

Регистры сведений так же, как и константы, работу с которыми мы рассмотрели в предыдущем юните, относятся к *прикладным объектам* системы.

Константа способна хранить только одно значение. Что делать, если значение, которое необходимо сохранять в ИБ, должно изменяться с течением времени (быть *развернуто* по времени) или если оно (значение) зависит от ряда факторов?

В таких случаях можно говорить о том, что нам необходима некоторая *многомерная* структура хранения, способная сохранять различные сведения в зависимости от конкретных значений своих измерений.

Такой структурой хранения в «1С: Предприятии» являются регистры сведений, которые, выражаясь математическим языком, позволяют описать зависимость значения некоторой функции от своих аргументов в табличной форме.

В постановке задачи (п. 3) сказано, что «...необходимо пересчитывать рублевые суммы в условные единицы (УЕ)». Тогда для выполнения подобных операций необходимо помнить соответствующие значения курса. И зависимость этого курса от даты может быть выражена таблицей (рис. 39.1).

Объединение "Содружество"	
История курса УЕ	
Дата	Значение
28.12.2007	25.2600
01.01.2008	24.8500
01.01.2009	23.7844
31.01.2009	23.6222
02.03.2009	23.5729
01.04.2009	23.3801
01.05.2009	23.1021
31.05.2009	23.7090
30.06.2009	23.3483
30.07.2009	23.2450
18.08.2009	23.3532

Рис. 39.1. Значения курса УЕ

Из этой таблицы мы видим, что информация, представленная в колонке «Значение», *развернута* во времени.

Поэтому мы можем приступить к созданию соответствующего регистра сведений.

Регистры сведений располагаются в отдельной ветви дерева окна «Конфигурация». Для того чтобы создать новый регистр, необходимо, используя правую кнопку мыши, на ветви «Регистры сведений» дерева окна «Конфигурация» выбрать пункт выпадающего меню «Добавить».

39.1. ОКНО РЕДАКТИРОВАНИЯ РЕГИСТРА СВЕДЕНИЙ

Сразу после этого Конфигуратор создаст в дереве новый объект «РегистрСведений1» и откроет специальное *окно редактирования* для его настройки (рис. 39.2).

Сам регистр сведений — достаточно сложный объект, он содержит ряд подчиненных объектов. Кроме того, он должен быть связан с остальной конфигурацией в единое целое. Поэтому закладки слева позволяют не только настроить свойства и состав самого регистра, но и связать его с другими объектами в конфигурации.

Кнопка «Далее>» позволяет последовательно пройти по всем этапам настройки регистра.

В этом юните мы рассмотрим только этапы «Подсистемы», «Основные», «Данные» и «Формы», а изложение остальных этапов в настройке регистра сведений мы рассмотрим в последующих юнитах.

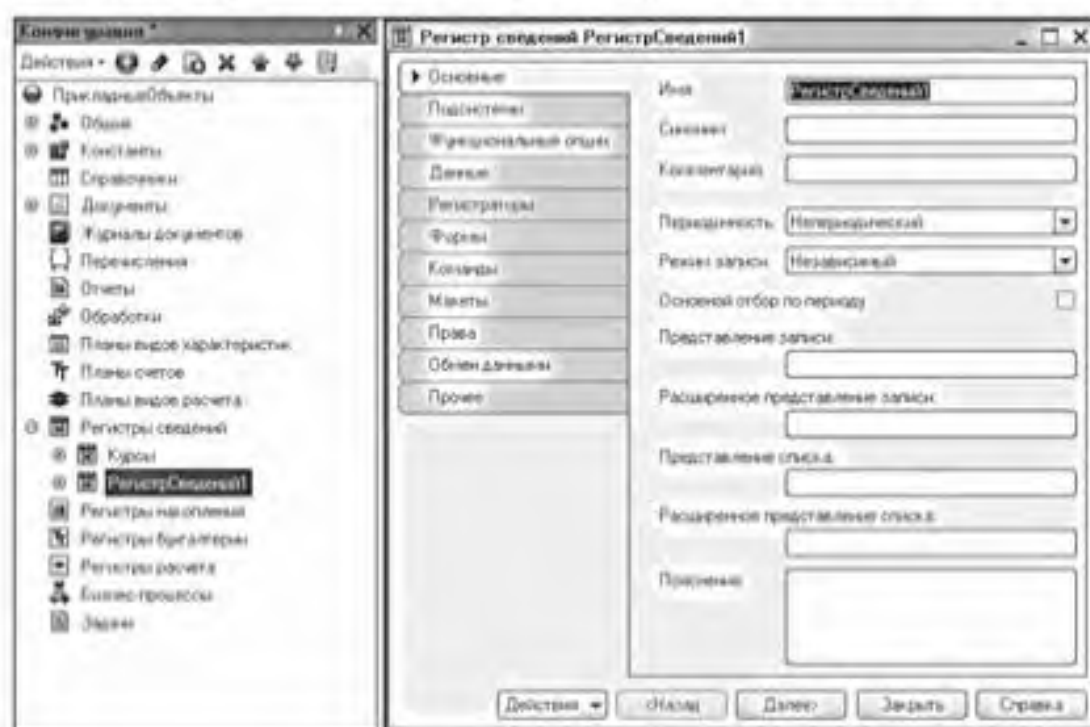


Рис. 39.2. Окно настройки Регистра сведений

39.1.1. Закладка «Основные»

На этой закладке собраны основные свойства регистра сведений, определение которых необходимо сделать до начала настройки остальных свойств.

Так же, как и у других объектов, у регистров сведений есть свойства: *Имя*, *Синоним* и *Комментарий*, которые мы уже рассматривали с вами ранее. Новыми для нас будут свойства *Периодичность* и *Режим записи*.

Периодичность. Это свойство позволяет определить, с какой степенью детализации необходимо помещать в регистр значения в зависимости от времени. Возможны следующие значения:

- ✓ **Непериодический;**
- ✓ **В пределах секунды;**
- ✓ **В пределах дня;**
- ✓ **В пределах месяца;**
- ✓ **В пределах квартала;**
- ✓ **В пределах года.**

Если регистр сделать непериодическим, то у него не будет разреза (изменения) «Период».

Режим записи. Это свойство определяет, каким образом будут вноситься изменения в регистр: вручную или документами.

Пока мы будем использовать только *независимый режим записи*, так как в нашей конфигурации еще нет ни одного документа.

Для самостоятельного освоения материала выполните следующее упражнение.

Упражнение 39.1

Создайте в конфигурации новый регистр сведений со следующими свойствами:

Свойство	Значение
Имя	Курсы
Синоним	Курсы
Комментарий	Содержит курсы УЕ
Периодичность	В пределах дня
Режим записи	Независимый

Перейдите к закладке «Данные».

39.1.2. Закладка «Данные»

На этой закладке (рис. 39.3) мы должны определить структуру хранения нашего регистра.



Рис. 39.3. Закладка «Данные»

Измерения представляют собой, по сути, аргументы функции, задаваемой в табличной форме.

Если регистр развернут по времени, то он уже имеет одно измерение — «Период». И такие регистры сведений называются *периодическими*.

Отметим, что в самом общем случае у непериодических регистров измерения могут отсутствовать.

Ресурсы будут содержать искомые значения в зависимости от конкретных значений измерений.

Так же, как и в случае с измерениями, отметим, что в регистре могут отсутствовать и ресурсы.

Согласно таблице, которая изображена на рис. 39.1, нам потребуются:

- ✓ одно измерение — «Период» (уже есть!);
- ✓ один ресурс, содержащий курс УЕ на конкретную дату.

Поэтому давайте создадим новый ресурс, выполнив следующее упражнение.

Упражнение 39.2

Выделите в окне «Регистр сведений: Курсы» закладку «Данные», затем выделите в окне ветвь «Ресурсы» и нажмите на кнопку Ins. После чего заполните свойства нового ресурса.


Основные	
Свойство	Значение
Имя	Курс
Синоним	Курс
Комментарий	Содержит курс валюты
Тип данных	
Свойство	Значение
Тип	Число
Длина	10
Точность	4
Неотрицательное	<input checked="" type="checkbox"/>

Нажмите на кнопку «Далее>», для того чтобы перейти к следующей закладке.

39.1.3. Закладка «Формы»

Наш регистр сведений будет содержать отдельные записи, которые будут упорядочены по измерению «Период». Причем каждая запись будет иметь две колонки — «Период» и «Курс».

Используем эту закладку для создания хотя бы одной формы для регистра сведений. Поскольку на данном этапе разработки конфигурации нам нужно обеспечить возможность ввода курсов УЕ на заданную дату, то мы создадим только *основную форму списка*.

Чтобы создать эту форму, необходимо щелкнуть мышью на кнопке  поля «Основная форма списка» (рис. 39.4).

После таких действий Конфигуратор вызовет уже известный нам конструктор форм, с первым шагом которого мы уже знакомы.

Самостоятельно заполните поля Конструктора формы регистра сведений следующими значениями.

Упражнение 39.3

В Конструкторе укажите следующие значения в полях его диалога:

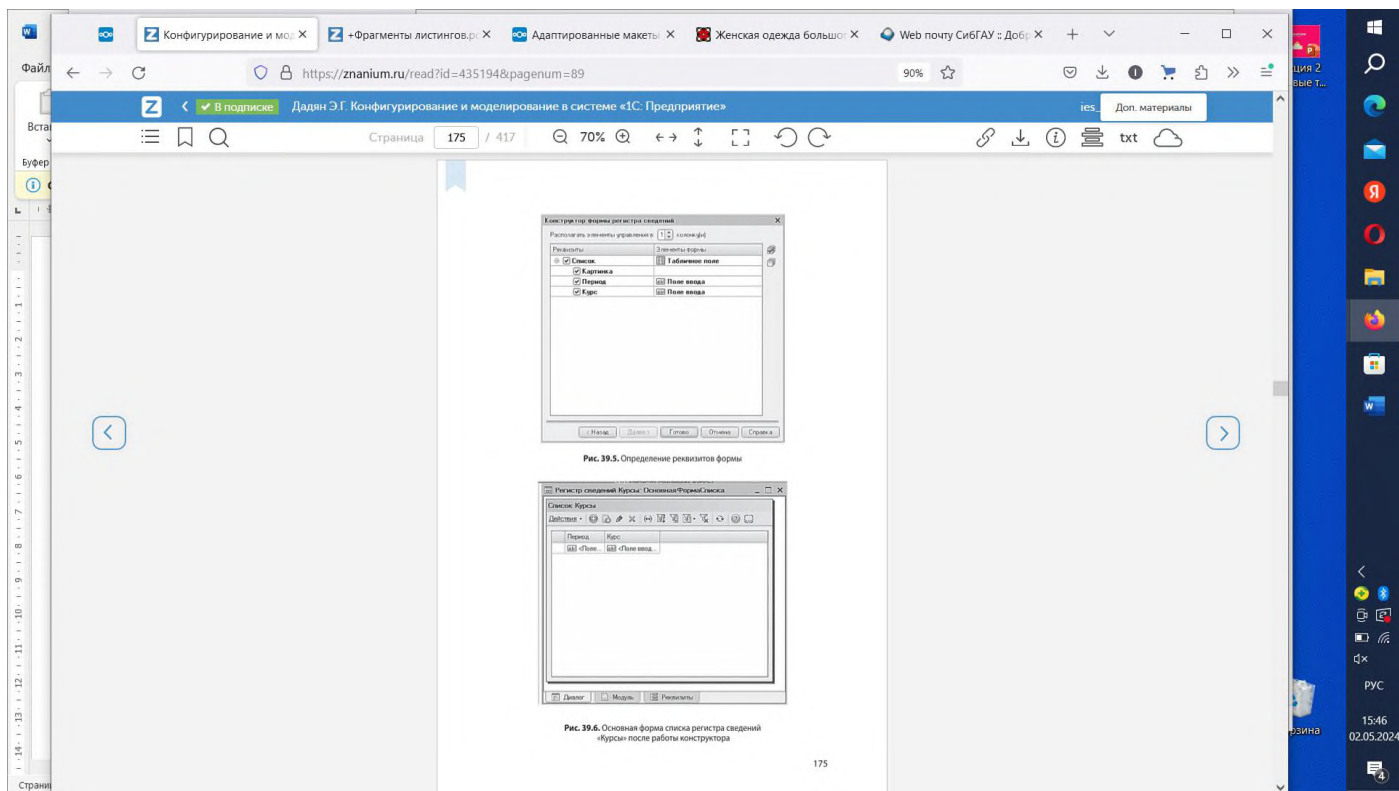




Рис. 39.4. Создание новой формы в закладке «Формы»

Свойство	Значение
Выберете тип формы	Форма списка регистра сведений
Тип формы	Обычная
Назначить форму основной	<input checked="" type="checkbox"/>
Имя	ОсновнаяФормаСписка
Синоним	Основная форма списка
Комментарий	Основная форма списка регистра «Курсы»

Нажмите на кнопку «Далее>», для того чтобы рассмотреть следующий шаг конструктора более подробно.

После указанных выше действий форма Конструктора формы списка примет вид, изображенный на рис. 39.5.

Как мы видим теперь, как и в Основной форме констант, эта форма содержит табличное поле, в котором мы можем по своему желанию (в некоторых пределах) настроить состав колонок, которые она будет содержать.

Конфигурирование и мо... «Фрагменты листингов.р... Адаптированные макеты Женская одежда больш... Web почту СибГАУ - Доб... 90%

https://znanium.ru/read?id=4351948&pagenum=89 ies_vel@mail.ru

Страница 176 / 417 70%

Упражнение 39.4

Нажмите на кнопку «Готово», чтобы оценить результат работы конструктора (рис. 39.6). В верхней части управляемой формы расположена командная панель, которая уже содержит необходимый набор кнопок управления для работы со списком записей регистра сведений, которые будут отображаться в табличном поле (оно расположено ниже). Сохраните конфигурацию.

Закройте окно редактирования регистра сведений и обновите конфигурацию (обратите внимание на появление нового объекта в ИБ).

39.2. ВВОД ДАННЫХ В РЕГИСТР СВЕДЕНИЙ

Как уже отмечалось, в Конфигураторе мы только создали структуру хранения, а ее заполнением конкретными значениями будет заниматься пользователь в режиме «ИС:Предприятие».

Упражнение 39.5

Запустите режим «ИС:Предприятие».

Выберите пункт меню «Операции — регистр сведений», а в открывшемся диалоговом окне выбора объекта нажмите кнопку ОК.

Введите в регистр новые значения, используя следующую таблицу.

Дата	Значение
28.12.2007	25,2600
01.01.2008	24,8500
01.01.2009	23,7844
31.01.2009	23,6222
02.03.2009	23,5729
01.04.2009	23,3801
01.05.2009	23,1021
31.05.2009	23,7090
30.06.2009	23,3483
30.07.2009	23,2450
18.08.2009	23,3532

Совет: для ввода новых записей удобно использовать кнопку Ins на клавиатуре. Кроме того, все возможные действия, которые можно выполнять в табличном поле, можно узнать, используя правую кнопку мыши.

176

Страница 176 / 417

15:47 02.05.2024

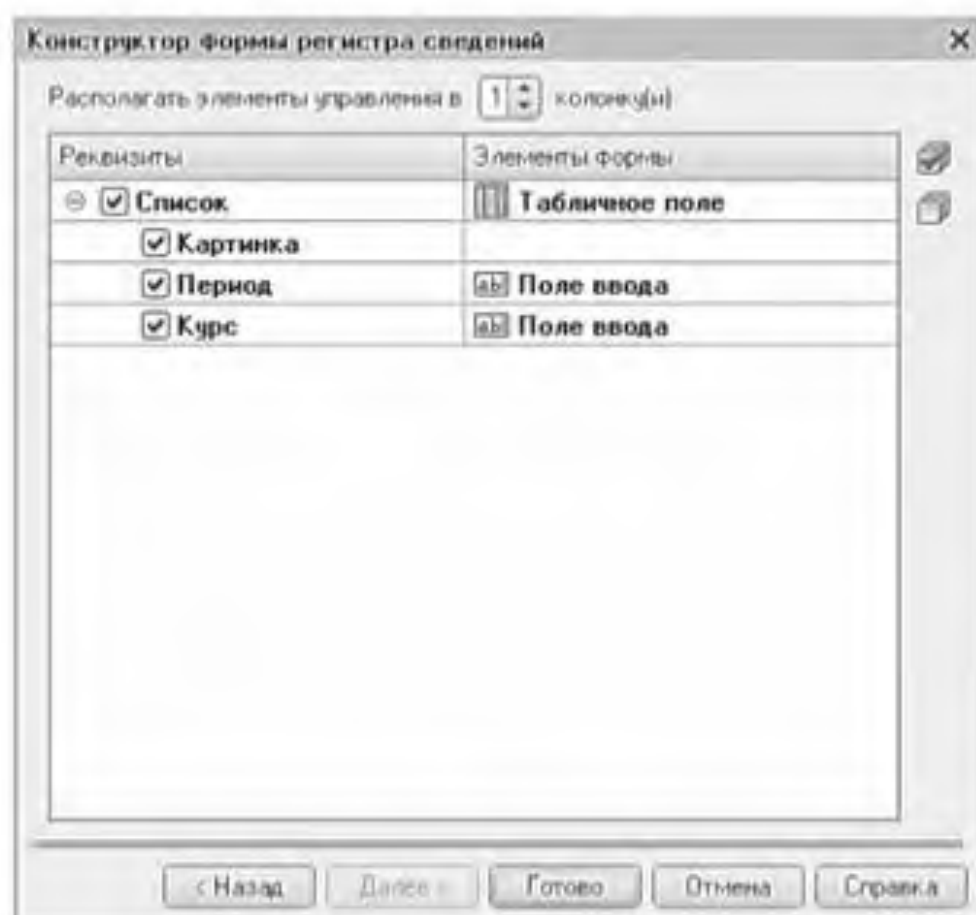


Рис. 39.5. Определение реквизитов формы

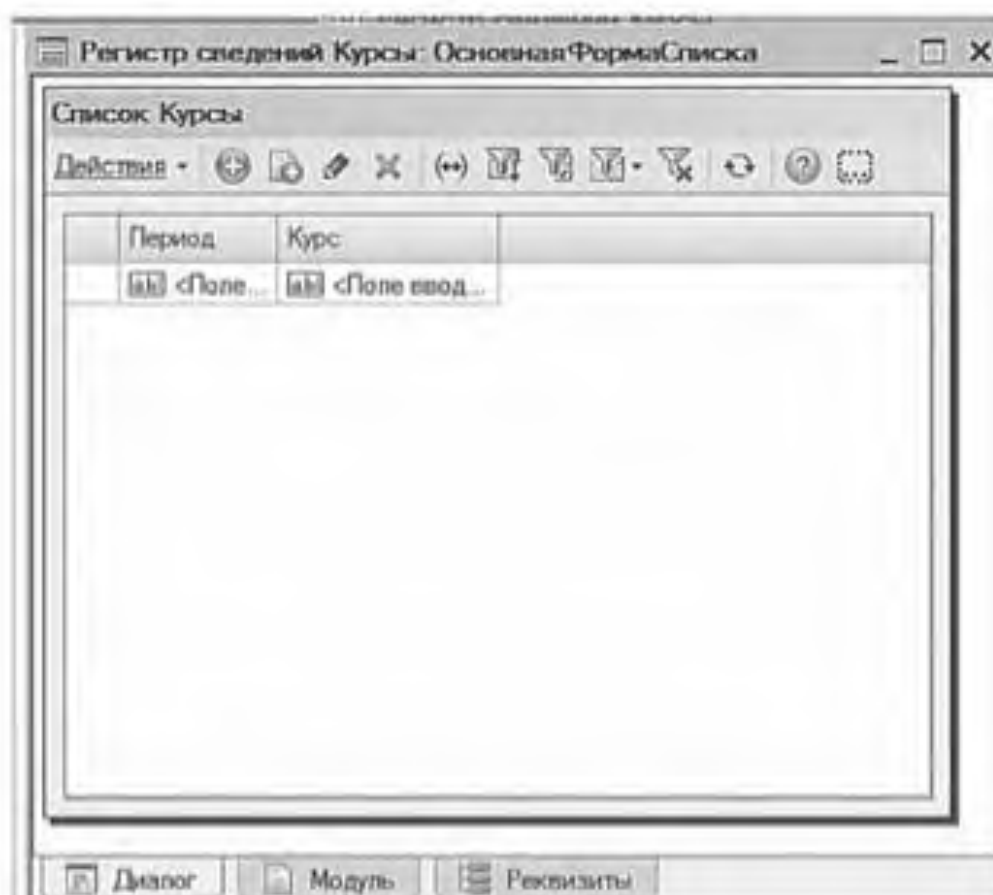


Рис. 39.6. Основная форма списка регистра сведений

Упражнение 39.4

Нажмите на кнопку «Готово», чтобы оценить результат работы конструктора (рис. 39.6). В верхней части управляемой формы расположена командная панель, которая уже содержит необходимый набор кнопок управления для работы со списком записей регистра сведений, которые будут отображаться в табличном поле (оно расположено ниже).

Сохраните конфигурацию.

Закройте окно редактирования регистра сведений и обновите конфигурацию (обратите внимание на появление нового объекта в ИБ).

39.2. ВВОД ДАННЫХ В РЕГИСТР СВЕДЕНИЙ

Как уже отмечалось, в Конфигураторе мы только создали структуру хранения, а ее заполнением конкретными значениями будет заниматься пользователь в режиме «1С:Предприятие».

Упражнение 39.5

Запустите режим «1С: Предприятие».

Выберите пункт меню «Операции — регистр сведений», а в открывшемся диалоговом окне выбора объекта нажмите клавишу ОК.

Введите в регистр новые значения, используя следующую таблицу.

Дата	Значение
28.12.2007	25,2600
01.01.2008	24,8500
01.01.2009	23,7844
31.01.2009	23,6222
02.03.2009	23,5729
01.04.2009	23,3801
01.05.2009	23,1021
31.05.2009	23,7090
30.06.2009	23,3483
30.07.2009	23,2450
18.08.2009	23,3532

Совет: для ввода новых записей удобно использовать кнопку Ins на клавиатуре. Кроме того, все возможные действия, которые можно выполнять в табличном поле, можно узнать, используя правую кнопку мыши.

Конфигурирование и мо... X Фрагменты листингов.р... X Адаптированные макеты X Женская одежда большо... X Web почту СибГАУ - Доб... X

https://znanium.ru/read?id=435194&pagenum=89 90%

Дадин Э.Г. Конфигурирование и моделирование в системе «1С: Предприятие» ies_vel@mail.ru

Страница 178 / 417 70%

А что делать, если записи на конкретный период нет? Тут возможны две стратегии:

- ✓ либо брать первую запись с более поздним периодом;
- ✓ либо брать последнюю запись с более ранним периодом.

Если записи, удовлетворяющие необходимому условию, нет, то, как и в предыдущем случае, возвращается структура, содержащая все пустые значения.

За первый способ получения записи отвечает метод **ПолучитьПервое(...)**, а за второй — **ПолучитьПоследнее()**.

Приведем пример.

Процедура ПримерыОбращенияКРегиструСведений()
//
// Найдем курс на сегодняшнее число:
Запись:=РегистрыСведений.Курсы.ПолучитьПоследнее(ТекущаяДата());
Сообщить(«Сегодня курс УЕ=>»+Запись.Курс);
КонецПроцедуры

Для закрепления материала выполните следующее упражнение.

Упражнение 39.6

Включите в процедуру — обработчик **ПриНачалеРаботыСистемы()** вызов процедуры **ПримерыОбращенияКРегиструСведений()**.

В этой процедуре напишите фрагмент программного кода, который рассчитывал бы в окне «Службные сообщения» первое значение валютного курса в 2009 г.

Каким методом Вы воспользуетесь?

Обязательно проверьте работу фрагмента путем запуска режима «1С:Предприятие».

Для надежности приводим текст данного фрагмента программного кода (сравните со своим вариантом).

ЛИСТИНГ Ю. 39.1

Процедура ПримерыОбращенияКРегиструСведений()
// Найдем первое значение валютного курса в 2009 г.
Запись:=РегистрыСведений.Курсы.ПолучитьПервое(20090101);
Сообщить(«Сегодня курс УЕ=>»+Запись.Курс);
КонецПроцедуры

39.3.2. Менеджер записи

Для того чтобы поместить в регистр сведений новую запись или заменить существующую, необходимо сначала создать объект типа «РегистрСведенийМенеджерЗаписи».

178

39.3. РАБОТА С РЕГИСТРОМ СВЕДЕНИЙ ИЗ ВСТРОЕННОГО ЯЗЫКА СИСТЕМЫ

Методы, предназначенные для получения значений из регистра сведений, часто содержат так называемые фактические параметры. Многие из них возвращают в качестве результата тип данных «Структура».

Этот тип данных представляет собой запись в оперативной памяти (ОП) с произвольным числом полей (свойств) следующего вида.

Ключ:	Свойство1	Свойство2	...	СвойствоN
Значение:	Значение1	Значение2	...	ЗначениеN

При этом количество свойств и их имена (*ключи*), а также сами значения определяются соответствующими методами объекта типа данных «Структура».

Обращение к свойствам структуры осуществляется обычным для объектной технологии способом «через точку»: **ИмяСтруктуры.ИмяСвойства**.

39.3.1. Получение отдельных значений

Как уже было отмечено выше, в периодическом регистре значения зависят от того периода времени, на который они были введены. У объекта данного типа имеются три схожих по применению метода, позволяющих получить значение типа «Структура», состав свойств которого полностью соответствует составу *ресурсов* регистра сведений, спроектированных на этапе конфигурирования.

Все эти методы начинаются со слова «Получить».

Метод **Получить(...)** пытается найти запись с указанным в качестве первого параметра моментом времени. Если запись есть, то возвращается структура, содержащая значения всех ресурсов. Если нет, то структура будет содержать пустые значения.

Приведем пример получения значения курса УЕ на конкретную дату:

```
//ПримерыОбращенияКРегиструСведений() — содержит примеры
// обращения к регистру сведений
Процедура ПримерыОбращенияКРегиструСведений()
// Найдём значение курса на конкретную дату:
Запись =РегистрыСведений.Курсы.Получить('20090101');
Сообщить(«Курс УЕ на 01.01.2009г.»+Запись.Курс);
КонецПроцедуры
```

А что делать, если записи на конкретный период нет? Тут возможны две стратегии:

- ✓ **либо брать первую запись с более поздним периодом;**
- ✓ **либо брать последнюю запись с более ранним периодом.**

Если записи, удовлетворяющие необходимому условию, нет, то, как и в предыдущем случае, возвращается структура, содержащая все пустые значения.

За первый способ получения записи отвечает метод **ПолучитьПервое(...)**, а за второй — **ПолучитьПоследнее()**.

Приведем пример:

Процедура ПримерыОбращенияКРегиструСведений()

//

// Найдем курс на сегодняшнее число:

Запись=РегистрыСведений.Курсы.ПолучитьПоследнее(ТекущаяДата());

Сообщить(«Сегодня курс УЕ=»+Запись.Курс);

КонецПроцедуры

Для закрепления материала выполните следующее упражнение.

Упражнение 39.6

Включите в процедуру — обработчик **ПриНачалеРаботыСистемы()** вызов процедуры: **ПримерыОбращенияКРегиструСведений()**.

В этой процедуре напишите фрагмент программного кода, который распечатывал бы в окне «Служебные сообщения» первое значение валютного курса в 2009 г.

Каким методом Вы воспользуетесь?

Обязательно проверьте работу фрагмента путем запуска режима «1С:Предприятие».

Для надежности приводим текст данного фрагмента программного кода (сравните со своим вариантом).



ЛИСТИНГ Ю_39.1

Процедура ПримерыОбращенияКРегиструСведений()

// Найдем первое значение валютного курса в 2009 г.

Запись=РегистрыСведений.Курсы.ПолучитьПервое('20090101');

Сообщить(«Сегодня курс УЕ=»+Запись.Курс);

КонецПроцедуры

39.3.2. Менеджер записи

Для того чтобы поместить в регистр сведений новую запись или заместить существующую, необходимо сначала создать объект типа «РегистрСведенийМенеджерЗаписи».

Делается это путем вызова соответствующего метода **СоздатьМенеджерЗаписи()**: для объекта типа **РегистрСведенийМенеджер**:

Объект=РегистрыСведений.ИмяРегистра.СоздатьМенеджерЗаписи();

После того как объект такого типа создан, Вы сможете применять его методы:

- ✓ **Прочитать();**
- ✓ **Записать();**
- ✓ **Удалить(),**

которые выполняют одноименные действия.

Перед вызовом этих методов необходимо предварительно заполнить свойства этого объекта, которыми будут измерения, ресурсы и реквизиты конкретного регистра сведений, менеджер записи которого Вы создали. Для периодических реквизитов у этого менеджера есть свойство «Период», которое позволяет определить конкретную запись, содержащую это значение.

Предположим, что мы хотим при **открытии формы** записать в регистр сведений «Курсы» значение 23.7244 на 9 января 2009 г., причем если запись на такое число уже есть, то мы должны оставить ее без изменений.

С точки зрения здоровой логики алгоритм решения этой задачи прост.

Попытаться прочитать запись из регистра на 09.01.2009 г.

Убедиться, что такой записи нет.

Записать новую запись на указанную дату.

В решении второго пункта нам поможет метод **.Выбран()**, который поможет определить, считана ли запись.

ЛИСТИНГ Ю_39.2

Процедура ПримерыОбращенияКРегиструСведений()

// Сначала создадим менеджер записи:

РегСведКурсы=РегистрыСведений.Курсы.СоздатьМенеджерЗаписи();

// Затем определим период в который будем писать данные:

РегСведКурсы.Период='20090109'; //9 января 2009 г.

// 1. Пытаемся прочитать запись

РегСведКурсы.Прочитать();

// 2. Если записи нет

Если Не РегСведКурсы.Выбран() Тогда

// 3. Запишем новую

РегСведКурсы.Период='20090109'; //9 января 2009 г.

РегСведКурсы.Курс=23.7244;

РегСведКурсы.Записать();

КонецЕсли;

Сообщить(«Сегодня курс УЕ=»+ РегСведКурсы.Курс);

КонецПроцедуры;
Процедура ПриНачалеРаботыСистемы()
 ИмяПроцедуры();
КонецПроцедуры

39.3.3. Использование выборки

В предыдущих разделах мы обращались к конкретным записям (по одной), а что делать, если нужно обработать все или какую-то часть записей, удовлетворяющих каким-либо условиям?

Данная постановка реализуется с помощью открытия так называемой *выборки*.

Для регистра сведений это делается следующим образом:

ВыборкаСведений=РегистрыСведений.<ИмяРегистраСведений>.Выбрать(...);
Пока ВыборкаСведений.Следующий() Цикл
КонецЦикла;

По методу **.Выбрать()** мы получаем объект типа **РегистрСведенийВыборка**, для которого в цикле применяем его метод **.Следующий()**, чтобы извлечь из выборки очередную запись.

Внутри цикла Вы можете обращаться к значениям измерений и ресурсов просто как к свойству этого объекта.

Например: **ВыборкаСведений.Период**.


39.4. САМОСТОЯТЕЛЬНАЯ РАБОТА

Для закрепления только что рассмотренного материала выполните следующие упражнения самостоятельно.

Упражнение 39.7

Добавьте в процедуру Модуля Приложения **ПримерыОбращенияКРегиструСведений()** фрагмент, позволяющий записать в регистр сведений «Курсы» значение 23.7244 на 9 января 2009 г. Причем если запись на такое число уже есть, ее необходимо оставить без изменений.

Используя дерево окна «Конфигурация», откройте форму «**Основная-ФормаСписка**» в редакторе форм.

Выделите мышью кнопку  панели инструментов (рис. 39.7) и заполните ее свойства следующими значениями.

Свойство	Значение
Имя	ПечатьКурсаУЕ
Тип	Действие
Действие	ПечатьКурсаУЕ
Текст	Печать курса УЕ
Подсказка	Печать курса УЕ
Пояснение	Печать курса УЕ
Картинка	 Картинка: bmp
Отображение	Надпись и картинка

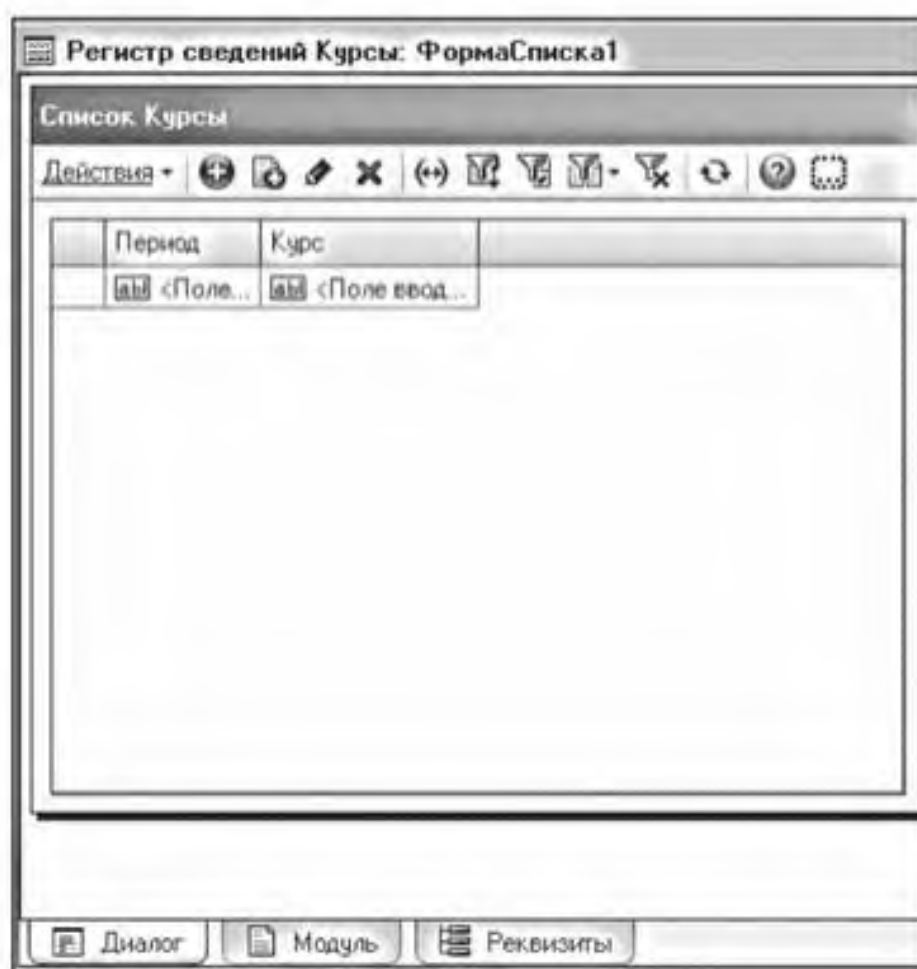


Рис. 39.7. Форма списка регистра сведений перед началом редактирования


Для надежности приводим текст данного фрагмента программы (сравните со своим вариантом).



ЛИСТИНГ Ю_39.3

Процедура ПримерыОбращенияКРегиструСведений()

```
// записать в регистр сведений «Курсы» значение 23.7244 на 9 //января 2009г.
// Сначала создадим менеджер записи:
РегСведКурсы=РегистрыСведений.Курсы.СоздатьМенеджерЗаписи();
// Затем определим период в который будем писать данные:
РегСведКурсы.Период='20090109'; // 9 января 2009 г.
// 1. Пытаемся прочитать запись
РегСведКурсы.Прочитать();
// 2. Если записи нет
Если Не РегСведКурсы.Выбран() Тогда
// 3. Запишем новую
РегСведКурсы.Период='20090109'; // 9 января 2009 г.
РегСведКурсы.Курс=23.7244;
РегСведКурсы.Записать();
КонецЕсли;
КонецПроцедуры
```

Совет: для выбора картинки необходимо в диалоге «Выбор картинки» сначала выбрать закладку «Из библиотеки», а затем на закладке «Стандартные» картинку «PRINT» .

Упражнение 39.8

Переключитесь на закладку «Модуль» и напишите процедуру **ПечатьКурсаУЕ()**, которая распечатывала бы в окне «Служебные сообщения» курс УЕ на каждую дату, что введена в регистре сведений «Курсы».

Проверьте работу кнопки  в режиме «IS:Предприятие».

По окончании отладки замените номер версии в свойствах конфигурации на значение «03».

Каким типом объекта Вы воспользуетесь для решения поставленной задачи?

Для надежности приводим текст данной процедуры (сравните со своим вариантом).

ЛИСТИНГ Ю_39.4

```
Процедура ПечатьКурсаУЕ(Кнопка)
ВыборкаСведений=РегистрыСведений.Курсы.Выбрать();
Пока ВыборкаСведений.Следующий() Цикл
Сообщить(«»+ВыборкаСведений.Период+Символы.Таб+ВыборкаСведений.Курс);
КонецЦикла;
КонецПроцедуры
```

39.5. ИСПОЛЬЗОВАНИЕ НАБОРА ЗАПИСЕЙ

Во всех предыдущих случаях (кроме выборки) мы считывали значения по одному, что нельзя признать эффективным способом обработки при большом или очень большом числе записей.

Использование выборки, которую Вы должны были освоить самостоятельно, позволяет несколько оптимизировать процесс чтения записей за счет того, что записи считываются отдельными *порциями*.

Но выборка, к сожалению, осуществляется только в прямом направлении. А в постановке задачи пример отчета содержит обратную сортировку.

Решить эту проблему можно путем использования *набора записей* регистра сведений. Методика применения здесь следующая:

```
НабКурсы=РегСведКурсы.СоздатьНаборЗаписей();
НабКурсы.Прочитать();
Для Каждого ТекСтрока из НабКурсы Цикл
Сообщить(«»+ТекСтрока.Период+Символы.Таб+ТекСтрока.Курс);
КонецЦикла;
```

Метод **.СоздатьНаборЗаписей()** создает объект типа **РегистрСведений-НаборЗаписей**, который будет содержать коллекцию (массив) отдельных записей.

Объект этого типа имеет достаточное число методов:

```
Вставить(<Индекс>);
Добавить();
Записать(<Замещать>);
Индекс(<Запись>);
Количество();
Очистить();
Прочитать();
Сдвинуть(<Запись>, <Смещение>);
Удалить(<Запись>),
```

которые позволяют работать как со всем набором записей в целом, так и с каждой из них в отдельности.

В частности, метод **.Прочитать()** позволяет считать всю таблицу из ИБ в оперативную память. Для последовательного перебора отдельных записей можно воспользоваться циклом «Для Каждого» (см. пример выше), который позволит последовательно обойти все записи из набора.

Однако это перебор записей в прямом направлении сортировки. Для того чтобы распечатать курсы в обратном порядке, нужно вспомнить, что набор записей — это коллекция. А следовательно, для доступа к отдельным записям (в «прямую») можно воспользоваться

операцией индексирования ([...]). Индексы в коллекциях отсчитываются от нуля. Поэтому, чтобы получить самую первую и самую последнюю запись, следует записать следующие выражения:

ПерваяЗапись = НабКурсы[0];

ПоследняяЗапись = НабКурсы[НабКурсы.Количество()-1];

А цикл распечатывания курсов в обратном направлении будет выглядеть следующим образом:

РегСведКурсы=РегистрыСведений.Курсы;

НабКурсы=РегСведКурсы.СоздатьНаборЗаписей();

НабКурсы.Прочитать();

к=НабКурсы.Количество();

Для н=1 по НабКурсы.Количество() Цикл

ТекСтрока=НабКурсы[к-н];

Сообщить(«»+ТекСтрока.Период+Символы.Таб+ТекСтрока.Курс);

КонецЦикла;

Упражнение 39.9

Измените текст процедуры **ПечатьКурсаУЕ()** в модуле формы списка регистра сведений, включив в него текст приведенный выше.

Проверьте работу процедуры после внесенных изменений.

Для надежности приводим текст данной процедуры (сравните со своим вариантом).



ЛИСТИНГ Ю_39.5

Процедура ПечатьКурсаУЕ(Кнопка)

РегСведКурсы=РегистрыСведений.Курсы;

НабКурсы=РегСведКурсы.СоздатьНаборЗаписей();

НабКурсы.Прочитать();

к=НабКурсы.Количество();

Для н=1 по НабКурсы.Количество() Цикл

ТекСтрока=НабКурсы[к-н];

Сообщить(«»+ТекСтрока.Период+Символы.Таб+ТекСтрока.Курс);

КонецЦикла;

КонецПроцедуры

39.5.1. Обработка событий

Объект типа **РегистрСведенийНаборЗаписей** предоставляет возможность обрабатывать два события:

✓ **ПередЗаписью;**

✓ **ПриЗаписи.**

Оба события позволяют контролировать процесс записи в ИБ.

Для того чтобы обработать нужное событие, необходимо написать процедуру с идентификатором соответствующего события в модуле набора записей регистра сведений.

В п. 1 постановки задачи сказано: «Ввод всех данных и документов должен выполняться не ранее *даты регистрации объединения*», для того чтобы решить поставленную задачу, выполните следующее упражнение.

Упражнение 39.10

В окне редактирования регистра сведений «Курсы» переключитесь на закладку «Прочее» (рис. 39.8).



Рис. 39.8. Закладка «Прочее» окна редактирования

Нажмите на кнопку «Модуль набора записей» и наберите следующий текст.



ЛИСТИНГ Ю_39.6

```
// Модуль набора записей регистра сведений «Курсы»  
// ПередЗаписью() — проверяет дату периода перед записью в ИБ  
// Параметры
```



```
// Отказ (Булево) позволяет отказаться от записи (Ложь-по умолчанию-  
// выполнение операции, Истина — отказ от операции.  
// Замещение (Булево) режим записи набора (Ложь-по умолчанию-дописывание  
// записи, Истина — замещение записи.
```

```
Процедура ПередЗаписью(Отказ, Замещение)  
ДатаРегистрации = Константы.ДатаРегистрации.Получить();  
Для каждого ТекЗапись Из ЭтотОбъект Цикл  
// Если период раньше даты регистраци  
Если ТекЗапись.Период < ДатаРегистрации Тогда  
// откажемся от записи  
Отказ = Истина;  
// сообщим об ошибке  
Сообщить(«Запись от «+ТекЗапись.Период+  
» с курсом «+ТекЗапись.Курс+  
» раньше даты регистрации объединения!»,  
СтатусСообщения.Важное);  
КонецЕсли;  
КонецЦикла;  
Если Отказ Тогда  
Сообщить(«Запись не выполнена!!!», СтатусСообщения.ОченьВажное);  
КонецЕсли;  
КонецПроцедуры
```

Проверьте работу процедуры путем ввода новой или редактирования существующей записи в регистре сведений «Курсы».

Дадим необходимые пояснения к нашим действиям.

Причина, по которой мы использовали именно такую процедуру, как обработчик событий, состоит в том, что запись новых данных и исправление существующих могут выполняться как вследствие интерактивных действий пользователя, так и программным путем.

Если бы проверка значения периода выполнялась только в модуле формы, то тогда мы смогли бы контролировать только интерактивные действия пользователя, связанные с непосредственным редактированием. При этом сохранялась бы возможность записи в регистр программным путем значений, которые не удовлетворяли бы условиям поставленной задачи.

Можно сказать, что модуль набора записей — это «последняя линия обороны» и его задача контролировать те данные, которые помещаются в регистр.

Формальный параметр **Отказ** анализируется системой после обработки события, поэтому

Отказ = Истина;

приведет к отмене транзакции.

При этом нужно помнить, что если Вы отменяете транзакцию записи в процедуре **ПередЗаписью(...)** или в **ПриЗаписи(...)**, то метод **Записать()** следует вызвать внутри конструкции:

```
Попытка
...
Исключение
...
КонецПопытки;
```

В противном случае система будет генерировать ошибку времени выполнения.

39.6. ЧТО МЫ УЗНАЛИ

В этой главе мы познакомились с прикладным объектом «Регистры сведений», который предназначен для сохранения в ИБ значений, зависящих от значений различных параметров. Если для регистра свойство периодичность уставлено в какое-либо из значений «В пределах ...», то значения такого регистра будут развернуты во времени.

С помощью окна редактирования регистра мы последовательно определили: его свойства, структуру хранения (какие данные будет содержать регистр сведений) и сформировали, используя конструктор, его основную форму списка.

Эту основную форму списка мы использовали для ввода новых и редактирования существующих записей регистра сведений «Курсы», заполнив его в режиме «ИС:Предприятие» конкретными значениями.

Мы также рассмотрели вопросы получения и записи отдельных записей для периодического регистра сведений, познакомились с методами организации выборки записей в цикле.

Работа с набором записей позволит достичь еще большей эффективности за счет того, что все (или часть) записей из таблицы, расположенной в ИБ, считываются в оперативную память по одной операции чтения. Обработка записей, образующих коллекцию, возможна как путем последовательного циклического перебора с использованием конструкции цикла для каждого, так и путем произвольного обращения по индексу в массиве.

В этой главе мы рассмотрели новую возможность — обработку событий записи в регистр, которая может выполняться в модуле набора записей регистра сведений.