

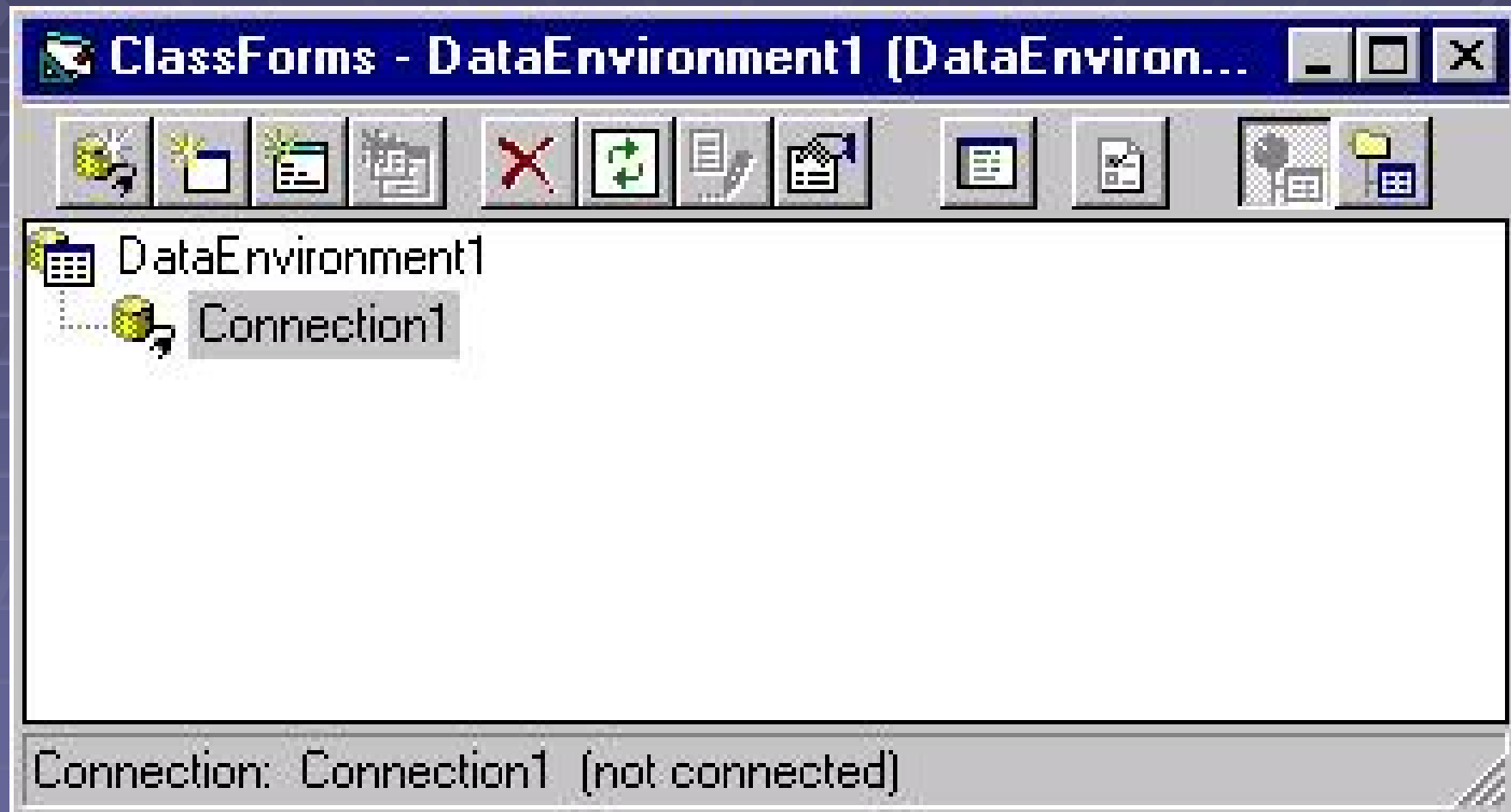
Visual Basic - связь с базами данных. Формы

Конспект лекций, прочитанный на
кафедре ФМЭГ
ФТ-факультета НТУ «ХПИ»
к.ф.-м.н. доцентом
Шкалето Владимиром Ивановичем

Создание соединений с серверами баз данных

Visual Basic позволяет использовать базы данных, размещенные на сервере. Прежде чем работать с такой базой данных, необходимо создать соединение с ней. Для создания соединения используется окно «**DataEnvironment**». Соединение создается при выполнении следующих действий:

Создание соединений с серверами баз данных

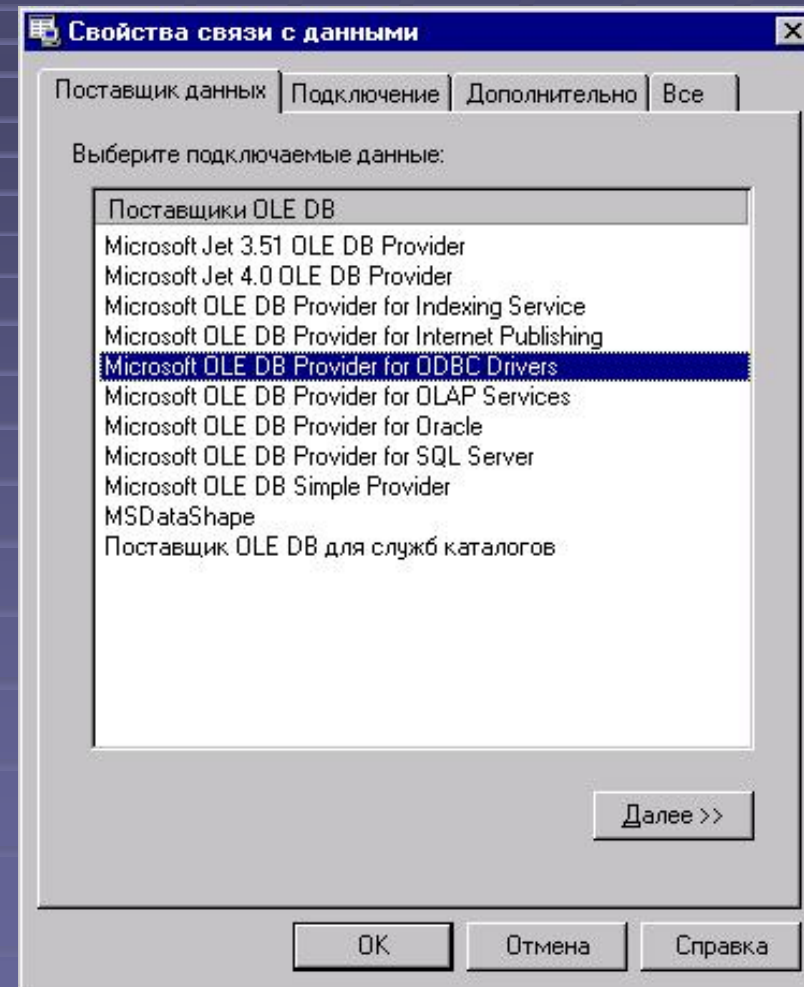


Создание соединений с серверами баз данных

1. Откройте окно «**DataEnvironment**». Для этого в меню «**Project**» (Проект) выберите команду **Add Data Environment** (Среда окружения). В окне свойств можно изменить ее имя.
2. Для добавления нового соединения нажмите кнопку «**Add Connection**» (Добавить соединение) на панели инструментов окна «**DataEnvironment**». В окне появится новый элемент, который по умолчанию имеет наименование «**Connection 1**». В окне свойств можно изменить его имя.

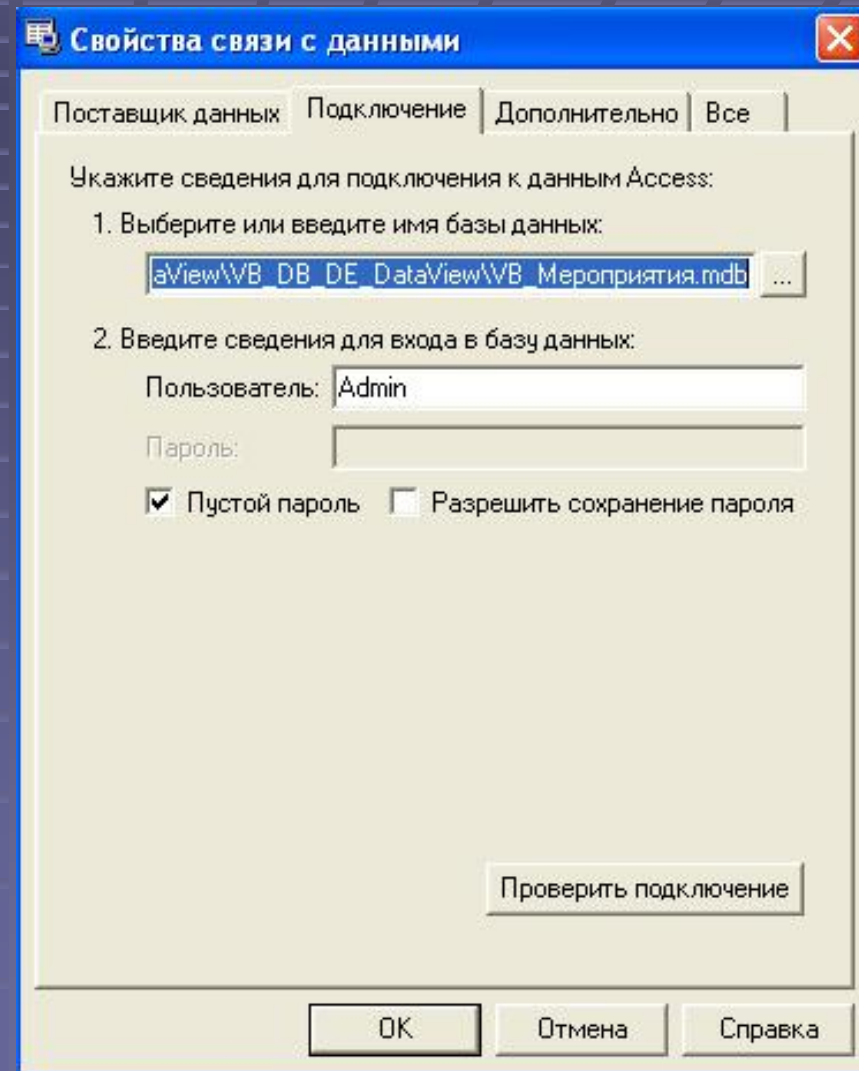
Создание соединения

3. Выберите для данного соединения команду контекстного меню «**Properties**» (**Свойства**).
Открывается диалоговое окно «**Свойства связи с данными**», предназначенное для ввода параметров соединения с сервером.



Создание соединений с серверами баз данных

4. На вкладке «**Поставщик данных**» диалогового окна «**Свойства связи с данными**» выберите из предложенного списка тип подключаемых данных.



Создание соединений с серверами баз данных

6. Перейдите на вкладку «**Подключение**» и введите **имя сервера**, **имя и пароль пользователя** (если в этом есть необходимость), выберите **базу данных** на сервере. Чтобы убедиться в правильности задания параметров соединения, нажмите кнопку «**Проверить подключение**». Если подключение выполнено успешно, появится сообщение, что проверка подключения выполнена.

7. Завершив установку параметров, нажмите кнопку **ОК** для закрытия диалогового окна.

Просмотр данных в открытой базе

Для просмотра входящих в соединение объектов откройте окно «**Data View**». Для этого в меню «**View**» (**Вид**) выберите команду «**Data View Window**» (**Окно просмотра данных**) или нажмите одноименную кнопку на стандартной панели инструментов **Visual Basic**.

Data View. Просмотр таблиц

При открытии таблицы в окне **Data View** появляется структура таблицы. Для открытия таблицы в конструкторе (только базы типа ODBC) установите курсор на наименование просматриваемой таблицы, нажмите правую кнопку мыши и выберите из контекстного меню команду «**Design**» (**Конструктор**).

Data View. Просмотр таблиц

Выделив в окне **Data View** нужную таблицу, разверните ее. Выделив нужное поле, из контекстного меню выберите пункт «**Properties...**» (свойства). В появившемся окне «**Properties: Имя поля**» будет отображена таблица. Столбцы этой таблицы отображают свойства полей открытой в конструкторе таблицы. Их назначение описано ниже в таблице.

Data View. Просмотр таблиц

The screenshot displays the Microsoft Access 'Data View' window for the 'prVB_DataBase - DE_prVB_DataBase (DataEnvironment)'. The left pane shows the database structure with 'DE_prVB_DataBase' expanded, revealing 'Connections', 'prVB_DB_DE', and 'Commands'. The 'prVB_DB_DE' connection is selected, and its 'Tables' folder is expanded, listing 'ActivityList', 'Equipment', 'EquipmentList', 'Placements', 'Type_Activity', 'Type_Placement', 'WorkerList', 'Views', and 'Stored Procedures'. The 'Equipment' table is expanded, showing fields: 'CodEquipment', 'Equipment Name', 'Number', 'Price One unit', 'Purpose', and 'Amortization Quota'. The 'Amortization Quota' field is selected, and a 'Properties: Amortization Quota' dialog box is open. This dialog box contains a table of properties and values:

Property	Value
Datatype	adSingle
Length	0
Precision	7
Scale	0
Name	Amortization Quota
Owner	
ADODatatype	4

The status bar at the bottom of the window indicates 'Connection: prVB_DB_DE (connected)'.

Свойства полей таблицы в базе данных

Свойство	Описание
Column Name	Имя поля таблицы
Datatype	Тип данных поля таблицы
Length	Максимальный размер поля
Precision	Максимальное число десятичных знаков в числовом поле
Scale	Максимальное количество десятичных знаков после разделителя

Свойства полей таблицы в базе данных

Свойство	Описание
Allow Nulls	Указывает, допустимо ли для данного поля значение NULL
Default Value	Значение, вводимое в поле по умолчанию
Identity	Если в данном поле установлен флажок, то при добавлении в таблицу нового значения в поле автоматически добавляется уникальное значение
Identity Seed	Значение, присваиваемое первой записи таблицы. Если свойство не содержит никаких значений, то по умолчанию присваивается 1
Identity Increment	Шаг, с которым изменяется значение в данном поле

12.04.2013

к.ф.-м.н., доц. Шкалето Владимир Иванович

13

Data View. Просмотр таблиц

Можно открыть для просмотра любую из таблиц, находящихся в окне «**Data View**». Для этого установите курсор на наименование выбранной таблицы, нажмите **правую кнопку** мыши и выберите из **контекстного меню** команду «**Open**» (**Открыть**). Для открытия таблицы можно также **дважды щелкнуть** мышью на ее наименовании.

Data View. Просмотр таблиц

Откроется окно «**Run table**» (таблица) в котором можно **изменять данные в различных полях разных записей таблицы, а также добавлять новые записи.** Изображение этого окна приведено на нижеследующем рисунке.

После закрытия окна «**Run table**» введенные данные автоматически сохраняются в базе данных.

Data View. Просмотр таблиц

Run Table: Equipment

	CodEquipment	Equipment Name	Number	Price One unit	Purpose	Amortization Quota
▶	1	Стол	36	763	за ним едят	0.15
	2	Стул	793	256	на нем сидят	0.15
*						

Data View

- Data Links
- Data Environment Connections
 - VB_DB_DE
 - Tables
 - ActivityList
 - Equipment
 - EquipmentList
 - Placements
 - Type_Activity
 - Type_Placement
 - WorkerList
 - Views
 - Stored Procedures

Создание простых форм

1. Вначале необходимо в окне конструктора «**Data Environment**» создать команды – соответствующие таблицам, для которых создаются формы ввода данных.
2. Для этого из окна «**Data View**» перетащите с помощью мыши и технологии **DragDrop** все таблицы в окно конструктора «**Data Environment**». В окне конструктора появятся команды с именами соответствующих таблиц базы данных.

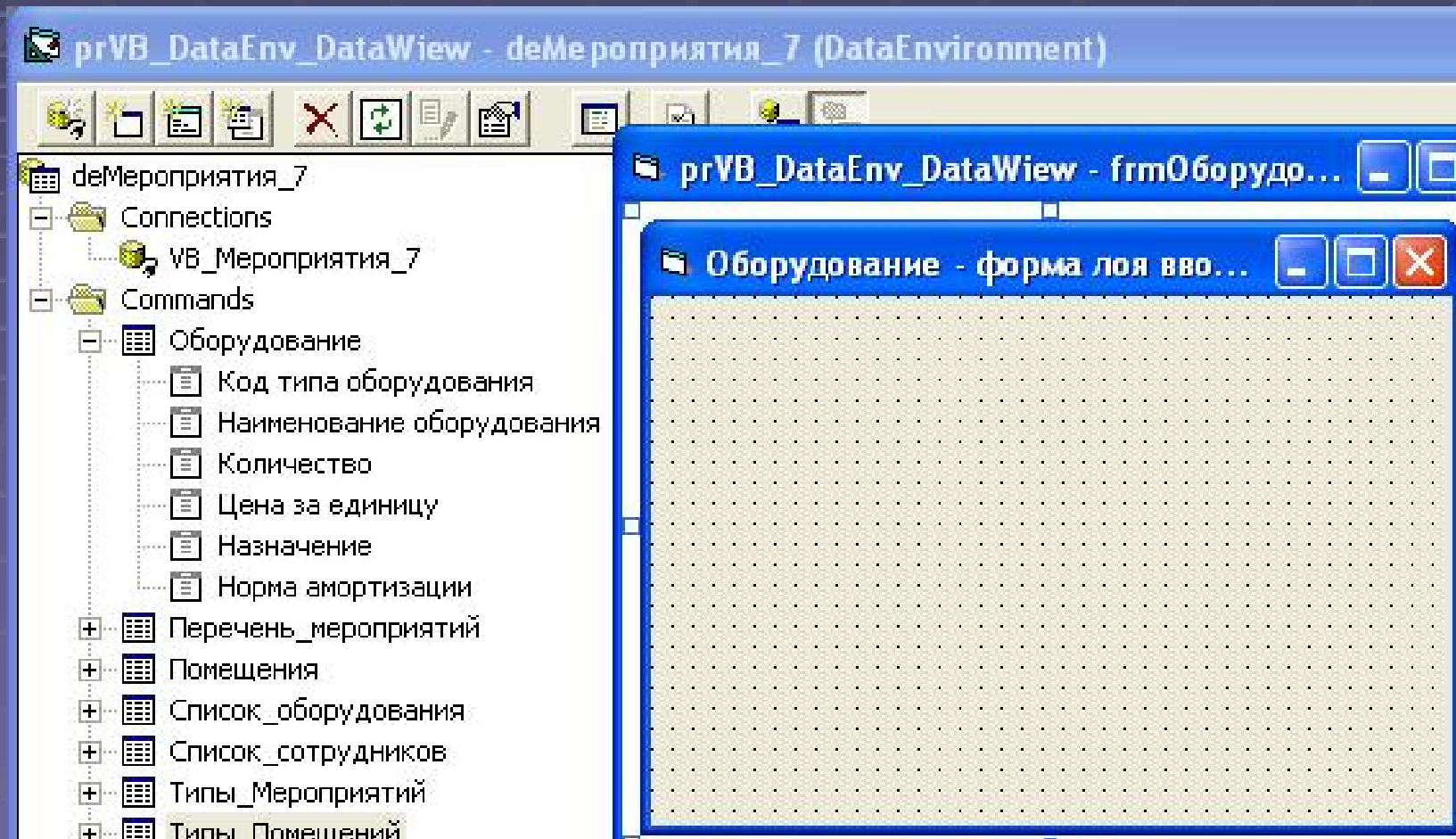
Создание простых форм

3. В окне конструктора «**Data Environment**» разверните какую-либо таблицу, чтобы были видны все её поля (если полей много, то измените соответствующим образом размеры окна конструктора).

4. В окне VB-проекта добавьте новую форму и задайте ей имя с префиксом **frm** и с основным именем – именем развернутой таблицы.

Создание простых форм

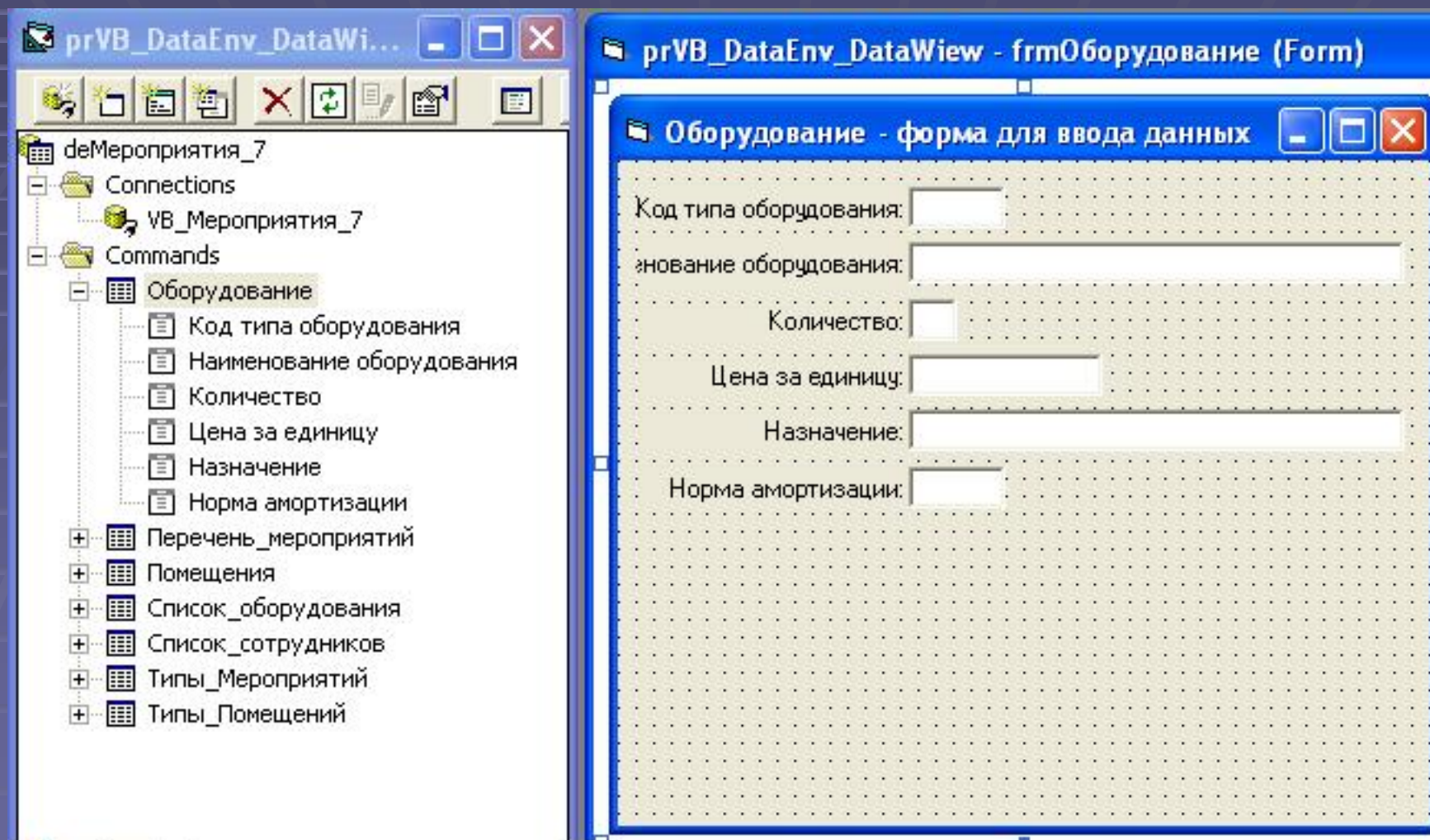
Измените размеры новой формы и конструктора «**Data Environment**» так, как показано на рисунке.



Создание простых форм

Затем перетащите команду (с именем таблицы) из конструктора «**Data Environment**» в верхнюю часть формы. При этом на ней разместятся несколько связанных с данными элементов управления «**TextBox**» и соответствующие им метки. Вид получившейся формы показан на следующем рисунке.

Создание простых форм



Создание простых форм

Можно настраивать элементы управления «**TextBox**» или «**FlexGrid**», при их связывании с данными. Если щелкнуть правой кнопкой мыши какое-либо поле в конструкторе «**Data Environment**» и выбрать команду «**Properties**», можно установить **соответствие** между этим полем и элементом управления любого типа.

Создание простых форм

При необходимости можно изменить оформление элементов управления: шрифты, размеры и т.п.

Для управления данными – переходам по записям, добавлением, удалением записей или их обновлением необходимо добавить на форму соответствующие кнопки и написать коды для обработки событий нажатия кнопок.

Примеры кодов перехода по записям

Коды для перемещения по записям имеют вид, аналогичный кодам в Access.

Дважды щелкните кнопку на форме для перехода в окно «**Code View**», где введите следующий код, например, для кнопки перемещения на одну запись вперед:

```
Private Sub cmdPrev_Click()  
If DataEnvironment1.rsCustomers.BOF Then  
    MsgBox «Вы находитесь в первой записи.»  
Else  
    DataEnvironment1.rsCustomers.MovePrevious  
End If  
End Sub
```


Пояснения к кодам:

Имя конструктора «**Data Environment**» – **DataEnvironment1**;

Имя набора данных (**Recordset**) – состоит из двух частей: префикса **rs** и имени таблицы (в примере это **Customers**).

Параметр **Recordset: BOF** - начало набора данных.

Добавление записей

Для добавления записи на форму необходимо добавить ещё две кнопки: «**Добавить**» (**Add**) и «**Обновить**» (**Update**). Коды для соответствующих процедур будут иметь вид:

DataEnvironment1.rsCustomers.AddNew

DataEnvironment1.rsCustomers.Update

Добавление записей

При нажатии на кнопку «**Добавить**» на форме очищаются все поля. Необходимо ввести значения во все поля, кроме счетчика, а затем нажать кнопку «**Обновить**». Как правило код (значение ключевого поля) новой записи появляется (почему-то) при следующем вызове формы (запуске проекта на выполнение).

Мастера в Visual Basic

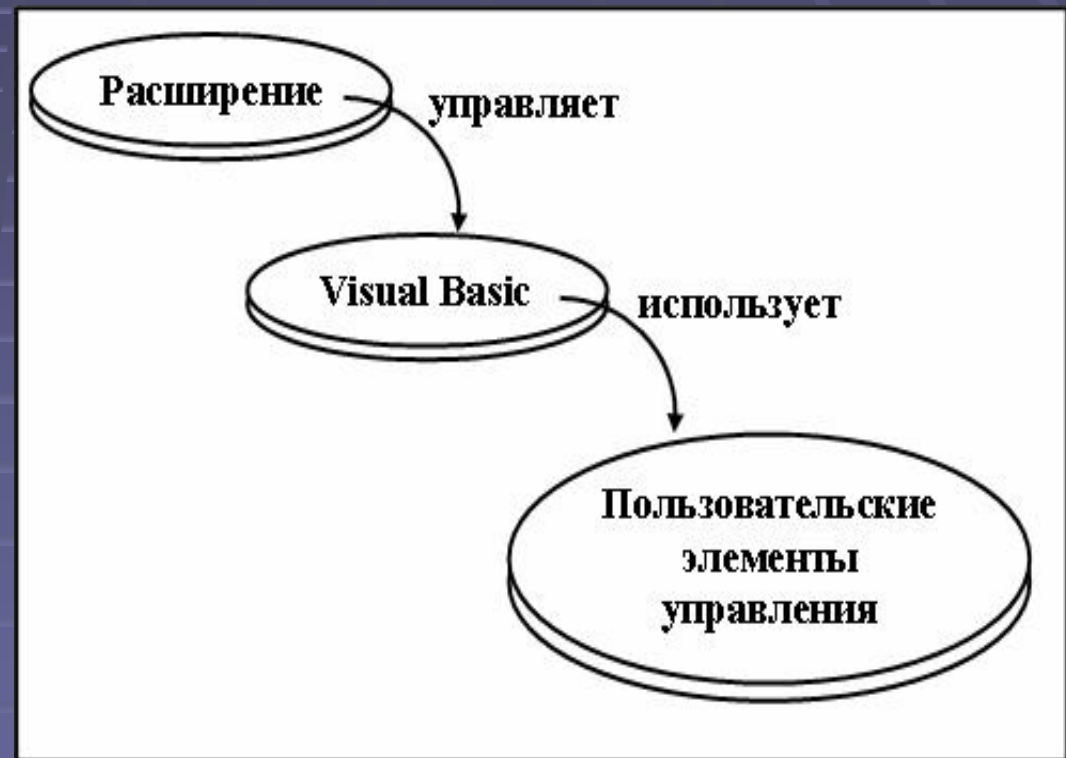
По своей сути, «**Мастера**» (**Wizard**) - это простые в использовании оболочки для автоматизации различных задач, начиная от простых, но утомительных, и заканчивая самыми сложными. Есть в VB «**Мастера**» на любой случай жизни: от создания программ установки дистрибутива программ до построения каркаса приложения или «скелета» программы.

Расширения

Любой «**Мастер**» не является составной частью среды разработки - **Visual Basic**, но может быть включен в нее для предоставления дополнительных возможностей пользователю. **Integrated Development Environment (IDE** - интегрированная среда разработки) **Visual Basic** снабжена фиксированным набором функций. **Add-ins** (расширения) - это инструменты, которые обогащают возможности **IDE**, делая среду более мощной.

Расширение

- **Wizard (Master, Мастер)** - это особый тип расширения. Как правило, Мастер последовательно «проводит» программиста по этапам некоторого задания, например, построения формы данных.



Wizard (Master, Мастер)

Мастер состоит из набора страниц, по одной для каждого шага задания, для выполнения которого он предназначен. В левом верхнем углу каждой страницы располагается **картинка**, а справа от нее - текст; **обычно он содержит инструкции для пользователя.**

Мастер форм данных Visual Basic

Мастер форм данных предназначен для автоматической генерации форм, то есть контейнеров, которые содержат отдельные связанные элементы управления, такие как поле ввода. С помощью Мастера форм данных вы можете создать запрос к таблице, результаты которого будут отображены в форме.

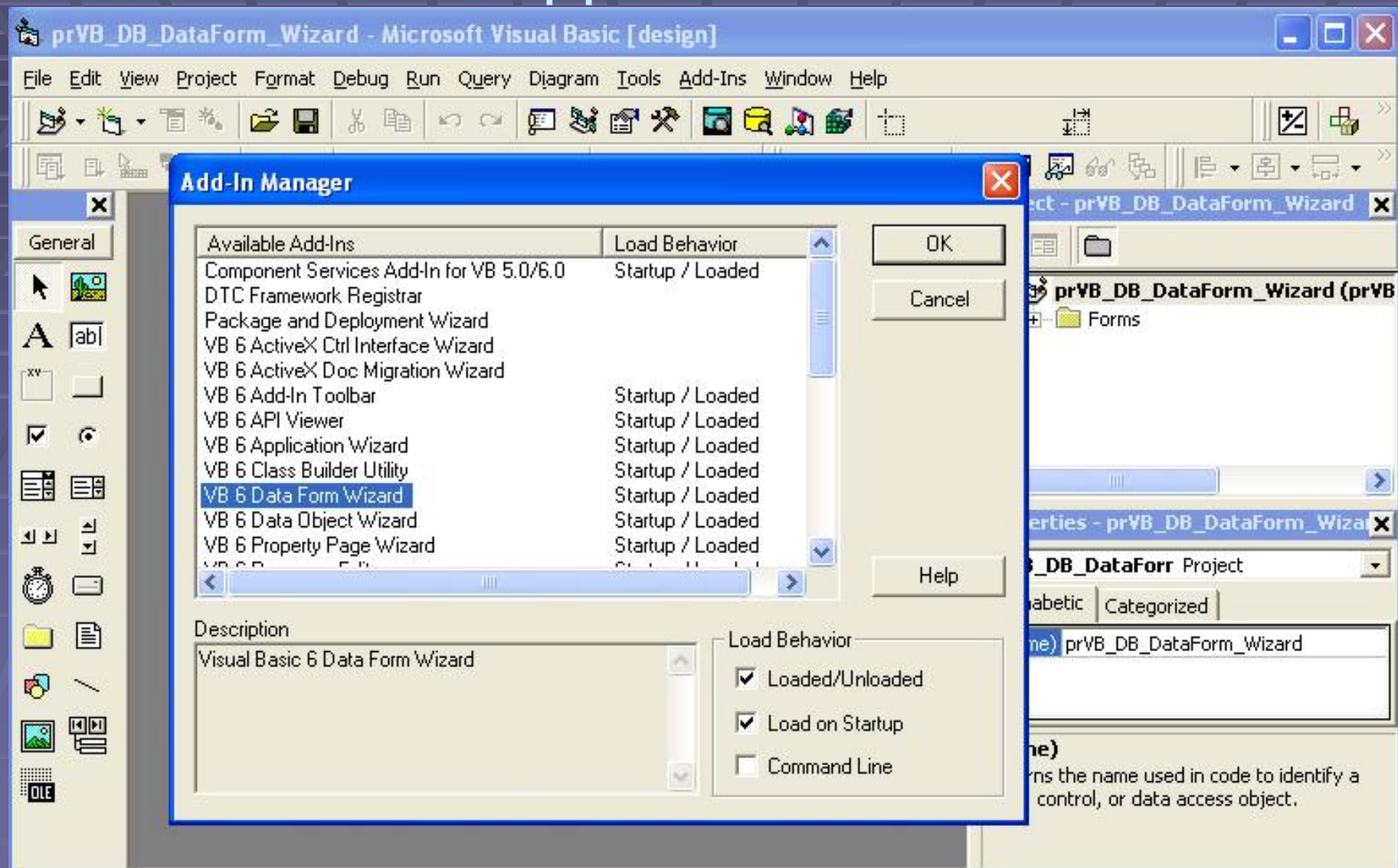
Мастер форм данных

Мастер, включенный в издания **Professional Edition** и **Enterprise Edition**, функционирует «внутри» **Visual Basic** и создает формы для работы с базами данных. Мастер создает в этих формах пользовательские элементы управления и даже самостоятельно генерирует базовый код для выполнения простых заданий.

Мастер форм данных

1. Перед запуском «**Мастера форм данных**» необходимо включить его в интегрированную среду разработки. Для этого из главного меню **Visual Basic** выберите пункт «**Add-Ins**», а затем «**Add-In Manager**».
2. На экране появится форма, где показаны все имеющиеся расширения, скопированные на ваш диск во время установки **Visual Basic**. Выберите строку «**VB 6 Data Form Wizard**».

Подключение «Мастера форм данных»



Мастер форм данных

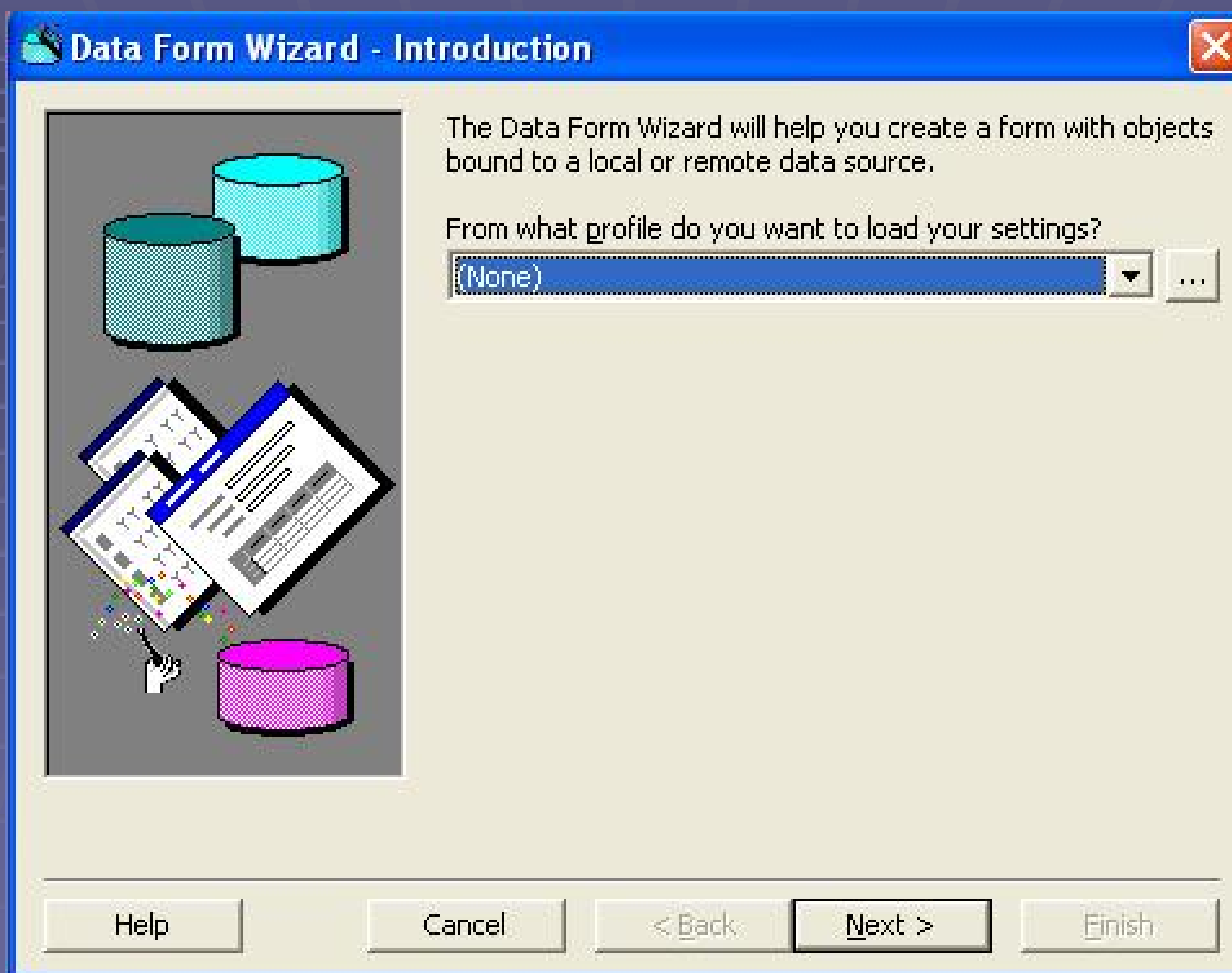
3. Включите флажок «**Loaded/Unloaded**», в результате чего можно будет работать с выбранным Мастером на протяжении всего сеанса. Если хотите, чтобы Мастер был доступен всякий раз, когда запускается **Visual Basic**, включите также флажок «**Load on Startup**».

4. Нажмите кнопку «**OK**»

Мастер форм данных

5. Перед началом работы с «**Мастером форм данных**» необходимо загрузить какой-нибудь проект, иначе появится сообщение об ошибке.
6. Выберите из меню панели инструментов «**Add-Ins**» или из панели инструментов **VB** мастера «**Data Form Wizard**». На экране появится страница «**Introduction**».

Страница «Introduction»



Мастер форм данных

5. Оставьте в поле ввода слово (**None**) - если вы запускаете **Мастер** в первый раз. Если же вы работали с ним раньше и сохранили свои установки, можно выбрать профиль из списка ранее сохраненных. Если файл профилей не пуст, вы можете выбрать любой из ранее созданных. Естественно, при сохранении вы можете присвоить профилю любое имя.

Профиль **Мастера** форм данных

Профиль - это файл с расширением **rwp**, в котором хранятся установки, сделанные и записанные во время предыдущих сеансов работы с «**Мастером форм данных**».

Профили удобно использовать, если необходимо создать несколько приложений с похожим внешним видом. При подготовке нескольких программ для взаимодействия с базой данных использование сохраненных профилей существенно ускоряет работу.

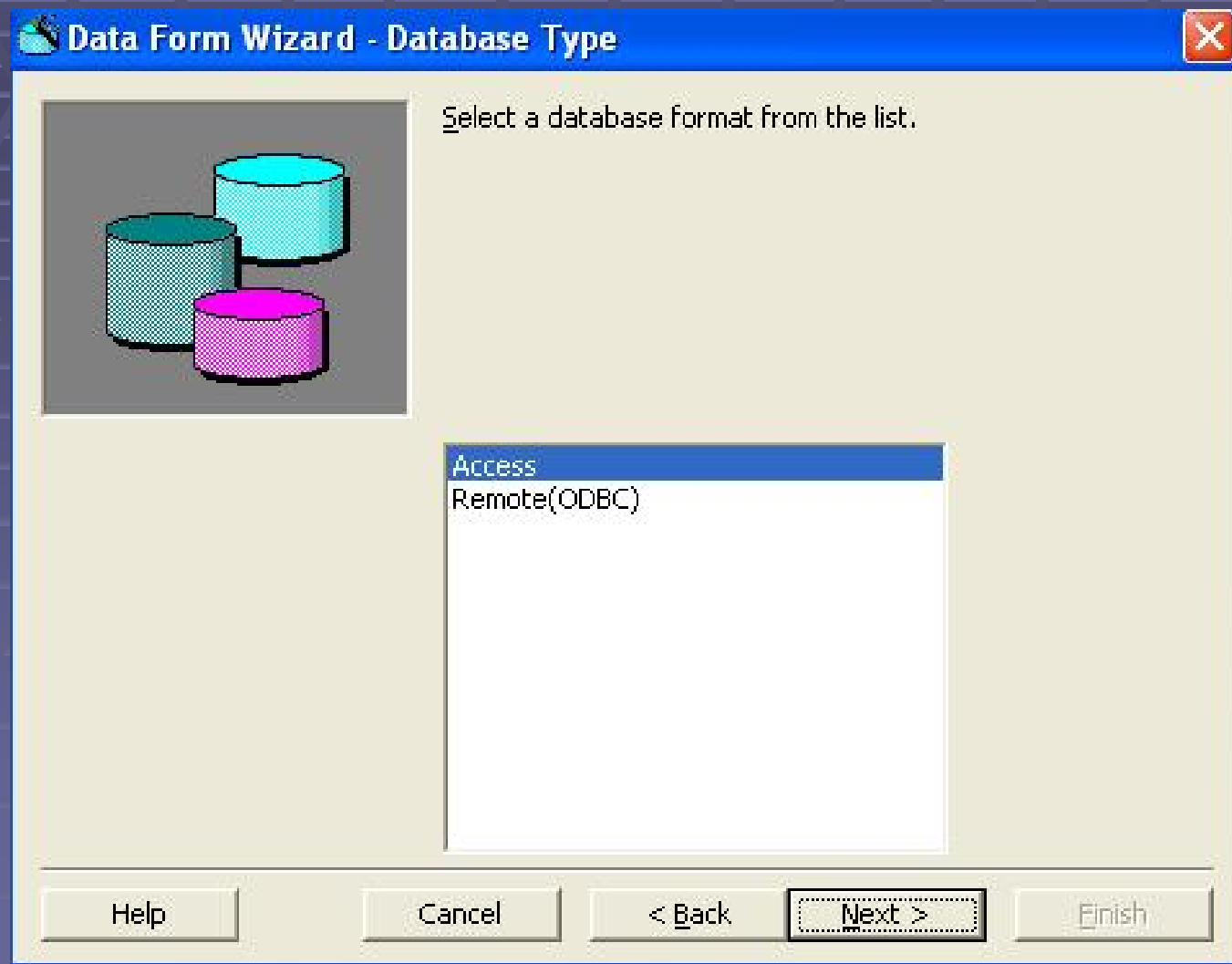
Формат базы данных

6. Щелкните по кнопке **Next >**.

7. Выберите на вкладке «**Database Type**» из списка «**Select a database format from the list**» тип базы данных. Элемент управления данными способен автоматически подключаться к следующим базам данных:

➤ **Access**, т.е. база, которая представляет собой файл с расширением mdb и создана в любой версии **Microsoft Access**;

Страница «Database Type»



Мастер форм данных

➤ удаленной (через **ODBC - Open Database Connectivity** (Открытый стандарт взаимодействия с базами данных) - это интерфейс, позволяющий Windows-приложениям обращаться к самым разным базам данных). Вы можете соединиться с любой базой, для которой существует **ODBC**-совместимый драйвер, например, с **SQL Server** или **Oracle**.

Мастер форм данных

Мастер форм данных может соединить **VB** проект практически с любым электронным хранилищем информации. **Мастер** предлагает выбрать Access по умолчанию, поскольку это «родная» база данных для **Visual Basic**.

Выберите тип базы данных и щелкните по кнопке «**Next >**».

Мастер форм данных

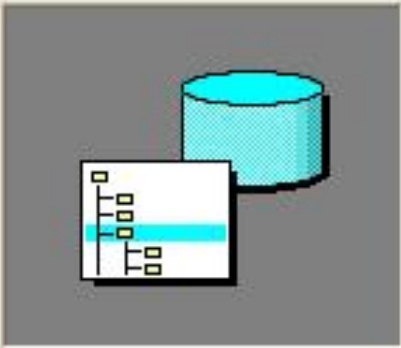
8. Теперь вы должны задать имя базы данных. Для этого на странице «**Database**» введите путь и имя файла базы данных. Воспользуйтесь при необходимости кнопкой «**Browse**» и найдите на диске нужный файл.

9. Нажмите кнопку «**Next >**».

Страница «Database»

Data Form Wizard - Database

Click the browse button to select a database file.



Database Name:

E:\Кафедра\УчебныйПроцесс\Преподаватели\Шкалето\С

Browse...

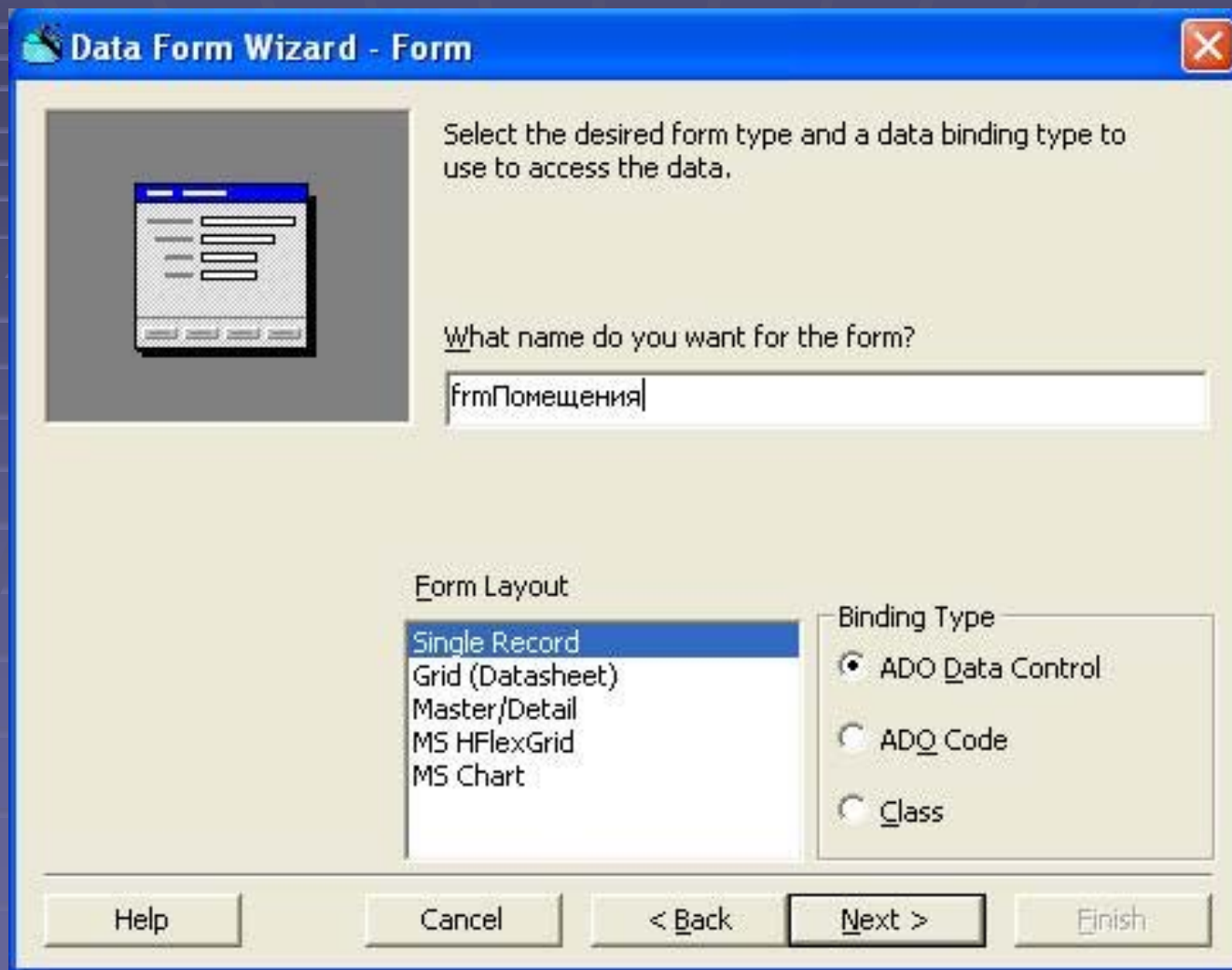
Help Cancel < Back Next > Finish

Мастер форм данных

9. На странице «**Form**» надо задать **имя формы**, которую будет создавать Мастер. Затем нужно из списка «**Form Layout**» выбрать вариант разметки формы, то есть способ размещения в ней данных:

➤ **Single record** - в форме в каждый момент работы будет видна только одна запись. Этот режим выбирается по умолчанию;

Страница «Form»



Data Form Wizard - Form

Select the desired form type and a data binding type to use to access the data.

What name do you want for the form?
frmПомещения

Form Layout

- Single Record
- Grid (Datasheet)
- Master/Detail
- MS HFlexGrid
- MS Chart

Binding Type

- ☒ ADO Data Control
- ☐ ADO Code
- ☐ Class

Help Cancel < Back Next > Finish

Варианты разметки формы

➤ **Grid (Datasheet)** - в форму вставляется элемент управления **DataGrid** (**Таблица данных**), так что выбранные вами поля будут отображаться в виде таблицы, которая представляет несколько записей;

Варианты разметки формы

➤ **Master/Detail** – в этом режиме **Мастер** отобразит **главную (master)** запись и множество связанных с ней **подчиненных (detail)** записей. Обычно в таком режиме в форме отображаются поля одной главной записи и элемент управления **Grid**, который содержит таблицу соответствующих подчиненных записей. При смене главной записи в таблице автоматически отображаются подчиненные записи, связанные с новой главной;

Варианты разметки формы

- **MSHFlexGrid** - Мастер будет создавать форму для отображения табличных данных;
- **MS Chart** - в созданной Мастером форме данные отображаются в виде графика.

Варианты разметки формы

Option Button «**Binding Type**»
позволяет выбрать один из трех способов
связывания данных с элементами
управления:

➤ **ADO Data Control** - для доступа к
информации Мастер будет использовать
ADO-элемент управления данными;

Варианты разметки формы

- **ADO Code** - для доступа к данным используется код программирования ADO. Мастер сам сгенерирует этот код;
- **Class** - для доступа будут использованы классы. Мастер применит так называемое объектно-ориентированное программирование. В этом случае код ADO помещается в специальный файл класса.

Страница «**Record Source**»

10. Щелкните по кнопке «**Next >**» для перехода к странице «**Record Source**». В комбинированном списке «**Record Source**» будут показаны все таблицы базы данных. Таблицы, имена которых начинаются с **Msys**, являются внутренними системными таблицами Access (в них хранится информация о различных атрибутах базы данных).

Страница «**Record Source**»

Выберите из списка «**Record Source**» таблицу.

Как только будет указано имя таблицы, в списке «**Available Fields**» покажутся все ее поля. Укажите **Мастеру**, какое поле или поля извлекать из ранее указанной таблицы. Если щелкните по кнопке «**Next >**», не указав ни одного поля, то **Мастер** сообщит об ошибке.

Страница «Record Source»

Data Form Wizard - Record Source

Select a record source and then select the fields to appear on the form.

Record Source:
Типы Мероприятий

Available Fields:

Selected Fields:
КодТипаМеро
ТипМероприятия

Column to Sort By:
(None)

Help Cancel < Back Next > Finish

Страница «**Record Source**»

Кнопки на форме позволяют выбирать поля таблицы и определять их положение друг относительно друга. Можно выбрать из списка «**Column to Sort By**» (**поля для сортировки**) поле для упорядочения записей.

Кнопка «**Finish**» (**финиш**) теперь активна, т.е. можно создавать форму, однако лучше нажать кнопку «**Next >**».

Страница «**Control Selection**»

11. При отображении данных удобно иметь кнопки для манипулирования текущей записью. Те кнопки, для которых поднят флажок, будут присутствовать в конечной форме. По умолчанию отмечены все:

- **Add Button** - в форму будет помещена кнопка «**Add**» для добавления записи в базу данных. Эта кнопка очищает все поля и добавляет в базу новую запись;
- **Update Button** - в форму будет помещена кнопка «**Update**» и код для обновления данных в базе. При щелчке по этой кнопке все изменения, сделанные в полях формы, сохраняются в базе данных;

Страница «Control Selection»



Страница «**Control Selection**»

- **Delete Button** - в форму будет помещена кнопка «**Delete**» и код для удаления текущей записи из базы;
- **Refresh Button** - в форму будет помещена кнопка «**Refresh**» и код для повторного считывания из базы. При щелчке по данной кнопке запрос выполняется заново. Это обычно применяется в приложениях, работающих в многопользовательской среде;

Страница «**Control Selection**»

➤ **Close Button** - в форму будет помещена кнопка «**Close**» и код для закрытия и выгрузки формы.

Обратите внимание: флажок «**Show Data Control**» (**Показать элемент управления данными**) включен и недоступен. Это связано с тем, что элемент управления данными должен присутствовать в форме, иначе навигация по набору записей будет невозможна. Поэтому сбросить флажок нельзя.

Щелкните по кнопке «**Next >**».

Страница «**Профиль**»

Если щелкните по кнопке с многоточием, появится диалоговое окно «**Save Profile**». Введите при необходимости имя файла в поле «**File name**», и профиль будет сохранен в файле с этим именем и расширением **rwp**. Щелкните по кнопке «**Finish**» и подождите несколько секунд. **Мастер**, следуя вашим указаниям, автоматически создаст форму.

Страница «Профиль»



Генерация формы с «Grid»

Для создания формы в табличном формате необходимо на странице мастера «Form» выбрать из списка «Form Layout» тип представления **Grid (Datasheet)**. При этом создаваемая форма будет иметь вид таблицы:

Форма с таблицей

Список сотрудников

	Фамилия	Имя	Отчество	телефон
▶	Шариков	Полиграф	Полиграфович	8-015-598-15-63
	Иванов	Ефим	Григорьевич	736-15-22
	Гришина	Ольга	Павловна	369-98-98
	Григорьева	Светлана	Леонидовна	268-55-63
	Павлов	Павел	Павлович	3-12-11
	Иванов	Григорий	Ефимович	736-15-22
	Сидоров	Илья	Израилевич	3-12-10
	Петренко	Иван	Сидорович	702-55-22
	Коган	Соломон	Абрамович	8-05747-3-15-15
	Изяксон	Абрам	Соломонович	8-096-756-66-55
*				

◀ ▶

Add Update Delete Refresh Close

◀◀ Record: 1 ▶▶

Подчиненные формы

Это тип формы — «**Master/Detail**» - для отображения одной главной и множества подчиненных записей. Этот вариант отличается от предыдущих тем, что для него требуется указать источник как главных записей, так и подчиненных, то есть определить связь «**один-ко-многим**». Каждой записи в главной таблице может соответствовать несколько записей в подчиненной.

Источники дополнительных сведений

- Коннэлл Дж. Visual Basic 6. Введение в программирование баз данных: Пер. с англ. - М.: ДМК, 2000. - 720 с: ил. (Серия «Для программистов»).
- Культин Н. Б. Visual Basic. Освой на примерах. — СПб.: БХВ-Петербург, 2004. - 288 с.
- VB6 – книга в формате файлов .html