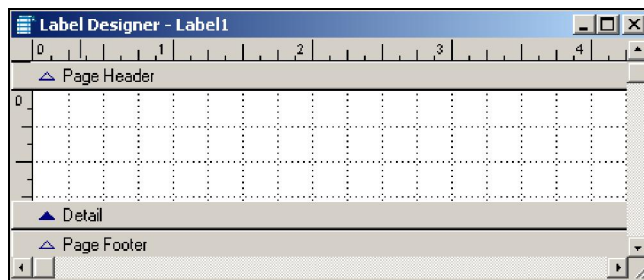


Рис. 13.26. Макет этикетки

Этикетки применяются не только в качестве наклеек на дела или в качестве наклеиваемых адресов на конверты, но и в качестве прайс-листов, писем со стандартным текстом, в который вставляются различные значения табличных полей, результаты вычислений и т. д.

Особенности многополосных отчетов

Visual FoxPro позволяет добавлять области отчета, например, добавить области детализации. Для чего это может понадобиться? Например, для включения в отчет дополнительных таблиц. Как вы уже видели на рис. 13.1, в отчет включены две области детализации: одна для номеров и наименований отделов, вторая — для сотрудников.

Следующие области отчета могут иметь несколько полос:

- ◆ **Title;**
- ◆ **Summary;**
- ◆ **Detail;**
- ◆ **Page Header** и **Page Footer** для **Summary;**
- ◆ **Detail Header** и **Detail Footer** для **Detail.**

Для того чтобы добавить дополнительную полосу, вы должны открыть отчет в режиме редактирования и в главном меню выбрать пункт **Reports | Optional Bands Documents Properties** (рис. 13.27).

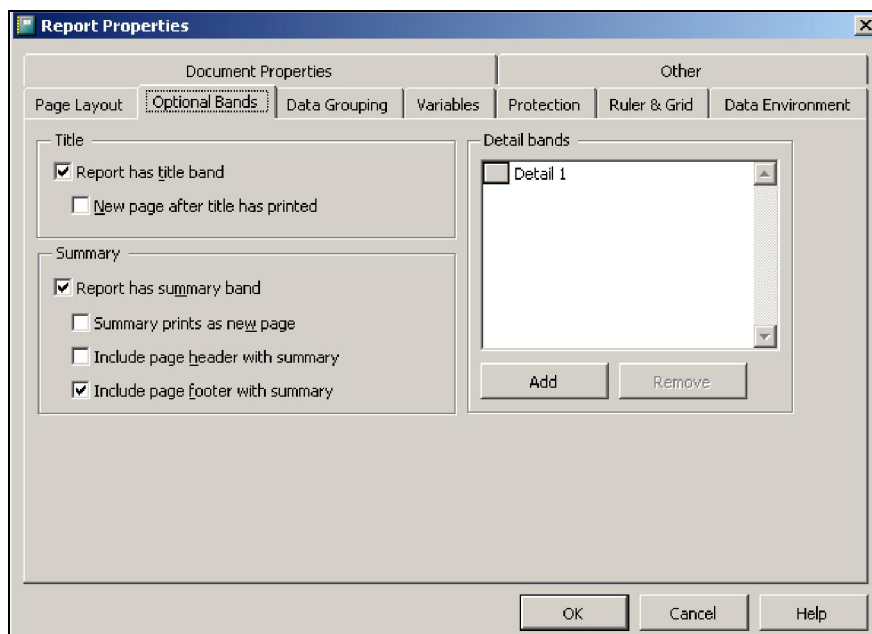


Рис. 13.27. Добавление дополнительных полос отчета

Если вы хотите добавить **Title Band** (заголовок), установите флажок **Report has title band**. Если необходимо добавить **Summary band**, установите флажок **Report has summary band**. Если нужно установить еще одну полосу детализации, нажмите кнопку **Add** в области **Detail bands**, в этом случае будет добавлена еще одна полоса детализации.

Скорее всего, для каждой полосы детализации вам понадобится указать свою управляющую таблицу. Сделать это можно следующим образом:

1. Откройте **Отчет** или **Этикетку** в соответствующем Конструкторе.
2. В меню **Report** выберите опцию **Edit Bands**.
3. В появившемся диалоговом окне **Edit Bands** выделите нужную полосу детализации и нажмите **OK**.
4. В диалоговом боксе **Detail Band Properties** выберите закладку **Band**.
5. В соответствующем окне **Target alias expression** (выражение для управляющего псевдонима) введите нужное имя. Для построения выражения нажмите многоточие (...), откроется соответствующий построитель выражения (**Expression Builder**).
6. После завершения нажмите кнопку **OK**.

Допустим, вы не хотите повторять печать для каждого сотрудника номера отдела и его наименования. В этом случае вам придется настроить печать полосы:

- ◆ на полосе детализации *в каждом поле*, печать которого вы не хотите повторять, нужно нажать правую кнопку мыши, вызвать в контекстном меню пункт **Properties** и выбрать вкладку **Print when** (рис. 13.28);
- ◆ в области **Print repeated values** (Печать повторяющихся значений) установите переключатель в значение **No**;
- ◆ в области **Also print** установите флажок **Remove line if blank** (Удалить строку, если она пуста).

Возможно, вам потребуется удалить излишне введенную область. В этом случае вы должны помнить, что нельзя удалить все области. Вы не можете уменьшить число областей детализации до нуля. Раскладка страницы должна всегда содержать хотя бы одну область детализации. Когда вы уменьшаете количество областей детализации, область под наивысшим номером, которая была добавлена последней и разместилась на нижнем уровне, будет удалена.

1. Откройте отчет или этикетку в соответствующем дизайнере.
2. В меню **Report** выберите **Optional Bands**.
3. В диалоговом окне **Report Properties** щелкните на вкладке **Optional Bands**.
4. На вкладке **Optional Bands** выберите область, которую вы хотите удалить:
 - чтобы удалить область заголовка (Title band), надо сбросить чек-бокс **Report_has_title_band**;
 - чтобы удалить область итогов (Summary band), надо сбросить чек-бокс **Report_has_summary_band**;
 - чтобы удалить область детализации, в списке **Detail bands** выберите нужную область и нажмите кнопку **Remove**. Когда дизайнер потребует подтвердить удаление, нажмите кнопку **Yes**.
5. Когда закончите удаление, нажмите кнопку **OK**.

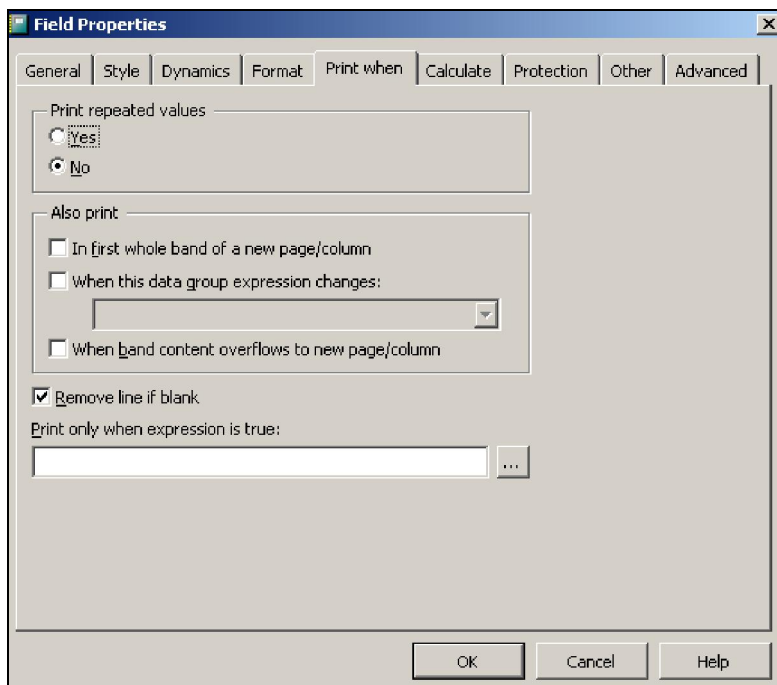


Рис. 13.28. Настройка печати дополнительной полосы детализации

Управление отчетами — объект *ReportListener*

Система отчетов Visual FoxPro 9.0 содержит новый базовый класс *ReportListener*, который позволяет выполнять отчеты в новом, объектно-ориентированном виде.

Класс *ReportListener* специально разработан для двухсторонней связи с механизмом отчетов на всем протяжении процесса генерации отчета. С помощью *ReportListener* вы можете:

- ◆ написать код для каждого процесса отображения элемента и событий полосок/областей как процесс генерации отчета — проходя вперед по области записи отчета;
- ◆ менять позицию элементов раскладки, которые изображаются, или их содержание;
- ◆ предоставлять обратную связь пользователю о ходе исполнения отчета;
- ◆ делать запросы картинок страниц для их отображения на экране или сохранения на диск;

- ◆ обслуживать частные сессии данных, хранить информацию из таблиц определения отчета или этикетки (FRX или LBX) для исследования атрибутов объектов отчета и полосок/областей (bands);
- ◆ непосредственно контактировать с устройством, на которое система отчетов отображает вывод;
- ◆ исследовать все опции команды `REPORT FORM` и менять некоторые из этих опций динамически.

Когда обрабатываются данные отчета или этикетки, информация посылается объекту `ReportListener` до того, как будет произведен непосредственный вывод на принтер.

Visual FoxPro предоставляет несколько различных возможностей для подключения объекта `ReportListener` к командам `REPORT FORM` или `LABEL FORM`:

- ◆ использовать в команде опцию `OBJECT`.

```
REPORT FORM FileName1 | ? [ENVIRONMENT] [Scope]
[FOR lExpression1] [WHILE lExpression2] [NOOPTIMIZE]
[RANGE nStartPage [, nEndPage]]
[HEADING cHeadingText] [SUMMARY] [NORESET] [PLAIN]
[NOCONSOLE | OFF] [PDSETUP]
[NAME ObjectName]
[OBJECT oReportListener | TYPE iExpression]
[TO OutputDestination [NODIALOG]]
[PREVIEW [PreviewDestination] [NOWAIT] [WINDOW WindowName]]
```

- ◆ использовать опцию `OBJECT TYPE <N>`, указывая номер `ListenerType` для типа вывода, который вы желаете создать. Report Engine запрашивает `ReportListener` об подходящем `ListenerType` из приложения Report Output Application.

```
REPORT FORM Myreport OBJECT TYPE 1 && PreviewListener
или
```

```
REPORT FORM Myreport OBJECT TYPE 5 && HTMLListener
```

Возможные значения предложения `Object Type`:

```
-2 Debug Listener
0 Print Listener
1 Preview Listener
4 XML Listener
5 HTML Listener
```

`Print` — представляет собой стандартный вывод на принтер;

`Preview` — улучшенная программа предварительного просмотра;

`XML` — создает XML-документ, который может состоять как из только ваших данных, так и из ваших данных плюс схема данных;

HTML — создает HTML-файл;

Debug — создает текстовый файл для отладки;

- ◆ использовать SET REPORTBEHAVIOR 90 для активизации объектного метода всем командам REPORT FORM и LABEL FORM в приложении.

```
SET REPORTBEHAVIOR 90
```

```
REPORT FORM Myreport PREVIEW
```

Используя ReportListener, вы можете добавить формат вывода, такой как HTML или XML, которые отчеты Visual FoxPro не поддерживают по своей природе.

Еще одна полезная особенность этого класса — "размноженный" вывод (multiply output). Это значит, что при выполнении одной и той же команды REPORT FORM вы можете одновременно печатать отчет и создавать его HTML-копию. VFP загружает все эти "многократные" выводы в новую системную переменную oReportOutput[] (не путайте с системной переменной _ReportOutput!). VFP может затем получить доступ к любому из определенных объектов через массив.

Физически библиотека класса _REPORTLISTENER.vcx находится в подкаталоге FFC каталога Visual FoxPro и содержит следующие классы:

- ◆ _ReportListener (базовый класс, является классом-родителем для всех последующих классов);
- ◆ _FrxCursor (имеет свойства и методы для манипулирования файлами FRX);
- ◆ _GDIPlus (Новая библиотека GDI);
- ◆ ReportListener:
 - UpdateListener (обеспечивает обратную связь с пользователем во время выполнения отчета);
 - UtilityReportListener (манипулирует таблицами конфигурации и выходными файлами):
 - DebugListener (обеспечивает отладку отчета);
 - XMLListener (обеспечивает вывод в формате XML);
 - XMLDisplayListener (настраивает XML-установки);
 - HTMLListener (производит HTML-представление отчета).

Переменные отчета доступны только в указанной вами сессии данных. Если вы хотите, чтобы они были видны не только в определенной сессии данных вашего отчета, вы должны позаботиться об этом.

Свойства, методы и события объекта *ReportListener*

Свойства, методы и события объекта *ReportListener* представлены в табл. 13.8—13.10.

Таблица 13.8. Некоторые важные свойства объекта *ReportListener*

Свойство	Описание
<code>AllowModalMessages</code>	Определяет, может ли <i>ReportListener</i> обеспечить модальные сообщения как часть своего интерфейса пользователя
<code>CommandClauses</code>	Обеспечивает получение списка опций команд <code>REPORT FORM</code> или <code>LABEL</code>
<code>Comment</code>	Хранит информацию об объекте
<code>CurrentDataSession</code>	Предоставляет <i>ReportListener</i> идентификатор сессии данных (<code>datasession ID</code>) отчета во время исполнения команд <code>LABEL</code> или <code>REPORT FORM</code> . Default Data Session=1
<code>CurrentPass</code>	Определяет текущий проход по данным отчета
<code>DynamicLineHeight</code>	Указывает, должен ли <i>ReportListener</i> использовать стандартное пространство между строками GDI+ или фиксированное пространство между строками
<code>FRXDataSession</code>	Содержит ID сессии данных
<code>GDIPlusGraphics</code>	Обеспечивает доступ к объекту <code>GDIPlusGraphics</code>
<code>ListenerType</code>	Определяет тип объекта вывода -2 Debug Listener 0 Print Listener 1 Preview Listener 4 XML Listener 5 HTML Listener
<code>Name</code>	Определяет имя для доступа к объекту
<code>OutputPageCount</code>	Число страниц, которое было обработано и к которым можно обратиться, используя метод <code>OutputPage</code>
<code>OutputType</code>	Обеспечивает значение, которое механизм отчетов передает <code>_REPORTOUTPUT</code> , чтобы представлять во фразе <code>OBJECT TYPE</code> в командах <code>LABEL</code> или <code>REPORT FORM</code>
<code>PageNo</code>	Представляет текущий номер страницы для вывода в отчете
<code>PageTotal</code>	Общее количество страниц отчета
<code>PrintJobName</code>	Показывает определенное пользователем значение как имя текущего печатаемого документа в диалоге Windows Print Queue

QuietMode	Определяет, сможет ли ReportListener предоставлять любую обратную связь пользователю или интерфейсу. .T. — обратная связь с пользователем разрешена
-----------	--

Таблица 13.8 (окончание)

Свойство	Описание
SendGDIPlusImage	Определяет, будет ли источник изображения для ReportListener загружен по объектной ссылке или из поля General. Не предназначено для работы с изображениями, хранящимися в файлах
Tag	Хранит всю вспомогательную информацию, необходимую для программы
TwoPassProcess	Определяет, требует ли отчет нескольких проходов через область данных. По умолчанию .F.

Таблица 13.9. События объекта ReportListener

Событие	Описание
AdjustObjectSize	Происходит <i>перед</i> тем, когда объект ReportListener начинает обрабатывать объекты отчета, такие как Shape или Picture
AfterBand	Происходит непосредственно после того, как Система отчетов завершит обработку полосы отчета
AfterReport	При завершении отчета, после заключительного события AfterBand, движок отчета запускает событие AfterReport. В это время собственный сеанс данных (data session), указанный свойством FRXDataSession, все еще содержит копию только для чтения файла определения отчета (frx). Задание по выводу на печать (если отчет печатается) является все еще открытым
BeforeBand	Происходит непосредственно перед тем, как система отчетов начинает обрабатывать область отчета
BeforeReport	Происходит непосредственно перед тем, как движок отчета начинает обрабатывать форму отчета
Destroy	Происходит, когда освобождается объект
Error	Происходит, когда появляется ошибка метода в run time. Вы можете добавить код для обработки события Error так, чтобы объект мог обрабатывать ошибки
EvaluateContents(nFRXReco, oObjProperties)	Происходит перед тем, как ReportListener начнет предоставлять объекты для полосы
Init	Имеет место при создании объекта

LoadReport	Происходит до выполненного отчета, перед событием BeforeReport. Обеспечивает доступ к опциям команды отчета
UnloadReport	Происходит после окончания выполнения отчета

Обратите внимание на событие EvaluateContents. В него передается ссылка на текущий обрабатываемый объект (только типа Filed). Через свойства этого объекта Text Value, Fontname, FontStyle и пр. можно менять вид поля типа Filed.

```
WITH toObjProperties
IF empty(.Text)
.Text = "----"
.Reload    = .T.
ENDIF
ENDWITH
```

Таблица 13.10. Методы объекта ReportListener

Метод	Описание
CancelReport	Позволяет пользователю или программе аварийно завершать отчет, подобно нажатию <Esc>
ClearStatus	Удаляет элементы интерфейса, подготовленные методом DoStatus
DoMessage	Предоставляет модальный диалоговый блок, когда AllowModalMessages установлен в True (.T.), в противном случае вызывает DoStatus
GetPageHeight	Возвращает высоту страницы отчета, кратную 960 точек на дюйм (dpi)
GetPageWidth	Возвращает ширину страницы отчета в dpi, кратных величине 960
IncludePageInOutput	Позволяет ReportListener определять диапазон страниц вывода
OnPreviewClose	Происходит, когда пользователь закрывает окно предварительного просмотра отчета
OutputPage	Обеспечивает доступ к текущей странице или полному диапазону страниц для выполнения отчета, в зависимости от значения свойства ListenerType

Render Render(nFRXRecno, nLeft, nTop, nWidth, nHeight, nObjectContinuationType, cContentsToBeRendered, GDIPlusImage)	Вызывается для каждого объекта визуализации. Некоторые аргументы подходят только для некоторых типов объектов формата. nFRXRecno nLeft nTop nWidth nHeight nObjectContinuationType cContentToBeRendered GDIPlusImage
SupportsListenerType	Определяет независимо от ReportListener поддержку конкретного типа построения и отображения отчета

Таблица 13.10 (окончание)

Метод	Описание
UpdateStatus	Запускает изменение информации в элементах пользовательского интерфейса, поставляемое методом DoStatus

Приведем далее примеры кода.

Пример 1

```
** Определение Preview Listener
oListener=NewObject("ltListener")
oListener.ListenerType=PreviewListener
```

Пример 2

```
** Создание нового HTML-приемника и установка его как "приемника" вашего
Preview-приемника
DO (_ReportOutput) WITH HTMLListener
oListener.Successor = _oReportOutput[TRANSFORM(HTMLListener)]
_oReportOutput[TRANSFORM(HTMLListener)].TargetFileName=("c:\output\MyHTML")
```

Пример 3. Вывод отчета в виде HTML

```
loHTMLListener = .NULL.
do (_reportoutput) with 5, loHTMLListener
loHTMLListener.TargetFileName = 'MyReport.html'
loHTMLListener.QuietMode      = .T.
loListener = newobject('_ReportListener', home() + 'ffc\_reportlistener.vcx')
loListener.ListenerType = 1
loListener.Successor      = loHTMLListener
report form EmployeesMD object loListener
```

Пример 4. Объединение двух отчетов

```
REPORT FORM (lcReportFullName1) OBJECT loListener NOPAGEEJECT NOWAIT
REPORT FORM (lcReportFullName2) OBJECT loListener NORESET NOWAIT
```

Пример 5. Вертикальное отображение текста

```
LOCAL oListener
oListener = CREATEOBJECT("rotateText")
oListener.ListenerType = 1
REPORT FORM ? OBJECT oListener

#define FRX_OBJCOD_PAGEHEADER 1

DEFINE CLASS rotateText AS ReportListener

    IsInPageHeader = .F.

    PROCEDURE Init()
        DECLARE integer GdipRotateWorldTransform In GDIPlus.Dll ;
            integer graphics, single angle, integer enumMatrixOrder_order
        DECLARE integer GdipTranslateWorldTransform In GDIPlus.Dll ;
            integer graphics, single dx, single dy, ;
            integer enumMatrixOrder_order
        DECLARE integer GdipSaveGraphics IN GDIPlus.DLL ;
            integer graphics, integer @xx
        DECLARE integer GdipRestoreGraphics IN GDIPlus.DLL ;
            integer graphics, integer xx
    ENDPROC

    PROCEDURE BeforeBand(nBandObjCode, nFRXRecNo)
        DODEFAULT(nBandObjCode, nFRXRecNo)
        IF (nBandObjCode = FRX_OBJCOD_PAGEHEADER)
            THIS.IsInPageHeader = .T.
        ENDIF
    ENDPROC

    PROCEDURE AfterBand(nBandObjCode, nFRXRecNo)
        IF (nBandObjCode = FRX_OBJCOD_PAGEHEADER)
            THIS.IsInPageHeader = .F.
        ENDIF
        DODEFAULT(nBandObjCode, nFRXRecNo)
    ENDPROC

    PROCEDURE Render(nFRXRecNo, nLeft, nTop, nWidth, nHeight, ;
        nObjectContinuationType, cContentsToBeRendered, GDIPlusImage)
        LOCAL xx, x, y, z
        xx = 0
        IF THIS.IsInPageHeader
            * get appropriate versions of coords
```

```

        x = nLeft
        y = nTop

        * save the current state of the graphics handle
        z = GdipSaveGraphics(this.GDIPlusGraphics, @xx)

        * now move the 0,0 point to where we'd like it to be
        * so that when we rotate we're rotating around the
        * appropriate point
        z = GdipTranslateWorldTransform(this.GDIPlusGraphics,x,y,0)
        * should check z here -- will be 0 if no error occurred

        * now change the angle at which the draw will occur
        z = GdipRotateWorldTransform ( this.GDIPlusGraphics,-20,0)
        * should check z as above

        * restore the 0,0 point
        z = GdipTranslateWorldTransform(this.GDIPlusGraphics,-x,-y, 0)
        * should check z as above

    ENDIF

    * explicitly call the baseclass behavior
    * when we are ready for it
    DODEFAULT(nFRXRecNo, nLeft, nTop, nWidth, nHeight, ;
              nObjectContinuationType, cContentsToBeRendered, ;
              GDIPlusImage)

    * put back the state of the graphics handle
    IF THIS.IsInPageHeader
        GdipRestoreGraphics(this.GDIPlusGraphics, xx)
    ENDIF

    * don't let the baseclass render
    * when and how it would otherwise do it
    NODEFAULT

ENDPROC

ENDEDEFINE

```