

## Лабораторная Работа № 2

### Средства Ввода и редактирования данных и создания меню приложения

Цель лабораторной работы:

1. Изучение возможностей СУБД по вводу и редактированию данных;
2. Построение вариантов форм для ввода и редактирования данных;
3. Разработка формы для заданного варианта

В Visual FoxPro существует два формата отображения данных таблицы – в виде таблицы и в виде формы.

Привычный способ отображения данных в таблицах с помощью форм, которые позволяют как вводить данные, так и редактировать их.

Для создания экранных форм используются:

- Автоформат (Auto Form Wizard), создающий форму для одной таблицы, с расположенными по умолчанию полями. Выделите имя таблицы в проекте, нажмите кнопку Auto Form Wizard на главном меню.
- Мастер форм (Form Wizard) для одной или двух связанных таблиц. Задаются формы полей, стиль их отображения и тип кнопок управления.
- Построитель (Builder), позволяющий создать объекты с определенными свойствами.
- Конструктор форм (Form Designer), в котором вы самостоятельно разрабатываете собственные формы с заданными свойствами для просмотра, ввода и редактирования данных.

Самый простой способ создания формы для одной таблицы является **автоформат**. В конструкторе проекта необходимо установить курсор мыши на таблицу, для которой вы создаете форму, и нажать на пиктограмму **Autoform Wizard** стандартной панели инструментов. Запускается мастер по созданию автоформы, и на экране появляется готовая форма (Рис.1).

Если созданная форма вас не полностью устраивает, нажмите пиктограмму **Modify Form** на стандартной панели инструментов для перехода в режим конструктора формы. Находясь в окне конструктора форм, вы можете улучшить созданную форму и сохранить ее командой меню **File → Save as** в вашей папке.

Форму, созданную одним из этих способов, надо включить в ваш проект:

Откройте вкладку **Documents** окна **Project Manager**

Нажмите кнопку **Add**,

Найдите папку, где находится форма,

Нажмите кнопку **OK**.

В проект будет включена созданная вами форма.

Также достаточно просто можно создать форму с помощью мастера. В лабораторной работе мы используем мастер для создания однотабличной формы.

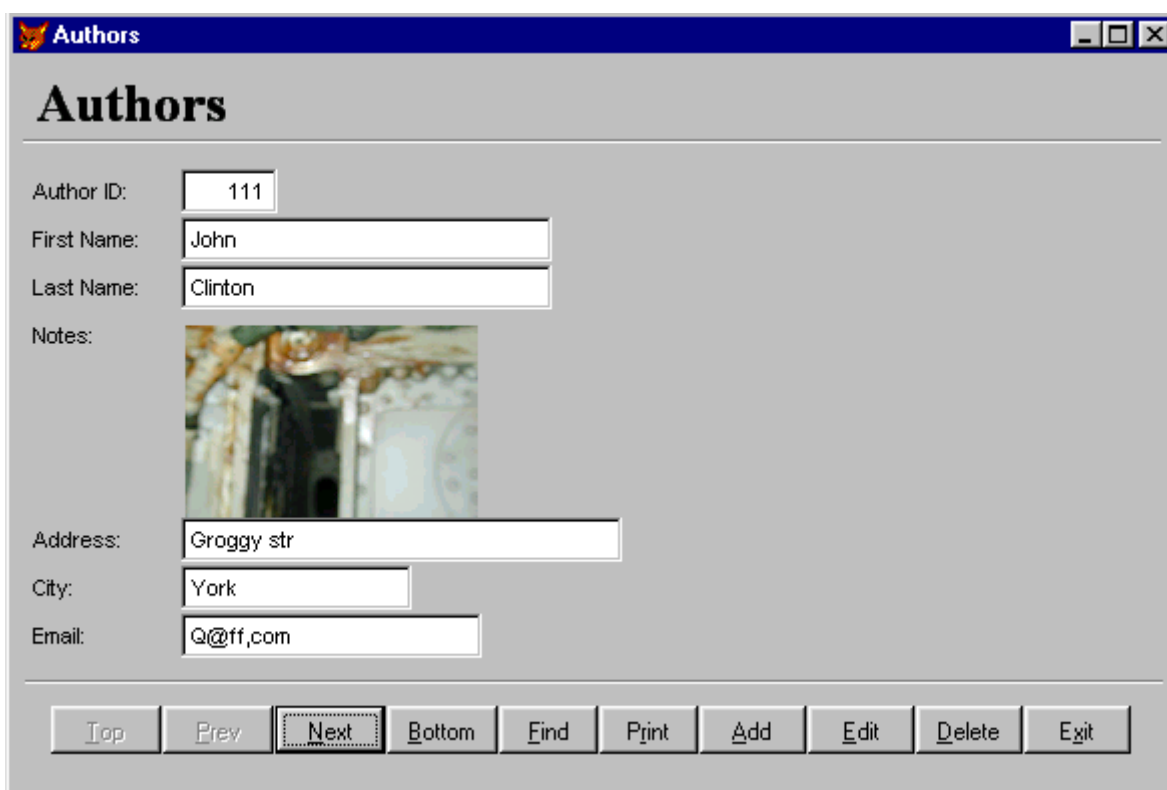
Экранная форма, созданная с помощью мастера, предназначена для просмотра и модификации данных таблиц. Эта форма содержит ряд кнопок для выбора режима просмотра, редактирования, печати данных.

В качестве примера запустим мастер форм из окна проекта. Для этого выполните следующие действия:

1. Откройте проект с именем вашего варианта.

2. Нажмите ярлык **Docs** для перехода на вкладку **Docs**. Затем выберите группу **Forms** и нажмите кнопку **New**, в результате чего откроется окно диалога **New Form**.

3. Выберите опцию **Form Wizard** (Мастер форм). Следуйте подсказкам Мастера.



The screenshot shows a window titled "Authors" with a blue header bar. Below the header, the title "Authors" is displayed in a large, bold font. The form contains several input fields: "Author ID" with the value "111", "First Name" with "John", "Last Name" with "Clinton", "Notes" with a small image of a classical building, "Address" with "Groggy str", "City" with "York", and "Email" with "Q@ff.com". At the bottom of the form, there is a horizontal bar with ten buttons: "Top", "Prev", "Next", "Bottom", "Find", "Print", "Add", "Edit", "Delete", and "Exit". The "Next" button is highlighted with a dashed border.

Рис. 1. Пример экранной формы для таблицы Authors

Для модификации формы в Конструкторе проекта выделите имя формы и нажмите кнопку **Modify**. Все поля формы доступны для модификации.

Рассмотрим основные элементы форм.

## Размещение текстовой информации на форме

Размещение текста в экранной форме осуществляется с помощью инструмента **Label**, который находится на панели инструментов **Form Controls**. Под текстом понимается любая текстовая информация: заголовки, наименования полей и поясняющая информация.

Для размещения текста выполните следующие действия:

1. Выберите инструмент **Label** на панели инструментов **Form Controls**. Если данная панель отсутствует на экране, для ее отображения выполните команду меню **View → Form Control Toolbar**.

2. Установите указатель мыши на место предполагаемого расположения текстового объекта и, удерживая кнопку мыши в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемого размера. В дальнейшем вы сможете скорректировать размер объекта.

3. Для открытия окна свойств созданного объекта выделите его и выполните команду меню **View → Properties**. На экране раскроется окно **Properties**.

4. Если вы хотите, чтобы фон текста не отличался от фона экранной формы, установите для свойства **BackColor** значение **Transparent**.

5. Текстовая информация задается свойством **Caption**. Выделите курсором данное поле, после чего в поле ввода свойства введите нужную информацию. Для переноса информации в поле свойства нажмите клавишу **Enter**.

## Размещение полей ввода на форме

Следующим шагом в создании экранной формы является добавление в нее полей различных типов. Наиболее простым типом поля является поле ввода. Для размещения поля ввода в экранной форме выполните следующие действия:

1. Выберите инструмент **Text Box** на панели инструментов **Form Controls**.

2. Установите указатель мыши на место, в котором вы предполагаете разместить поле. Удерживая кнопку мыши в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемого размера.

3. Для открытия окна свойств созданного объекта выделите его и выполните команду меню **View | Properties**. На экране раскроется окно **Properties**.

4. Чтобы связать созданное поле с полем таблицы, выберите свойство **Control Source** из вкладки **Data**. В поле ввода свойства воспользуйтесь кнопкой раскрытия списка и из списка всех полей открытой таблицы выберите поле, которое хотите добавить в форму.

5. Установите свойство **Alignment**, позволяющее задать несколько вариантов выравнивания информации, отображаемой в поле: по центру, по левому или правому краю поля.

## Ввод изображений – объектов типа General.

1. Для добавления графического поля типа **General** в экранную форму нажмите кнопку **OLE Bound Control** на панели инструментов **Form Controls**.

2. После выбора инструмента установите указатель мыши на то место, где вы хотите добавить поле для изображения. Удерживая кнопку мыши в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемого размера.

3. Для открытия окна свойств созданного объекта выделите его и выполните команду меню **View | Properties**. На экране раскроется окно **Properties**.

4. Чтобы связать созданное поле с полем таблицы, выберите свойство **ControlSource**. В поле ввода свойства воспользуйтесь кнопкой раскрытия списка и из списка всех полей открытой таблицы с вашим именем выберите поле типа **General** с именем **Notes**, которое будет добавлено в форму.

1. Запустите созданную экранную форму на выполнение, выбрав команду меню **Form → | Run Form**. Вид формы с полем типа **General** представлен на рис 4. Если рисунок не помещается в рамке целиком, увеличьте в конструкторе форм размер рамки. Для масштабирования рисунка в окне **Properties** можно использовать свойство **Stretch** закладки **Layout**. Выберите значение **Stretch**, чтобы рисунок целиком появился в окне формы.

### Создание раскрывающегося списка ComboBox

Рассмотрим еще один тип объектов, предназначенных для отображения на экране элементов списка — объект типа **ComboBox** или раскрывающийся список. (Рис.2)

Рассмотрим создание раскрывающегося списка для редактирования поля **Author\_id** таблицы **Bookauth**. Значения этого поля должны вводиться из таблицы **Authors**.

1. Откройте экранную форму **Bookauth** в окне конструктора форм.

2. Удалите список для ввода значения поля **Author\_id**, созданный в Автоформате.

3. Выберите инструмент **Combo Box** на панели инструментов **Form Controls**.

4. Установите указатель мыши на место удаленного объекта. Удерживая кнопку мыши в нажатом состоянии, переместите курсор по диагонали так, чтобы получилась рамка требуемых размеров.

5. Откройте окно **Properties** размещенного в форме раскрывающегося списка.

2. Свяжите раскрывающийся список с полем **Author\_id** таблицы **Authors**.

Нажмите правой кнопкой мыши на выделенное поле **Combo Box**.

Из открывшегося меню выберите пункт **Builder**.

Откроется окно (Рис. 3)

В закладке **List Items** укажите имя таблицы **Authors** и имя поля **Author ID** откуда будут выбираться значения для списка.

В закладке **Value** укажите поле, куда записываются значения из этого окна (Рис.3)

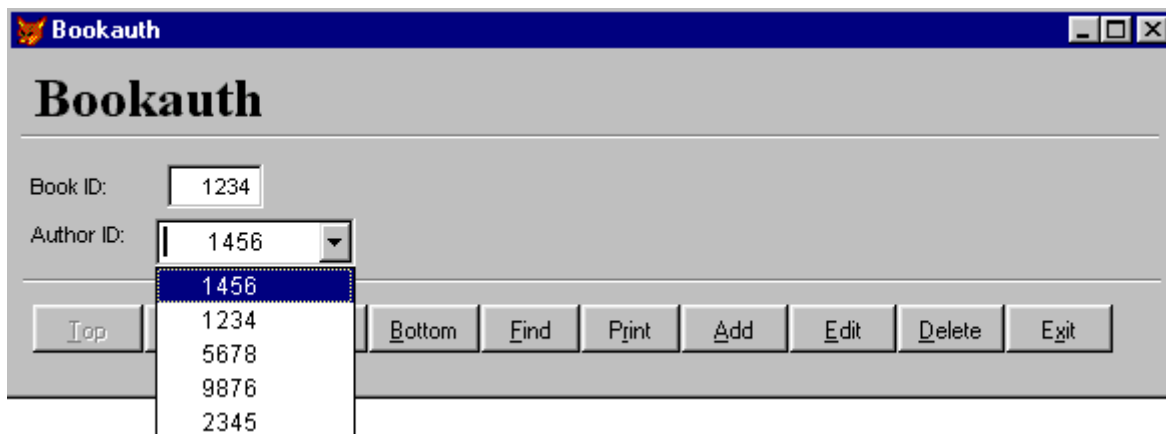


Рис. 2. Форма с полем Author ID типа Combo Box

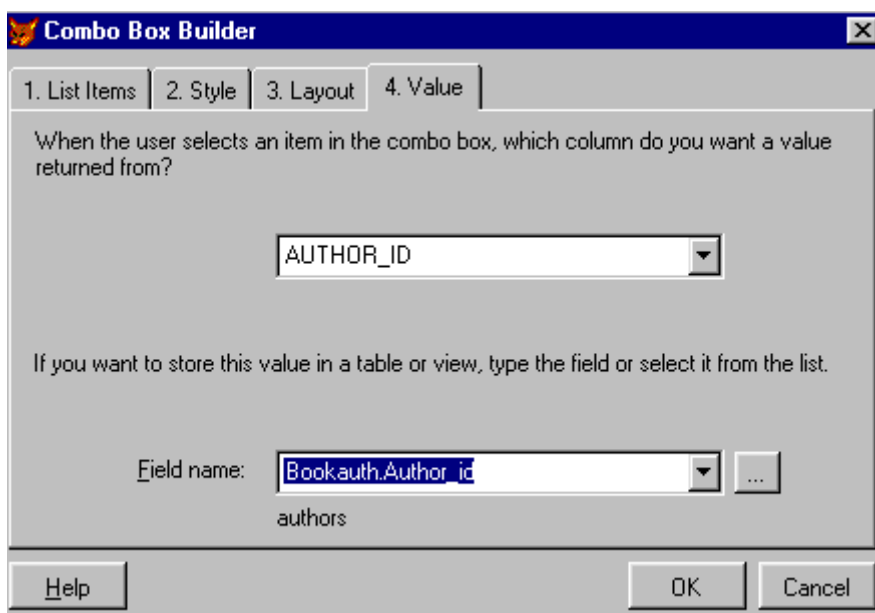


Рис. 3. Построитель поля типа ComboBox

### Создание кнопок управления

В приведенных выше формах кнопки управления выполнены стандартно с помощью автоформата. Можно создать кнопки с определенными функциями.

Для создания кнопок можно использовать два инструмента на панели инструментов Form Controls: Кнопку Command Button для одиночных кнопок и кнопку Command Group для групповых.

Для построения одиночной кнопки:

1. Выберите инструмент Command Button на панели инструментов Form Controls и поместите кнопку в требуемом месте.
2. Откройте окно свойств созданного объекта Command.

3. На кнопке можно расположить текст или графическое изображение. При расположении текстовой информации скорректируйте свойство **Caption**, разместив в поле ввода значения текст, который будет отображаться на кнопке.

4. Для расположения на кнопке графического изображения выберите свойство **Picture** и нажмите кнопку, расположенную правее поля ввода. В результате откроется окно диалога **Open**. В окне диалога вы можете выбрать файл с графическим изображением и просмотреть его вид в области **Picture**. Для просмотра файла нажмите кнопку **Preview**. После выбора файла нажмите кнопку **OK** для перенесения его на кнопку.

5. Кнопка создана. Осталось определить команды, которые будут выполняться при нажатии кнопки. Для этой цели служит метод объекта **Click**, который автоматически вызывается при нажатии на созданную кнопку мышью. Для отображения в окне свойств объекта списка всех методов перейдите на вкладку **Methods**.

6. Для определения текста метода установите курсор на метод **Click** и два раза нажмите кнопку мыши. На экране откроется окно процедур для описания функции **Click**. В текстовой области окна разместите команды, которые должны выполняться при нажатии на данную кнопку. Например, при нажатии на кнопку выхода из формы на экран будет выдаваться запрос о том, хотите ли вы действительно выйти из формы. При утвердительном ответе форма будет закрыта. Для выполнения этих действий используется функция **Visual FoxPro MESSAGEBOX**. Текст процедуры обработки события нажатия на кнопку определяется вариантом задания.

### Создание меню приложения

Система имеет средства для создания меню приложения

Горизонтальное меню, расположенное в верхней части экрана, называется *строкой меню*.

Для запуска **Menu Designer** существует несколько способов.

- Выберите команду **File → New** и в окне **New** – опцию **Menu**.
- Выполните в командном окне команду **Create Menu**.
- Если запущен **Project Manager**, то во вкладке **Other** выделите элемент **Menu** и щелкните **New**.

Откроется диалоговое окно Конструктора Меню (**Menu Designer**) (Рис. 10), в котором вы наберете пункты вашего меню. На рисунке показан готовящийся первый пункт меню, например, Ввод данных. В первой колонке **Prompt** вы указываете название пункта меню, во второй **Result** из выпадающего меню выберите то, что будет выполняться при нажатии на этот пункт меню. В нашем случае это – **Command**, выполнится команда, набранная в следующей колонке, например, **do form Samolet**. Это означает, что при нажатии на этот пункт меню, запустится форма **Самолет**.

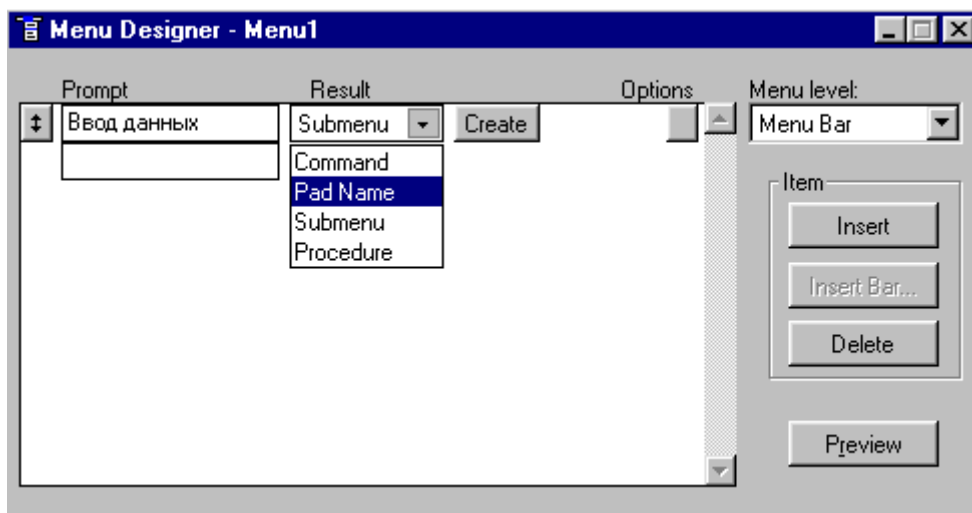


Рисунок 10. Окно Конструктора Меню

### Выполнение лабораторной работы

Для вариантов базы данных, созданной в первой лабораторной работе, разработайте формы для всех таблиц, а также главную форму (Рис.11), по которой вы будете входить в базу данных и, с помощью которой автоматически будет вноситься в таблицы сегодняшняя дата и код оператора.

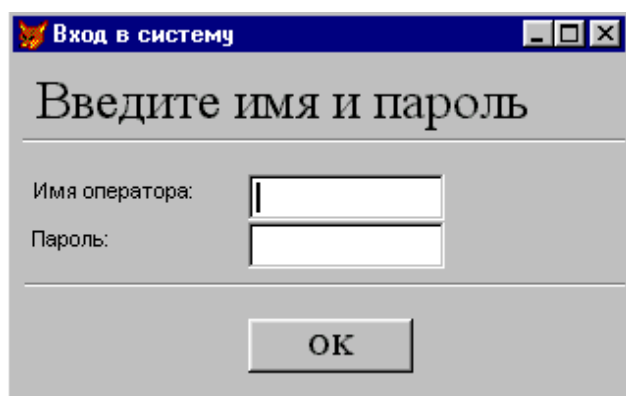


Рис. 11 Форма для проверки наличия оператора и его пароля

Главная форма должна содержать поля фамилии оператора и его пароль, которые хранятся в главной таблице. Создайте сначала эту таблицу с именем Operators, которая содержит два поля с именами operatname и password и введите туда фамилии операторов и их пароли. При нажатии кнопки ОК главной формы, система должна проверить есть ли в таблице указанная фамилия и правильно ли набран пароль. Примерное содержание функции Click кнопки ОК:

```
Local cArr1(1), cArr2(1)@@ имена локальных переменных
select * from operators where operatname ==
ThisForm.text1.text into array cArr1
if vartype(cArr1(1)) == "L"
    cErrorMessage = "Неверное имя"
do form "Сообщение об ошибке"
```

```

else
    select password from operators where operatname
== ThisForm.text1.text into array cArr1
    if alltrim(cArr1(1)) != alltrim(ThisForm.text2.text)
        cErrorMessage = "Неверный пароль"
        do form "Сообщение об ошибке"
        _screen.ActiveForm.Release()
    else
        do m2.mpr
        _screen.ActiveForm.Release()
    endif
endif
endif

```

При нажатии на кнопку ОК формы выполняется эта программа проверки правильности фамилии и пароля оператора, а также запускается программа формирования меню (do m2.mpr). Переменные ThisForm.text1.text и ThisForm.text2.text – это обозначения соответственно первого и второго текстовых полей этой формы. Переменные operatname и password – названия полей таблицы operators, в которых хранятся фамилия и пароль оператора. Названия полей могут быть заменены данными варианта.

Форма «Сообщение об ошибке» выглядит так:



Создайте ее при выполнении работы. В ней одно поле – текстовое, у которого свойство ControlSource имеет название cErrorMessage и кнопка ОК с содержанием: \_screen.ActiveForm.Release().

Для восстановления основного меню наберите в командном окне:

```

set sysmenu to default @@ восстановить системное
меню

```

### Варианты заданий

Создать главную форму и все формы, необходимые для ввода и просмотра данных в соответствии с вариантом, указанным в первой лабораторной работе. Главная форма должна запустить меню, в котором будут созданы пункты запуска всех форм варианта и, при нажатии на который, форма будет выполняться.

1. Разработать формы: Кодификатор особых случаев эксплуатации, Инцидент, Операторы. Введите из формы 10-12 записей в таблицы.

2. Разработать формы: Ремонтный завод, Ремонт, Самолет, Операторы. Введите 10-12 записей в таблицы.



3. Разработать формы: Отдел\_универмага, Продавец, Товар, Продажи. Введите из формы 10-12 записей в таблицы.
4. Разработать формы: Кафедра, Преподаватель, Дисциплины, Расписание. Введите из формы 10-12 записей в таблицы.
5. Разработать формы: Работник Автосервиса, Автомобиль на ремонте, Запчасти, Оператор. Введите из формы 10-12 записей в таблицы.
6. Разработать формы: Кодификатор Зон контроля, Самолет, Дефекты, Оператор. Введите из формы 10-12 записей в таблицы.
7. Разработать формы: Автор, Книга, Издательство, Издания. Введите из формы 10-12 записей в таблицы.
8. Разработать формы: Карта, Врач, Прием, Болезнь. Введите из формы 10-12 записей в таблицы.
9. Разработать формы: Промежуточные пункты маршрута (ППМ), Трасса, Маршрут. Введите из формы 10-12 записей в таблицы.
10. Разработать формы: Авиакомпания, Вылет, Прилет, Аэропорт. Введите из формы 10-12 записей в таблицы.
11. Разработать формы: Студенты, Дисциплины, Преподаватель, Оценки. Введите из формы 10-12 записей в таблицы.
12. Разработать формы: Авиакомпания, Расписание, Рейс, Список пассажиров. Введите из формы 10-12 записей в таблицы.

### Отчет

Отчет по лабораторной работе должен содержать:

Вариант задания

Краткое описание выполненной работы.

Окна Конструкторов, использованных в работе.

Изображения всех форм с комментариями и содержанием программ кнопок.

Изображение структуры базы данных.

Распечатку введенных записей базы данных.

### Контрольные вопросы

Какова последовательность проектирования базы данных?

Какие основные компоненты таблицы базы данных?

Как определяются индексы таблицы?

Как определить поле, необязательное для заполнения?

Как устанавливаются связи между таблицами?

Как реализовать связь между таблицами с одинаковыми индексами?

Как ввести изображение в таблицу из формы?