

Лабораторная работа №2

Справочники. Иерархия справочников. Управление интерфейсом. Модуль формы

Цель работы – изучить основные директивы компиляции «1С: Предприятие 8.2», получить практические навыки работы по созданию справочников, перечислений, групп и печатных форм.

Теоретические сведения

Справочники

Предназначены для хранения множества характеристик некоторого объекта реального мира. Например, информации об учащихся и преподавателях, товарах и услугах и др. Название и состав конкретного справочника определяется при его создании в конфигураторе.

Справочник состоит из элементов. Например, для справочника преподаватели элементом является конкретный преподаватель, для справочника услуги - услуга и т. д. Пользователь при работе в системе сам формирует список элементов в справочнике: например, добавляет новых преподавателей, а также корректирует их данные или удаляет из списка данные о преподавателе.

Справочники могут быть иерархическими и подчиненными. Справочник может содержать при необходимости дополнительный состав реквизитов (дополнительных данных), табличных частей и других настроек.

Иерархия элементов

Иерархия — это возможность установить связи между записями в рамках одного справочника.

После выбора «Иерархический справочник» в стандартных реквизитах появиться поле «Родитель», кроме это в режиме отладки при создании новой записи в справочнике также появиться поле «Родитель. Тип иерархии «Иерархия элементов» означает, что любая запись в справочнике может быть «родителем» для другой записи, при этом платформа отслеживает, чтобы записи не были «закольцованы».

Перечисления

Предназначены для создания в системе неизменяемого пользователем списка элементов. Например, список форм обучения студентов, пола человека и т.п.

Перечисления создаются с использованием панели конфигуратора, после чего в разделе «Данные» формируется необходимый перечень значений объекта «Перечисления». При необходимости в процессе программирования объекты перечисления могут использоваться в коде программы обработки событий.

Иерархия групп

При установке флага «иерархия групп» в справочнике отобразится дополнительное поле «ЭтоГруппа» с типом «Булево». В данном виде иерархии все записи справочника делятся на группы и элементы, которые могут иметь различные свойства. Поэтому «родителем» будут группы.

Подчиненные справочники

Для подчиненного справочника в поле «Владелец» содержится ссылка на запись из другого справочника. Система не запишет элемент подчиненного справочника, если владелец не будет выбран. Рекомендуется заполнять подчиненный справочник, начиная с объекта, который является владельцем.

Табличные части

Если в справочнике один из элементов должен содержать несколько данных, например каждый преподаватель может иметь несколько членов семьи, то необходимо создавать так называемую табличную часть, которая по своей сути является дополнительной таблицей в базе данных системы.

Печатные формы

Реализация печатных форм основывается на объектах конфигурации макет и табличный документ. Макет представляет собой набор именованных областей, в которых определены элементы оформления и набор параметров. При реализации алгоритма печати в процедуре создается «ТабличныйДокумент». Из макета берется текстовая часть, а выбранные данные для печати записываются в параметры области. Заполненная данными область включается в табличный документ.

Директивы компиляции

Поскольку настройка конфигурации невозможна без написания собственного кода, то необходимо использовать директивы компиляции.

Создаваемая разработчиком информационной системы подпрограмма (процедура/функция) или описание переменной «Модуля формы» должны начинаться одной из нижеприведенных директив компиляции:

&НаКлиенте – подпрограмма выполняется на стороне клиента, а переменная существует все время жизни клиентской части управляемой формы. Из клиентского метода допустимыми являются вызовы клиентских, серверных и серверных внеконтекстных методов.

Переменная, описанная в процедуре или функции существует пока выполняется эта процедура или функция.

&НаСервере – подпрограмма выполняется на стороне сервера, а переменная существует только во время выполнения серверного или серверного внеконтекстного вызова. Для серверных методов допустимыми являются вызовы просто серверных и серверных внеконтекстных методов.

&НаСервереБезКонтекста – подпрограмма выполняется на сервере вне контекста формы.

При выполнении заданий лабораторной работы №1 была создана такая функция, т.к. клиент не может напрямую взаимодействовать с сервером.

Переменные не могут быть внеконтекстными. В таких методах недоступен контекст формы (включая данные формы). Допустимыми являются вызовы только других внеконтекстных методов.

Примером, такой работы может быть использование браузера в режиме инкогнито.

При вызове этих подпрограмм передача данных формы на сервер и обратно не выполняется. Применение внеконтекстных подпрограмм позволяет сократить объем передаваемых данных при вызове серверной процедуры из клиентского приложения.

&НаКлиентеНаСервереБезКонтекста – процедура/функция может выполняться в управляемом клиенте или на сервере, при этом контекст формы не доступен.

По умолчанию, т.е. если директива компиляции отсутствует, используется директива «&НаСервере».

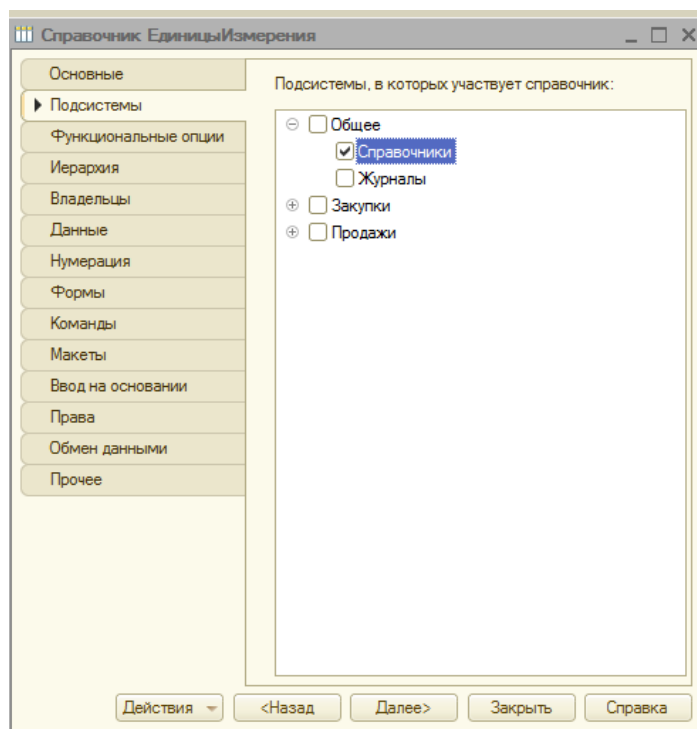
Директивы ограничивают количество передаваемой информации с целью уменьшения расхода ресурсов.

Контекст модуля – называется связь данного модуля с остальной частью конфигурацией. Это тот набор процедуры и функции, свойства и методы которые

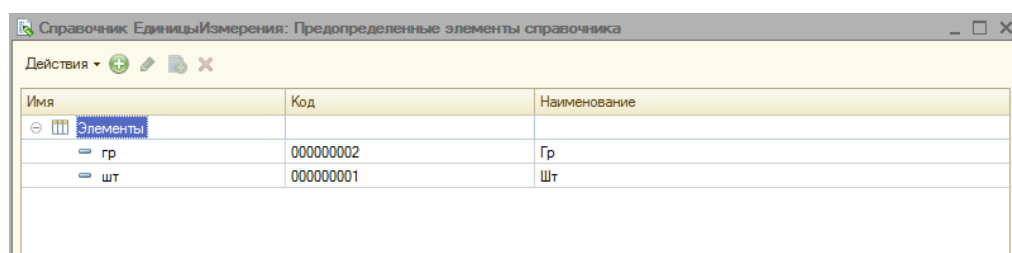
доступны во время выполнения модуля.

Порядок выполнения работы

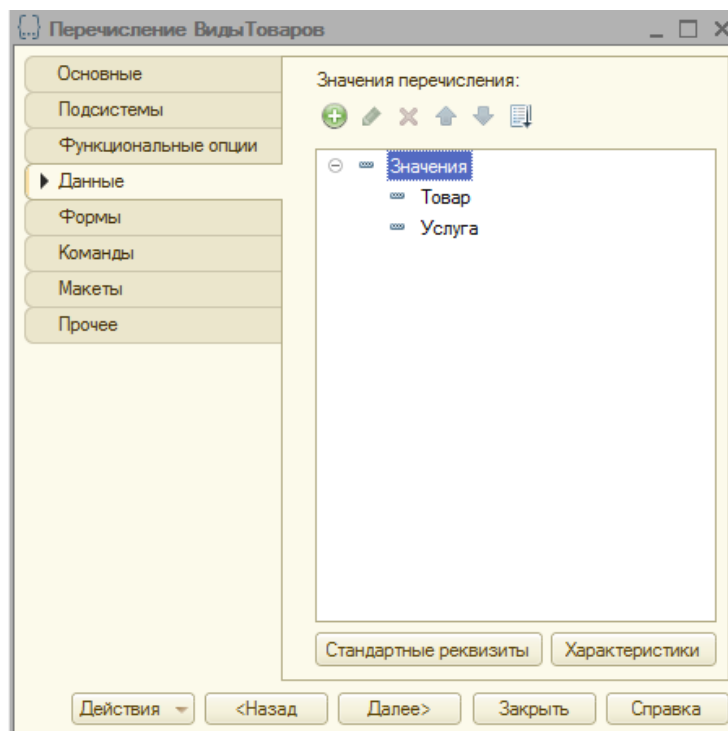
1. Создать справочник «ЕдиницыИзмерения».
2. Отнести его к подчиненной подсистеме «Справочники» подсистемы «Общее».



3. Во вкладке «Прочее» определить predetermined элементы справочника – «шт» и «гр».



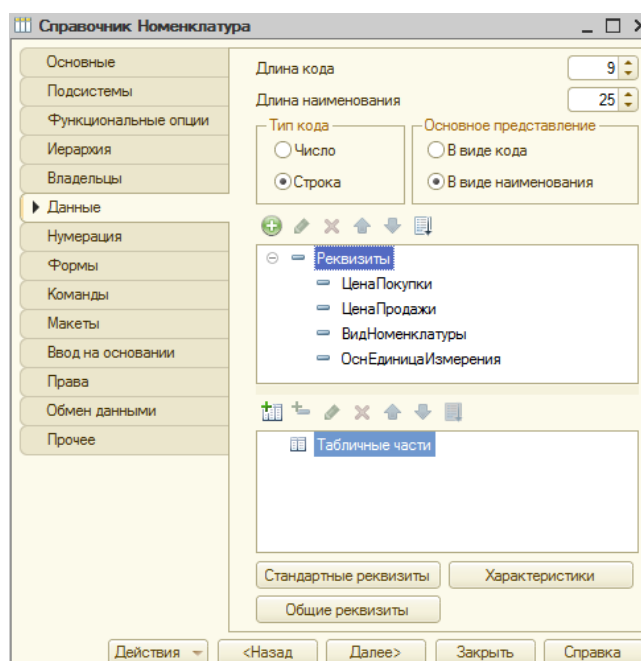
4. Создать иерархический справочник «Подразделения» с иерархией элементов. Отнести его к подчиненной подсистеме «Справочники» подсистемы «Общее».
5. Создать справочник «Склады». Отнести его к подчиненной подсистеме «Справочники» подсистемы «Общее».
6. Создать перечисление «ВидыТоваров» с данными «Товар» и «Услуга».



7. Создать иерархический справочник «Номенклатура» с иерархией групп и элементов. Отнести его к подчиненной подсистеме «Справочники» подсистемы «Общее».

8. Задать для данного справочника дополнительные реквизиты:

- ЦенаПокупки и ЦенаПродажи – неотрицательное число, длина 15, точность 2
- ВидНоменклатуры – ссылка на перечисление «ВидыТоваров»
- ОснЕдиницаИзмерения – ссылка на справочник «ЕдиницыИзмерения»



9. Создать для данного справочника форму элемента.

10. Добавьте в список элементов формы группу (тип «Группа – обычная группа») и переместите в нее элементы «ЦенаПокупки» и «ЦенаПродажи». Задайте в свойствах группы заголовок «Цены», а также задайте режим отображения «Рамка группы» и группировку «Горизонтальная».

The screenshot displays the 'Справочник Номенклатура: ФормаЭлемента' window. The left sidebar shows a tree structure with 'Цены' selected. The main area on the right shows a list of requisites with 'Объект' selected. The bottom pane shows a form with fields for 'Код', 'Наименование', 'Вид номенклатуры', 'Осн единица измерения', and a group 'Цены' containing 'Цена покупки' and 'Цена продажи'.

11. Создать справочник «Серии». Установить владельцем данного справочника справочник «Номенклатура». Отнести справочник к подчиненной подсистеме «Справочники» подсистемы «Общее».

12. Создать справочник «ФизическиеЛица». Задать для данного справочника дополнительные реквизиты (тип реквизитов должен соответствовать их смыслу):

- Фамилия.
- Имя.
- Отчество.
- ДатаРождения.

И табличную часть «ТрудоваяДеятельность» с реквизитами (тип реквизитов должен соответствовать их смыслу):

- Организация.

- Должность.
- НачалоРаботы.
- ОкончаниеРаботы.

Переопределить стандартный реквизит наименование, задав ему синоним «ФИО».

13. Создать форму элемента справочника «ФизическиеЛица».

14. Удалить элемент формы «Наименование». Создайте две страницы на форме (сперва необходимо добавить «Группа страницы», затем уже «Группа - страница»). На вторую страницу поместите табличную часть, на первую – остальные элементы. Внутри табличной части создайте обычную группу и поместите в неё дату начала и окончания работы.

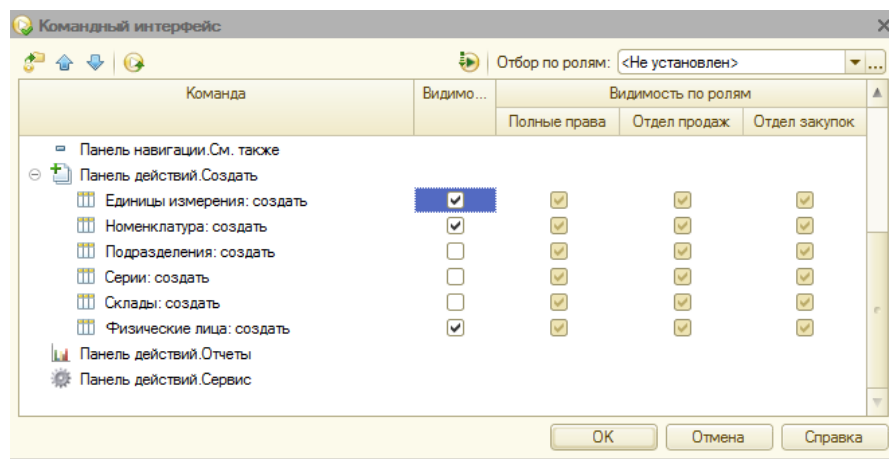
15. Перейдите в модуль формы и напишите следующий программный код.

```

&НаКлиенте
Процедура ФИОПриИзменении(Элемент)
    Стр = Элементы;
    фамилия = Стр.фамилия.ТекстРедактирования;
    Имя = Стр.Имя.ТекстРедактирования;
    Отчество = Стр.Отчество.ТекстРедактирования;
    ФИО = "" + фамилия + " " + Имя + " " + Отчество;
    Объект.Наименование= ФИО;
КонецПроцедуры
  
```

16. В свойствах полей «Фамилия», «Имя» и «Отчество» назначьте обработчиком события ПриИзменении данную процедуру.

17. Откройте командный интерфейс подчиненной подсистемы «Справочники» подсистемы «Общее». Сделайте видимыми команды создания элементов для справочников «ЕдиницыИзмерения», «Номенклатура» и «ФизическиеЛица».



18. Выполните компиляцию. Заполните справочники данными.

19. Продемонстрируйте результаты работы преподавателю. Выгрузите информационную базу.

Контрольные вопросы

1. Для чего предназначены справочники?
2. Что такое иерархия элементов справочника?
3. Для чего предназначены перечисления и в чем отличие их от справочников?
4. Для чего нужна иерархия групп?
5. В каких случаях надо создавать табличные формы?
6. Что из себя представляет печатная форма?
7. Для чего нужны директивы компиляции?