

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

**Методические рекомендации для
самостоятельной работы обучающихся по дисциплине**

Б1.В.07_ Основы программирования и конфигурирования в корпоративных
информационных системах (I часть)

Направление подготовки (специальность) 09.03.01 Информатика и вычислительная техника

Профиль подготовки (специализация) Автоматизированные системы обработки информации и управления

Форма обучения заочная

СОДЕРЖАНИЕ

1. Организация самостоятельной работы.....	3
2. Методические рекомендации по самостоятельному изучению вопросов	4
3. Методические рекомендации по подготовке к занятиям.....	25
3.1 Лабораторная работа №1 Создание новой информационной базы данных.....	25
3.2 Лабораторная работа №2 Классификация объектов конфигурации.....	25
3.3 Лабораторная работа №3 Введение в язык запросов.....	25
3.4 Лабораторная работа №4 Роли и права пользователей.....	25

1. ОРГАНИЗАЦИЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

1.1. Организационно-методические данные дисциплины

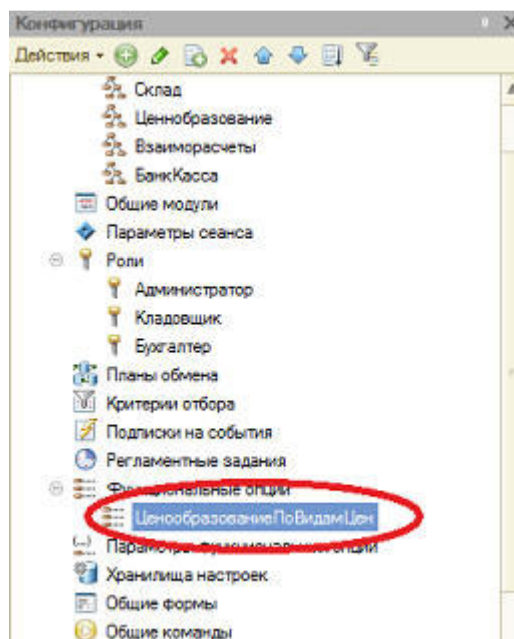
№ п.п.	Наименование темы	Общий объем часов по видам самостоятельной работы (из табл. 5.1 РПД)				
		подготовка курсового проекта (работы)	подготовка реферата/эссе	индивидуальные домашние задания (ИДЗ)	самостоятельное изучение вопросов (СИБ)	подготовка к занятиям (ПкЗ)
1	2	3	4	5	6	7
1	Создание и настройка информационной базы данных				20	4
2	Работа со справочниками					2
3	Группы панели навигации					2
4	Документы					2
5	Классификация объектов конфигурации					3
6	Определение режима запуска				10	2
7	Введение в язык запросов					2
8	Пользовательские настройки отчета				10	2
9	Разбор примера отчета о закупках товаров					1
10	Роли и права пользователей				20	10

2.МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО САМОСТОЯТЕЛЬНОМУ ИЗУЧЕНИЮ ВОПРОСОВ

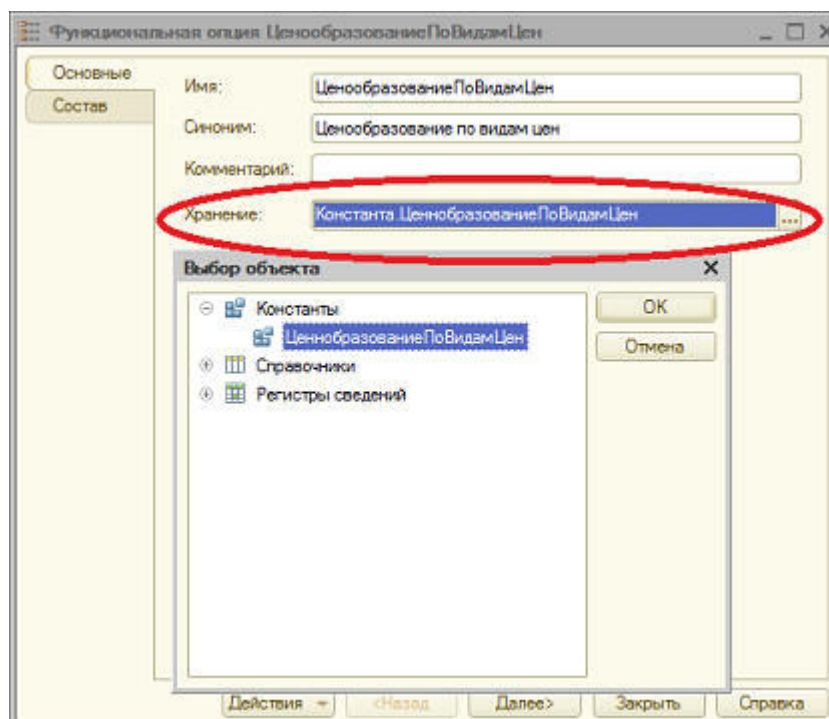
2.1 Константы и функциональные опции

При изучении вопроса необходимо обратить внимание на следующие особенности:

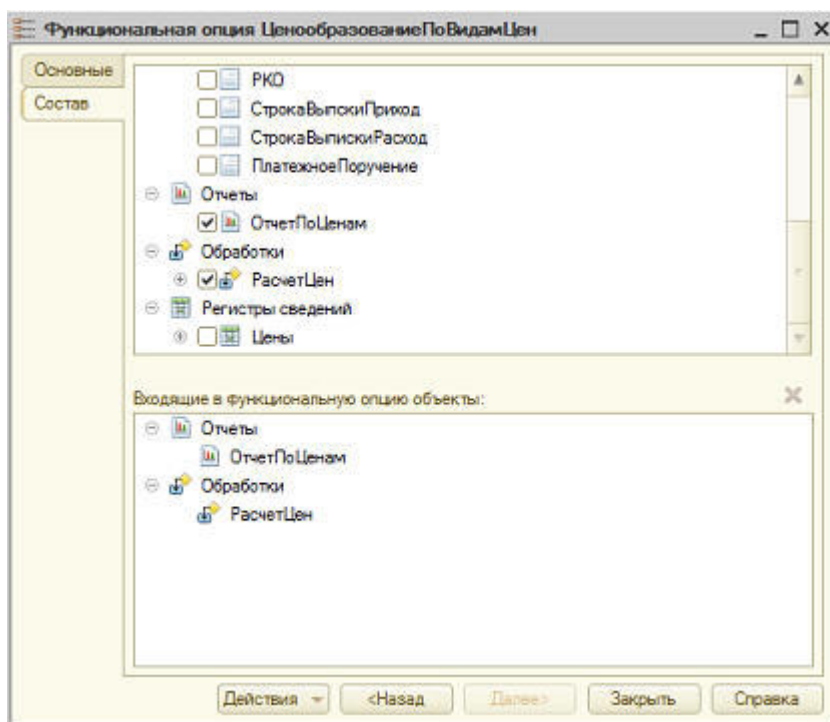
В 1С:Бухгалтерии 8.2 можно элементы интерфейса сгруппировать и вывести для включения/выключения в режиме пользователя. Для этого предусмотрен такой объект конфигурации, как функциональные опции:



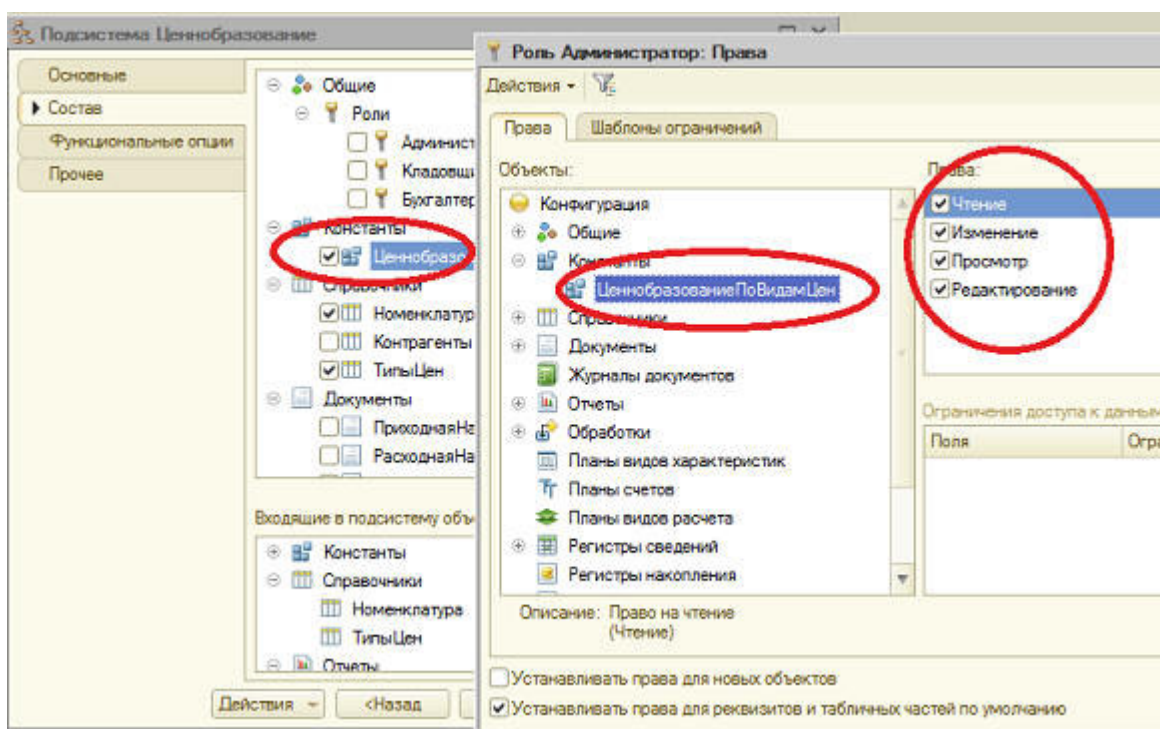
Опция "хранение" показывает, с каких объектом должно быть связано включение/выключение функциональной опции, рекомендуется использовать для этих целей константу типа "Булево":



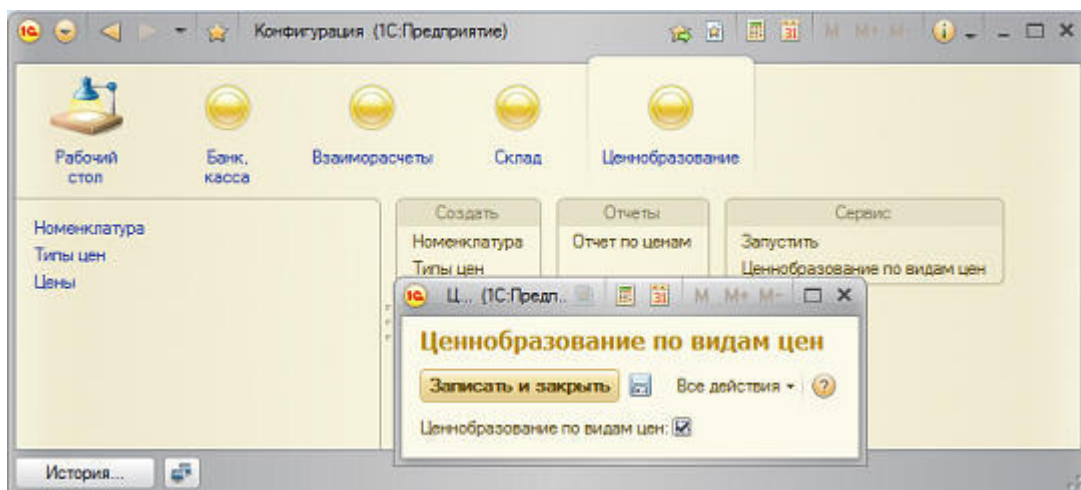
Список объектов, которые включаются/выключаются при включении/выключении функциональной опции настраивается на вкладке "состав":



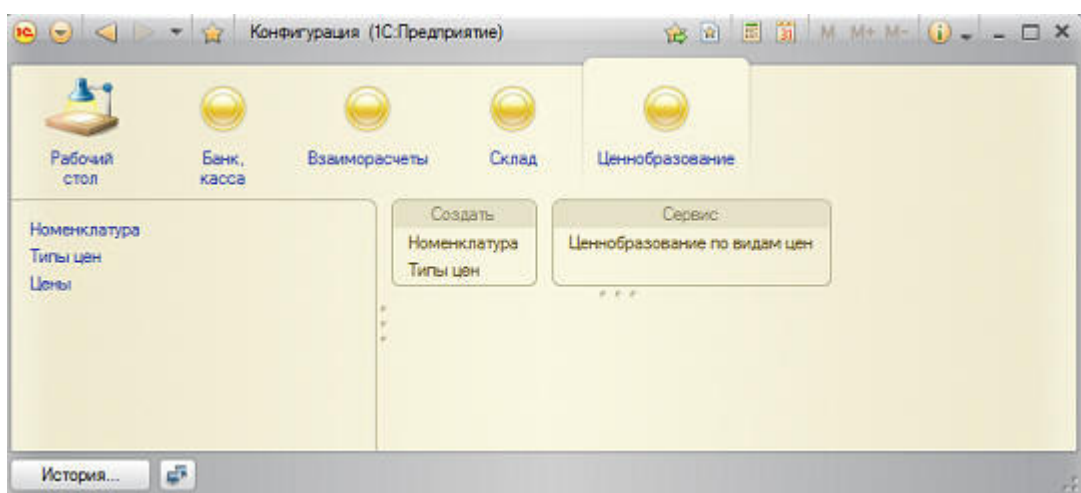
Кстати, добавляя новую константу, не забудьте включить ее в соответствующую подсистему и назначить на нее права:



После этого, изменяя значение константы, мы можем включать и выключать показ команд функциональной опции:



Отключим константу, и команды функциональной опции исчезнут:



Изменение вступят в силу, если из программы выйти и снова зайти.

2.2 Кнопочная форма. Командные кнопки

При изучении вопроса необходимо обратить внимание на следующие особенности:

Как только мы перейдем в режим *Конструктор* формы, тут же на ленте появляется вкладка *Конструктор*, а на ней мы увидим все доступные нам элементы управления (рис. 1). Именно эти элементы мы можем помещать на формы.

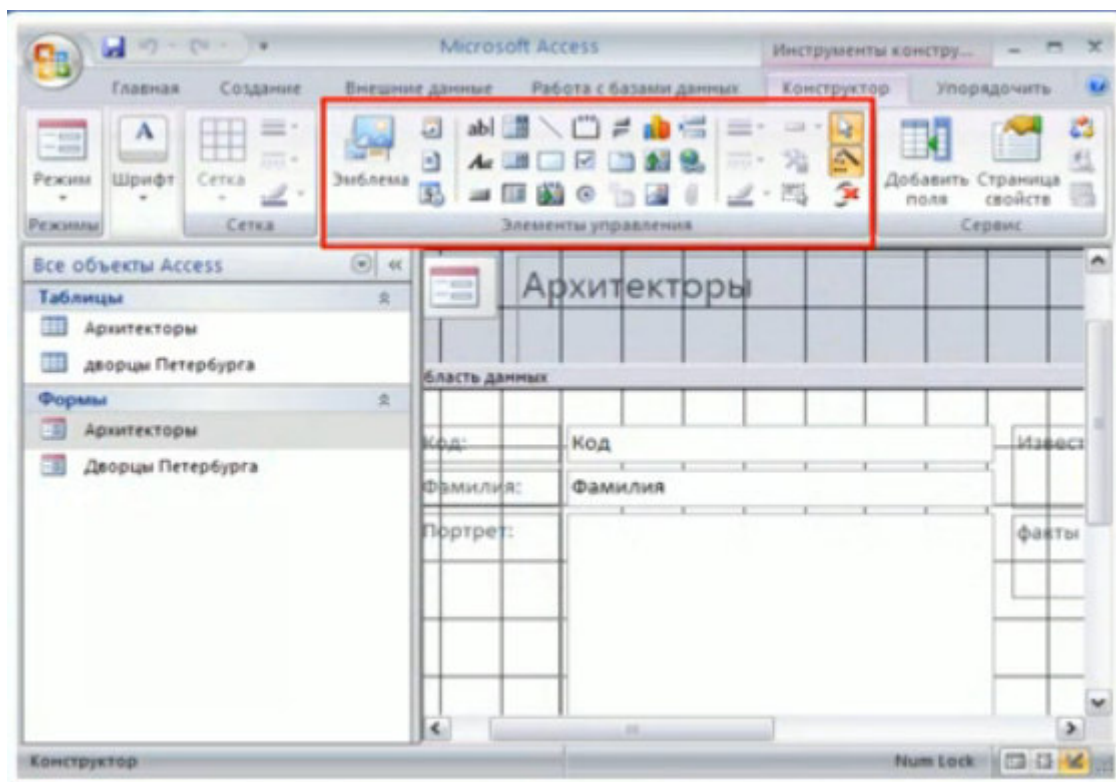


Рис. 1

Посмотрим, как же на практике нам добавить в форму эти элементы. Для этого перейдем в режим Конструктор (рис. 2), тогда на ленте автоматически появится вкладка Конструктор, и мы увидим все доступные элементы (рис. 1).

Выберем элемент *кнопка* (рис. 3), щелкнем на ней ПКМ и обозначим на форме размеры будущей кнопки (рис. 4). Автоматически появится диалоговое окно создания кнопок, где мы можем выбрать категорию и действие (рис. 5). Например, найти запись, и нажимаем Далее. На кнопку мы можем поместить либо текст, либо соответствующее стандартное изображение или иконку по обзору (рис. 6). Щелкнем далее и мы увидим, что на следующем этапе задается имя кнопки, после чего щелкнем Готово (рис. 7). Кнопка появилась, мы можем подкорректировать размеры и перейти в режим *Форма* для того, чтобы увидеть, как кнопка будет отражена на форме, и опробовать ее действие (рис. 8). Щелкнем на ней; да, действительно, как и должно было, появилось окно *Поиск и замена* (рис. 9). Пусть образцом будет фамилия Блок, поиск пусть осуществляется в категории «музей-квартиры Санкт-Петербурга», совпадение – с любой частью поля и просматриваются все записи (рис. 10). Щелкнем далее, действительно кнопка работает правильно, и нужная нам информация в таблице найдена (рис. 11).

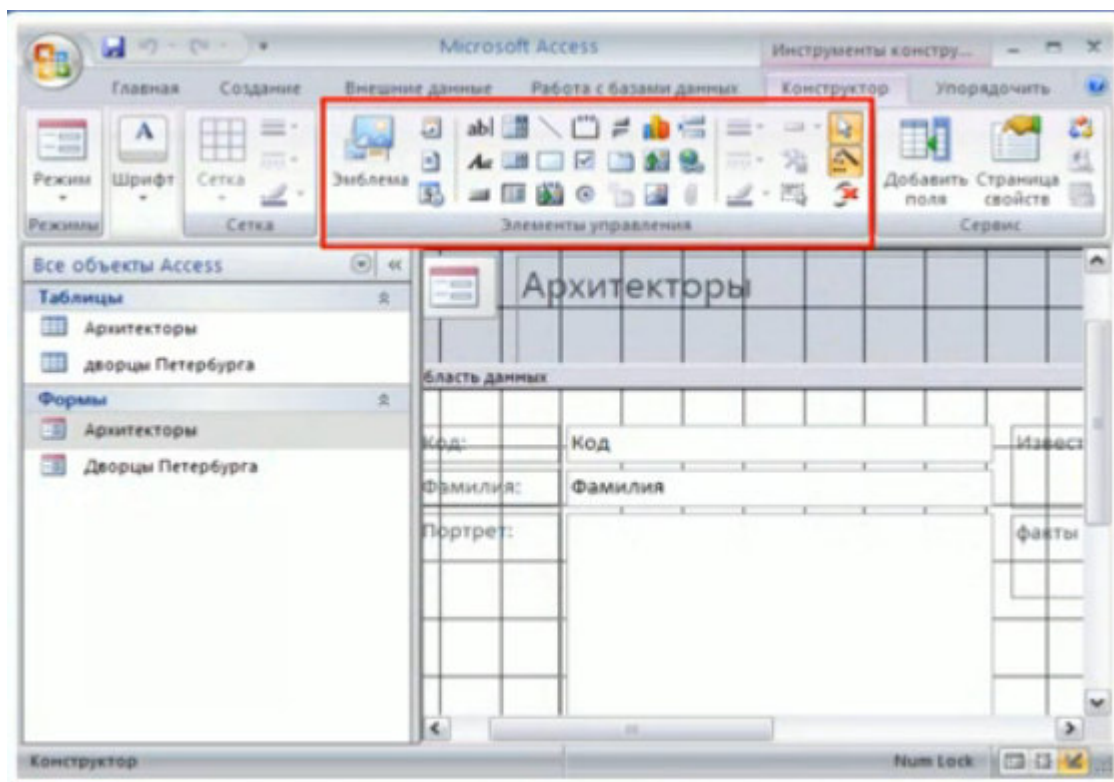


Рис. 2

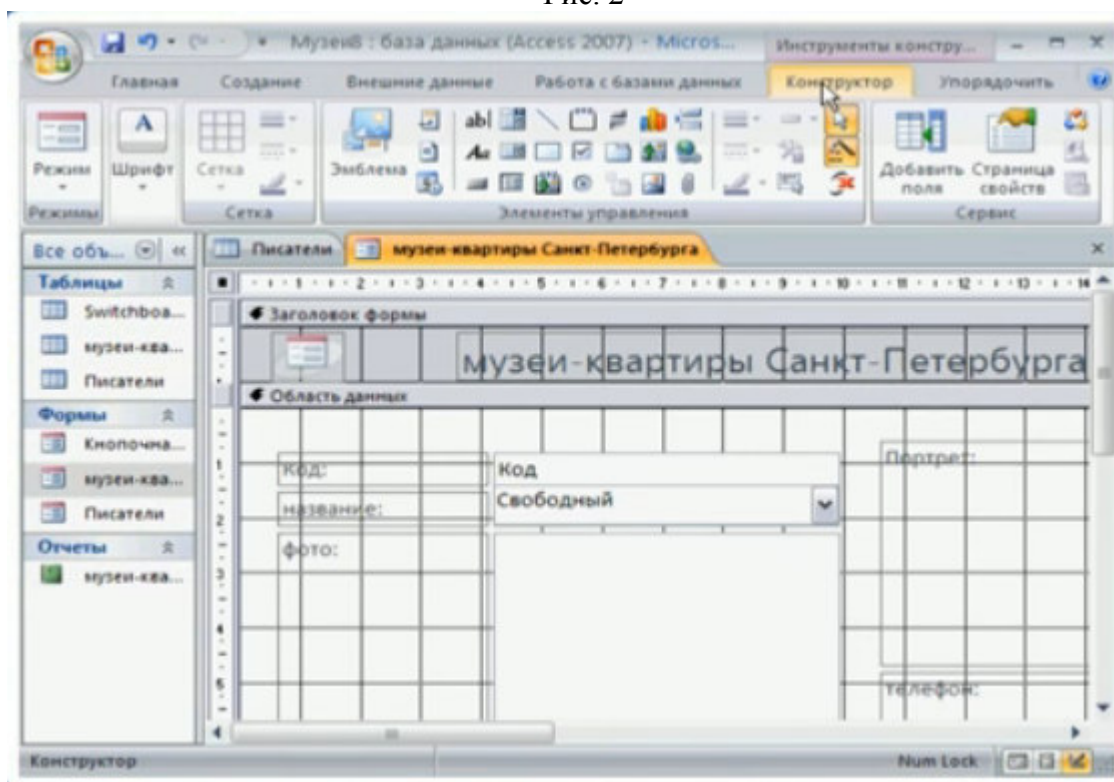


Рис. 3

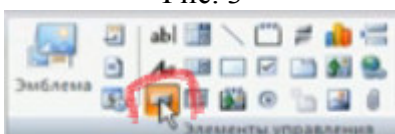


Рис. 4

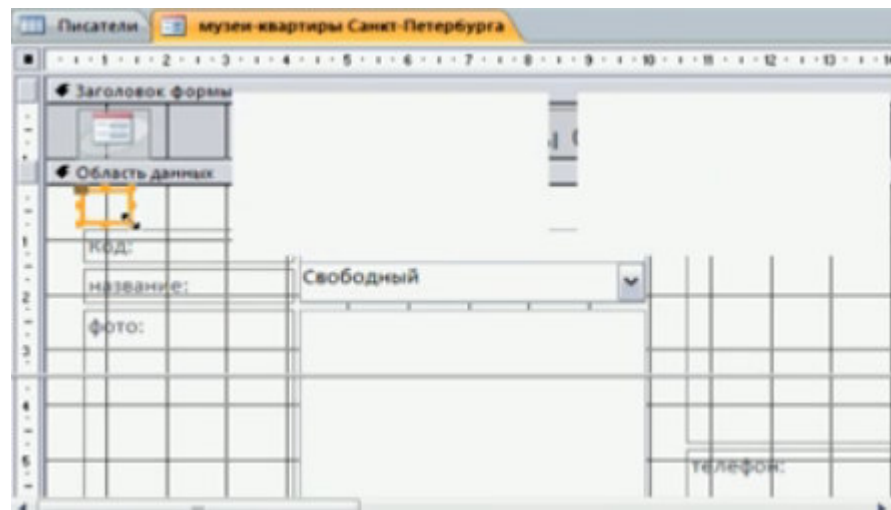


Рис. 5

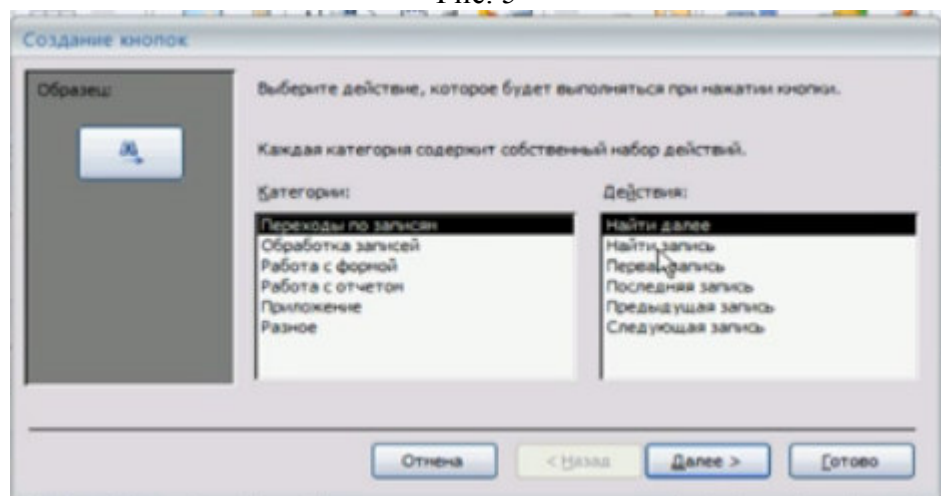


Рис. 6

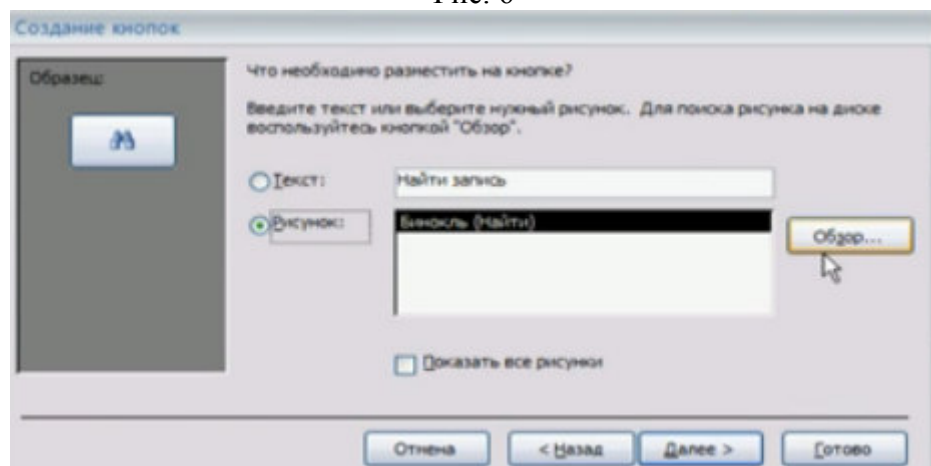


Рис. 7

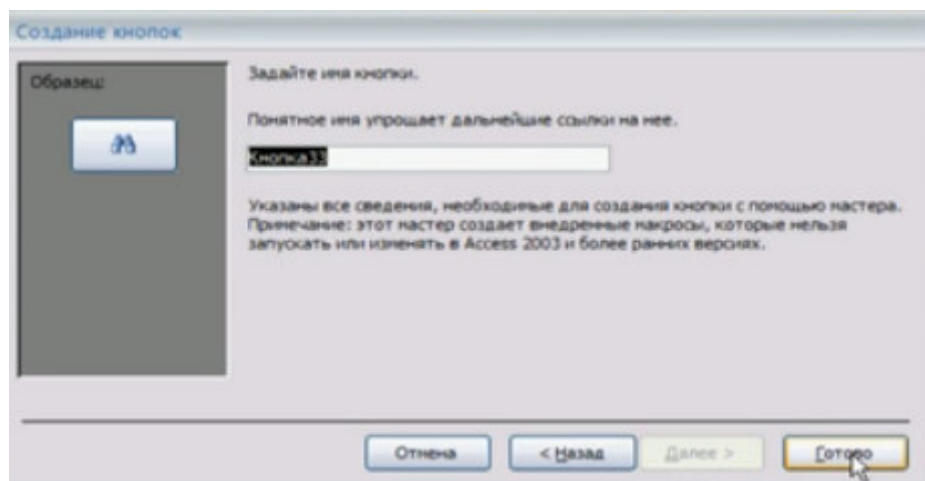


Рис. 8

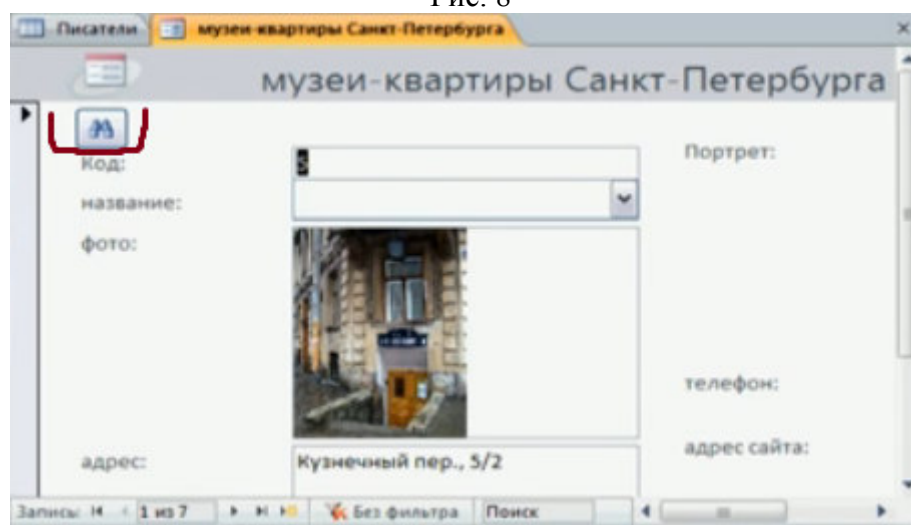


Рис. 9

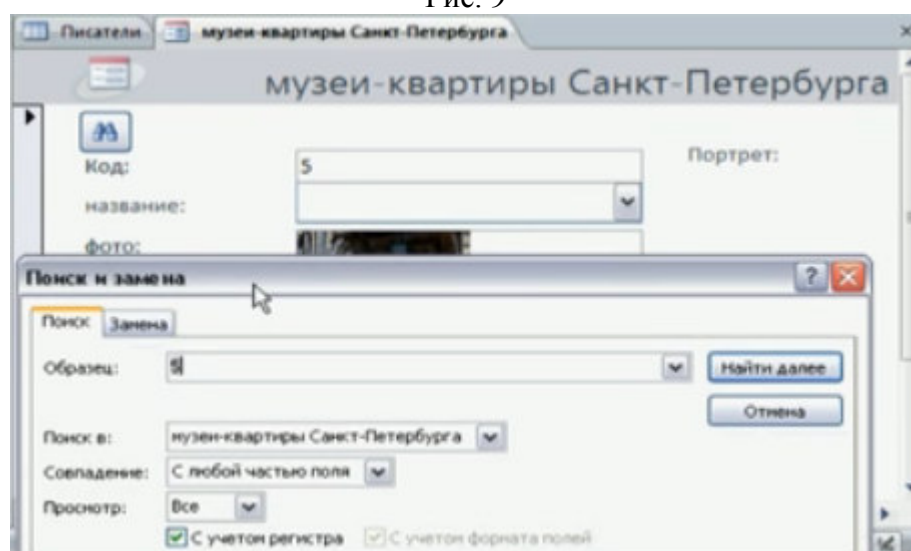


Рис. 10



Рис. 11

Добавление списка на форму

Кроме командных кнопок на форму можно также помещать списки. Выбор из списка обеспечивает более быстрый переход к нужной записи. Давайте посмотрим, как на форме можно создать выпадающий список.

Создадим еще один элемент управления на форме, перейдем в режим *Конструктор* и выберем элемент управления *Поле со списком* (рис. 12), обозначим его расположение и размеры (рис. 13). После этого появится диалоговое окно *Создание полей со списком*, в котором мы можем выбрать, каким образом поле со списком будет получать значение, выберем, например, третий вариант и щелкнем далее (рис. 14). Выберем поле *Название* и щелкнем Далее (рис. 15) и Готово. Но мы видим, что название поле уже существовало, получилось наложение, но мы можем просто удалить второе название. Перейдем в режим *Форма*, для того чтобы опробовать список. Список открывается, мы выбираем название *музей-квартира А.С. Пушкина* (рис. 16) и видим, что происходит переход на ее картинку (рис. 17). Т. е. поле со списком работает.



Рис. 12

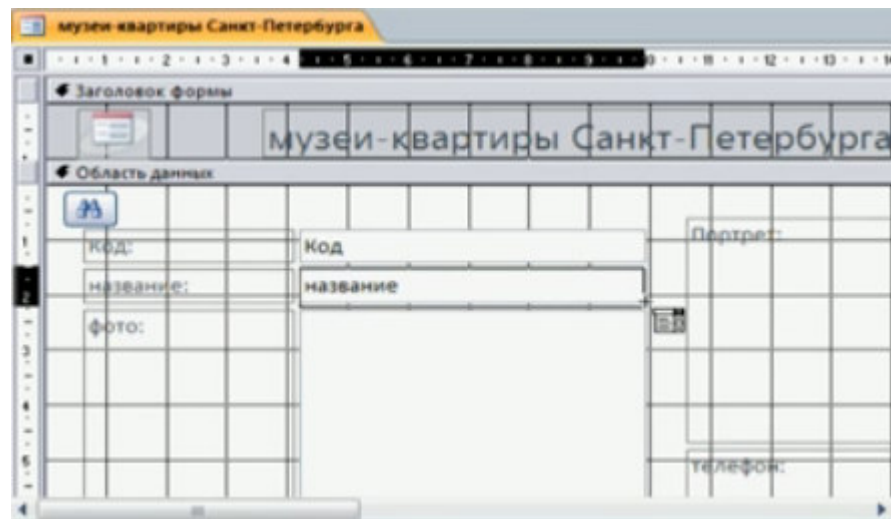


Рис. 13

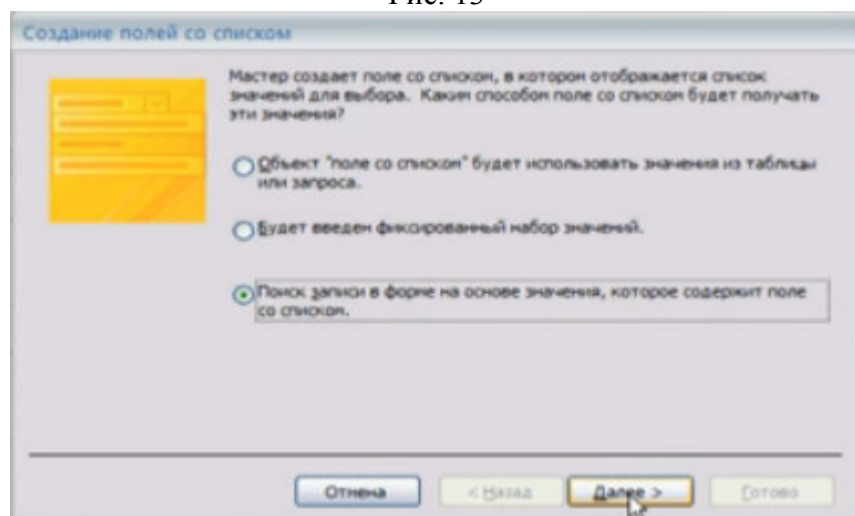


Рис. 14

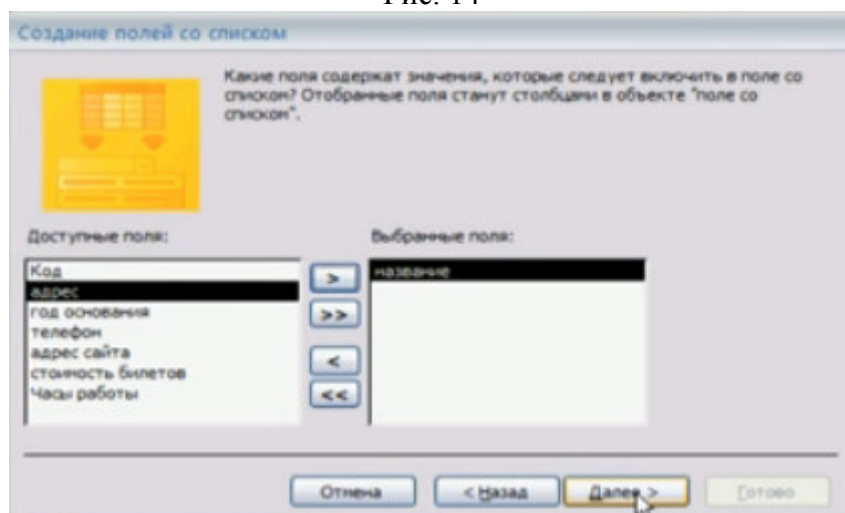


Рис. 15

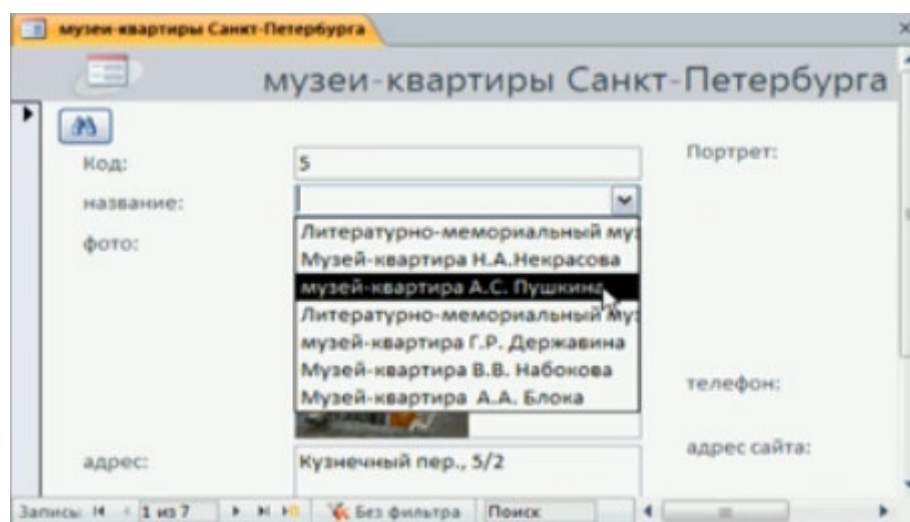


Рис. 16

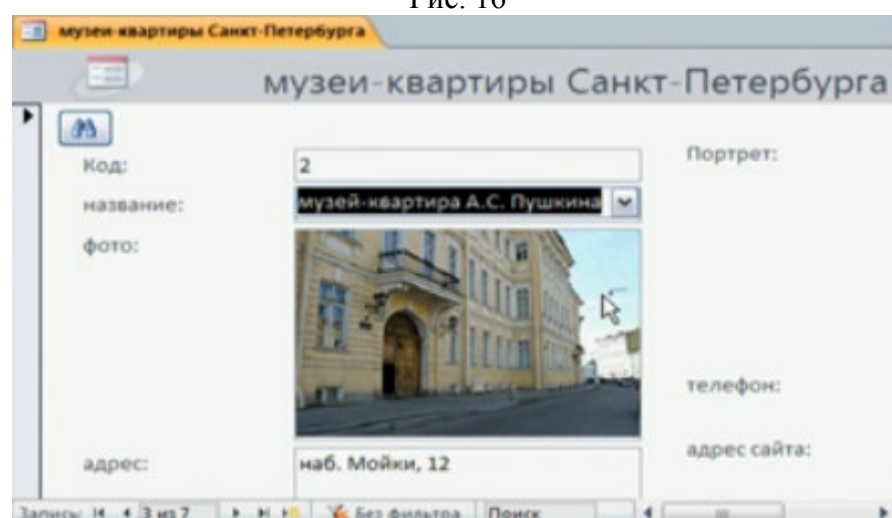


Рис. 17

3. Связывания нескольких форм и таблиц базы данных, создание главной кнопочной формы

Если в базе данных не одна таблица, а несколько и, соответственно, несколько форм, то можно создать **главную кнопочную форму**, содержащую кнопки, причем щелчком на кнопки мы можем загружать ту или иную форму.

Давайте посмотрим, из каких этапов состоит создание главной кнопочной формы.

Если в базе данных несколько таблиц и соответствующее количество форм, то имеет смысл создать главную кнопочную форму, которая будет служить своеобразной заставкой для базы данных, в которой можно загружать все остальные формы. Щелчком на ленте вкладку *Работа с базами данных* (рис. 18), выберем там команду диспетчер кнопочных форм, при этом выдается сообщение: *Не удастся найти кнопочную форму в этой базе данных*, поэтому мы нажимаем кнопку *Создать* (рис. 19). Появится диалоговое окно *Диспетчер кнопочных форм*, в котором мы можем выбрать команду *Изменить*, появится окно *Изменение страницы кнопочной формы*. Создадим на ней первую кнопку (рис. 20). Пусть текст этой кнопки будет *музеи-квартиры* и будет выполняться коман-

да *Открыть форму для изменения*, а форма – *музеи-квартиры Санкт-Петербурга*. Создадим еще одну кнопку, которую назовем *писатели*, по щелчку также будет выполняться команда *Открыть форму для изменения* и форма *писатели* (рис. 21). Закроем открытые диалоговые окна, и мы увидим в списке объектов, что появились новые формы и соответствующие им таблицы (рис. 22).

Откроем кнопочную форму. По умолчанию она имеет вот такое оформление (рис. 23), на ней находятся две кнопки, при нажатии на первую кнопку происходит переход на форму *музеи-квартиры Санкт-Петербурга*, щелчком на второй мы откроем форму *писатели*, т. е. кнопочная форма работает правильно. Теперь попробуем изменить ее стандартное оформление, перейдем в режим *Конструктор*, и мы здесь увидим окно свойств. Таким образом, мы можем выбрать любой из элементов и изменить, например, цвет фона (рис. 24). А затем мы можем поместить, например, изображение (рис. 25). Для этого выберем произвольное изображение на нашем компьютере и перейдем в режим *Форма*. Теперь наша главная кнопочная форма приняла следующий вид (рис. 26). У вас может появиться не обязательно изображение компьютера, а тот рисунок, который вы выбрали на предыдущем этапе. Можем отметить, что наше оформление изменилось к лучшему.

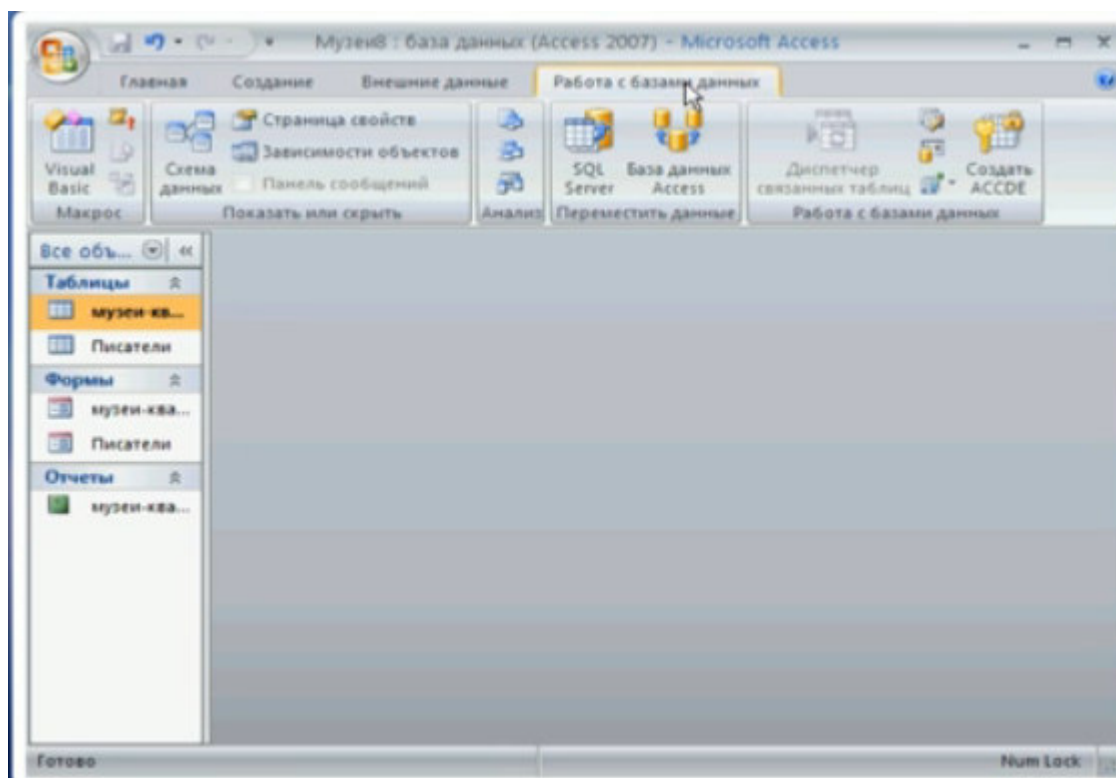


Рис. 18

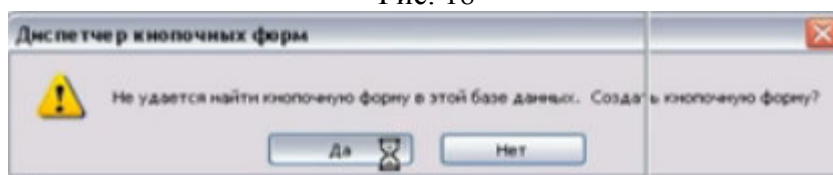


Рис. 19

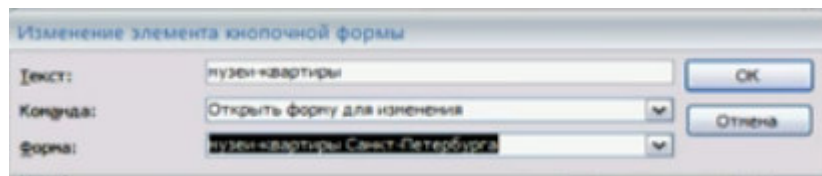


Рис. 20



Рис. 21

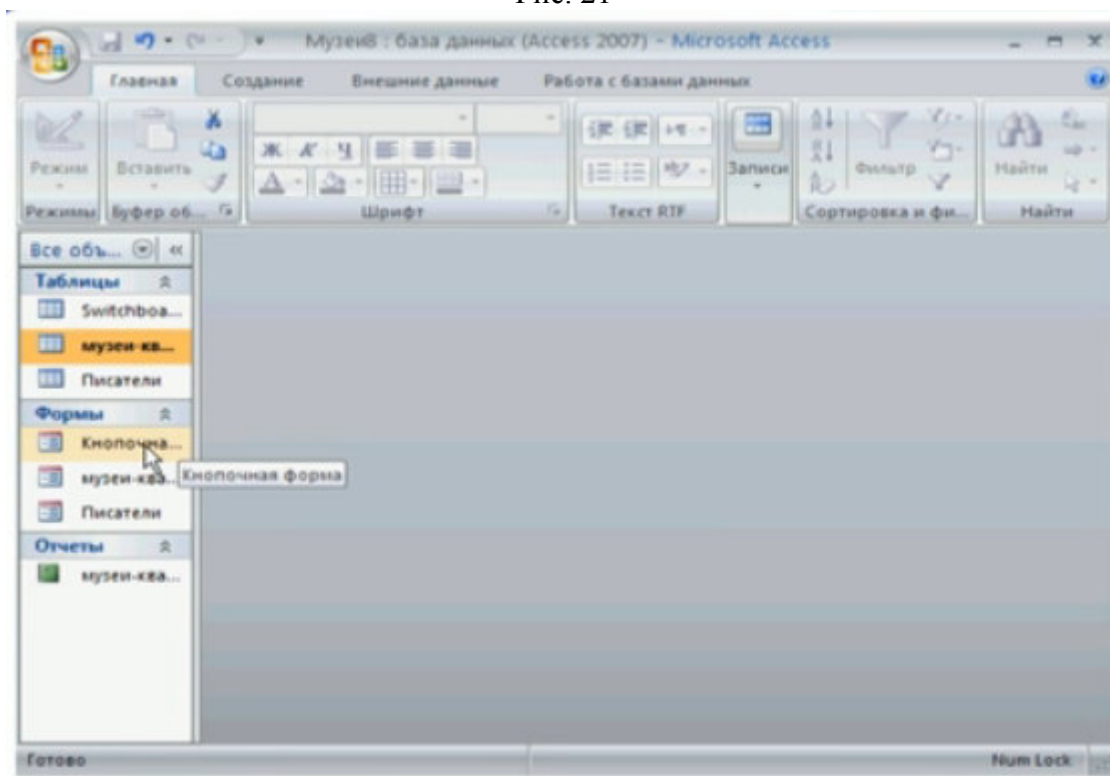


Рис. 22

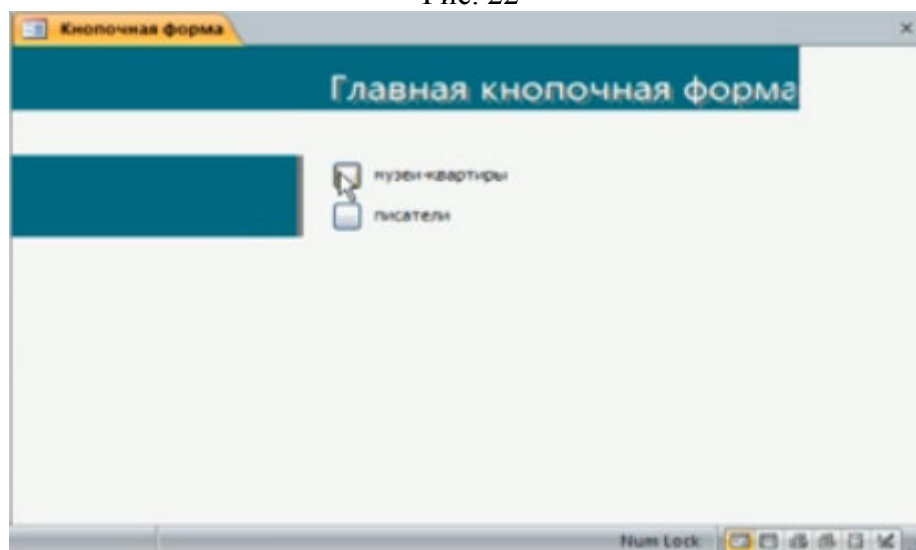


Рис. 23

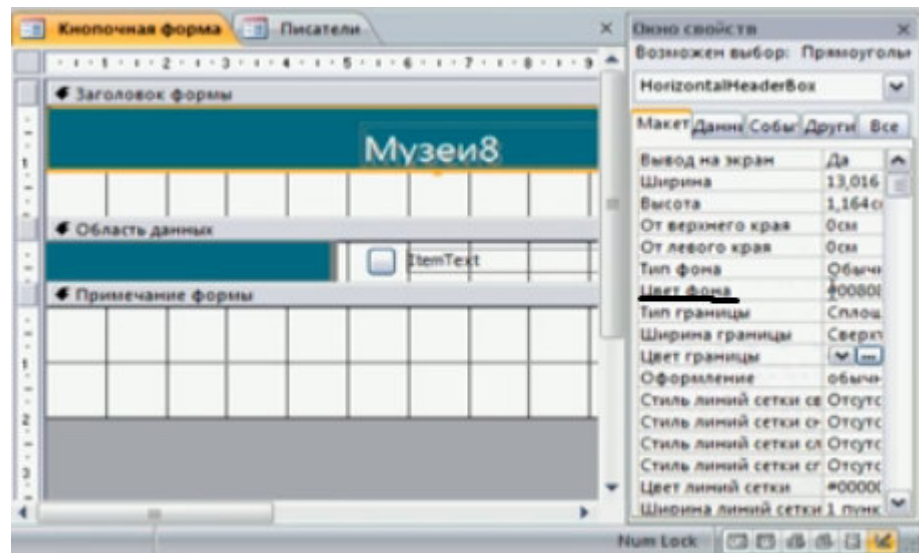


Рис. 24

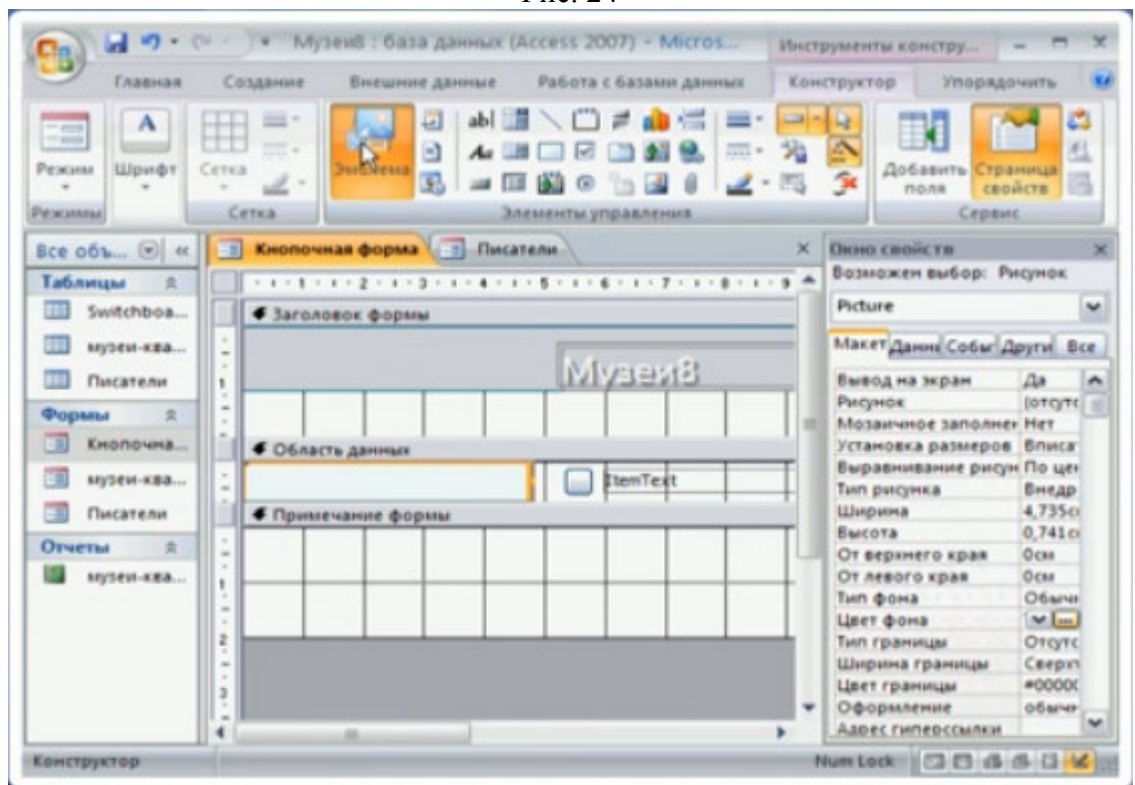


Рис. 25

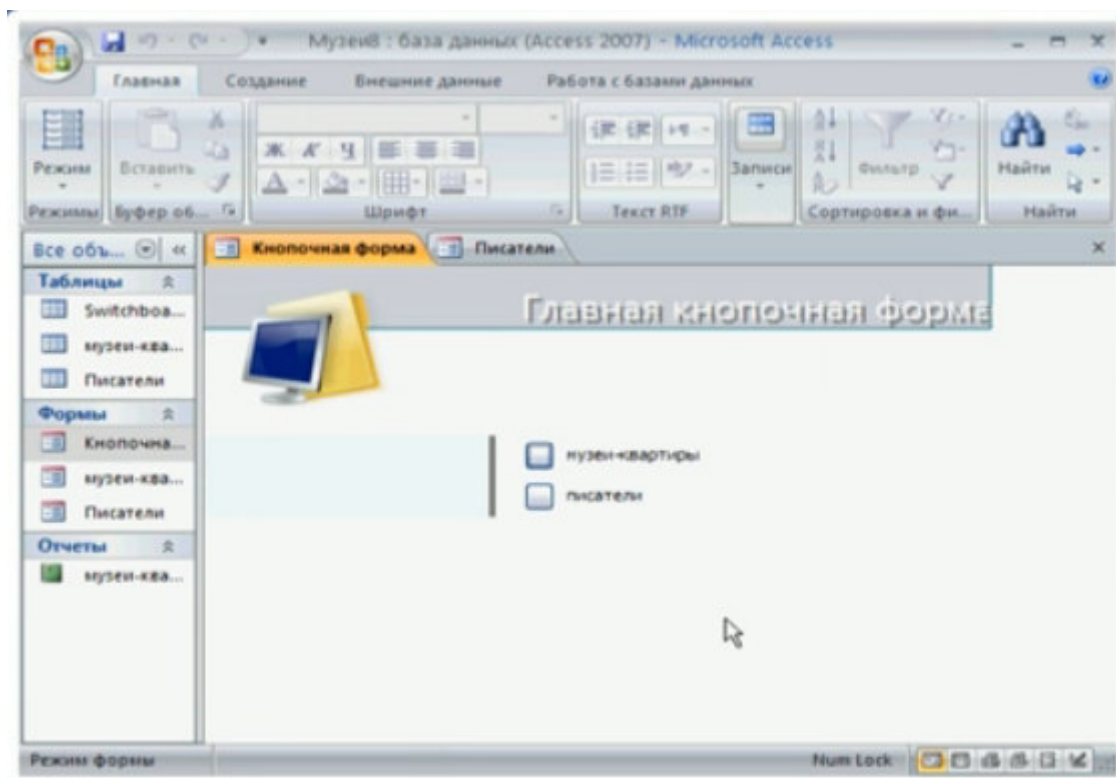


Рис. 26

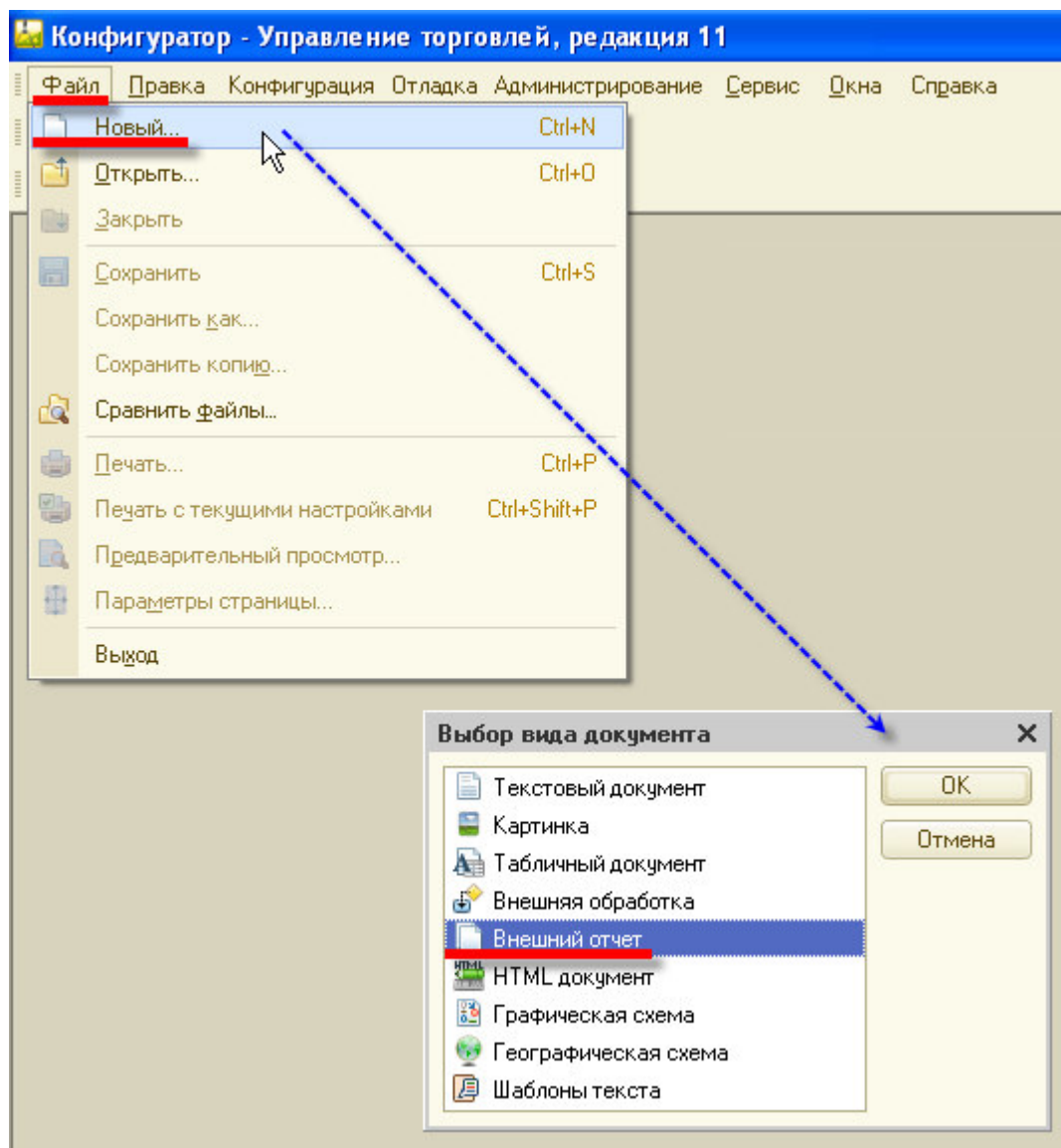
2.3 Внешние отчеты

При изучении вопроса необходимо обратить внимание на следующие особенности:

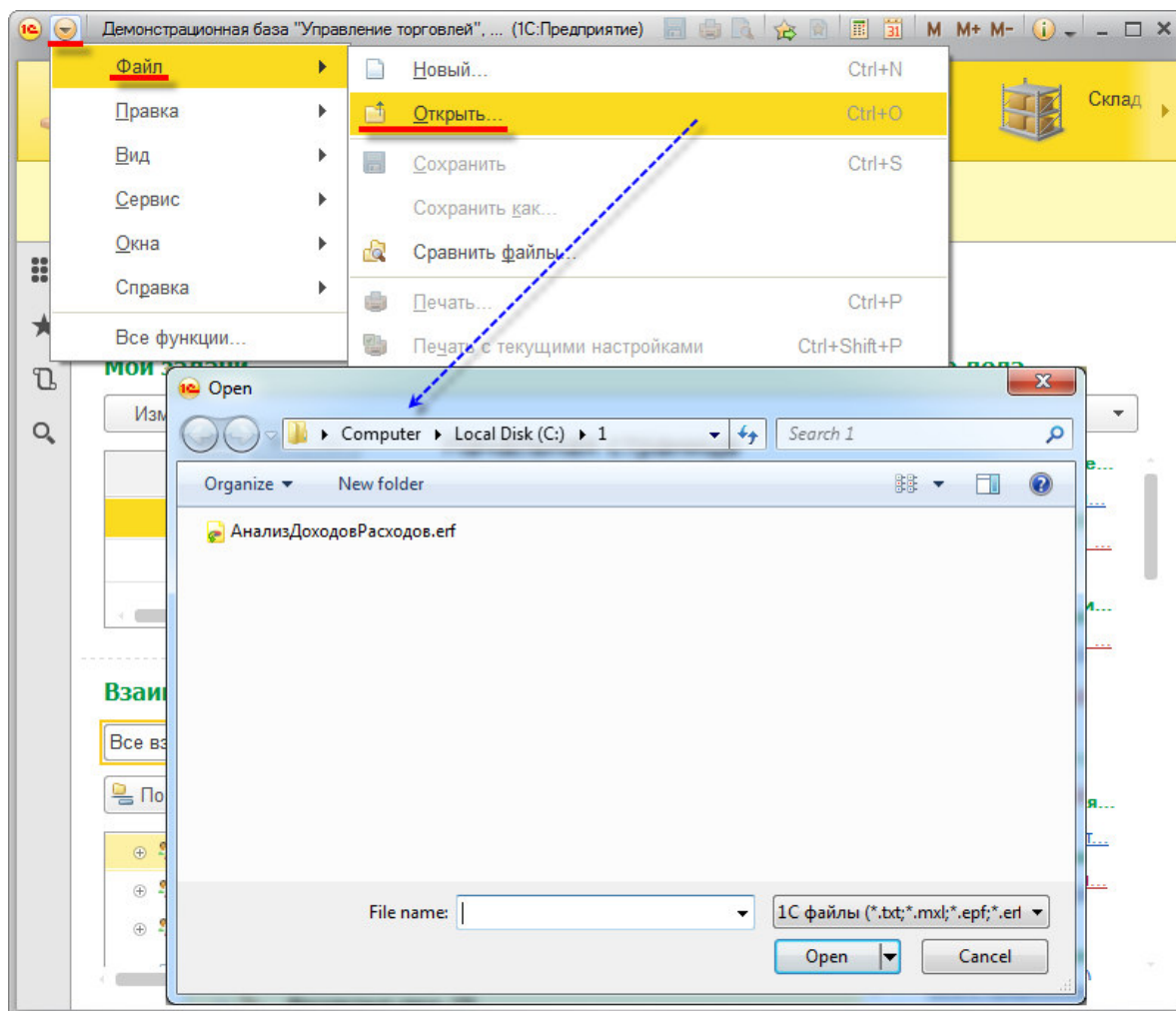
Внешние отчеты представляют собой отчеты, которые не входят в состав прикладного решения и хранятся в отдельных файлах с расширением *.erf.

Основное их преимущество заключается в том, что такие отчеты можно использовать в различных прикладных решениях без изменения структуры самих решений. Кроме того, важным преимуществом внешних отчетов является возможность проектировать и отлаживать их в процессе работы 1С:Предприятия, без необходимости сохранения каждый раз конфигурации прикладного решения.

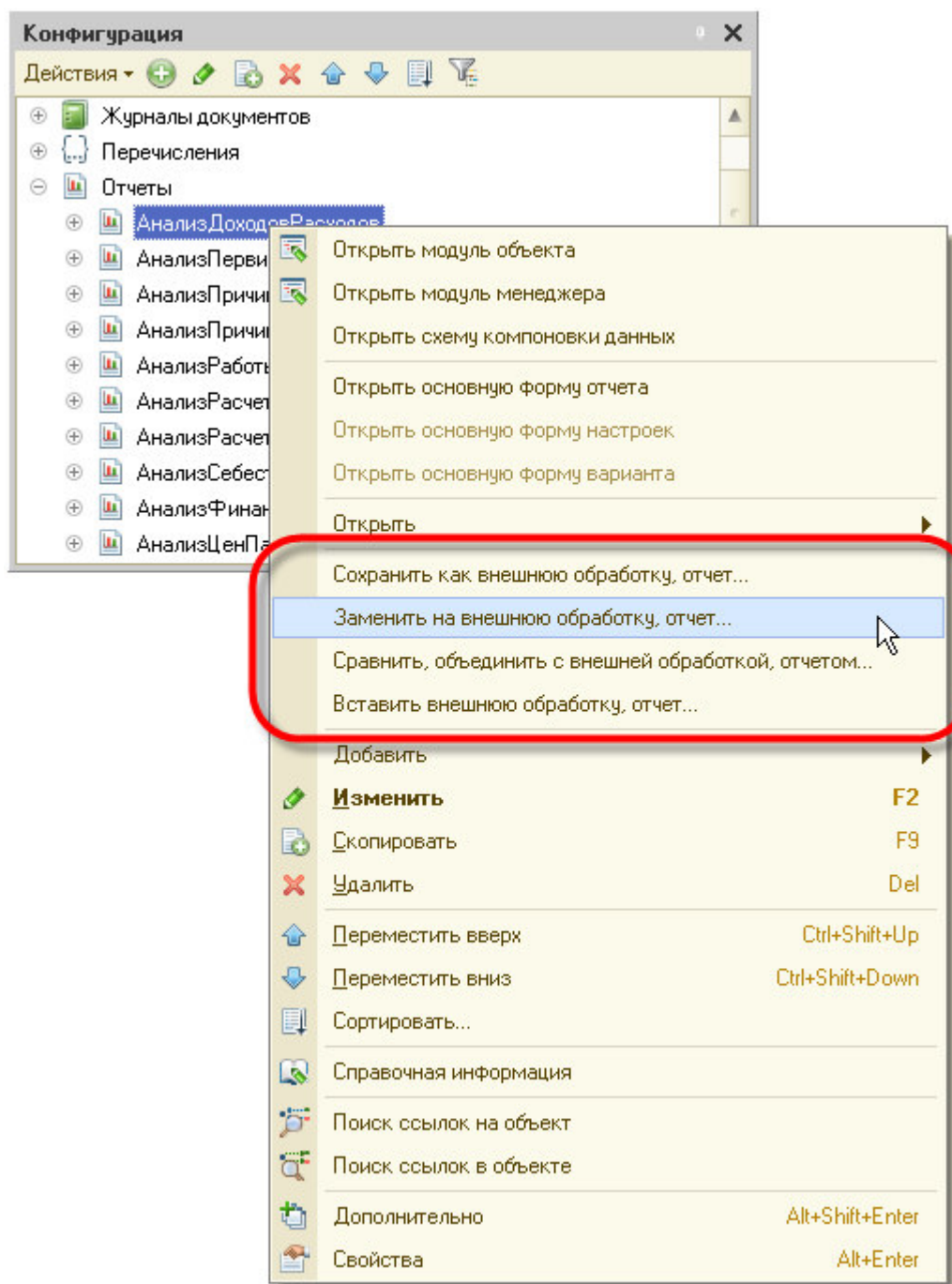
Создать внешний отчет можно в конфигураторе:



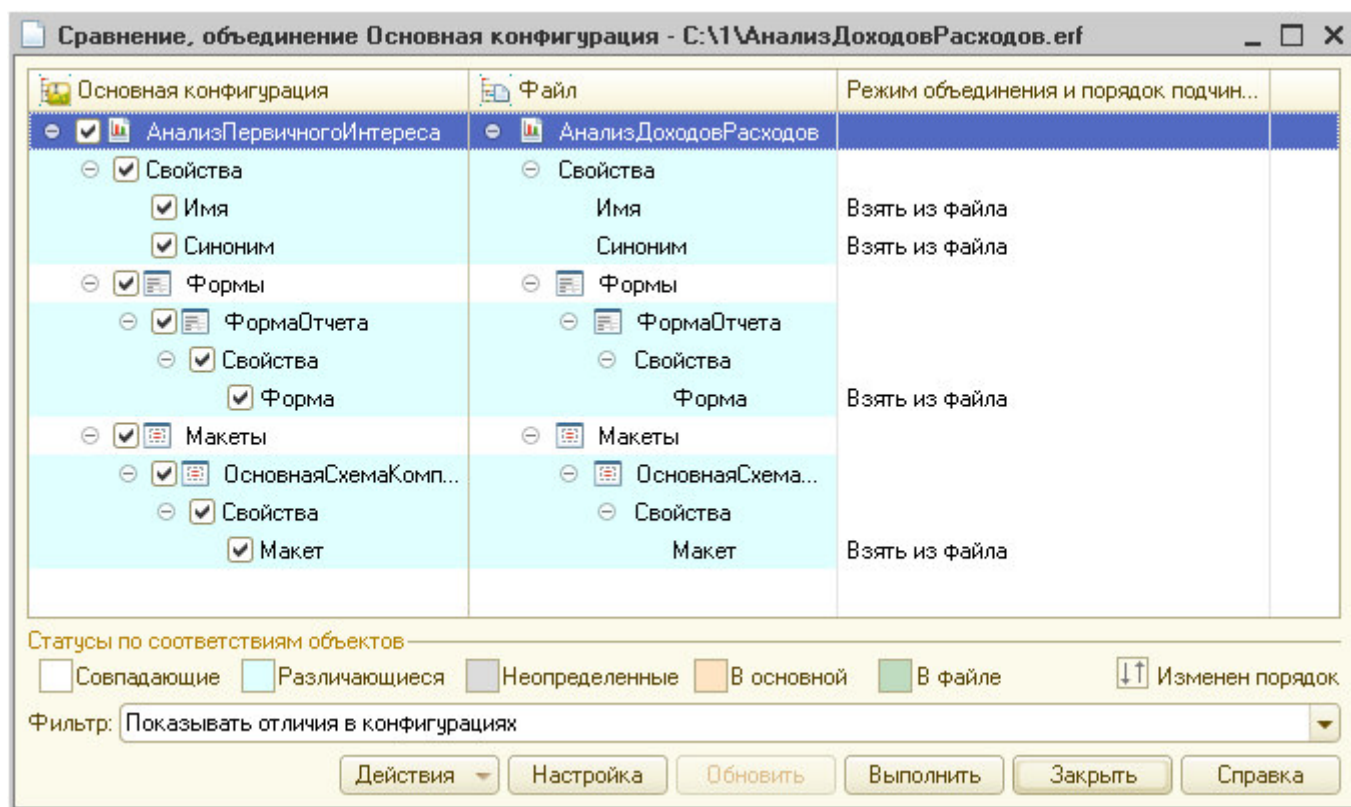
В режиме 1С:Предприятие внешний отчет можно запустить на выполнение, открыв его как любой другой файл, хранящийся на диске. Такой отчет будет работать точно так же, как если бы он входил в состав прикладного решения.



Любой существующий в конфигурации отчет или обработка может быть преобразован во внешний, и наоборот. Кроме этого внешние обработки и отчеты могут быть добавлены в структуру прикладного решения как новые объекты:



Конфигуратор поддерживает также возможность сравнения и объединения существующей в конфигурации обработки или отчета с внешней обработкой и отчетом, а также сравнение и объединение двух внешних отчетов или обработок:

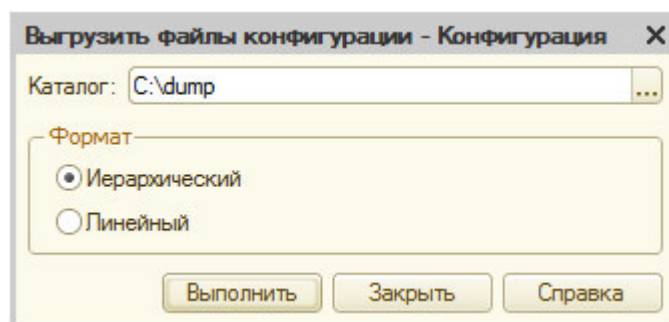


2.4 Выгрузка, загрузка и конфигурация базы данных

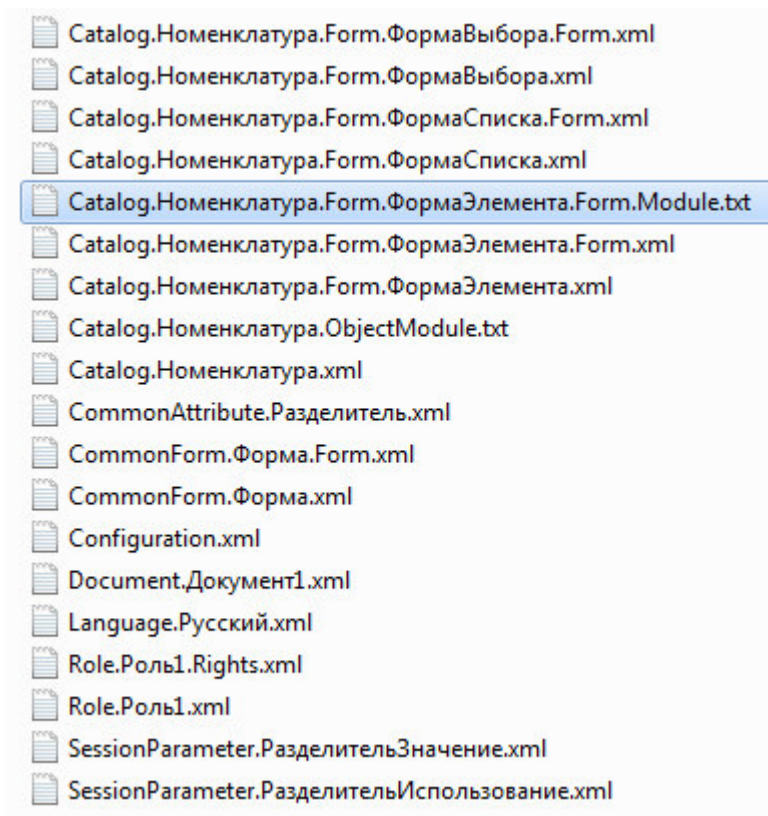
При изучении вопроса необходимо обратить внимание на следующие особенности:

Иерархическая выгрузка в файлы XML

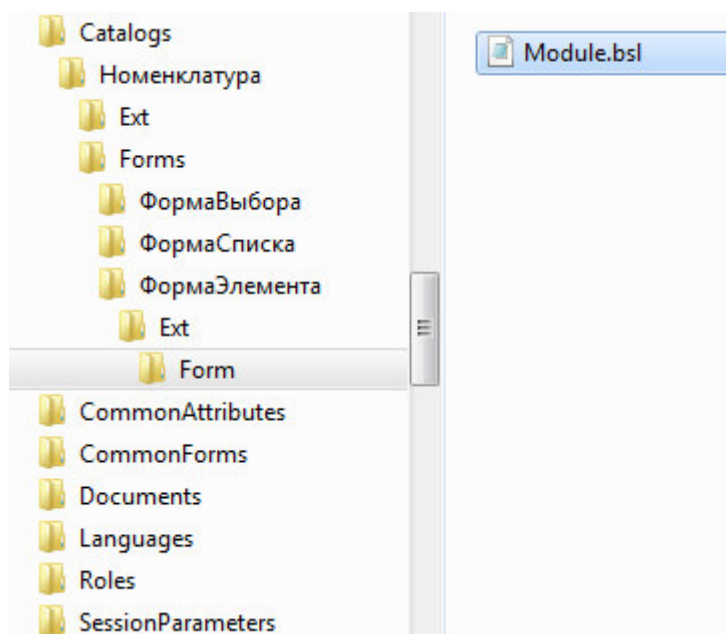
Для выгрузки конфигурации в XML мы добавили новый формат, - **Иерархический**. Теперь это стандартный формат, который предлагает платформа. Чтобы выгрузить файлы в прежнем формате, нужно это указать в явном виде:



В отличие от старого формата, линейного, в котором полное имя объекта конфигурации содержалось в имени результирующего файла ...



... новый формат выгрузки, иерархический, формирует структуру каталогов с файлами:



Иерархическая выгрузка позволяет избавиться от проблем, связанных с очень длинными именами файлов. Такие проблемы могли возникать раньше как при выгрузке, так и при переносе файлов между разными файловыми системами (FAT, NTFS, EXT).

Сам по себе иерархический формат выгрузки не гарантирует того, что в выгрузке не появится очень длинных имён, или очень длинных путей. Поэтому мы ввели ряд ограничений и рекомендаций. Например, в конфигураторе нельзя создавать имена объектов длиннее 80 символов, не рекомендуется использовать подсистемы большой

вложенности, каталог выгрузки следует располагать как можно ближе к корню устройства и так далее.

Вы можете управлять форматом выгрузки при запуске конфигулятора в пакетном режиме. Для этого прежнему параметру **DumpConfigToFiles** мы добавили новую опцию **Format**. Если вы хотите выгрузить в старом, линейном формате, это нужно указать в явном виде:

```
"C:\Program Files (x86)\lcv8\8.3.7.1759\bin\lcv8.exe" DESIGNER /IBName
"TestBase" /DumpConfigToFiles "C:\dump" -Format Plain
```

Без указания этой опции выгрузка выполняется в иерархическом формате. А при загрузке формат определяется автоматически, и никаких дополнительных опций не требуется.

Частичная загрузка конфигурации из файлов XML

Мы реализовали возможность загружать из файлов XML не всю конфигурацию, а только её часть. В первую очередь эта возможность востребована в новой среде разработки **1C:Enterprise Development Tools**. Ведь Development Tools ориентированы на работу с крупными конфигурациями, а частичная загрузка помогает ускорить процесс разработки, сократить цикл «редактирование - отладка».

Однако вы можете использовать эту возможность и независимо от Development Tools. Потому что для загрузки отдельных файлов конфигурации используется запуск конфигулятора из командной строки в пакетном режиме. А значит, используя частичную загрузку, вы можете:

- изменять свойства конфигурации,
- добавлять, изменять и удалять объекты конфигурации,
- загружать только некоторые свойства объектов конфигурации без загрузки самих объектов. Например, модули объектов, формы, модули форм, роли и так далее.

Мы сразу хотим обратить ваше внимание на то, что мы реализовали только частичную загрузку, и только из командной строки. Частичной выгрузки нет так же, как нет интерактивных команд конфигулятора, позволяющих загружать часть конфигурации.

Поэтому загрузить только модуль справочника Номенклатура вы можете, например, следующей командой:

```
"C:\Program Files (x86)\lcv8\8.3.7.1759\bin\lcv8.exe" DESIGNER /IBName
"TestBase" /LoadConfigFromFiles "C:\dump" -Files
"C:\dump\Catalogs\Номенклатура\Ext\ObjectModule.bsl"
```

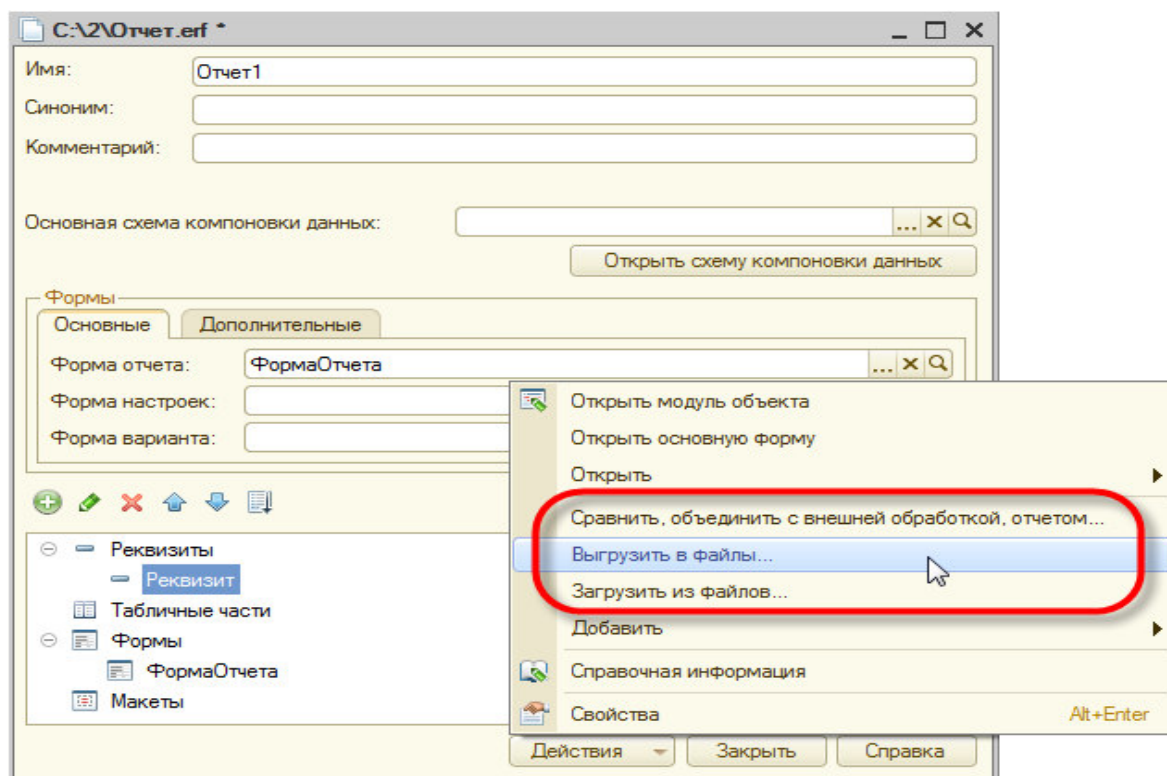
Для частичной загрузки используется прежний параметр **LoadConfigFromFiles**, и две новых опции: **Files** и **ListFiles**. **Files** позволяет вам перечислить через запятую те файлы, которые нужно загрузить, если таких файлов немного. А если их много, тогда вы можете использовать опцию **ListFiles**. Она указывает на файл, в котором перечислены XML файлы, которые нужно загрузить.

Кроме этого, для повышения удобства работы, мы разрешили совместное использование в одной строке параметров **LoadConfigFromFiles** и **UpdateDBCfg**. Таким

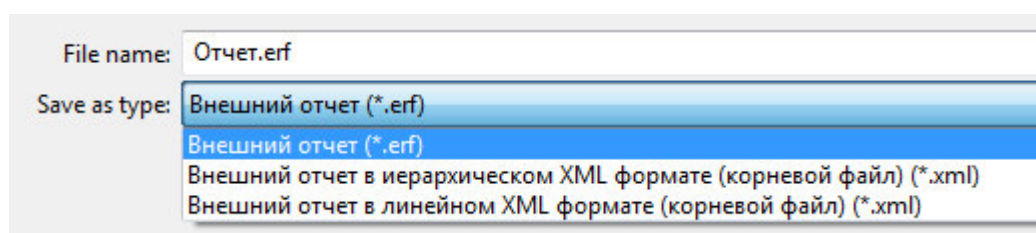
образом, теперь за один вызов вы можете загрузить изменения и принять их (обновить конфигурацию базы данных).

Выгрузка/загрузка внешних отчётов и обработок в/из XML

В версии 8.3.8 мы добавили возможность выгружать в XML и загружать внешние отчёты и обработки:



Более того, при работе в конфигураторе вы можете сохранять их сразу в формате XML (Файл - Сохранить как...). То же самое относится и к открытию:



Также мы добавили возможность сравнить внешний отчёт или обработку с XML выгрузкой. Все эти изменения мы сделали в первую очередь для того, чтобы в новой среде разработки **1C:Enterprise Development Tools** обеспечить полноценную работу с внешними отчётами и обработками. Однако и отдельно от Development Tools эти возможности могут быть вам полезны для любых автоматизированных изменений выгруженных XML файлов.

Выгрузить/загрузить внешние отчёты/обработки вы можете не только интерактивно, но и автоматически, запуская конфигуратор в пакетном режиме. Для этого мы добавили два новых параметра: **Dump External Data Processor Or Report To Files** и **Load External Data Processor Or Report From Files**.

3. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ПОДГОТОВКЕ К ЗАНЯТИЯМ

3.1 Лабораторная работа №1 Создание новой информационной базы данных

При подготовке к занятию необходимо обратить внимание на следующие особенности.

Если объекту установить имя предопределенного элемента из метаданных, то он станет предопределенным элементом, связанным с метаданными. В пределах одной области информационной базы допустимо использование только одного объекта данных, связанных с конкретным предопределенным элементом метаданных.

3.2 Лабораторная работа №2 Классификация объектов конфигурации

При подготовке к занятию необходимо обратить внимание на следующие особенности.

Общие объекты играют вспомогательную, с точки зрения разработки прикладного решения роль. Результат их работы интересен скорее программисту, чем пользователю, хотя нельзя сказать, что эти объекты не участвуют в работе пользователя или неважны для целей решения прикладной задачи. Они - важная часть конфигурации.

3.3 Лабораторная работа №3 Введение в язык запросов

При подготовке к занятию необходимо обратить внимание на язык запросов.

При подготовке к занятию необходимо обратить внимание на следующие особенности.

Программа на языке SQL представляет собой простую линейную последовательность операторов языка SQL. Язык SQL в своем чистом виде операторов управления порядком выполнения запросов к БД (типа циклов, ветвлений, переходов) не имеет.

3.4 Лабораторная работа №4 Роли и права пользователей

При подготовке к занятию необходимо обратить внимание на следующие особенности.

В типовых решениях 1С Предприятия для платформы 8.1 предусмотрено разграничение прав доступа пользователей к информации. С одной стороны, это обеспечивает определенную степень информационной безопасности предприятия (пользователю доступна только информация необходимая для выполнения его задач), с другой - облегчает работу пользователям информационной системы (пользователь не видит лишней информации, которая может препятствовать качественному и

своевременному выполнению своей работы и оперировать только необходимыми ему данными).