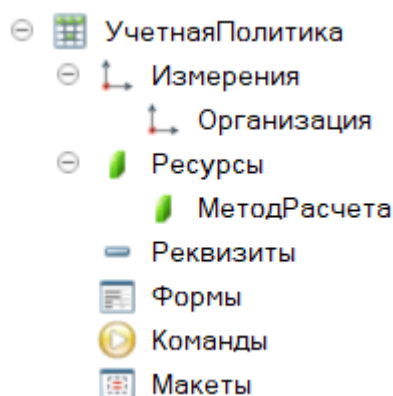
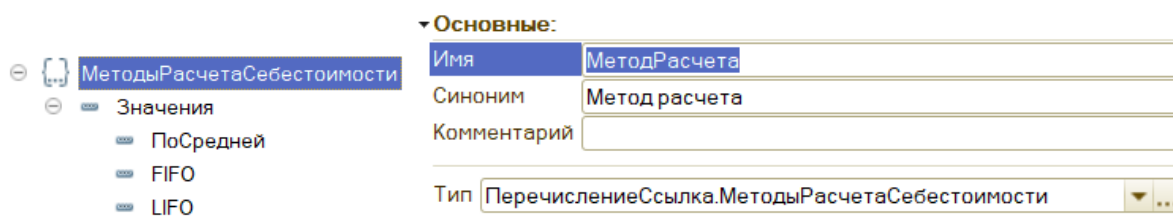


# Организация учета себестоимости



Был создан регистр сведений для смены учетной политики с периодичностью указанной в задании.



Создано перечисление и присвоено ресурсу МетодРасчета

функция ПолучитьУчетнуюПолитикуОрганизации(Дата, Организация) Экспорт

```
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
|   УчетнаяПолитикаСрезПоследних.МетодРасчета КАК МетодРасчета
|ИЗ
|   РегистрСведений.УчетнаяПолитика.СрезПоследних (&Дата, Организация = &Организация) КАК УчетнаяПолитикаСрезПоследних";

Запрос.УстановитьПараметр("Дата", Дата);
Запрос.УстановитьПараметр("Организация", Организация);

РезультатЗапроса = Запрос.Выполнить();

Если НЕ РезультатЗапроса.Пустой() Тогда
    Выборка = РезультатЗапроса.Выбрать();
    Выборка.Следующий();
    МетодРасчета = Выборка.МетодРасчета;
Иначе
    Сообщение = Новый СообщениеПользователю;
    Сообщение.Текст = "Для организации " + Организация + " не установлена учетная политика";
    Сообщение.УстановитьДанные();
    Сообщение.Сообщить();
    МетодРасчета = Неопределено;

КонецЕсли;

Возврат МетодРасчета;

КонецФункции
```

В общем серверном модуле создал запрос для получения конкретной учетной политики

Процедура ОбработкаПроведения (Отказ, Режим)

```
МетодРасчета = ОбщегоНазначенияСервер.ПолучитьУчетнуюПолитикуОрганизации (Дата, Организация);

Движения.ТоварыНаСкладах.Записывать = Истина;
Для Каждого ТекСтрокаТовары Из Товары Цикл
    Движение = Движения.ТоварыНаСкладах.Добавить ();
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Организация = Организация;
    Движение.Склад = Склад;
    Движение.Номенклатура = ТекСтрокаТовары.Номенклатура;
    Движение.Количество = ТекСтрокаТовары.Количество;
КонецЦикла;

Движения.ПартииТоваровНаСкладах.Записывать = Истина;
Для Каждого ТекСтрокаТовары Из Товары Цикл
    Движение = Движения.ПартииТоваровНаСкладах.Добавить ();
    Движение.ВидДвижения = ВидДвиженияНакопления.Приход;
    Движение.Период = Дата;
    Движение.Организация = Организация;
    Движение.Склад = Склад;
    Движение.Номенклатура = ТекСтрокаТовары.Номенклатура;

Если МетодРасчета = Перечисления.МетодыРасчетаСебестоимости.FIFO или
МетодРасчета = Перечисления.МетодыРасчетаСебестоимости.LIFO Тогда

    Движение.Партия = Ссылка;

КонецЕсли;

Движение.Количество = ТекСтрокаТовары.Количество;
Движение.Стоимость = ТекСтрокаТовары.Сумма;

КонецЦикла;

КонецПроцедуры
```

В модуле проведения добавил проверку на принадлежность политики к учету по среднему или FIFO, LIFO

Если НЕ Отказ Тогда

ОтразитьСписаниеСебестоимости ();

КонецЕсли;

- КонецПроцедуры

Теперь мы перешли в модуль документа расхода и описали, что если режим проведения соответствует контролю остатка и ОТКАЗ != Истина, опишем процедуру выбора метода учета

Процедура ОтразитьСписаниеСебестоимости()

МетодРасчета = ОбщегоНазначенияСервер.ПолучитьУчетнуюПолитикуОрганизации(Дата, Организация);

Если МетодРасчета = Перечисления.МетодыРасчетаСебестоимости.ПоСредней Тогда

```
Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
|   РасходнаяНакладнаяТовары.Номенклатура КАК Номенклатура,
|   СУММА(РасходнаяНакладнаяТовары.Количество) КАК Количество
|ПОМЕСТИТЬ ВТ_Товары
|ИЗ
|   Документ.РасходнаяНакладная.Товары КАК РасходнаяНакладнаяТовары
|ГДЕ
|   РасходнаяНакладнаяТовары.Ссылка = &Ссылка
|СГРУППИРОВАТЬ ПО
|   РасходнаяНакладнаяТовары.Номенклатура
|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|   ПартииТоваровНаСкладахОстатки.Номенклатура КАК Номенклатура,
|   ПартииТоваровНаСкладахОстатки.КоличествоОстаток КАК КоличествоОстаток,
|   ПартииТоваровНаСкладахОстатки.СтоимостьОстаток КАК СтоимостьОстаток
|ПОМЕСТИТЬ ВТ_Остатки
|ИЗ
|   РегистрНакопления.ПартииТоваровНаСкладах.Остатки(
|       &МоментВремени,
|       (Номенклатура, Склад, Организация) В
|       (ВЫБРАТЬ
|           ВТ_Товары.Номенклатура КАК Номенклатура,
|           &Склад КАК Склад,
|           &Организация КАК Организация
|       ИЗ
|           ВТ_Товары КАК ВТ_Товары)) КАК ПартииТоваровНаСкладахОстатки
|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|   ПартииТоваровНаСкладахОстатки.Номенклатура КАК Номенклатура,
|   ПартииТоваровНаСкладахОстатки.КоличествоОстаток КАК КоличествоОстаток,
|   ПартииТоваровНаСкладахОстатки.СтоимостьОстаток КАК СтоимостьОстаток
|ПОМЕСТИТЬ ВТ_Остатки
|ИЗ
|   РегистрНакопления.ПартииТоваровНаСкладах.Остатки(
|       &МоментВремени,
|       (Номенклатура, Склад, Организация) В
|       (ВЫБРАТЬ
|           ВТ_Товары.Номенклатура КАК Номенклатура,
|           &Склад КАК Склад,
|           &Организация КАК Организация
|       ИЗ
|           ВТ_Товары КАК ВТ_Товары)) КАК ПартииТоваровНаСкладахОстатки
|;
|
|////////////////////////////////////
|ВЫБРАТЬ
|   ВТ_Товары.Номенклатура КАК Номенклатура,
|   ВТ_Товары.Количество КАК КоличествоВДокументе,
|   ЕСТЬNULL(ВТ_Остатки.КоличествоОстаток, 0) КАК КоличествоОстаток,
|   ЕСТЬNULL(ВТ_Остатки.СтоимостьОстаток, 0) КАК СтоимостьОстаток
|ИЗ
|   ВТ_Товары КАК ВТ_Товары
|   ЛЕВОЕ СОЕДИНЕНИЕ ВТ_Остатки КАК ВТ_Остатки
|   ПО ВТ_Товары.Номенклатура = ВТ_Остатки.Номенклатура";

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

Блокировка = Новый БлокировкаДанных;

ЭлементБлокировки = Блокировка.Добавить("РегистрНакопления.ПартииТоваровНаСкладах");
ЭлементБлокировки.УстановитьЗначение("Организация", Организация);
ЭлементБлокировки.УстановитьЗначение("Склад", Склад);
ЭлементБлокировки.ИсточникДанных = Товары;
ЭлементБлокировки.ИспользоватьИсИсточникаДанных("Номенклатура", "Номенклатура");

Блокировка.Заблокировать();

РезультатЗапроса = Запрос.Выполнить();
```

```

Выборка = РезультатЗапроса.Выбрать ();

Движения.ПартииТоваровНаСкладах.Записывать = Истина;
Пока Выборка.Следующий () Цикл

    НоваяЗапись = Движения.ПартииТоваровНаСкладах.ДобавитьРасход ();
    НоваяЗапись.Период = Дата;
    НоваяЗапись.Организация = Организация;
    НоваяЗапись.Склад = Склад;
    НоваяЗапись.Номенклатура = Выборка.Номенклатура;
    НоваяЗапись.Количество = Выборка.КоличествоВДокументе ();

    Если Выборка.КоличествоОстаток = 0 Тогда
        СебестоимостьЕдиницы = 0;
    Иначе
        СебестоимостьЕдиницы = Выборка.СтоимостьОстаток / Выборка.КоличествоОстаток;
    КонецЕсли;

    НоваяЗапись.Стоимость = Выборка.КоличествоВДокументе * СебестоимостьЕдиницы;

КонецЦикла;

ИначеЕсли МетодРасчета = Перечисления.МетодыРасчетаСебестоимости.FIFO Тогда

    Запрос = Новый Запрос;
    Запрос.Текст =
        "ВЫБРАТЬ
        |   РасходнаяНакладнаяТовары.Номенклатура КАК Номенклатура,
        |   СУММА (РасходнаяНакладнаяТовары.Количество) КАК Количество
        |ПОМЕСТИТЬ ВТ_Товары
        |ИЗ
        |   Документ.РасходнаяНакладная.Товары КАК РасходнаяНакладнаяТовары
        |ГДЕ
        |   РасходнаяНакладнаяТовары.Ссылка = &Ссылка
        |
        |СТРУППИРОВАТЬ ПО
        |   РасходнаяНакладнаяТовары.Номенклатура
        |;
        |
        |////////////////////////////////////
        |ВЫБРАТЬ
        |   ПартииТоваровНаСкладахОстатки.Номенклатура КАК Номенклатура,
        |   ПартииТоваровНаСкладахОстатки.Партия КАК Партия,
        |   ПартииТоваровНаСкладахОстатки.КоличествоОстаток КАК КоличествоОстаток,
        |   ПартииТоваровНаСкладахОстатки.СтоимостьОстаток КАК СтоимостьОстаток
        |ПОМЕСТИТЬ ВТ_ОстаткиПоПартиям
        |ИЗ
        |   РегистрНакопления.ПартииТоваровНаСкладах.Остатки (
        |       &МоментВремени,
        |       (Номенклатура, Склад, Организация) В
        |       (ВЫБРАТЬ
        |           ВТ_Товары.Номенклатура КАК Номенклатура,
        |           &Склад КАК Склад,
        |           &Организация КАК Организация
        |       ИЗ
        |           ВТ_Товары КАК ВТ_Товары)) КАК ПартииТоваровНаСкладахОстатки
        |

```

```

|;
|
|////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
|ВЫБРАТЬ
|    VT_Товары.Номенклатура КАК Номенклатура,
|    VT_Товары.Количество КАК КоличествоВДокументе,
|    ЕСТЬNULL(VT_ОстаткиПоПартиям.Партия, ЗНАЧЕНИЕ(Документ.ПриходнаяНакладная.ПустаяСсылка)) КАК Партия,
|    ЕСТЬNULL(VT_ОстаткиПоПартиям.КоличествоОстаток, 0) КАК КоличествоОстаток,
|    ЕСТЬNULL(VT_ОстаткиПоПартиям.СтоимостьОстаток, 0) КАК СтоимостьОстаток
|ИЗ
|    VT_Товары КАК VT_Товары
|    ЛЕВОЕ СОЕДИНЕНИЕ VT_ОстаткиПоПартиям КАК VT_ОстаткиПоПартиям
|        ПО VT_Товары.Номенклатура = VT_ОстаткиПоПартиям.Номенклатура
|
|УПОРЯДОЧИТЬ ПО
|    VT_ОстаткиПоПартиям.Партия.МоментВремени
|ИТОГИ
|    МАКСИМУМ(КоличествоВДокументе),
|    СУММА(КоличествоОстаток),
|    СУММА(СтоимостьОстаток)
|ПО
|    Номенклатура";

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

Блокировка = Новый БлокировкаДанных;

ЭлементБлокировки = Блокировка.Добавить("РегистрНакопления.ПартииТоваровНаСкладах");
ЭлементБлокировки.УстановитьЗначение("Организация", Организация);
ЭлементБлокировки.УстановитьЗначение("Склад", Склад);
ЭлементБлокировки.ИсточникДанных = Товары;
ЭлементБлокировки.ИспользоватьИсточникаДанных("Номенклатура", "Номенклатура");

Блокировка.Заблокировать();

РезультатЗапроса = Запрос.Выполнить();

ВыборкаНоменклатура = РезультатЗапроса.Выбрать(ОбходРезультатаЗапроса.ПоГруппировкам);

Движения.ПартииТоваровНаСкладах.Записывать = Истина;
Пока ВыборкаНоменклатура.Следующий() Цикл

    ОсталосьСписать = ВыборкаНоменклатура.КоличествоВДокументе;

    ВыборкаПартии = ВыборкаНоменклатура.Выбрать();

    Пока ВыборкаПартии.Следующий() и ОсталосьСписать <> 0 Цикл

        Если ОсталосьСписать < ВыборкаПартии.КоличествоОстаток Тогда
            КоличествоСписания = ОсталосьСписать;
        Иначе
            КоличествоСписания = ВыборкаПартии.КоличествоОстаток;
        КонецЕсли;

        Если КоличествоСписания = ВыборкаПартии.КоличествоОстаток Тогда
            Себестоимость = ВыборкаПартии.СтоимостьОстаток;
        Иначе
            Себестоимость = КоличествоСписания / ВыборкаПартии.КоличествоОстаток *
                ВыборкаПартии.СтоимостьОстаток;
        КонецЕсли;

        НоваяЗапись = Движения.ПартииТоваровНаСкладах.ДобавитьРасход();
        НоваяЗапись.Период = Дата;
        НоваяЗапись.Организация = Организация;
        НоваяЗапись.Склад = Склад;
        НоваяЗапись.Номенклатура = ВыборкаПартии.Номенклатура;
        НоваяЗапись.Партия = ВыборкаПартии.Партия;
        НоваяЗапись.Количество = КоличествоСписания;
        НоваяЗапись.Стоимость = Себестоимость;

        ОсталосьСписать = ОсталосьСписать - КоличествоСписания;

    КонецЦикла;

КонецЦикла;

ИначеЕсли МетодРасчета = Перечисления.МетодыРасчетаСебестоимости.LIFO Тогда

```

```

Запрос = Новый Запрос;
Запрос.Текст =
"ВЫБРАТЬ
|     РасходнаяНакладнаяТовары.Номенклатура КАК Номенклатура,
|     СУММА (РасходнаяНакладнаяТовары.Количество) КАК Количество
| ПОМЕСТИТЬ ВТ_Товары
| ИЗ
|     Документ.РасходнаяНакладная.Товары КАК РасходнаяНакладнаяТовары
| ГДЕ
|     РасходнаяНакладнаяТовары.Ссылка = &Ссылка
|
| СГРУППИРОВАТЬ ПО
|     РасходнаяНакладнаяТовары.Номенклатура
| ;
|
| //////////////////////////////////////
| ВЫБРАТЬ
|     ПартииТоваровНаСкладахОстатки.Номенклатура КАК Номенклатура,
|     ПартииТоваровНаСкладахОстатки.Партия КАК Партия,
|     ПартииТоваровНаСкладахОстатки.КоличествоОстаток КАК КоличествоОстаток,
|     ПартииТоваровНаСкладахОстатки.СтоимостьОстаток КАК СтоимостьОстаток
| ПОМЕСТИТЬ ВТ_ОстаткиПоПартиям
| ИЗ
|     РегистрНакопления.ПартииТоваровНаСкладах.Остатки (
|         &МоментВремени,
|         (Номенклатура, Склад, Организация) В
|         (ВЫБРАТЬ
|             ВТ_Товары.Номенклатура КАК Номенклатура,
|             &Склад КАК Склад,
|             &Организация КАК Организация
|         ИЗ
|             ВТ_Товары КАК ВТ_Товары)) КАК ПартииТоваровНаСкладахОстатки
| ;
|
| //////////////////////////////////////
|
|     ВТ_Товары.Номенклатура КАК Номенклатура,
|     ВТ_Товары.Количество КАК КоличествоВДокументе,
|     ЕСТЬNULL (ВТ_ОстаткиПоПартиям.Партия, ЗНАЧЕНИЕ (Документ.ПриходнаяНакладная.ПустаяСсылка)) КАК Партия,
|     ЕСТЬNULL (ВТ_ОстаткиПоПартиям.КоличествоОстаток, 0) КАК КоличествоОстаток,
|     ЕСТЬNULL (ВТ_ОстаткиПоПартиям.СтоимостьОстаток, 0) КАК СтоимостьОстаток
| ИЗ
|     ВТ_Товары КАК ВТ_Товары
|     ЛЕВОЕ СОЕДИНЕНИЕ ВТ_ОстаткиПоПартиям КАК ВТ_ОстаткиПоПартиям
|     ПО ВТ_Товары.Номенклатура = ВТ_ОстаткиПоПартиям.Номенклатура
|
| УПОРЯДОЧИТЬ ПО
|     ВТ_ОстаткиПоПартиям.Партия.МоментВремени УБЫВ
| ИТОГИ
|     МАКСИМУМ (КоличествоВДокументе),
|     СУММА (КоличествоОстаток),
|     СУММА (СтоимостьОстаток)
| ПО
|     Номенклатура";

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

Блокировка = Новый БлокировкаДанных;

ЭлементБлокировки = Блокировка.Добавить ("РегистрНакопления.ПартииТоваровНаСкладах");
ЭлементБлокировки.УстановитьЗначение ("Организация", Организация);
ЭлементБлокировки.УстановитьЗначение ("Склад", Склад);
ЭлементБлокировки.ИсточникДанных = Товары;
ЭлементБлокировки.ИспользоватьИсходникаДанных ("Номенклатура", "Номенклатура");

Блокировка.Заблокировать ();

РезультатЗапроса = Запрос.Выполнить ();

ВыборкаНоменклатура = РезультатЗапроса.Выбрать (ОбходРезультатаЗапроса.ПоГруппировкам);

Движения.ПартииТоваровНаСкладах.Записывать = Истина;
Пока ВыборкаНоменклатура.Следующий () Цикл

    ОсталосьСписать = ВыборкаНоменклатура.КоличествоВДокументе;

    ВыборкаПартии = ВыборкаНоменклатура.Выбрать ();

```

```

Пока ВыборкаПартии.Следующий() и ОсталосьСписать <> 0 Цикл

    Если ОсталосьСписать < ВыборкаПартии.КоличествоОстаток Тогда
        КоличествоСписания = ОсталосьСписать;
    Иначе
        КоличествоСписания = ВыборкаПартии.КоличествоОстаток;
    КонецЕсли;

    Если КоличествоСписания = ВыборкаПартии.КоличествоОстаток Тогда
        Себестоимость = ВыборкаПартии.СтоимостьОстаток;
    Иначе
        Себестоимость = КоличествоСписания / ВыборкаПартии.КоличествоОстаток *
            ВыборкаПартии.СтоимостьОстаток;
    КонецЕсли;

    НоваяЗапись = Движения.ПартииТоваровНаСкладах.ДобавитьРасход();
    НоваяЗапись.Период = Дата;
    НоваяЗапись.Организация = Организация;
    НоваяЗапись.Склад = Склад;
    НоваяЗапись.Номенклатура = ВыборкаПартии.Номенклатура;
    НоваяЗапись.Партия = ВыборкаПартии.Партия;
    НоваяЗапись.Количество = КоличествоСписания;
    НоваяЗапись.Стоимость = Себестоимость;

    ОсталосьСписать = ОсталосьСписать - КоличествоСписания;

КонецЦикла;

КонецЦикла;

КонецЕсли;

КонецПроцедуры

```