

Низамутдинов Ильяс

ПРОГРАММИРОВАТЬ В 1С – ЗА 11 ШАГОВ

СБОРНИК ЗАДАЧ

Простой пошаговый курс обучения программированию
в 1С для всех!

Ижевск 2018

© Январь 2018 г. Все права защищены.

Глава 1. Вводная информация

Часть 1. Установка платформы и создание пустой базы

- 1) Установите учебную платформу 1С и создайте пустую базу для Вашей учебы.

Часть 7. Создание внешних обработок.

- 1) Создайте несколько внешних обработок, форм у этих обработок, разместите на формах команды. В командах выводите предупреждения или сообщения (процедура Сообщить).

Глава 2. Прimitivesкие типы и переменные

Часть 3. Переменные типа Булево

- 1) Поставьте одно *НЕ*, чтобы в результате получилась *Истина* (можно использовать скобки).

A = Ложь;

B = Ложь;

C = Ложь;

E = Ложь;

D = A и B и C и E;

- 2) Используйте два *НЕ* и две пары скобок, чтобы в результате получилась *Истина*.
A, B, C, E – те же что и в первом задании

D = A и B или C и E;

- 3) Поставьте одно *И*, одно *НЕ* и одну пару скобок (скобки должны закрывать только две переменные) чтобы в результате получилась *Ложь*

A = Истина;

B = Истина;

C = Истина;

D = A Или B Или C;

- 4) Поставьте одну пару скобок (скобки должны закрывать только две переменные) и одно *НЕ*, чтобы в результате получилась *Ложь*;

A = Истина;

Б = Ложь;

С = Ложь;

Е = Истина;

Д = А или Б и С или Е;

- 5) Даны цифры 5, 7, 4. Используя оператор > и оператор *ИЛИ* создайте выражение, результатом которого будет *Истина*. Цифры в выражении должны идти в том порядке, который дан в задаче.
- 6) Даны цифры 1, 0, 10, 27. Используя оператор =, оператор <, оператор *ИЛИ*, создайте выражение, результатом которого будет *Истина*. Цифры в выражении должны идти в том порядке, который дан в задаче.
- 7) Даны цифры 1, 7, 5. Используя оператор =, получите значение *Истина* любым способом. Напишите 5 вариантов решения задачи. Цифры в выражении должны идти в том порядке, который дан в задаче.
- 8) Даны цифры 6, 6, 7, 7. Используя оператор <>, получите значение *Истина* любым способом. Напишите 5 вариантов решения задачи. Цифры в выражении должны идти в том порядке, который дан в задаче.

Часть 4. Переменные типа Число

Вычислить:

1. $(\frac{A}{B} + C^3) * D$, где A, B, C, D – даны и задаются в модуле.

2. $(\sqrt{A} + B^C) * (\frac{D - E}{F * G})$, где A, B, C, D, E, F, G – даны и задаются в модуле.

3. $\frac{\sqrt{A^3 + B^4 + C^5}}{B^3}$, где A, B, C – задаются пользователем

4. $\frac{\sqrt{A-1} - \sqrt[3]{B-5}}{1 + \frac{A^2}{2} + \frac{B^5}{4}}$, где A, B – задаются пользователем

5. Округлите результат в выражениях 1-4 до 3 знака после запятой, до знака после запятой, до целого числа.

6. Получите остаток от деления $\frac{A + B - D}{A * D}, \frac{(A + D) * A}{A - D}$, где A, B, D задаются модуле.

7. Получите целое число от деления: $\frac{A + D * 10}{A - D}$, $\frac{A * D}{5 - D}$, $\frac{A + D}{A - D}$, где A,D задаются пользователем.

Часть 5. Переменные типа Строка

- 1) Запросите у пользователя поочередно город, улицу, дом, квартиру и выведите его адрес: г. Москва, ул. Гагарина, д. 98, кв. 34.
- 2) Введите интерактивно любой многострочный текст, разбитый на предложения (предложения разделяются точками). Разнесите все предложения на каждую отдельную строку.
- 3) Имея автомобильный номер формата хн344яRU63, извлеките: первые две буквы, цифры, последнюю букву и код региона. Выведите все это отдельно в виде «Номер хн344яRU63, состоит:

Первый буквы – хн

Цифры - 344

Последняя буква – я

Код региона - 63»

Ввод номера должен быть интерактивным.

- 4) Пусть имеется адресная страница вида <http://lenta.ru/news/2013/10/02/betterkittens/>

Вам необходимо извлечь из нее домен (lenta.ru).

Адресную строку вставляйте любую интерактивно. Домен должен быть чистый без «www», а адресная строка может быть как и с «www», так и без них

- 5) Пусть пользователь вводит интерактивно с клавиатуры какую-нибудь дату строкой вида «21.04.1988 г.», извлеките из строки отдельно год, отдельно день, отдельно месяц (цифрами). Выведите в виде

«День – 21

Месяц – 04

Год - 1988»

- 6) Пусть название товара задается одной строкой в таком виде «12321 Карандаш серый, шт»

Где 12311 – артикул

Карандаш серый – название

шт – единица измерения.

Необходимо всю эту информацию извлечь из данной строки, артикул всегда 5 символов, единица измерения может быть как шт, так и м (один символ)., но всегда в конце строки и после запятой с пробелом. Запятые в названии тоже могут присутствовать.

- 7) Имеем фамилия, имя, отчество человека (Иванов Петр Васильевич). Необходимо в каждую отдельную переменную написать фамилию, имя, отчество. Вывести на экран. ФИО нужно вводить интерактивно. Пользователь ФИО может ввести как угодно (и ИвАноВ, и иванов), необходимо все привести в вид Иванов.
- 8) Пусть имеется полное название файла вида C:\Program Files\1Cv82\8.2.19.68\Readme.html, извлеките из него название Readme.html. Полное название файла вводите интерактивно, расширение файла может быть любое.
- 9) Есть название организации в виде: ЗАО "Торговый дом Комплексный", ИНН 7705260699, извлеките из него отдельно ИНН, отдельно название.
- 10) Есть название номенклатуры в виде:
В-987 Вентилятор JIPONIC (Тайв.), настольный

ВН-900 Вентилятор настольный

Где В-987 и ВН-900 – артикулы (разной длины), извлеките из данных названий артикулы.

Часть 6. Переменные типа Дата

- 1) Пользователь вводит интерактивно дату, определите начало и конец месяца, начало и конец квартала, и начало и конец года, выведите эту информацию в виде

Данная дата (<Дата>) находится в интервале от <ДатаНачалоГода> по <ДатаКонецГода>.

Данная дата (<Дата>) находится в интервале от <ДатаНачалоКвартала> по <ДатаКонецКвартал>.

Данная дата (<Дата>) находится в интервале от <ДатаНачалоМесяца> по <ДатаКонецМесяца>.

Формат даты вида: 2012-04-20.

- 2) Введите интерактивно любую дату и выведите в окно сообщений в следующем виде:
2012-02-сентября.

02(09)-2012г.

02/09/2012г.

02/сентября/2012г.

02-сентября-2012г.

02(сентября)2012г.

- 3) Подсчитайте, сколько дней осталось от текущей даты до нового года.
- 4) Подсчитайте, сколько месяцев осталось от текущей даты до нового года.

- 5) Интерактивно введите две даты, посчитайте разницу в днях между этими датами.
- 6) Интерактивно введите два значения времени, подсчитайте разницу в часах и минутах между этими значениями, также подсчитайте равное количество часов между этими значениями.
- 7) Введите интерактивно время начала рабочего дня (08:00, 07:00), введите интерактивно время на обед (тип число, где единица это 1 час, а 0.5 – 30 минут, 0.25 – 15 минут). Вычислите время конца рабочего дня (рабочий день длится 8 часов). Усложните задачу, введите длину рабочего дня интерактивно.
- 8) Введите интерактивно время начала рабочего дня, время на обед (тип число, где единица это 1 час, а 0.5 – 30 минут, 0.25 – 15 минут), константно задается, что обед начинается через 4 часа после рабочего дня, весь рабочий день 8 часов. Выведите в окно сообщения следующую информацию в виде: Офис работает с 8:00 по 17:00, обед с 12:00 по 13:00 (с вашими значениями времени начала и окончания рабочего дня и перерыва на обед).

Часть 7. Функции преобразования и общие функции для примитивных типов

- 1) Введите интерактивно два числа (Например 2 и 5), выведите следующие строки
 $2 + 5 = 7;$
 $2 * 5 = 10;$
 $2 - 5 = -3;$
 $2 / 5 = 0,4.$

Результат операции получите внутри строки вывода.

- 2) Введите интерактивно три числа, вычислите самое минимальное и выведите на экран в удобном виде.
- 3) Введите интерактивно три даты, вычислите самую позднюю и самую раннюю дату.
- 4) Введите интерактивно три строки, вычислите самую короткую и самую длинную. Двумя способами (сравнение строк и чисел).
- 5) Преобразуйте строку вида «20.09.2012» в дату.
- 6) Даны строки вида [НазваниеАртикула] [НазваниеТовара] кол-во [КоличествоТовара], напишите метод который возвращает количество товара (тип число из строк)
Например: 23452 Карандаш зеленый кол-во 34

22111 Ручка шариковая кол-во 45

- 7) Введите интерактивно число дня, число месяца, и число года. Возвратите дату в любом удобном для вас формате.

Глава 3. Основные операторы

Часть 1. Условие: Если...Тогда...ИначеЕсли

- 1) Введите интерактивно числа X, Y.

Вычислите $Z = X - Y$, если $X > Y$

$Z = X + Y$, если $X \geq Y$.

- 2) Введите интерактивно X, Y.

Вычислите $Z = X/Y$. Если $Y = 0$, выведите предупреждение, что деление на 0 невозможно.

- 3) Введите интерактивно любое число. Проверьте, является ли оно целым (подсказка: используйте функцию *Цел* из 4 части урока 2).

- 4) Вычислите $(\sqrt{A} + B^C) * (\frac{D-E}{F * G})$, где D, E, задаются в модуле (суть константы), остальные переменные задаются интерактивно, обработайте все ситуации, чтобы не генерировались ошибки.

- 5) $\frac{\sqrt{A-1} - \sqrt[3]{B-5}}{1 + \frac{A^2}{2} + \frac{B^5}{4}}$, где A, B – задаются интерактивно, обработайте ситуации, чтобы не генерировались ошибки.

- 6) Введите интерактивно номер вида ху344яRU63, где 63 – код региона, код региона может быть любой, состоящий из двух или трех цифр, последние две цифры определяют регион. Определите код региона и выведите сообщение, что данный номер принадлежит такому-то региону (или код региона неизвестен). Должно быть не менее 5-ти известных регионов.

- 7) Запросите у пользователя поочередно город, улицу, дом, квартиру и выведите его адрес в виде: г. Москва, ул. Гагарина, д. 98, кв. 34. Если пользователь не ввел город, улицу, дом и т.д., заменяйте этот реквизит «---» (Например: г. Москва, ---, д. 98, кв. ---,).

- 8) Запросите у пользователя любое число (ввод сделайте целочисленным от 0 до 9). Если пользователь ввел 1, выведите свою ФИО в верхнем регистре; если 2 - в нижнем регистре; если 3 - в титульном регистре. В любом другом случае выведите просто фамилию.

- 9) Некоторый сотрудник должен отработать с 8 до 17 (норма), включая час на обед, почасовая ставка - 100 р. Сотрудник интерактивно вводит время начала рабочего дня, и время окончания рабочего дня. Если он превысил рабочий день, то ставка первых двух сверхнормативных часов увеличивается в 1,5 раза, а последующих часов в 2 раза. По условиям задачи в любом рабочем дне сотрудника есть 1 час на обед, который не оплачивается. Вычислите з/п сотрудника за рабочий день.

- 10) Введите интерактивно две любые даты, выведите соответствующие сообщения если эти даты: а) внутри одной недели; б) внутри одного месяца; в) внутри одного года; г) принадлежат к разным годам.

Часть 2. Цикл: Для...Цикл

- 1) Выведите следующий ряд $x_i + x_i^2 + x_i^3$, где $i = 1 \dots 10$;

- 2) Выведите следующий ряд $\frac{1}{20 - x_i - x_i^2}$, где $i = -10 \dots 10$, обработайте ситуацию когда знаменатель может быть равен 0.

- 3) Интерактивно введите дату, выведите в окно сообщений каждый день месяца, дата которого была задана.
- 4) Интерактивно введите дату, выведите в окно сообщений название дней недели той недели на которую приходится эта дата (используйте внутри цикла конструкцию *Если...ИначеЕсли*, и функции из части 6 2-го урока).
- 5) Задайте интерактивно две даты. Первая дата должна быть ранее второй, если это не так, то сообщайте об этом пользователю (см часть 2 второго урока). Выведите все даты с первой по последнюю, кроме субботы и воскресенья.
- 6) Выведите 12 месяцев, используя цикл *Для...Цикл*, и функцию *Формат*.
- 7) Введите интерактивно многострочную строку, используя цикл *Для...Цикл* и функции из части 5 второго урока, выведите каждую строку по отдельности.
- 8) Введите какое-нибудь предложение, используя цикл *Для...Цикл* и функции из части 5 урока 2 выведите в окно сообщений каждое слово по отдельности.

Часть 3. Рекурсия переменных Цикл: Пока...Цикл

- 1) Пусть пользователь задает два значения времени, первое значение должно быть раньше второго. Выведите на экран время в формате ЧЧ:мм, от начального до конечного значения с шагом 15 минут.
- 2) Пусть пользователь задаст две любых даты (начальная должна быть ранее конечной), выведите каждый день этого интервала дат в окно сообщений в таком виде: «01/10/2012 режим работы с 08-00 по 18-00», причем для выходных (суббота и воскресенье) должно быть «07/10/2012 режим работы с 08-00 по 16-00».
- 3) Введите два любых целых числа, проверьте, чтобы значение числа были от 1 до 12, и первое было меньше второго, иначе не выполняйте код. Выведите месяцы (в формате ММММ) по порядку начиная с номера первого числа и заканчивая номером второго числа.
- 4) Введите интерактивно некоторое слово. Выведите ряд сообщений сформированный из двух данных слов рядом. В каждом последующем элементе ряда, первое слово уменьшайте на одну букву слева, пока не останется одна буква, второе слово уменьшайте слово на одну букву справа, пока не останется одна буква.
- 5) Введите интерактивно некоторое предложение (слова разделены пробелами). Выведите ряд, сформированный из данного предложения, где каждый последующий элемент ряда уменьшается на одно слово. Например:
Программировать очень интересно и занимательно

Программировать очень интересно и

Программировать очень интересно

Программировать очень

Программировать

- 6) Пусть имеется адресная строка <http://lenta.ru/news/2013/10/02/betterkittens/>. Выведите в окно сообщений каждое слово разделенное флешем.

ДЛЯ ПРОДВИНУТЫХ

- 1) Вычислите следующую сумму $\sum_{k=1}^n (-1)^k / k$, т.е. (для тех кто не знаком с мат.анализом)
 $(-1)^1 / 1 + (-1)^2 / 2 + (-1)^3 / 3 + \dots + (-1)^n / n$, n – задайте интерактивно целое число.

2) Рассчитайте $\frac{1}{x + x^2 + x^3}$, где x = от -10 до 10, с шагом 0,1 (не забудьте про деление на 0).

3) Рассчитайте сумму $\sum_{k=1}^n \sum_{j=1}^m \frac{k+1}{j} * k$, где m и n – целые числа, задаются пользователем

интерактивно. Для тех кто не знаком с мат.анализом для $m = 3$, а $n=2$:

$$\frac{1+1}{1} * 1 + \frac{1+1}{2} * 1 + \frac{1+1}{3} * 1 + \frac{2+1}{1} * 2 + \frac{2+1}{2} * 2 + \frac{2+1}{3} * 2$$

Часть 4. Работа с процедурами и функциями

1) Пользователь интерактивно вводит любое целое число (например, 75). Выведите на экран следующие строки

000000000000075

000000000000075

000000000000075

000000000000075

000000000000075

2) Пользователь вводит собственное ФИО (Например, Иванов Иван Иванович), выведите следующий список

Иван

Иванов Иван

Иванов Иван Иванович

Иванов Иван

Иван

3) Напишите функцию ДобавитьРабочийДень(ДатаНачала,КоличествоРабочихДней), данная функция добавляет рабочие дни (с 1 по 5 день недели, праздники учитывать не будем), и возвращает дату. ДатаНачала изменяться не должна. КоличествоРабочихДней – необязательный параметр, по умолчанию равен 0.

4) Напишите функцию ДобавитьКоличествоРабочихЧасов(ДатаНачала, КоличествоЧасов). Задается, что рабочий день равен 8 часов. ДатаНачала - некоторая дата со временем (например, 13.02.2012 10-00), причем интервал времени должен быть от 8 часов до 17, а сама дата должна быть рабочая (с 1 по 5 день недели, праздники не учитываем), иначе ничего не считаем и возвращаем эту же дату. КоличествоЧасов – любое количество часов, по умолчанию = 1. Вычислите следующую дату, исходя из того сколько рабочих часов должно пройти. Например, ДобавитьКоличествоРабочихЧасов(13.02.2012 10.00,5) = 13.02.2012 15-00, а ДобавитьКоличествоРабочихЧасов(13.02.2012 10.00,10) = 14.02.2012 11-00. ДатаНачала – не должна измениться.

5) Пользователь интерактивно вводит любую дату (Например, 10.01.12). Выведите следующий список

05.01.12

07.01.12

10.01.12

13.01.12

15.01.12

ДЛЯ ПРОДВИНУТЫХ:

- 1) Вычислите следующее выражение $\frac{5!+2!}{3!} * \sqrt{7!}$, напишите функцию, которая считает факториал числа.
- 2) Пользователь вводит интерактивно числа a,b,c; вычислите $y = F(a,b*12,c/4) + F(a/2,c*5,0)$.
Где $F = \frac{\sqrt{|a-c|} + b * \sqrt{b-c}}{a + c * (a - \sqrt{|b-c|})}$, отработайте исключительные ситуации (деление на ноль, корень из отрицательного числа – выведите пользователю сообщение, что вычисление данной функции невозможно и приведите причину.).
- 3) Пользователь вводит интерактивно a,b,c, n – причем n – целое число; вычислите $y = (S(a,b/2) - S(c,a*5)) * S(b/5,c+3)$. Где

$$S = F(a,b,n) + T(a,c,n); F(a,b,n) = \sum_{i=1}^n (a+b)^n; T(a,c,n) = \sum_{i=2}^{n+1} (a^n - c^{n-1})$$

Часть 5. Попытка...Исключение

- 1) Пользователь вводит интерактивно числа a,b,c; вычислите $y = F(a,b*12,c/4) + F(a/2,c*5,0)$.
Где $F = \frac{\sqrt{|a-c|} + b * \sqrt{b-c}}{a + c * (a - \sqrt{|b-c|})}$, отработайте исключительные ситуации с помощью оператора *Попытка* (деление на ноль, корень из отрицательного числа – выведите пользователю сообщение, что вычисление данной функции невозможно и приведите причину.).
- 2) Получите целое число от деления: $\frac{A + D * 10}{A - D}, \frac{A * D}{5 - D}, \frac{A + D}{A - D}$, где A,D задаются пользователем. Обработайте исключительную ситуацию с помощью оператора *Попытка...Исключение*.
- 3) Пользователь вводит интерактивно числа a,b,c; вычислите $y = F(a/6,b+5,c-7) / F(a*2,c+5,0)$.
Где $F = \frac{\sqrt{c} + b * \sqrt{b-c}}{\sqrt[3]{(a-c)} + \sqrt[5]{b-c}}$, отработайте исключительные ситуации (деление на ноль, корень из отрицательного числа – выведите пользователю сообщение, что вычисление данной функции невозможно и приведите причину.).
- 4) Выведите ряд $\frac{1}{(x-5)*x}$, где x = -100 до 100, шаг 0.5 .

Глава 4. Основы конфигурирования

Часть 1. Основы конфигурации

Разработайте с нуля примитивную конфигурацию по складскому учету. В этой конфигурации должна быть следующая структура данных.

Перечисления:

- Вид контрагента. Значения:
 - Юр. Лицо
 - Физ. Лицо
- Вид номенклатуры
 - Товар
 - Услуги

Справочники:

- Номенклатура (иерархический). Реквизиты: Вид номенклатуры (с соответствующим типом)
- Склады
- Помещения (подчинен справочнику Склады)
- Контрагенты (иерархический). Реквизиты: Вид контрагента (с соответствующим типом).
- Пользователи
- ТипыЦен

Документы:

- Установка цен.
 - Реквизиты: Тип цен (с соответствующим типом);
 - Табличная часть: Цены.
 - Реквизиты табличной части:
 - Номенклатура (с соответствующим типом);
 - Цена (тип Число(10,2));
- Поступление товаров
 - Реквизиты:
 - Контрагент (с соответствующим типом);
 - Склад (с соответствующим типом);
 - Помещение (с соответствующим типом, сделать чтобы показывались в отборе помещения только выбранного склада)
 - Табличная часть: Товар
 - Реквизита табличной части
 - Номенклатура (с соответствующим типом);
 - Количество (тип Число(10,2));
 - Цена (тип Число(10,2));
 - Сумма (тип Число(10,2));
- Продажа товаров. Реквизиты и табличная часть аналогичны документу «Поступление товаров»
- Приходно-кассовый ордер
 - Реквизиты:
 - Контрагент (с соответствующим типом);
 - Сумма (тип Число(10,2));

- Расходно-кассовый ордер. Реквизиты и табличная часть аналогичны документу Приходно-кассовый ордер.

Регистры сведений:

- Установка цен номенклатуры.
 - Измерения
 - Тип цены (с соответствующим типом);
 - Номенклатура (с соответствующим типом);
 - Ресурс
 - Цена (тип Число(10,2)).

Регистры накоплений:

- Остатки товаров на складах (остатков).
 - Измерения:
 - Номенклатура (с соответствующим типом);
 - Склад (с соответствующим типом);
 - Ресурсы:
 - Количество (тип Число(10,2))
 - Документ «Поступление товаров» делает движение на плюс, а документ «Продажа товаров» делает движение на минус.
- Движения по кассе (остатков):
 - Измерения
 - Контрагент
 - Ресурс
 - Сумма
 - Документ «Приходно-кассовый ордер» делает движения на плюс, а документ «Расходно-кассовый ордер» делает движения на минус.
- Продажи (оборотов)
 - Измерения:
 - Контрагент (с соответствующим типом);
 - Номенклатура (с соответствующим типом);
 - Ресурс:
 - Сумма (тип Число(10,2));
 - Документ «Продажа товаров» делает движение по этому регистру.

Общий реквизит для всех документов: Пользователь (тип спр. Пользователи).

Реализуйте возможность создания документа «Продажа товаров» на основании документа «Поступление товаров».

Часть 2. Модули

- 1) Создайте внешнюю обработку, в которой в модуле формы создайте две глобальных переменных. Одной присвойте числовое значение, а второй значение типа Строка. Создайте процедуру, в которой эти глобальные переменные будут меняться значениями. Если было А = 10, Б = «Привет», то должно стать А = «Привет», Б = 10.
- 2) Создайте внешнюю обработку, в которой в модуле формы создайте две глобальных переменных. Одной присвойте числовое значение, а второй значение типа Строка. Создайте процедуру, в которой будут собственный переменный имеющие такое же

- название, как и глобальные. Проверьте, будут ли меняться значения глобальных переменных после выполнения этой процедуры.
- 3) Создайте общий модуль «РаботаСПеременнымиТипаСтрока» (выполнение на сервере, вызов сервера). Перенесите в этот модуль все выполненные задания для пятой части второй главы (сделайте отдельные процедуры(функции)). Переделаете все обработки из домашних работ для пятой части второй главы, так, чтобы осуществлялся вызов этого модуля и работа процедур (функций) этого модуля вместо работы кода на форме обработки.
 - 4) В конфигурации из книги задайте глобальную экспортную переменную в модуле приложения *Плановое время прибытия* (тип дата). Там же присваивайте ей значение по умолчанию 17 часов. При записи документа *Прибытие в гараж*, проверяйте время прибытия в гараж (в модуле документа). Если оно больше 17 часов, то выдавайте соответствующее предупреждение.
 - 5) В конфигурации из книги при записи элемента справочника *Автомобили*. Проверяйте номер автомобиля, чтобы он принадлежал 50 региону, иначе не давайте записывать элемент справочника. Номер региона задавайте глобальной переменной в модуле приложения.
 - 6) В конфигурации из книги проверяйте при открытии программы, чтобы системное время было в интервале от 8 до 17 часов, иначе не давайте открывать программу.
 - 7) В конфигурации из книги создайте общий модуль *Проверка дат прибытия и отбытия* (не глобальный), в котором опишите функции, сравнивающие параметр *Время* с плановым временем отбытия (8 часов) и плановым временем прибытия (17 часов). Используйте эти функции в при записи документов *ОтбытиеИзГаража* и *ПрибытиеВГараж*. Если оно не соответствует плановому, выдавайте сообщение.

Глава 5. Работа с управляемыми формами

Часть 1. Конструирование форм

- 1) Создайте основные формы для всех документов и справочников, созданных в домашнем задании для первой части четвертой главы. У документов

Часть 2. Командный интерфейс формы

- 1) У всех форм документов, созданных в предыдущей части сделайте возможность в командной панели просматривать движения по регистрам накопления (или сведений), которые делают эти документы.
- 2) Добавьте на форму документа «Поступление товаров» локальную команду, разместите её в командной панели формы. Создайте обработчики этой команды в клиентском контексте. При выполнении команды выводите произвольное сообщение.
- 3) Добавьте на форму документа «Продажа товаров» локальную команду, произвольно поместите её в любое удобное место формы. Создайте обработчики этой команды в клиентском контексте. При выполнении команды выводите произвольное сообщение.

Часть 3. Реквизиты и элементы формы

Все задания в этой части нужно выполнять на конфигурации, созданной в четвертой главе домашнего задания.

- 1) Для всех основных форм элементов справочников установите режим открытия - *БлокироватьОкноВладельца*. А для всех основных формы документов установите режим открытия окна – *Независимый*.
- 2) У форм документов «Приходно-кассовый ордер» и «Расходно-кассовый ордер» создайте две группы: «НомерИДата» и «КонтрагентИСумма». В первую соответственно поместите элементы «Номер», «Дата», а во вторую – «Контрагент» и «Сумма».
- 3) У форм документов «Поступление товаров» и «Продажа товаров». Создайте группу с видом *Страницы* с двумя страницами – Шапка и ТабличнаяЧасть. В первой разместите все реквизиты. А во второй табличную часть. Создайте в странице Шапка как минимум еще одну группу, в которой (или которых) разместите реквизиты документа, близкие по смыслу (например, номер и дата). Создайте в странице шапка группу с видом *КоманднаяПанель*, источником команд в которой будет форма.
- 4) Добавьте на все форм документа новое поле Ответственный, с которым будет связан реквизит документа Пользователь.
- 5) Создайте обработку, на которой разместите реквизиты со всеми четырьмя примитивными типами (число, дата, булево и строка). Разместите эти реквизиты на форме в виде элементов. Реквизиты с типами Число и Дата расположите горизонтально друг относительно друга. Под ними расположите реквизит с типом Булево (установите для него переключатель с видом *Тумблер*), и реквизит с типом Строка (сделайте возможность многострочного ввода).
- 6) У документа «Установка цен» создайте поле надписи, в котором будет отображаться текущая номенклатура из табличной части.
- 7) У всех документов создайте кнопку «Печать» со стандартной картинкой «Печать», поместите эту кнопку в командную панель документов. К кнопке привяжите команду, при выполнении происходит эмуляция печати.
- 8) У всех документов с табличными частями создайте над табличными частями декорации с названиями этих табличных частей.

Часть 4. Программирование управляемых форм

- 1) Создайте обработку, на форме которой создайте четыре реквизита. Первый *ФИО*, остальные: *Фамилия*, *Имя*, *Отчество*. Разместите их на форме в виде элементов. Последние три сделайте недоступными для редактирования (свойство поля ввода *ТолькоПросмотр*). Создайте рядом с полем *ФИО* кнопку с заголовком *Разбить*. Пользователь вводит в данное поле *ФИО* и нажимает кнопку *Разбить*, автоматически в полях *Фамилия*, *Имя*, *Отчество* появляются фамилия, имя и отчество соответственно. Для продвинутых: поработайте с событиями поля ввода и сделайте заполнение остальных полей после окончания редактирования поля *ФИО*.
- 2) Создайте обработку, на форме которой создайте шесть реквизита с типом Число. В первые два будем вводить значения с формы, а в остальных вычислять какое-то выражение. Поместите эти реквизиты на форму. Первые два доступны для редактирования. А остальные нет. Введите через поля ввода числа *X*, *Y*, а в остальных полях вычисляйте

- значения $X+Y$, $X-Y$, $X*Y$, X/Y (учтите, что Y может быть равен 0). Для вычисления всех значений создайте кнопку.
- 3) Сделайте форму расчета з/п сотрудника за день: есть форма, на ней поля *Начало рабочего дня эталон* и *Конец рабочего дня эталон*, они заполняются соответственно 08.00 и 17.00 пользователем. Есть поле *Почасовая ставка* она равна 500 р и также заполняется пользователем. Все эти поля расположены на странице *Настройки* (она вторая). На первой странице основная форма расчета з/п. Пользователь вводит начало рабочего дня (подсказка: для времени выбирайте тип даты *Время* при создании реквизита), вводит конец рабочего дня. Выведите в отдельное поле (закрывается для редактирования) расчет з/п. Исходя из того, что имеется восьмичасовой рабочий день (обед входит по умолчанию), и первые два часа переработки расцениваются по полуторной ставке, а последующие по двойной ставке. Для продвинутых: проверяйте заполнение всех полей (метод *ЗначениеЗаполнено*), а так же правильность заполнения всех полей: дата начала должна быть меньше даты конца.
 - 4) Вычислите деление числа X/Y . Создайте две страницы «Основная» и «Настройки». В странице «Настройки» разместите реквизит с типом *Булево (тумблер)*, и поле ввода (тип *число*). Поля назовите на ваше усмотрение. Если флажок нажат, то результат деления должен округляться с точностью до числа в поле ввода, если отжат, то результат не округляется. Исключите возможность деления на 0.
 - 5) Создайте 7 полей надписей, возле каждого поля ввода установите флажок (тумблер). А так же одно поле с типом *Дата*. Пользователь вводит любую дату, вычислите дни недели с понедельника по воскресенье той недели, на которую приходится эта дата, и выведите их в поля надписей, рядом с которыми флажок нажат. Т.е. если флажок отжат (тумблер выключен), то поле надписи должно быть пустым. Т.е. если пользователь ввел дату 06.12.2017, то нужно отобразить следующие даты: ПН – 04.12.2017, ВТ – 04.12.2017, СР – 05.12.2017 и т.д.

Глава 6. Объектные типы

Часть 1. Объекты метаданных

При выполнении заданий этой главы используйте конфигурацию из книги

- 1) Создайте обработку, которая будет содержать следующие поля: *Марка автомобиля (тип Строка)*, *Гос.номер (тип Строка)*, *Дата выпуска (тип Дата)*, *Коробка передач* с соответствующими типами. При нажатии на кнопку создайте элемент справочника *Марка автомобиля* (перед этим проверьте, есть ли уже марка с таким названием, если есть - то не добавляйте), после создайте сам элемент справочника *Автомобиль*.
- 2) Напишите обработку, полями которой будут *Марка автомобиля* и *Модель автомобиля*, пусть данная обработка создает элемент справочника *Марки автомобиля*, и элемент справочника *Модели Автомобиля*. Причем новый элемент справочника *Модели Автомобиля* подчиненный, его владелец это вновь созданный элемент справочника *Марка автомобиля* (Для установки владельца используйте свойство объекта *Владелец*).
- 3) Создайте обработку с двумя полями: *Название гаража* (тип значения строка) и *Гараж* (тип значения *Ссылка на справочник Гараж*). Пользователь вводит в поле название гаража и нажимает на кнопку. При нажатии кнопки второе поле заполняется ссылкой на гараж наименование, которого соответствует полю *Название гаража*, иначе выходит предупреждение.

- 4) Сделайте обработку, которая у модели автомобилей будет менять владельца. Нужную модель находите при помощи поиска. После поиска на форму выходит владелец найденной модели (Марка автомобиля), если модель не найдена, то выходит предупреждение. Пользователь в форме устанавливает нового владельца, нажимает кнопку *Записать* и если владелец отличен от текущего, то объект перезаписывается.
- 5) Создайте обработку, которая меняет коробку передач у автомобиля. Разместите на обработке поле с типом *Ссылка на справочник Автомобили*. И поле со ссылкой на перечисление. При нажатии на кнопку, у данного автомобиля поменяется коробка передач, если значение в поле отличается от текущего.
- 6) Создайте обработку, которая меняет год выпуска автомобиля. На обработке размещается поле с типом дата, и поле с типом значения *Ссылка на справочник Автомобили*. Пользователь выбирает автомобиль, устанавливает дату. И при нажатии на кнопку, дата выпуска автомобиля перезаписывается.
- 7) Создайте обработку с тремя полями: *Группа гаражей* (*Ссылка на справочник Гаражи*, в палитре свойств поля укажите свойство *ВыборГруппИЭлементов* как *Группы*), *Название гаража* (тип Строка) и *Ссылка на гараж* (*Ссылка на справочник Гаражи*). После нажатии кнопки, найдите элемент с таким названием внутри указанной группы. Если группа не заполнена, поиск не осуществляйте и выведите соответствующее сообщение. Если ничего не найдено, выведите сообщение.
- 8) Создайте обработку с тремя полями: *Ссылка на марку автомобиля*, *Название марки автомобиля*, и *Ссылка на модель автомобиля*. Пользователь выбирает ссылку на марку, пишет название модели, и осуществляет поиск по названию для тех моделей, у которых владелец указанная марка. Если ничего не найдено, выведите сообщение.
- 9) Сделайте так, чтобы нельзя было записать два гаража с одинаковыми наименованиями внутри одной группы.
- 10) Сделайте так, чтобы нельзя было записать два модели с одинаковым наименованием с одним владельцем.

Часть 2. Работа с документами

- 1) Разработайте самостоятельно обработку, в которой будут формироваться два документа: *Прибытие в гараж*, и *Выбытие из гаража*. Предоставьте пользователю выбирать дату прибытия, дату выбытия, сам автомобиль и гараж, а также то, оперативно будет проводиться данный документ или нет.
- 2) Добавьте реквизит *Ремонт* с типом Булево в справочник *Автомобили*. Запретите запись документа *Выбытие из гаража*, если у автомобиля данный реквизит *Истина*. (Используйте метод *ПриЗаписи* модуля объекта).
- 3) Создайте документ *Время на линии*, реквизиты: *Автомобиль*, *Гараж*, *Дата прибытия*, *Дата выбытия*, *Время на линии в часах*. Разместите данный документ в подсистеме *УчетАвтомобилей*. Добавьте права на создание и проведение этого документа роли Диспетчер. Пользователь не должен интерактивно создавать документ, он может его только просматривать (запретить интерактивные операции с документом правами). Документ должен создаваться встроенной обработкой, с двумя полями: *Документ выбытия* (тип значения *Ссылка на документ Выбытие из гаража*) и *Документ прибытия* (тип значения *Ссылка на документ Прибытие в гараж*). При выборе *Документа выбытия*, необходимо проверять, чтобы *Автомобиль* и *Гараж* совпадали, а *Дата прибытия* была позже *Даты выбытия*. Создавайте документ *Время на линии* только в том случае, когда поля обработки будут заполнены. Все реквизиты в документе *Время на линии* берите из двух выбранных документов, реквизит *Время на линии* вычисляйте. Для продвинутых: Настройте обработку так, чтобы поле *Документ прибытия* было недоступно для редактирования, пока не будет заполнено поле *Документ выбытия*.

- 4) Доработайте справочник *Автомобили*, внеся два реквизита: *Зимний расход* и *Летний расход* (тип число). Создайте документ *Нормативный расход топлива*, который должен вводится на основании документа *Прибытие в гараж*. В данном документе три поля: *Автомобиль*, *Пробег* и *Общий расход*. *Автомобиль* заполняется автоматически из документа основания, *Пробег* вводит пользователь, а *Общий расход* должен рассчитываться автоматически (при вводе данных о пробеге), при этом данные по расходу должны браться из справочника *Автомобили*. Причем учитывайте, что зима начинается с 15 ноября, а кончается 15 апреля. Проведите данный документ по регистру накопления *Нормативный расход топлива* (разработайте самостоятельно: регистр оборотный с одним измерением *Автомобиль* и одним ресурсом *Расход*). Определите самостоятельно, должен данный документ проводиться оперативно или нет.
- 5) Создайте документ *Выплата водителю за бензин*, данный документ вводится на основании документа *Нормативный расход топлива*, и имеет следующие поля: *Автомобиль*, *Основание* (тип *Ссылка на документ Нормативный расход топлива*) *Цена бензина* и *Сумма выплаты*. *Автомобиль* и *Основание* заполняются автоматически, стоимость бензина вводит пользователь, а *Сумма выплаты* вычисляется: $\text{Цена бензина} * \text{Общий расход}$ (извлекается из *Основания*). *Сумма выплаты* должна вычисляться автоматически при вводе пользователем цены за бензин. Данный документ должен проводиться по регистру накопления *Выплата за ГСМ*, который имеет одно измерение *Автомобиль* и один ресурс *Сумма выплаты*. Самостоятельно определите, должен документ проводиться оперативно или нет.

Часть 3. Форма как объект

- 1) Создайте обработку, на форме которой будут размещены следующие элементы: поле ввода (тип реквизита *Ссылка на справочник Автомобили*), поле надписи *Год выпуска* с пустым заголовком, поле надписи *Гос.номер* с пустым заголовком. Пользователь должен выбрать элемент справочника *Автомобили* в соответствующем поле. После выбора надпись *Год выпуска* должна приобрести вид: «Год выпуска автомобиля: 20 января 2001 г.», надпись *Гос.номер* вид: «Гос. номер автомобиля хх222х».
- 2) В конфигурации из книги, в форме документа *Прибытие в гараж* внесите следующие коррективы: если пользователь вводит дату прибытия более раннюю чем дата документа, то автоматически исправьте дату прибытия на дату документа.
- 3) В конфигурации, разработанной в главе 4 домашнего задания, у всех форм документов «Приходно-кассовый ордер» и «Расходно-кассовый ордер». Если заполнено поле *Контрагент*, то выводите заголовки по шаблону: «ПКО [НазваниеКонтрагента] на сумму [СуммаДокумента]», «РКО [НазваниеКонтрагента] на сумму [СуммаДокумента]».
- 4) В конфигурации, разработанной в главе 4 домашнего задания, для документов «Поступление товаров» и «Продажа товаров» в модуле объекта создайте процедуру *Печать*, которая будет эмулировать печать этого документа. На формах документов разместить команду «Печать», при выполнении которой будет запускаться процедура *Печать* из модуля объекта.
- 5) Создайте обработку, на форму которой поместите поле *Автомобиль* (тип *Строка*), а также кнопку «Поиск автомобиля». При нажатии на эту кнопку по справочнику *Автомобили* будет вестись поиск по названию, и если искомый автомобиль найден, то будет открыта форма этого элемента. Иначе выведено сообщение. Форма элемента должна открываться в отдельном окне.
- 6) Создайте обработку, в которой будет вестись расчет дневного заработка сотрудника. В данной обработке будут поля: *Ставка* (руб/ч), *Время начала рабочего дня*, *Время окончания рабочего дня*, *Время начала обеда*, *Время окончания обеда*, и *Сумма за день* (закрыта от редактирования). Если хотя бы один из параметров меняется (все параметры

должны быть заполнены), автоматически должна рассчитываться *Сумма за день*. Сумма за день равна количеству отработанных часов минус часы на обед, умноженному на ставку. Разница между временем начала обеда и временем окончания обеда должна быть не больше часа, в противном случае выдавать предупреждение и делать интервал в час. Время начала обеда и окончания обеда можно устанавливать только в том случае, если заполнены время начала рабочего дня и окончания рабочего дня. Время окончания рабочего дня должно заполняться, когда установлено время начала рабочего дня (и быть позже по времени минимум на час), иначе выводить предупреждение. Время обеда (начало и конец) должно быть между временем начала рабочего дня и временем окончания рабочего дня, иначе не давать устанавливать.

Глава 7. Универсальные коллекции значений

Часть 1. Массивы

- 1) Создайте одномерный массив, заполните его датами, начиная с первого числа текущего месяца, и заканчивая последним числом текущего месяца. Выведите элементы массива в окно сообщений.
- 2) Создайте одномерный массив, заполните его датами, начиная с первого числа текущего месяца, и заканчивая последним числом текущего месяца. Удалите все элементы, даты которых приходятся на субботу и воскресенье. Выведите элементы массива в окно сообщений.
- 3) Создайте многомерный массив с двумя измерениями: первое измерение будет содержать 12 массивов, соответствующих двенадцати месяцам года, каждый массив будет содержать все дни данного месяца. Выведите элементы массива в окно сообщений.
- 4) Создайте обработку, на которой будут два поля с типом значения *Дата* и с видом даты *Время*. Пользователь вводит время в первое поле и время во второе поле. Сделайте проверку ввода времени в поля: чтобы всегда время, введенное во второе поле, было больше времени в первом поле. Разместите на поле переключатель с тремя пунктами: *Заполнить с шагом в час*, *Заполнить с шагом в полчаса*, *Заполнить с шагом в 10 минут*. Создайте и заполните одномерный массив, в котором первым элементом будет время, введенное в первом поле, второй элемент будет больше первого элемента на то количество минут, которое указано в переключателе, и так далее. Последний элемент будет соответствовать времени, указанному в последнем поле. Выведите элементы массива в окно сообщений.
- 5) Заполните массив только нечетными числами от 1 до 100. Выведите элементы массива в окно сообщений.
- 6) Создайте обработку, разместите на ней два поля, первое *Начальная дата* с типом значения *Дата*, второе *Число дней* с типом значения *Число*. Создайте безразмерный массив. И добавляйте в него элементы, начиная с начальной даты, пока их количество не будет равно значению поля *Число дней*. Выведите элементы массива в окно сообщений.
- 7) Создайте обработку, в данной обработке создайте и заполните массив всеми датами, начиная с первого дня текущего года и кончая последним днем текущего года (массив создавайте и заполняйте в модуле объекта). Разместите на форме поле ввода с типом значения *Число*. Введите в поле ввода число, и с помощью данного массива узнайте, на какую дату в году приходится данное число.
- 8) Создайте многомерный массив заполненный лесенкой: в основном массиве первый элемент будет строка «1.1», второй элемент массив из 2-х элементов - строки «2.1.» и «2.2», третий элемент массив из 3-х элементов – строки «3.1», «3.2», «3.3», и т.д.

Количество элементов основного массива вводит пользователь. Выведите элементы массива в окно сообщений.

- 9) Заполните массив цифрами от 1 до 100, удалите каждый нечетный десяток цифр. Т.е. цифры от 1 до 10 удаляем, потом от 30 до 40, потом от 50 до 60 и т.д. Выведите элементы массива в окно сообщений.
- 10) Создайте функцию, которая копирует один произвольный одномерный массив в другой.
- 11) Заполните одномерный массив от 1 до 100, вставьте вместо каждого числа кратного 10 прочерк «-». Выведите элементы массива в окно сообщений.
- 12) Заполните массив датами января текущего года, введите любую дату текущего года не приходящуюся на месяц январь. Вставьте данную дату в массив, так чтобы ее номер по порядку в массиве совпадал с номером дня в году. Остальные элементы массива, которые будут иметь тип *Неопределено*, заполните датами по порядку. Выведите элементы массива в окно сообщений.

Часть 2. Структура

- 1) Создайте обработку с полями: *Фамилия, Имя, Отчество*. При нажатии на кнопку «Выполнить», создайте структуру, содержащую три элемента с ключами, названия которых соответствуют названию полей. Информацию из каждого поля записывайте в соответствующий элемент.
- 2) На обработке разместите два поля: «*Прибытие в гараж*» (тип значения *ссылка на документ «Прибытие в гараж»*) и «*Выбытие из гаража*» (тип значения *ссылка на документ «Выбытие из гаража»*). При нажатии на кнопку «Выполнить» создайте структуру с двумя элементами, один с ключом «*ПрибытиеВГараж*», второй – «*ВыбытиеИзГаража*». Получите доступ к объектам документов через ключи, и проведите их оперативно.
- 3) Создайте в обработке переменную типа *Структура*, которая будет глобальным объектом обработки (разместите в модуле обработки). Создайте две формы – *основная форма* (она будет открываться при запуске обработки) и *форма поиска* (она будет открываться при нажатии на кнопку на основной форме). В форме поиска разместите поле ввода (тип значения *строка*) и заполните структуру следующим образом: введите в форму поля ввода название марки автомобиля, и, если есть марка с таким названием, то создайте новый элемент структуры, ключ которого будет название марки, а значение - ссылка на элемент справочника. В основной форме разместите два поля: поле поиска (тип *строка*) и поле «*Марка*» (тип *ссылка на справочник Автомобили*). Введите название марки в поле поиска основной формы, выполните поиск одноименного ключа в структуре. Если есть ключ с таким названием, то значение данного элемента введите в поле «*Марка*». Если нет такого значения, то сообщите об этом пользователю.
- 4) Создайте два массива с количеством элементов 100, заполните один четными числами, другой нечетными числами. Создайте структуру с двумя элементами, первый с ключом «*Четные*», в значении которого массив с четными числами; второй с ключом «*Нечетные*», в значении которого массив с нечетными числами. Разместите на форме два переключателя: *Вывести четные числа, Вывести нечетные числа*. В зависимости от того какой переключатель стоит, получите из структуры соответствующий массив, и выведите элементы массива в окно сообщений.
- 5) Создайте четыре глобальных переменных в модуле формы, присвойте одной переменной тип *Структура*, а трем другим тип *Массив*. При открытии формы заполните массивы любыми значениями и создайте элементы структуры с произвольными ключами, так, чтобы значения всех элементов указывали на каждый из массивов. Создайте кнопку «*Проверить*» на форме, которая выводит данные массивов в окно сообщений. Создайте кнопку «*Изменение*» на форме, которая получает доступ к массивам через структуру. При

нажатии очистите массивы и заполните их другими данными. Проверьте, изменятся ли данные в массивах, с помощью кнопки «Проверить».

Часть 3. Соответствие

- 1) Создайте обработку с двумя формами. Первая *основная* (открывается при запуске обработки), вторая *форма ввода значений*. В первой разместите поле ввода с *ссылкой на справочник Автомобиля*, и кнопку «Создать элемент». Во второй разместите поля, соответствующие названиям реквизитов справочника *Автомобили*, причем поле *Марка* должна иметь текстовый тип. При нажатии кнопки «Создать элемент» на первой форме, открывается форма ввода значений в отдельном окне. Пользователь вводит все значения, которые передаются при нажатии на кнопку «ОК» в основную форму в виде соответствия (ключи соответствуют названиям реквизитов). В том случае, если значение передалось (обработайте ситуацию, когда пользователь просто закрыл форму), создавайте новый элемент справочника *Автомобили*, значение реквизитов берите из соответствия. Для реквизита *Марка* ищите, есть ли марка автомобиля с таким названием, если нет - создавайте новый объект справочника, иначе просто берите ссылку на реквизит. Ссылку на вновь созданный объект передайте в поле ввода.
- 2) Создайте два массива с произвольным количеством элементов, заполните их любыми данными по усмотрению, разместите на форме поле ввода с типом значения *ссылка на перечисление Коробка передач*. Создайте соответствие, у которого ключи будут значения перечисления, а значения элементов – массивы. При выборе того или иного значения перечисления, выведите в окно сообщений элементы соответствующего массива.
- 3) Разместите на форме четыре переключателя, каждый будет представлять собой какой-нибудь формат даты, разместите на форму поле ввода с типом значения *дата* (вид даты *Дата-Время*). Создайте соответствия, где в качестве ключей будут цифры от 1 до 4, а в качестве значений шаблоны формат дат (к примеру, "ДФ = ""дд- ММММ-гггг "" или "ДЛФ = ДДВ", на Ваш выбор). Названия переключателей должны быть в виде примера дат по тому или иному шаблону, который задан в соответствии. Номера переключателей должны соответствовать номерам в соответствии. После того как пользователь ввел дату, в зависимости от того, какой переключатель установлен, выведите в окно сообщений дату в соответствующем формате, если пользователь устанавливает другой переключатель, опять выводите в окно сообщений дату в соответствующем формате (формат при этом необходимо извлекать из соответствия в зависимости от номера переключателя.).
- 4) Разработайте обработку для создания нового элемента справочника *Автомобили*, разместите на форме поля ввода: *Наименование*, *Гос.номер* и т.п. (все реквизиты справочника кроме реквизита *Марка*). Разместите на форме четыре переключателя с названиями разных марок. Создайте соответствие, в котором номеру каждого переключателя будет соответствовать ссылка на нужный элемент справочника *Марка*. При нажатии на кнопку «Создать», создайте новый элемент справочника *автомобили*, а марку для данного элемента извлеките из соответствия, в зависимости от того какой переключатель установлен.

Часть 4. Список Значений

- 1) Создайте форму, на которой разместите три поля ввода (тип *текст*): *Фамилия*, *Имя*, *Отчество* и один список значений: *ФИО* и кнопку «Заполнить список». При нажатии кнопки, данные из полей ввода переносятся в список значений (*Фамилия* – отдельный

- элемент, *Имя* – отдельный элемент, *Отчество* – отдельный элемент), причем если данные уже были, то они должны быть очищены.
- 2) Создайте форму со списком значений, при открытии формы заполняйте список числами начиная с первого числа текущего месяца и заканчивая последним числом текущего месяца. Разместите на данной форме две кнопки – «Удалить четные дни», и при нажатии на нее удаляйте все элементы, которые соответствуют четным дням, и «Удалить нечетные дни» – при нажатии на нее удаляйте все элементы, которые соответствуют нечетным дням.
 - 3) Создайте форму со списком значений, разместите на данном списке все месяцы года (любым способом), создайте две кнопки рядом со списком: «Вверх» и «Вниз». При нажатии на кнопку «Вверх», текущий элемент переносится на одну позицию выше, при нажатии на кнопку «Вниз» - на одну позицию ниже. Не забудьте обработать граничные условия.
 - 4) Создайте на форме список значений и три переключателя и кнопку «Заполнить». Создайте на форме три глобальных переменных *Группа1*, *Группа2*, *Группа3*. Эти переменные должны быть массивами и при открытии формы должны быть заполнены произвольными фамилиями и инициалами. При нажатии на кнопку «Заполнить» в зависимости от того, какой переключатель установлен, список значений будет заполняться из соответствующего массива.
 - 5) Создайте форму и разместите на ней три списка значений. Первый при открытии заполняйте датами текущего квартала (причем в значении должна храниться дата, а в представлении дата в формате *дд-ММММ-гггг*). Разместите кнопку удаления текущих строк данного списка. Второй список значений заполняется только четными датами из первого списка (при нажатии определенной кнопки), а третий список заполняется нечетными датами из первого списка (при нажатии какой-нибудь кнопки). Поместите кнопку «Заполнить обратно», при нажатии на которую первый список очищается и заполняется данными сначала из второго, потом из третьего списка. Поместите кнопку «Сортировка», при нажатии на которую будут сортироваться элементы всех списков по значению.

Глава 8. Таблицы

Часть 1. Таблицы значений

- 1) Создайте программным способом таблицу значений со следующими полями
(переменная *Таблица* должна быть глобальной)
 - *Дата* (тип *дата*)
 - *Автомобиль* (тип *Ссылка на справочник Автомобили*)
 - *Пробег* (тип *число*)
 - *Расход бензина* (тип *число*).(таблица и колонки должны быть уже созданы при открытии формы).

Разместите на форме четыре поля ввода соответствующие трем полям таблицы, и кнопку «Ввод». При нажатии кнопки «Ввод», если все поля заполнены, вносите соответствующую информацию.

Разместите кнопку «Подсчитать итог по автомобилю». При нажатии на эту кнопку, необходимо подсчитать по данным из таблицы, каков был общий пробег и общий расход бензина по автомобилю, который на данный момент указан в поле *Автомобиль*.

Создайте на форме таблицу значений (поля: *Дата*, *ФИО*, *Сумма*). Поле *Дата* закройте от редактирования. Создайте кнопку «Загрузить», которая заполнит данную таблицу датами, причем так, что первая строка – первое число текущего месяца, вторая строка – второе число и т.д. Сделайте заполнение данной колонки, используя метод *Загрузить колонку*.

- 2) Создайте таблицу с двумя колонками – первая *Дата* (тип *дата*), вторая *Формат даты* (тип *строка*). Заполните ее датами текущего месяца с первой по последнюю, и соответствующим форматом даты (на ваше усмотрение). На форме разместите поле *Дата*, при выборе даты ищите данную дату в таблице. И если она найдена, то выведите в окно сообщений соответствующий формат, а если нет, то сообщайте об этом пользователю.
- 3) Создайте на форме табличное поле со следующими полями: *Дата*, *Автомобиль*, *Пробег* (км), *Норма расхода* (л/100 км), *Расход* (л). Пересчитайте *Расход топлива* при изменении полей *Пробег* и *Норма расхода* по следующей формуле:
$$\text{Расход топлива} = \text{Пробег} * \text{Норма} / 100.$$
 Сделайте по данной формуле обратный расчет *Пробега* при изменении поля *Расход топлива*.
- 4) Создайте табличное поле на форме, с полями *Дата*, *ФИО* и *Сумма*. Заполните его автоматически при открытии, причем *ФИО* и *Сумма* останутся пустыми, а *Дата* заполнится датами текущего месяца (начиная с первого числа месяца по последнее), в таблице должно соответствующее количество строк. Сделайте так, чтобы строки с датами, на которые приходятся выходные, окрашивались в красный цвет. *ФИО* и *Сумму* заполняйте по своему усмотрению, но если *Сумма* больше 1000 р, то шрифт в колонке должен быть синего цвета, подчеркнутый и жирный.
- 5) Создайте таблицу значений на форме с четырьмя любыми полями на ваш выбор (хотя бы одно поле должно быть объектного типа). Все колонки в таблице должны быть заблокированы. Разместите на форме четыре поля ввода, которые будут соответствовать колонкам таблицы плюс поле *Индекс* (тип *целое число*, *положительное*), и кнопку «Ввести». При нажатии на кнопку «Ввести» добавляйте строку по нужному индексу. Если *Индекс* больше имеющегося количества, выводите соответствующее сообщение.
- 6) Создайте на форме таблицу со следующими полями: *Дата*, *Автомобиль*, *Сумма за топливо*. Создайте меню «Свертка» в командной панели таблицы с тремя кнопками: «Свернуть по дате», «Свернуть по автомобилю» и «Восстановить данные». Соответственно при нажатии на кнопку сворачивайте таблицу значений либо по *Дате*, либо по *Автомобиле*, суммируя поле *Сумма*. При этом скрывайте в таблице на форме либо поле *Дата*, либо *Автомобиль*. При нажатии на кнопку «Восстановить данные», возвращайте таблицу, которая была до свертки.
- 7) Создайте на форме таблицу значений и список значений. Таблица значений с полями *ФИО* (тип *строка*), *Автомобиль* и *Дата*. При выделении курсором поля *Автомобиль* (используйте событие *ПриАктивизацииКолонки* табличного поля), вводите в список

- значений все *ФИО*, которые приходятся на данный *Автомобиль* в таблице, предварительно свернув их.
- 8) Создайте на форме две таблицы: *Норма расхода автомобиля* (поля: *Автомобиль* и *Норма расхода*) и *Пробег автомобиля* (поля: *Дата*, *Автомобиль*, *Норма расхода*, *Пробег*, *Расход*). При выборе *Автомобиля* в таблице *Пробег*, находите *Норму расхода* в таблице *Норма расхода автомобиля* и подставляйте в соответствующее значение, *Расход* рассчитывайте по формуле из задания 4.
- 9) Создайте два идентичных табличных поля на форме, с полями: *Дата*, *Автомобиль*, *Сумма затрат*. Первую таблицу может заполнять пользователь. Вторая таблица закрыта от редактирования. При нажатии на строку первой таблицы (используйте событие табличного поля *ПриАктивизацииСтроки*), заполняйте второе табличное поле всеми строками из первого поля, на которых имеется *Автомобиль*, присутствующий на активированной строке.

Часть 2. Табличные части справочников и документов

- 1) Создайте для конфигурации из книги документ *Пробег автомобилей*, с табличной частью *Автомобили*, у которой есть следующие реквизиты: *Автомобиль*, *Норма расхода* (л/100 км), *Пробег*, *Расход*. Разместите на форме элемента данную табличную часть. *Расход* должен считаться автоматически $\text{Расход} = (\text{Пробег} / \text{Норма}) * 100$. Для справочника *Автомобили* создайте новый реквизит *Норма расхода*. При выборе элемента справочника в табличную часть, если *Норма расхода* не равна нулю, она должна подставляться в табличную часть. Создайте кнопку «Пересчет нормы», при нажатии на которую будет пересчитана *Норма расхода* для всех строк табличной части. Не забудьте также пересчитать *Расход*.
- 2) Создайте обработку, на форме которой будут три поля ввода: *Автомобиль* (ссылка на справочник *Автомобили*), *Пробег автомобиля* (ссылка на документ *Пробег автомобиля*) и *Расход* (число). Создайте процедуру, которая будет срабатывать на выбор автомобиля или документа, и, если данные заполнены, подсчитывать общий пробег по автомобилю в данном документе.
- 3) Создайте обработку, на форме которой разместите табличное поле с теми же реквизитами что и у табличной части документа *Пробег автомобилей* (не забудьте сделать расчет *Расхода*). Создайте возможность загружать данную табличную часть в любой документ (соответствующее поле ввода разместите на форме).
- 4) Создайте обработку со списком значений и полем ввода (ссылка на документ *Пробег автомобиля*). При выборе документа выводите в список все автомобили, которые есть в табличной части документа (без повторов).
- 5) Создайте обработку с двумя полями ввода (тип *ссылка на документ Пробег автомобиля*) и кнопкой «Замена». При нажатии на кнопку поменяйте значение табличной части одного документа на значение табличной части другого документа.

Глава 9. Получение данных

Часть 1. Понятие выборки

- 1) Создайте на форме элемент Переключатель с двумя пунктами: Наименование и Код, создайте табличное поле с двумя колонками – код и наименование. Заполните табличное поле элементами справочника *Марки автомобилей*, причем в зависимости от того какой

- на данный момент установлен переключатель, сортируйте либо по коду, либо по наименованию.
- 2) Создайте на форме элемент поле ввода с типом ссылка на перечисление Коробка передач. По умолчанию устанавливайте «автоматическая». Создайте табличное поле с колонками: автомобиль (тип ссылка на справочник Автомобили) и коробка передач (тип ссылка на перечисление Коробка передач). При открытии формы автоматически заполняйте данное табличное поле элементами справочника *Автомобили*, у которых есть автоматическая коробка передач, в колонку коробка передач заполняйте значение реквизита соответствующего элемента. При изменении поля коробка передач, перезаполняйте табличное поле теми элементами справочника у которых реквизит коробка передач совпадает с новым реквизитом поля ввода.
 - 3) Создайте на форме две таблицы– *Марки* и *Автомобили*. Заполняйте таблицу *Марки* всеми элементами справочника «Марки автомобилей», а при выделении строки в таблице *Марка*, заполняйте табл. поле автомобили всеми автомобилями данной марки. (Подумайте, как необходимо доработать конфигурацию для исполнения данных возможностей).
 - 4) Создайте форму с таблицей, в эту таблицу выводите все группы и элементы справочника *Гаражи*. На данной форме разместите три переключателя с пунктами: все, группы, элементы. В зависимости от того какой установлен переключатель выводите либо группы и элементы, либо только группы, либо только элементы.
 - 5) Создайте форму с двумя таблицами, в первую таблицу выводите все группы справочника *Гаражи*. Создайте флажок «Выводить с иерархией». Когда флажок включен, при активации строки первого табличного поля выводите все элементы, которые входят в группу, включая элементы подчиненных групп, если он выключен, то выводите только собственные элементы указанной группы.
 - 6) Создайте форму с двумя таблицами: в первой выводите документы «Установка цен на топливо» за указанный период (используйте элемент выбор периода). При активации соответствующей строки первой таблицы выводите во второе табличное поле все элементы табличной части данного документа.
 - 7) Создайте форму с одной таблицей, и двумя полями ввода (Автомобиль и гараж). Заполняйте таблицу документами *Выбытие из гаража* по указанному периоду (используйте элемент выбор периода). Сделайте возможность отбора либо по автомобилю, либо по гаражу (подправьте конфигурацию если надо). Причем при выборе в поле ввода Автомобиль значения, поле Гараж должно затираться, и наоборот.
 - 8) Создайте форму с одной таблицей. Создайте на форме переключатель с двумя пунктами: «Сортировка по дате документа», «Сортировка по дате прибытия», а также флажок «Убытие», Выводите в табличное поле документы *Прибытие в гараж* за указанный период (используйте элемент выбор периода), и отсортировывайте либо по дате документа, либо по дате прибытия (подправьте конфигурацию если надо).

Часть 2. Работа с запросами

- 1) Сделайте форму с табличным полем (колонки: автомобиль, номер, марка), и с полем ввода (тип значения Марка автомобиля). При выборе марки перезаполняйте табличное поле автомобилями только с указанной маркой.
- 2) При записи элемента справочника автомобиль (используйте метод ПередЗаписью модуля объекта справочник), проверяйте, есть ли автомобиль с таким номером в базе, если есть, то не записывайте.
- 3) Разместите на форме два элемента: список и таблицу, реализуйте возможность добавлять в поле списка и удалять из поля списка элементы справочника автомобиль, после при

нажатии на кнопку заполняйте табличное поле документами Прибытие в гараж и выбытие из гаража с теми автомобилями, которые есть в данном поле..

- 4) Создайте форму, на которой поместите таблицу «Количество путевых листов» (Колонки – Автомобиль, Количество путевых листов). Выведете, сколько документов печать путевых листов было оформлено на конкретный автомобиль.
- 5) Выведете в таблицу на форме все документы *Прибытие в гараж* с отбором по нужному гаражу, упорядоченные по дате прибытия.
- 6) Создайте таблицу на форме, в которой будут поля *Автомобиль* и *Прибытие в гараж*, выведете список автомобилей и документы прибытия, которые были у данных автомобилей. Если документов прибытия нет, то выводите сообщение «Документы прибытия не оформлены».
- 7) На форме подсчитывайте, сколько у Вас имеется проведенных документов *Прибытие в гараж* и сколько не проведенных. Выводите результаты в поле надписи.

Глава 10. Регистры сведений и накоплений

Часть 1. Регистры сведений. Работа с записями

- 1) Создайте форму, на которой разместите два поля ввода Коэффициент для лета и Коэффициент для зимы. При нажатии на кнопку изменяйте в регистре сведений Норма расхода значение ресурса норма. Умножайте на соответствующий коэффициент для с сезоном лето или зима.
- 2) Заполните регистр сведений *Основной гараж автомобиля* (создан в 5 главе). Следующим образом: на форме разместите список значений *Автомобили* и поле ввода *Гараж*. Автомобили загружайте автоматически в список значений (с помощью выборки или запроса), в списке значений установите возможность ставить флажки. Загружайте в регистр сведений *Основной гараж автомобиля* только те автомобили, которые отмечены флажками в списке, значение гаража берите из поля ввода. Проверяйте на уникальность записей.
- 3) Создайте форму, где с помощью менеджера записи будете создавать новую запись в регистре *Основной гараж автомобиля*. Проверяйте уникальность записи. Данные должны браться с формы.
- 4) Создайте обработку, с полями ввода: Дата, Автомобиль, Сезон. Работаем с регистром «Норма расхода». Если все поля заполнены, то удаляйте записи, которые соответствуют данным полям, если какое-нибудь одно поле не заполнено, то удаляйте записи, которые соответствуют двум другим полям (например, все записи для выбранного автомобиля и сезона). Если заполнено только одно поле, то удаляйте все записи, которые соответствуют данному полю (например, Автомобиль).
- 5) Создайте на форме таблицу с колонками Автомобиль, сезон, норма расхода. Загружайте данные из таблицы в регистр сведений «Норма расхода», проверяйте уникальность записей.
- 6) Создайте регистр сведений «Выбытие из гаража», данный регистр подчинен регистратору (документ «Выбытие из гаража»), имеет следующие измерения: автомобиль, гараж; ресурсы: дата выбытия. При проведении документы «Выбытие из гаража» делают соответствующую запись в регистре.

Часть 2. Регистры сведений. Получение данных

- 1) Создайте обработку, которая будет возвращать норму расхода указанного автомобиля за указанный сезон на текущую дату.

- 2) Создайте форму, которая будет подсчитывать пробег конкретного автомобиля за указанный интервал. Выполните эту задачу, не используя язык запроса 1С.
- 3) Создайте форму с таблицей (колонки автомобиль, норма расхода) и полем ввода сезон. Выводите в таблицу норму расхода всех автомобилей для указанного сезона на текущую дату.
- 4) Создайте форму с таблицей (колонки: дата, автомобиль, пробег), и полем ввода Дата. Необходимо вывести в табличное поле самую свежую информацию из регистра «Пробег автомобиля» на указанную дату. (Не используя язык запросов).
- 5) Создайте форму с таблицей (колонки: дата, автомобиль, пробег, норма расхода, расход), и полем ввода Дата. Получите последние данные на указанную дату о пробеге всех автомобилей, норму расхода для каждого автомобиля, и рассчитайте расход по формуле: $\text{пробег} * \text{норма} / 100$. Сделайте данную задачу двумя методами: используя язык запросов и используя методы менеджера регистров сведений.
- 6) Создайте форму с таблицей (колонки: дата, автомобиль, пробег, норма расхода, расход) и интервалом дат. Выведете за данный интервал дат пробег всех автомобилей, норму расхода для каждого автомобиля, и рассчитайте расход по формуле: $\text{пробег} * \text{норма} / 100$. Сделайте данную задачу двумя методами: используя язык запросов и используя методы менеджера регистров сведений.
- 7) В п 8. прошлого домашнего задания создан РС «Выбытие из гаража». Создайте форму с таблицей (колонки: автомобиль, гараж, дата выбытия), и с полями ввода: автомобиль, гараж и дата. При нажатии кнопки выполнить получайте данные о выбытии указанного автомобиля из указанного гаража на дату более близкую к выбранной.

Часть 3. Регистры накопления. Работа с записями

- 1) Создайте функцию, которая заполняет приход регистра накопления *Топливо в автомобилях* по документам *Заправка топлива* (по проведенным, и по которым нет движений). Подсказка: используйте метод Добавить приход набора записей.
- 2) Создайте функцию, которая заполняет расход регистра накопления *Топливо в автомобилях* по документам *Отчет о расходе топлива* (по проведенным, и по которым нет движений). Подсказка: используйте метод Добавить расход набора записей, и используйте свойство регистра вид движения: [Запись].ВидДвижения = ВидДвиженияНакопления.Приход;
- 3) Создайте функцию очистки регистра накоплений *Топливо в автомобилях* с параметром ТипТоплива, данная функция должна очистить приход и расход всех записей с указанным типом топлива.
- 4) Создайте обработку с табличным полем «Документы прибытия выбытия», имеющем две колонки: Признак (Булево), и Документ (Составной тип ссылка на документ Заправка топлива и Отчет о расходе топлива). Заполняйте при открытии всеми проведенными документам. Удаляйте движения из регистра *Топливо в автомобилях* для всех документов из списка отмеченных флажком.
- 5) Доработайте задание из п 4: создайте записи для документов помеченных флажком (приход и расход для соответствующих документов).

Часть 4. Регистры накопления. Получение данных

- 1) Разместите на форме поле ввода (тип значения перечисление коробка передач), и таблицу с колонками автомобиль и пробег. Выводите пробег только тех автомобилей, коробка передач которых такая же, как в поле ввода.

- 2) С помощью метода «Остатки» менеджера регистра накопления, выведите в таблицу по регистру накопления «Топливо в автомобилях» в разрезе автомобилей. Предусмотрите возможность сделать отбор по конкретному типу топлива.
- 3) С помощью метода «Обороты» менеджера регистра накопления, выведите оборот по топливу из регистра «Топливо в автомобилях» в разрезе автомобилей. Предусмотрите возможность сделать отбор по конкретному типу топлива.
- 4) Выведите в таблицу расход топлива автомобилей по маркам, за указанный период. Сделайте сортировку по расходу.
- 5) На форме разместите поля ввода тип топлива, период, и таблицу с колонками: автомобиль, приход, расход. Извлекайте из регистра накопления «Топлива в автомобилях» данные о приходе и расходе конкретного топлива по указанному в поле ввода по неделям (например, за неделю данный автомобиль заправился на 10 литров, а отчитался за 5 литров раз).
- 6) Используя регистр накопления «Топливо в автомобилях» выведите в табличное поле за указанный период следующую информацию: документ заправки топлива (или отчета о расходе), остаток на начало периода, заправка, расход, остаток на конец периода.

Глава 11. Вывод информации

Часть 1. Печатные формы

- 1) Создайте печатную форму для документа выбытие из гаража. Разместите на ней следующую информацию: название документа, номер документа, дата документа, дата выбытия, и гараж выбытия.
- 2) Доработайте печатную форму документа выбытие из гаража, сделав три области: шапка (Название документа, Дата и номер). Область пробег: гараж и информация о пробеге. Область подвал: подписи.
- 3) Создайте печатную форму документа печать путевых листов. Шапка: название документа (Путевой лист), номер документа (уберите все лишние нули), дата документа (в формате 1 января 2012 г.). Общая часть: дата выбытия (пустое поле), дата прибытия (пустое поле), водитель (пустое поле), показание спидометра на момент выбытия (пустое поле), показание спидометра на момент выбытия (пустое поле), пробег (пустое поле). Подвал: подписи.
- 4) конфигурации созданной в домашнем задании 1 части 4 главы, создайте печатные формы к документам поступление товаров и списание товаров. Структуру полей печатных форм придумайте самостоятельно.

Часть 2. Отчеты

- 1) Все домашние задания из 2-ой части 9 урока сделайте в виде отчетов. Простых и СКД.
- 2) Все домашние задание из 2-ой и 4-ой части 10 урока сделайте в виде отчетов. Простых и СКД.