

2. Цели занятия:

- формирование представления о способах и особенностях ведения баз данных.
- изучение основополагающих принципов разработки и использования современных информационных технологий.

3. Учебные вопросы:

1. Ввод, редактирование и удаление информации в базе данных.
2. Комплексная отладка элементов контроля информации.

Вопрос 1. Ввод, редактирование и удаление информации в базе данных.

Непосредственный ввод данных в таблицу осуществляется в режиме таблицы (Datasheet View). Переход в этот режим из окна базы данных выполняется нажатием кнопки **Открыть** (Open) при выделенной таблице в окне базы данных. Переход в режим таблицы из режима конструктора выполняется нажатием кнопки **Вид** (View) на панели инструментов или выполнением команды **Вид|Режим таблицы** (View|Datasheet).

В режиме таблицы предоставляется возможность вводить новые записи в таблицу, заполняя значениями ее поля. Дополнение таблицы новыми записями и редактирование записей разрешено по умолчанию, т.к. по умолчанию пользователь работает под логином администратора (Admin) и ему предоставлены *все права доступа* к объектам базы данных.

Просмотреть и изменить права доступа к данным таблицы можно по команде **Сервис|Защита|Разрешения** (Tools|Security|User And Group Permissions).

При дополнении таблицы новыми записями может быть использован *режим ввода записи*, при котором видна только вводимая запись. Выполнение команды **Записи|Ввод данных** (Records|Data Entry) позволяет перейти в этот режим;. Для возврата к просмотру всей таблицы надо выполнить команду **Записи | Удалить фильтр** (Records|Remove Filter/Sort).

Очевидно, что вводимые в поля таблицы значения данных должны соответствовать типам данных, определенным в структуре, а способ их записей — допустимым в текущий момент форматам данных. Значения должны удовлетворять накладываемым ограничениям. После ввода значений в ячее поля при попытке перейти к другой ячейке Access проверяет, являются ли введенные данные допустимыми для этого поля. Если введено значение, не соответствующее типу данных поля, Access пытается преобразовать его в правильный тип данных. Если значение не является допустимым и преобразование невозможно, например, нельзя преобразовать текст в число, и появляется предупреждающее сообщение. Для того чтобы выйти из ячейки следует ввести правильное значение или отменить внесенные изменения.

При заполнении связанных таблиц необходимо обеспечить правильность связей между записями этих таблиц. Так, при одно-многочисленных связях таблиц и вводе записей в подчиненную таблицу необходимо отслеживать наличие в главной таблице записей с соответствующими значениями ключевых полей. При изменении или удалении ключевых полей в записях главной

таблицы необходимо изменять или удалять связанные с ними записи в подчиненных таблицах.

Макет таблицы

Для удобства работы с таблицей можно изменить ее представление на экране. При этом можно менять ширину столбца, высоту строки, шрифт данных таблицы, цвет текста, линий сетки и фона, оформление, которое может быть обычным, приподнятым или утопленным. Можно выводить на экран только те столбцы, которые нужны для текущей работы, можно зафиксировать столбец при просмотре широких таблиц. Эти параметры отображения таблицы на экране называются *макетом* таблицы и сохраняются вместе с ней.

Настройка макета выполняется в режиме таблицы. При этом могут быть использованы команды меню **Формат** (Format) или панель инструментов **Формат (режим таблицы)** (Formatting (Datasheet)), которую можно вызвать из контекстного меню на доступной панели инструментов.

Многие операции настройки макета можно выполнить непосредственно в таблице с помощью мыши:

- *изменение ширины столбца.* Для изменения ширины столбца курсор мыши устанавливается на линию, разделяющую имена столбцов. При этом он превращается в планочку со стрелками влево и вправо. Далее границу столбца можно перетащить в нужное место;
- *изменение высоты строки.* Для изменения высоты строки курсор мыши устанавливается в области маркировки записи, расположенной слева, на границе между записями. Граница строки перетаскивается на требуемое расстояние. При этом изменяется высота всех строк таблицы;
- *удаление столбцов с экрана.* Убрать столбец можно, перетащив его правую границу влево до исчезновения столбца;
- *изменение порядка расположения столбцов на экране.* Столбец выделяется щелчком кнопки мыши на его имени. Протащив курсор мыши поперек столбцов, можно выделить несколько столбцов. Выделенный столбец перетаскивается в новое место при установке курсора на его имени (области маркировки столбца);
- *скрытие столбцов, ненужных для текущей работы, закрепление столбцов, а также отмена этих действий* могут быть выполнены при помощи соответствующих команд меню **Формат** (Format);
- *сохранение макета таблицы* выполняется командой **Файл|Сохранить** (File|Save Layout) или при закрытии таблицы после утвердительного ответа на вопрос "Сохранить изменения макета таблицы "<имя таблицы>"? (Do you want to save changes to the layout of table <имя таблицы>?).

Ввод данных в таблицы БД "Поставка товаров"

В режиме таблицы введем несколько записей в таблицу ТОВАР. В окне **Поставка товаров: база данных** установим курсор на таблице ТОВАР и нажмем кнопку **Открыть** (Open). Таблица откроется в режиме таблицы (Datasheet View). Заполним строки (записи) открывшейся таблицы (рис. 3.13) в соответствии с названиями столбцов (полей).

Корректность вводимых данных (соответствие заданному типу поля, размеру и условию на значение, которые определены в свойствах полей в режиме конструктора) проверяется автоматически при их вводе. Отслеживается уникальность значений ключевых полей.

Отменить ввод значения в поле до перехода к другому полю можно, нажав клавишу <Esc> или выполнив команду **Правка|Отменить** ввод. Переход от одного поля к другому можно выполнить клавишей <Tab> или переводом курсора. Отмена ввода значения в поле после перехода к другому полю текущей записи (до перехода к другой записи) происходит с помощью клавиши <Esc> или команды **Правка|Восстановить текущее поле/запись** (Edit| Undo Current Field/Record).

Код товара	Наименование товара	Цена	Единица измерения	Ставка НДС	ФОТО
001	Монитор17LG	6 587,00р.	штука	0,05	
002	FDD 3,5	363,00р.	коробка	0,2	
003	HDD Maxtor20GB	2 590,00р.	штука	0,1	
004	Корпус MiniTower	916,00р.	штука	0,2	
005	CD-ROM Panasonic Id	1 153,00р.	штука	0,3	
006	DIMM64MPC100	360,00р.	штука	0,15	
007	Принтер EPSON ST.A	2 432,00р.	штука	0,05	
008	СканерAser	2 338,00р.	штука	0,15	
▶ 009	Зв. картаGenius Liv	789,00р.	штука	0,05	
010	Модем Genius ext	1 295,00р.	штука	0,05	
*		0,00р.		0	

Рис. Ввод записей в таблицу ТОВАР

Завершение ввода новых значений записи или редактирования осуществляется при переходе к любой другой записи (при смене текущей записи). После перехода к другой записи можно отменить ввод (редактирование) всей записи, выполнив команду **Правка|Восстановить запись** (Edit|Undo Saved Record).

Команда **Восстановить** (Undo) занимает одну позицию в меню, но в зависимости от контекста приобретает различный смысл и наименование. Это же касается и соответствующей кнопки панели инструментов, всплывающая подсказка которой сообщает о текущем её назначении. По этой команде может быть отменено только одно последнее действие.

Добавление записи в таблицу начинается с заполнения пустой строки, размещенной в конце таблицы и помеченной звездочкой (*).

Переход к этой записи можно выполнить также по команде **Правка|Перейти|Новая запись** (Edit|Go To|New Record) или нажатием кнопки **Новая запись** (New Record) панели инструментов **Таблица в режиме таблицы** (Table Datasheet). Корректируемая запись помечается слева символом карандаша.

Для создания новой записи должны быть обязательно заполнены ключевые поля, обязательные поля, а также поля связи, если установлены связи таблицы с другими таблицами.

Сохранение новой записи происходит после перехода к другой записи или выполнения команды **Записи|Сохранить запись** (Records Save Record).

Для удаления записи в таблице ее нужно сделать текущей и нажать кнопку панели инструментов **Удалить запись** (Delete Record) или выполнить соответствующую команду **Правка|Удалить запись** (Edit|Delete Record). Для исключения ошибочного удаления в Access предусмотрен запрос на подтверждение удаления. После подтверждения на удаление восстановление удаления.

Изменение значений в полях записи осуществляется непосредственно в ячейках таблицы. Введенное значение проверяется Access при попытке перевода курсора в другое поле. Если значение не является допустимым, появляется предупреждающее сообщение. Для того чтобы выйти из поля, следует ввести правильное значение или отменить внесенные изменения. Отмена изменения Значения производится нажатием кнопки **Восстановить текущее поле/Запись** (Undo Typing). Откорректированная запись сохраняется после перехода к другой записи или принудительного сохранения командой **Записи|Сохранить запись** (Records|Save Record).

Размещение объекта OLE

Рассмотрим размещение объекта **OLE** на примере поля **ФОТО** в таблице **ТОВАР**. Пусть фотографии хранятся в формате графического редактора Paint в файлах с расширением bmp.

Рассмотрим вариант внедрения объекта в файл базы данных. Установим курсор в соответствующем поле таблицы. Выполним команду **меню Вставка | Объект** (Insert | Object). В открывающемся окне надо отметить переключатель **Создать из файла** (Create from File).

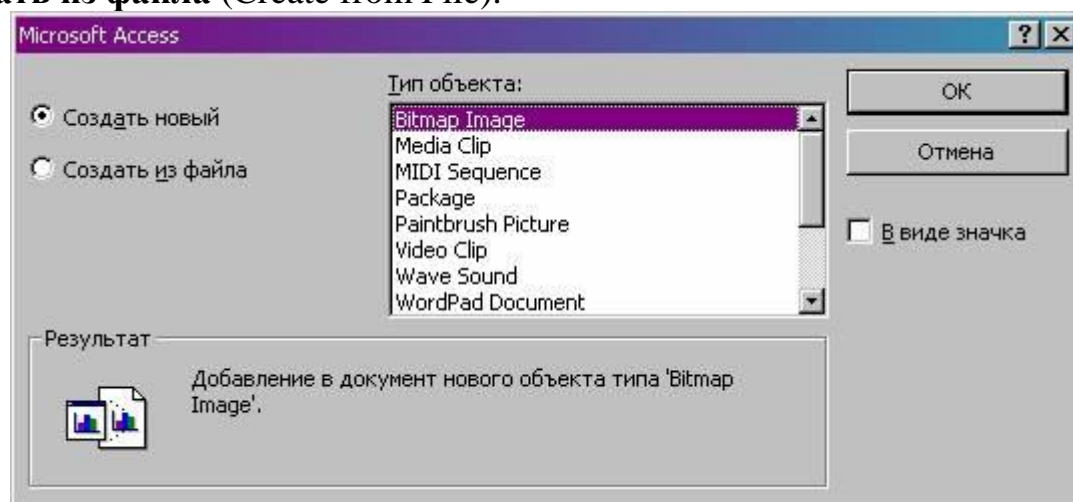


Рис. Окно для выбора варианта вставки объекта

В следующем окне можно ввести имя файла с фотографией. Для поиска файла можно воспользоваться кнопкой **Обзор** (Browse), после нажатия которой появится диалоговое окно, позволяющее Просмотреть диски и папки и выбрать необходимый файл.

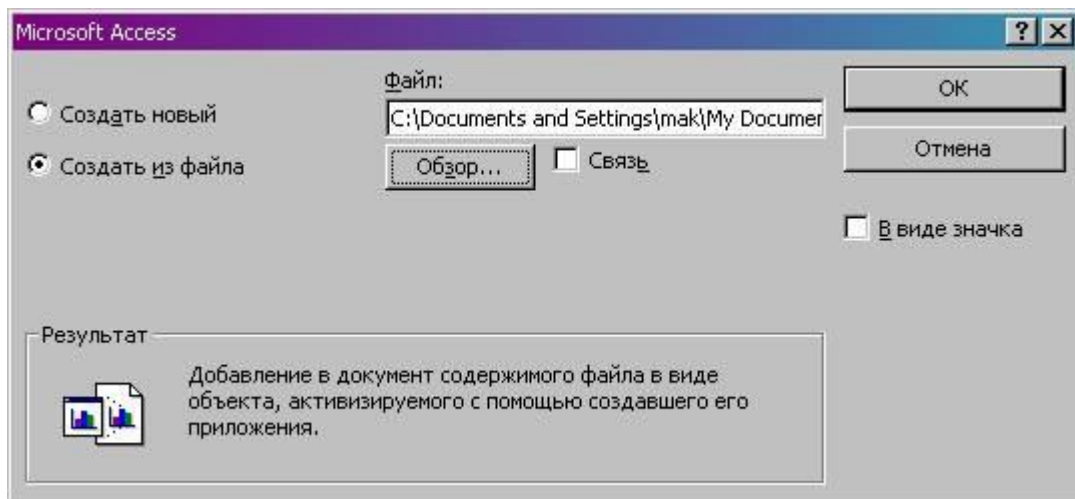


Рис. Окно для выбора вставляемого объекта из файла

Флажок **Связь** (Link) по умолчанию не отмечен и, следовательно, содержимое файла будет введено в поле как внедренный объект. Увидеть содержимое поля можно через форму или отчет. Дальнейшие изменения файла не будут отображаться на встроенном объекте. Для ввода в поле связанного объекта достаточно установить флажок **Связь** (Link). Это сэкономит место в базе данных и даст возможность отображать все изменения, вносимые в файл другими приложениями.

Результат заполнения таблицы приведенный ранее, где в соответствующем поле указан вид объекта — *точечный рисунок*. Для просмотра внедренного объекта достаточно установить в соответствующем поле курсор и дважды щелкнуть кнопкой мыши.

Для отображения содержимого поля в виде значка, представляющего файл с документом, в окне **Вставка объекта** (Insert Object) нужно установить флажок **В виде значка** (Display As Icon). Значок может быть использован для представления связанного объекта.

Ввод логически связанных записей

Введем несколько логически взаимосвязанных записей в таблицы ДОГОВОР^И ПОСТАВКА_ПЛАН. Логическая связь этих таблиц обеспечивается полем НОМ_ДОГ — номером договора, входящим в структуру обеих таблиц.

Объекты ДОГОВОР и ПОСТАВКА_ПЛАН связаны одно-многочисленными отношениями. Связь таких таблиц реализуется в схеме данных с помощью ключа главной таблицы ДОГОВОР. В подчиненной таблице это поле связи является в данном случае частью ее составного ключа. Пока не создана схема; данных, в которой устанавливаются связи между таблицами, система не может контролировать логическую взаимосвязь вводимых данных. Поэтому для получения целостной базы, в которой каждая запись подчиненной таблицы имеет логически связанную с ней главную запись, пользователю придется самому отслеживать взаимосвязи записей. При вводе подчиненной записи в таблицу ПОСТАВКА_ПЛАН необходимо проверять наличие записи в главной таблице ДОГОВОР с ключом, значение которого совпадает со значением поля связи (внешнего ключа) в подчиненной записи. То есть при добавлении строки

плана поставки для договора Д111 необходимо, чтобы договор с этим номером уже был представлен в таблице ДОГОВОР.

Для удобного отображения открытых таблиц можно воспользоваться командой меню **Сверху вниз** (Tile Horizontally) или **Слева направо** (Tile Vertically) в меню **Окно** (Windows).

Далее, в разд. "Отображение записей подчиненных таблиц в главной таблице", показано, что при наличии установленных связей в схеме данных между таблицами имеется возможность отображать связанные записи подчиненных таблиц при просмотре главной таблицы. Это позволяет осуществлять автоматический контроль за корректностью связей логически связанных таблиц при добавлении, удалении и редактировании записей.

Использование поля со списком при вводе записей

Задачу ввода подчиненных записей только при наличии связанной записи в главной таблице можно решить с помощью поля со списком. Для этого достаточно в подчиненной таблице преобразовать поле связи в поле со списком. В качестве источника строк для списка можно использовать значения ключевого поля из главной таблицы.

Создадим в таблице ПОСТАВКА_ПЛАН для поля НОМ_ДОГ (номер договора) поле со списком, построенным на основе ключа таблицы ДОГОВОР. Для этого откроем таблицу ПОСТАВКА_ПЛАН в режиме конструктора. Установим курсор в поле НОМ_ДОГ на столбце **Тип данных** (Data Type), откроем список и выберем строку **Мастер подстановок** (Lookup Wizard). В открывшемся окне мастера **Создание подстановки** (Create Lookup) установим флажок на строке, которая приведет к построению списка на основе значений из таблицы.

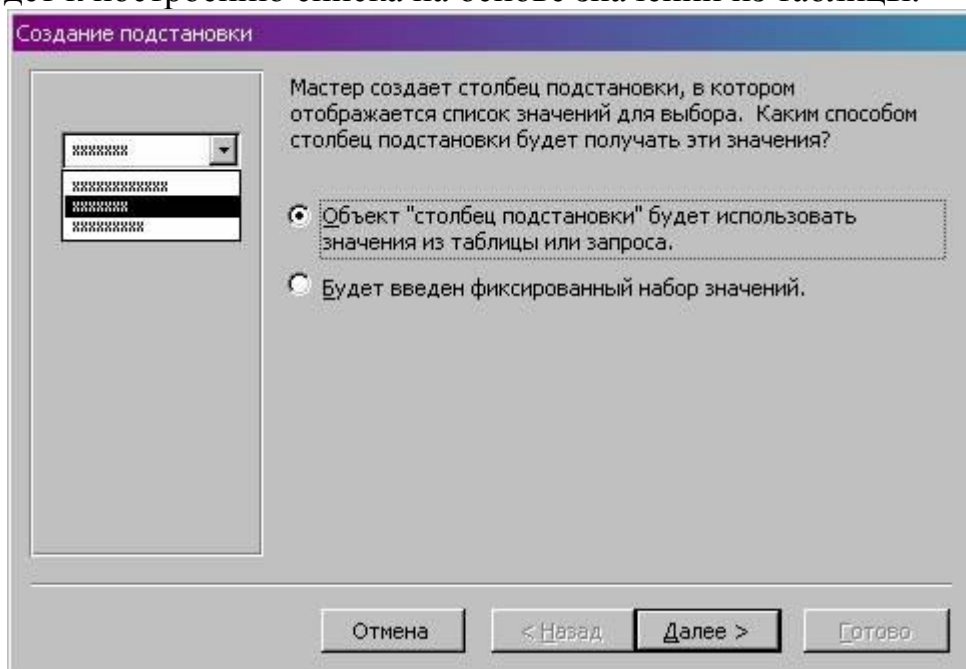


Рис. Первое окно Мастера подстановок

В следующем окне мастера выберем таблицу ДОГОВОР, на значениях из которой будет строиться список поля.

Далее выберем столбец, из данных которого будет формироваться список

В следующих окнах мастера выберем нужную ширину столбца списка и его подпись. На этом создание мастером поля со списком завершается. Параметры списка, подготовленные мастером, **отображаются в области Свойства поля** на вкладке **Подстановка**

Теперь при вводе данных в таблицу ПОСТАВКА_ПЛАН можно воспользоваться списком, в котором отобразятся все значения поля НОМ_ДОГ; содержащиеся в таблице ДОГОВОР. Выбором нужного значения из списка осуществляется ввод значения в поле НОМ_ДОГ.

Наличие поля со списком не запрещает ввод в поле НОМ_ДОГ значений, не указанных в списке. Однако если при определении параметров списка на вкладке Подстановка в окне конструктора выбрать в строке Ограничиться списком значение Да, ввод значений, не предусмотренных в списке, будет невозможен. В случае отклонения от значений списка система выдаст сообщение В результате пользователь вынужден будет придерживаться только значений, представленных в списке, и в подчиненную таблицу будут введены только связанные записи.

Таким образом, использование списков не только обеспечивает удобный ввод, но позволяет избежать многих ошибок.

Рассмотрим еще один пример использования поля со списком. Преобразуем в таблице ПОСТАВКА_ПЛАН поле КОД_ТОВ (код товара) в поле со списком. Для этого откроем таблицу в режиме конструктора, установим курсор в поле КОД_ТОВ на столбце **Тип данных (Data Type)**, откроем список этого столбца и выберем строку **Мастер подстановок (Lookup Wizard)**. В диалоге с мастером выберем главную по отношению к таблице ПОСТАВКА_ПЛАН таблицу ТОВАР, на основе которой создается список, и поля, включаемые в список: код_тов (код товара) и НАЙМИТОВ (наименование товара). Кроме того, оставим, как предлагает мастер, помеченным флажок **Скрыть ключевой столбец (рекомендуется)**.

При выборе параметра **Скрыть ключевой столбец (рекомендуется)** для первого столбца списка устанавливается ширина, равная нулю. При этом параметр **Ограничиться списком** может принимать только одно значение — Да

Определение поля со списком с такими параметрами приводит к отображению в поле КОД_ТОВ вместо значений ключевого поля значений второго поля списка — поля НАММТОВ (наименование товара), взятого из главной таблицы ТОВАР.

Таким образом, использование полей со списком поможет избежать ошибок при вводе связанных записей в таблицы как за счет того, что в таблицах вместо ключей могут быть отображены смысловые значения, так и за счет того, что ввод ограничивается набором значений в списке.

При создании полей со списком Мастер подстановки автоматически создает связь между таблицей, в которой создается поле со списком, и таблицей, являющейся источником значений списка. Эту связь можно увидеть в Схеме данных (Relationships), щелкнув на кнопке панели инструментов **Отобразить все связи (Show All)**.

Очевидно, что в базе данных сложной структуры при вводе данных непосредственно в таблицы не гарантируется надежное и корректное обслуживание данных. В рассматриваемом примере базы данных "Поставка товаров" при вводе данных в таблицы нижних уровней надо отслеживать несколько вышестоящих. При большом объеме данных это очень затруднительно.

Далее будут рассмотрены средства Access, позволяющие автоматизировать контроль связной целостности и сделать загрузку базы данных простой и удобной процедурой. К таким средствам относятся, прежде всего, **Схема данных** (Relationships) и **Формы** (Forms). Кроме того, может быть использовано такое простое средство, как отображение в главной таблице записей подчиненных таблиц нескольких уровней — подтаблиц.

Вопрос 2. Комплексная отладка элементов контроля информации.

Контрольные вопросы:

1. Назначение и состав объектов типа "Database", "Table".
2. Форматы ввода и вывода данных.
3. Ввод, редактирования и удаления информации в базе данных.
4. Комплексная отладка элементов контроля информации.
5. Региональные представления различных типов данных.

Литература

1. Попов В.Б. Основы информационных и телекоммуникационных технологий.- Финансы и статистика, 2007 г. – 336 с.
2. Угринович Н.Д. Информационные технологии и компьютеризация делопроизводства. – М.:БИНОМ ЛЗ.,2007.-394с.
3. Шамраев А.В. Правовое регулирование информационных технологий (анализ проблем и основные документы). – Статут, 2003. – 1013 с.
4. Хроленко А.Т., Денисов А.В. Современные информационные технологии для гуманитария. – Флинта, 2007. – 128 с.
5. Федотова Е.Л. Информационные технологии в профессиональной деятельности. – Форум, 2008. – 368 с.
6. Емельянова Н.З., Партыка Т.Л., Попов И.И. Основы построения автоматизированных информационных систем. – Форум, 2007. – 416 с.
7. Беляева Т. М., Важнов С. А., Вешняков В. В., Кудинов А. Т., Пальянова Н. В. Основы информатики и математики для юристов. – Элит, 2007. – 368 с.
8. Казанцев С.Я. Информатика и математика для юристов. Учебник. – Юнити, 2008. – 560 с.
9. Строганов М.П., Щербаков М.А. Информационные сети и телекоммуникации. – Высшая школа, 2008. – 151 с.

10. Микрюков В.Ю. Информация, информатика, компьютер, информационные системы, сети. – Феникс, 2007. – 448 с.