

Определение сложности работы алгоритмов платформы 1С: Предприятие

Алгоритмы решения задач на платформе 1С:Предприятие зависят от постановки и сложности самой задачи.

Стандартный алгоритм разработки БД, включает в себя:

1. Особенности условий задачи;
2. Настройка конфигурации;
3. Настройка интерфейса (через подсистемы);
4. Определение списка документов;
5. Построение начальной конфигурации задачи;
6. Настройка документов;
7. Обработка проведения документов;
8. Создание отчетов.

Рассмотрим построение алгоритма решения задачи на примере:

Особенности условия задачи

1. Складской учет не ведется.
2. Списание себестоимости товаров организовано по партиям, в зависимости от метода списания себестоимости (FIFO или LIFO), принятого на этот год.
3. В первую очередь должен списываться товар из партии, указанной в шапке документа Расходная накладная, а затем - в соответствии с текущим методом списания себестоимости.

Настройка конфигурации

1. Включить возможность использования обычных форм (нужно для Консоли запросов):
 - Сервис – Параметры – флаг «Управляемое приложение и обычное приложение»;
 - свойства конфигурации – флаг «Использовать обычные формы в управляемом приложении»;
2. Настройка конфигурации (свойства):

- свойства конфигурации – Режим управления блокировкой данных - Управляемый;
- добавить кнопку запуска отладки под Толстым клиентом (эта обработка Консоль запросов запускается только под ним).

Настройка интерфейса (через Подсистемы):

1. Создать Подсистемы (порядок – согласно порядку задачи: <конфигурация> – пкм – Открыть командный интерфейс):
 - Оперативный учет;
 - Бухгалтерский учет;
 - Расчеты;
 - Бизнес-процессы (или Управляемые формы) – по задаче;
 - Сервис.
2. Упорядочить подсистемы можно в режиме 1С Предприятие (Настройки – Настройка панели разделов).
3. В созданные подсистемы (Подсистемы – пкм – Все подсистемы) добавить те объекты, которые должны в них входить (отображаться в интерфейсе):
 - Создать Группы команд:
 - Справочники;
 - Документы;
 - ПланыВидовХарактеристик;
 - РегистрыСведений;
 - РегистрыОУ;
 - РегистрыБУ;
 - РегистрыСПР.
 - Например, подсистема "Оперативный учет" может включать:
 - Справочник Номенклатура;
 - Документ Приходная накладная;
 - Документ Расходная накладная;
 - Регистр накопления Остатки номенклатуры и т.д.
4. Распределить объекты подсистем по соответствующим группам команд.
5. Настроить интерфейс клиентского приложения (<конфигурация> – пкм – Открыть интерфейс клиентского приложения):
 - Сверху убрать Панель функций текущего раздела, добавить Панель открытых;
 - Слева – Панель разделов.

6. Включить возможность использования Консоли запросов (запускается только под Толстым клиентом):
- В модуле формы консоли изменить обработку исключения Предупреждение(Сред(ОписаниеОшибки(),69)); на ПоказатьПредупреждение(,Сред(ОписаниеОшибки(),69));

Определить список документов и выполнить начальную настройку конфигурации

1. Определяем список документов, используемых в решении. Все они явно описаны в задании, свои не добавлять.
2. Если в условии упоминаются услуги, то сразу готовим конфигурацию:
 - проверяем (создаем) Перечисления.ВидыНоменклатуры со значениями Товар, Услуга ;
 - в справочник Номенклатура добавляем реквизит ВидНоменклатуры , тип ПеречислениеСсылка.ВидыНоменклатуры ;
3. Если в условии упоминается учетная политика (ФИФО, ЛИФО, средняя), то сразу готовим конфигурацию:
 - проверяем (создаем) Перечисления.УчетнаяПолитика со значениями ФИФО, ЛИФО, Средняя;
 - создать регистр сведений (периодический) УчетнаяПолитика (по нему будем определять метод учетной политики на дату документа):
 - периодичность - по заданию (например, В пределах года);
 - режим записи - Независимый;
 - добавить его в подсистему (например, ОУ);
 - вкладка "Данные" - добавить ресурс МетодСписанияСебестоимости с типом ПеречислениеСсылка.УчетнаяПолитика .
4. Если ведется складской учет, то сразу готовим конфигурацию:
 - добавляем справочник Склады;
 - в регистры накопления добавить измерение Склад , тип СправочникСсылка.Склады;
5. (При желании, сэкономит время на правильный ввод данных в документы) Настроить автоподсчет сумм (строка и документ) в Приходной и Расходной накладных (общий модуль + модули форм

документов, в свойствах общего модуля проверить флаги «Клиент», «Сервер»):

- при изменении в строке табличной части Количество и Цена;
- при удалении или добавлении копированием.

Спроектировать регистры (место хранения данных)

Общий алгоритм:

1. Определить количество ресурсов – по накапливаемым данным (количество, сумма, себестоимость, сумма продажи и т.д.). Добавить в соответствующий регистр, проверив соответствие типа, длины и точности.
2. Определить количество измерений – по детализации ресурсов (склад, партия, контрагент, номенклатура и т.д.). Добавить соответствующие измерения.
3. Проверить установку флага «Запрет незаполненных значений» (при необходимости).
4. Регистр накопления:
 - Прочее - флаг «Разрешить разделение итогов» - включить.

Проанализируем условие задачи и таблицы из условия:

*Необходимо построить отчеты по **продажам** товаров за период и остаткам товара на указанную дату.*

Т.о. нам понадобятся 2 регистра накопления:

первый - оборотный, по продажам;

второй - остатков.

В регистрах будут храниться:

Регистр **ОстаткиНоменклатуры**:

- измерения:

Номенклатура, тип СправочникСсылка.Номенклатура ;

Партия, тип ДокументСсылка.ПриходнаяНакладная ;

- ресурсы:

Количество;

Себестоимость, тип Число (длина 12, точность 2);

Регистр Продажи (вид регистра Обороты) (*видео от Леонтьева, с 30 мин*):

- измерение:

Номенклатура, тип СправочникСсылка.Номенклатура ;

- ресурсы:

Количество;

Себестоимость (дублирование, НО упростит формирование **отчета**);

СуммаПродажи, тип **Число** (длина 12, точность 2);

Настройка документов

1. Добавить реквизиты по заданию (поле **Партия** в шапку документа **РасходнаяНакладная**).
2. Добавить и настроить форму:
 - форму документов создавать после добавления всех реквизитов документа (ускорение разработки).
 - в командном интерфейсе форм в пункт «Перейти» добавить ссылки на регистры, установив соответствующие флажки.
3. На вкладке «Движения»:
 - Проведение – Разрешить (при ручном вводе движений – Запрещено);
 - Оперативное проведение – Разрешить;
 - Удаление движений – Удалять автоматически при отмене проведения.
4. Движения формировать через конструктор (табличная часть выбирается только тогда, когда из неё заполняются измерения!) и определить методику проведения:
 - если для заполнения набора записей данные регистра не нужны – новая (если новая и приход – то контроль не нужен);
 - если для заполнения набора записей данные регистра нужны – старая.

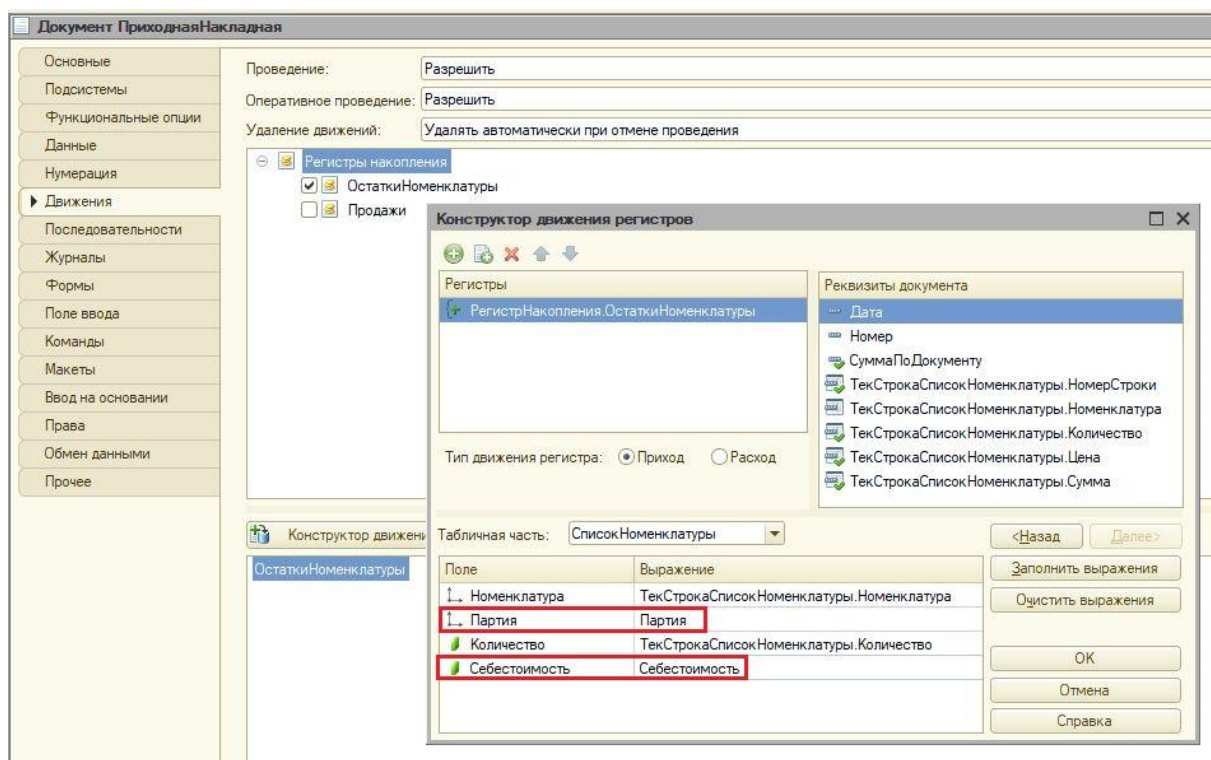
Обработка проведения документа ПриходнаяНакладная

Особенности:

1. Приходная накладная записывает движения только в регистр накопления **ОстаткиНоменклатуры**.
2. В запросе:
 - условие - по ссылке на документ;
 - группировка по **Номенклатура**,
 - суммируем **Количество** и **Себестоимость**.

Порядок действий:

3. Отметить **ПриходнаяНакладная** как регистратор для регистров **ОстаткиНоменклатуры**;
4. Сформировать движения (через конструктор), вручную заполнив в колонке "Выражение" поля **Партия** и **Себестоимость**;



- Сформировать запрос для выборки данных из документа для движений, указав:
 - условия: ссылка на документ, вид номенклатуры - Товар;
 - в качестве **Партия** - **Ссылка**;
 - в качестве **Себестоимость** - **Выборка.Сумма** (код процедуры **ОбработкаПроведения** см. ниже).
- Завести пользовательские данные в режиме 1С Предприятие, провести документ, проверить движения (добавить ссылку на соответствующий регистр в форму документа: (три верикальные точки) – Окно – Настройка панели навигации формы).

Код процедуры ОбработкаПроведения документа ПриходнаяНакладная:

```

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
| ПриходнаяНакладнаяСписокНоменклатуры.Номенклатура КАК Номенклатура,
| СУММА(ПриходнаяНакладнаяСписокНоменклатуры.Количество) КАК Количество,
| СУММА(ПриходнаяНакладнаяСписокНоменклатуры.Сумма) КАК Сумма
| ИЗ
| Документ.ПриходнаяНакладная.СписокНоменклатуры КАК ПриходнаяНакладнаяСписокНоменклат
уры
| ГДЕ
| ПриходнаяНакладнаяСписокНоменклатуры.Ссылка = &Ссылка
| СГРУППИРОВАТЬ ПО
| ПриходнаяНакладнаяСписокНоменклатуры.Номенклатура";

Запрос.УстановитьПараметр("Ссылка", Ссылка);
// отбор по виду номенклатуры Услуга в запрос не включен, т.к. производится в форме через
// "Параметр выбора" поля Номенклатура табличной части СписокНоменклатуры

РезультатЗапроса = Запрос.Выполнить();

Выборка = РезультатЗапроса.Выбрать();

Пока Выборка.Следующий() Цикл
    Движение = Движения.ОстаткиНоменклатуры.Добавить();
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Номенклатура = Выборка.Номенклатура;
    Движение.Партия = Ссылка;
    Движение.Количество = Выборка.Количество;
    Движение.Себестоимость = Выборка.Сумма;
КонецЦикла;

// регистр ОстаткиНоменклатуры Приход
Движения.ОстаткиНоменклатуры.Записывать = Истина;

КонецПроцедуры

```

Обработка проведения документа РасходнаяНакладная

Особенности:

1. При обработке проведения не забывать о проверке на "товар - услуга" (услуги в регистр накопления **ОстаткиНоменклатуры** не пишутся, и т.д.).
2. Сформировать движения (через конструктор, **табличная часть выбирается только тогда, когда из неё заполняются измерения!**) и определить методику контроля остатков:
 -
 - если для заполнения набора записей данные регистра не нужны – новая (если новая и приход – то контроль не нужен);
 - если для заполнения набора записей данные регистра нужны - старая (если используются два регистра, то **возможен вариант использования разных методик**).

Код процедуры:

Процедура ОбработкаПроведения(Отказ, Режим)

// определяем метод списания себестоимости

МетодСписания =
РегистрыСведений.МетодыСписания.ПолучитьПоследнее(Дата).МетодСписанияСебестоимости;

Если НЕ ЗначениеЗаполнено(МетодСписания) Тогда

Отказ = Истина;

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = "Не задан метод списания себестоимости (учетная политика)";

Сообщение.Сообщить();

КонецЕсли;

// очищаем движения, взводим флаги на запись

Движения.ОстаткиНоменклатуры.Записать();

Движения.ОстаткиНоменклатуры.Записывать = Истина;

Движения.Продажи.Записывать = Истина;

// устанавливаем блокировку данных

Блокировка = Новый БлокировкаДанных;

ЭлементБлокировки =

Блокировка.Добавить("РегистрНакопления.ОстаткиНоменклатуры");

ЭлементБлокировки.Режим = РежимБлокировкиДанных.Исключительный;

ЭлементБлокировки.ИсточникДанных = СписокНоменклатуры;

ЭлементБлокировки.ИспользоватьИзИсточникаДанных("Номенклатура",
"Номенклатура");

Блокировка.Заблокировать();

// формируем запрос для получения данных из документа и регистра остатков

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ


```

|      РасходнаяНакладнаяСписокНоменклатуры.Номенклатура      КАК
Номенклатура,
|      СУММА(РасходнаяНакладнаяСписокНоменклатуры.Количество)  КАК
Количество,
|      СУММА(РасходнаяНакладнаяСписокНоменклатуры.Сумма) КАК Сумма,
|
|      РасходнаяНакладнаяСписокНоменклатуры.Номенклатура.Представле
ние КАК НоменклатураПредставление,
|      РасходнаяНакладнаяСписокНоменклатуры.Ссылка.Партия      КАК
ПартияДок
| ПОМЕСТИТЬ вт_ТЧ_Товары
| ИЗ
|      Документ.РасходнаяНакладная.СписокНоменклатуры      КАК
РасходнаяНакладнаяСписокНоменклатуры
| ГДЕ
|      РасходнаяНакладнаяСписокНоменклатуры.Ссылка = &Ссылка
|
| СГРУППИРОВАТЬ ПО
|      РасходнаяНакладнаяСписокНоменклатуры.Номенклатура,
|
|      РасходнаяНакладнаяСписокНоменклатуры.Номенклатура.Представле
ние,
|      РасходнаяНакладнаяСписокНоменклатуры.Ссылка.Партия
|
| ИНДЕКСИРОВАТЬ ПО
|      Номенклатура
| ;
|
| //////////////////////////////////////
| ВЫБРАТЬ
|      вт_ТЧ_Товары.Номенклатура КАК Номенклатура,
|      вт_ТЧ_Товары.НоменклатураПредставление      КАК
НоменклатураПредставление,
|      вт_ТЧ_Товары.Количество КАК Количество,
|      вт_ТЧ_Товары.Сумма КАК Сумма,
|      ОстаткиНоменклатурыОстатки.Номенклатура.ВидНоменклатуры  КАК
ВидНоменклатуры,
|      ВЫБОР

```

```

|          КОГДА          ОстаткиНоменклатурыОстатки.Партия          =
вт_ТЧ_Товары.ПартияДок
|          ТОГДА 1
|          ИНАЧЕ 0
|          КОНЕЦ КАК ПорядокПартия,
|          ЕСТЬNULL(ОстаткиНоменклатурыОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток,
|          ЕСТЬNULL(ОстаткиНоменклатурыОстатки.СебестоимостьОстаток, 0)
КАК СебестоимостьОстаток,
|          вт_ТЧ_Товары.ПартияДок КАК ПартияДок,
|          ОстаткиНоменклатурыОстатки.Партия КАК Партия
| ИЗ
|          вт_ТЧ_Товары КАК вт_ТЧ_Товары
|          ЛЕВОЕ          СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиНоменклатуры.Остатки(
|          &МоментВремени,
|          Номенклатура В
|          (ВЫБРАТЬ
|          вт_ТЧ_Товары.Номенклатура          КАК
Номенклатура
|          ИЗ
|          вт_ТЧ_Товары КАК вт_ТЧ_Товары)) КАК
ОстаткиНоменклатурыОстатки
|          ПО          вт_ТЧ_Товары.Номенклатура          =
ОстаткиНоменклатурыОстатки.Номенклатура
|
| УПОРЯДОЧИТЬ ПО
|          ПорядокПартия УБЫВ,
|          ОстаткиНоменклатурыОстатки.Партия.МоментВремени
| ИТОГИ
|          МАКСИМУМ(Количество),
|          МАКСИМУМ(Сумма),
|          МАКСИМУМ(ВидНоменклатуры),
|          СУММА(КоличествоОстаток)
| ПО
|          Номенклатура";

```

```

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());
Запрос.УстановитьПараметр("Ссылка", Ссылка);

```

// корректируем запрос в соответствии с действующим методом списания себестоимости

Если МетодСписания = Перечисления.УчетнаяПолитика.ЛИФО Тогда

Запрос.Текст = СтрЗаменить(Запрос.Текст,
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени",
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени УБЫВ");

КонецЕсли;

РезультатЗапроса = Запрос.Выполнить();

// определяем остатки и выводим сообщение при недостатке товаров
ВыборкаНоменклатура =
РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);

Пока ВыборкаНоменклатура.Следующий() Цикл

СписаннаяСебестоимость = 0;

Если ВыборкаНоменклатура.ВидНоменклатуры =
Перечисления.ВидыНоменклатуры.Товар Тогда

Если ВыборкаНоменклатура.Количество >
ВыборкаНоменклатура.КоличествоОстаток Тогда

Сообщение = Новый СообщениеПользователю;
Сообщение.Текст = СтрШаблон("По номенклатуре %1 не хвататет товара в
количестве %2",
ВыборкаНоменклатура.Номенклатура,
ВыборкаНоменклатура.Количество -
ВыборкаНоменклатура.КоличествоОстаток);
Сообщение.Сообщить();

Отказ = Истина;

КонецЕсли;

Если Отказ Тогда
Продолжить;
КонецЕсли;

// при достаточном количестве - обходим детальные записи и списываем
товар согласно условиям
КоличествоВсегоСписать = ВыборкаНоменклатура.Количество;

Выборка = ВыборкаНоменклатура.Выбрать();

Пока Выборка.Следующий() И КоличествоВсегоСписать > 0 Цикл

КоличествоСписать = МИН(КоличествоВсегоСписать,
Выборка.КоличествоОстаток);

Если КоличествоВсегоСписать = Выборка.КоличествоОстаток Тогда
СебестоимостьСписать = Выборка.СебестоимостьОстаток;

Иначе

СебестоимостьСписать = КоличествоСписать / Выборка.КоличествоОстаток *
Выборка.СебестоимостьОстаток;

КонецЕсли;

Движение = Движения.ОстаткиНоменклатуры.Добавить();

Движение.ВидДвижения = ВидДвиженияНакопления.Расход;

Движение.Период = Дата;

Движение.Номенклатура = Выборка.Номенклатура;

Движение.Партия = Выборка.Партия;

Движение.Количество = КоличествоСписать;

Движение.Себестоимость = СебестоимостьСписать;

СписаннаяСебестоимость = СписаннаяСебестоимость +
СебестоимостьСписать;

КоличествоВсегоСписать = КоличествоВсегоСписать - КоличествоСписать;

КонецЦикла;

Иначе

СебестоимостьСписать = 0;

КонецЕсли;

// записываем данные в регистр Продажи для формирования отчета

Движение = Движения.Продажи.Добавить();

Движение.Период = Дата;

Движение.Номенклатура = ВыборкаНоменклатура.Номенклатура;

Движение.Количество = ВыборкаНоменклатура.Количество;

Движение.Себестоимость = СписаннаяСебестоимость;

Движение.СуммаПродажи = ВыборкаНоменклатура.Сумма;

КонецЦикла;

КонецПроцедуры

Порядок действий:

1. Определить текущий метод списания себестоимости.

// определяем метод списания себестоимости

МетодСписания =

РегистрыСведений.МетодыСписания.ПолучитьПоследнее(Дата).МетодСписанияСебестоимости;

Если НЕ ЗначениеЗаполнено(МетодСписания) Тогда

Отказ = Истина;

Сообщение = Новый СообщениеПользователю;

Сообщение.Текст = "Не задан метод списания себестоимости (учетная политика)";

Сообщение.Сообщить();

КонецЕсли;

2. Очистить движения регистров

// очищаем движения, взводим флаги на запись

Движения.ОстаткиНоменклатуры.Записать();

Движения.ОстаткиНоменклатуры.Записывать = Истина;

Движения.Продажи.Записывать = Истина;

3. Устанавливаем на регистры блокировку:

- копируем из Синтакс-помощника;
- пока закомментируем (т.к. данных пока нет, определим позже, по данным запроса);

// устанавливаем блокировку данных

Блокировка = Новый БлокировкаДанных;

ЭлементБлокировки

=

Блокировка.Добавить("РегистрНакопления.ОстаткиНоменклатуры");

ЭлементБлокировки.Режим = РежимБлокировкиДанных.Исключительный;

ЭлементБлокировки.ИсточникДанных = СписокНоменклатуры;

ЭлементБлокировки.ИспользоватьИзИсточникаДанных("Номенклатура",
"Номенклатура");

Блокировка.Заблокировать();

Получить из регистра данные для проведения:

- запросом с обработкой результатов получаем данные для проведения из табличной части и регистра накопления **ОстаткиНоменклатуры.Остатки**(с отбором ВидНоменклатуры <> Услуга):
 1. выбираем из табличной части **Номенклатура, Количество, Номенклатура.Представление, Сумма**;
 2. группируем по **Номенклатура** (убираем возможные дубли);
 3. суммируем по **Количество** (определяем общее количество списываемой номенклатуры для контроля остатков), **Сумма** (определяем сумму продажи);
 4. ставим условия:
 - **Ссылка** на эту РасходнуюНакладную;
 - условие на **ВидНоменклатуры** при отборе **ставить не будем**, т.к. в регистр Продажи должны писаться не только Товары, но и Услуги;
 5. помещаем во временную таблицу;
 6. индексируем по **Номенклатура** (будем далее соединять по этому полю);
 7. добавляем второй запрос пакета запросов для получения данных из регистра накопления **ОстаткиНоменклатуры.Остатки**;
 8. добавляем временную таблицу, выбираем **Номенклатура, Номенклатура.Представление**(представления нужны для вывода сообщений), **Количество, Сумма, ВидНоменклатуры**;

// корректируем запрос в соответствии с действующим методом списания себестоимости

Если МетодСписания = Перечисления.УчетнаяПолитика.ЛИФО Тогда

Запрос.Текст = СтрЗаменить(Запрос.Текст,
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени",
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени УБЫВ");

КонецЕсли;

// формируем запрос для получения данных из документа и регистра остатков

Запрос = Новый Запрос;

Запрос.Текст =

"ВЫБРАТЬ

| РасходнаяНакладнаяСписокНоменклатуры.Номенклатура КАК
Номенклатура,

| СУММА(РасходнаяНакладнаяСписокНоменклатуры.Количество) КАК
Количество,

| СУММА(РасходнаяНакладнаяСписокНоменклатуры.Сумма) КАК Сумма,
|

РасходнаяНакладнаяСписокНоменклатуры.Номенклатура.Представле
ние КАК НоменклатураПредставление,

| РасходнаяНакладнаяСписокНоменклатуры.Ссылка.Партия КАК
ПартияДок

| ПОМЕСТИТЬ вт_ТЧ_Товары

| ИЗ

| Документ.РасходнаяНакладная.СписокНоменклатуры КАК
РасходнаяНакладнаяСписокНоменклатуры

| ГДЕ

| РасходнаяНакладнаяСписокНоменклатуры.Ссылка = &Ссылка

|

| СГРУППИРОВАТЬ ПО

| РасходнаяНакладнаяСписокНоменклатуры.Номенклатура,


```

|
|      РасходнаяНакладнаяСписокНоменклатуры.Номенклатура.Представле
ние,
|      РасходнаяНакладнаяСписокНоменклатуры.Ссылка.Партия
|
|ИНДЕКСИРОВАТЬ ПО
|      Номенклатура
|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|      вт_ТЧ_Товары.Номенклатура КАК Номенклатура,
|      вт_ТЧ_Товары.НоменклатураПредставление          КАК
НоменклатураПредставление,
|      вт_ТЧ_Товары.Количество КАК Количество,
|      вт_ТЧ_Товары.Сумма КАК Сумма,
|      ОстаткиНоменклатурыОстатки.Номенклатура.ВидНоменклатуры  КАК
ВидНоменклатуры,
|      ВЫБОР
|          КОГДА          ОстаткиНоменклатурыОстатки.Партия          =
вт_ТЧ_Товары.ПартияДок
|          ТОГДА 1
|          ИНАЧЕ 0
|      КОНЕЦ КАК ПорядокПартия,
|      ЕСТЬNULL(ОстаткиНоменклатурыОстатки.КоличествоОстаток, 0) КАК
КоличествоОстаток,
|      ЕСТЬNULL(ОстаткиНоменклатурыОстатки.СебестоимостьОстаток, 0)
КАК СебестоимостьОстаток,
|      вт_ТЧ_Товары.ПартияДок КАК ПартияДок,
|      ОстаткиНоменклатурыОстатки.Партия КАК Партия
|ИЗ
|      вт_ТЧ_Товары КАК вт_ТЧ_Товары
|          ЛЕВОЕ          СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиНоменклатуры.Остатки(
|          &МоментВремени,
|          Номенклатура В
|          (ВЫБРАТЬ
|              вт_ТЧ_Товары.Номенклатура          КАК
Номенклатура

```

```

|                                     ИЗ
|                                     вт_ТЧ_Товары КАК вт_ТЧ_Товары)) КАК
ОстаткиНоменклатурыОстатки
|                                     ПО                                     вт_ТЧ_Товары.Номенклатура                                     =
ОстаткиНоменклатурыОстатки.Номенклатура
|
| УПОРЯДОЧИТЬ ПО
|     ПорядокПартия УБЫВ,
|     ОстаткиНоменклатурыОстатки.Партия.МоментВремени
| ИТОГИ
|     МАКСИМУМ(Количество),
|     МАКСИМУМ(Сумма),
|     МАКСИМУМ(ВидНоменклатуры),
|     СУММА(КоличествоОстаток)
| ПО
|     Номенклатура";

```

```

Запрос.УстановитьПараметр("МоментВремени", МоментВремени());
Запрос.УстановитьПараметр("Ссылка", Ссылка);

```

```

// корректируем запрос в соответствии с действующим методом списания
себестоимости

```

```

Если МетодСписания = Перечисления.УчетнаяПолитика.ЛИФО Тогда

```

```

Запрос.Текст                                     =                                     СтрЗаменить(Запрос.Текст,
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени",
"ОстаткиНоменклатурыОстатки.Партия.МоментВремени УБЫВ");

```

```

КонецЕсли;

```

выполняем запрос и обходим группы номенклатуры

```

РезультатЗапроса = Запрос.Выполнить();

```

```
ВыборкаНоменклатура                                     =  
РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппир  
овкам);
```

```
Пока ВыборкаНоменклатура.Следующий() Цикл
```

```
// для суммирования списанной себестоимости по номенклатуре  
для заполнения Стоимости регистра Продажи  
СписаннаяСебестоимость = 0;
```

```
// если услуга, то записываем только регистр Продажи  
Если                ВыборкаНоменклатура.Количество          >  
ВыборкаНоменклатура.КоличествоОстаток Тогда
```

```
// здесь будет код обхода ВыборкаНоменклатура (контроль  
остатков по общему количеству к списанию)
```

```
// здесь будет код обхода детальных записей (из  
ВыборкаНоменклатура)
```

```
Иначе  
// если услуга, то Себестоимость равна 0  
СебестоимостьСписать = 0;  
КонецЕсли;
```

```
// здесь будет запись движений регистра Продажи
```

```
КонецЦикла;
```

выполняем контроль остатков по сгруппированным в ИТОГАХ
номенклатурным позициям (детальные записи при нехватке обходить
смысла нет)

```
РезультатЗапроса = Запрос.Выполнить();
```

```
// определяем остатки и выводим сообщение при недостатке товаров
ВыборкаНоменклатура                                     =
РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);
```

Пока ВыборкаНоменклатура.Следующий() Цикл

СписаннаяСебестоимость = 0;

```
// если услуга, то записываем только регистр Продажи
Если              ВыборкаНоменклатура.ВидНоменклатуры      =
Перечисления.ВидыНоменклатуры.Товар Тогда
```

```
// контроль остатков по сгруппированным в ИТОГАХ номенклатурным
позициям
```

```
Если              ВыборкаНоменклатура.Количество          >
ВыборкаНоменклатура.КоличествоОстаток Тогда
```

```
Сообщение = Новый СообщениеПользователю;
Сообщение.Текст = СтрШаблон("По номенклатуре %1 не хватает %2 шт.",
ВыборкаНоменклатура.НоменклатураПредставление,
ВыборкаНоменклатура.Количество                      -
ВыборкаНоменклатура.КоличествоОстаток);
Сообщение.Сообщить();
```

Отказ = ИСТИНА;

КонецЕсли;

```
// если не хватает, то отменяем проведение (детальные записи не
обрабатываем)
```

Если Отказ Тогда

Продолжить;

КонецЕсли;

записываем движения в регистр Продажи и взводим флаги на запись:

```
// записываем данные в регистр Продажи для формирования отчета
```

Движение = Движения.Продажи.Добавить();

Движение.Период = Дата;

Движение.Номенклатура = ВыборкаНоменклатура.Номенклатура;
Движение.Количество = ВыборкаНоменклатура.Количество;
Движение.Себестоимость = СписаннаяСебестоимость;
Движение.СуммаПродажи = ВыборкаНоменклатура.Сумма;

Завести пользовательские данные в режиме 1С Предприятие, провести документ, проверить движения (добавить ссылку на соответствующий регистр в форму документа: (три вертикальные точки) – Окно – Настройка панели навигации формы).

Сделать отчет по задаче