

Лабораторная работа №6
**«СТРАТЕГИИ СПИСАНИЯ ВЗАИМОРАСЧЕТОВ (ПО СРЕДНЕЙ,
ФИФО, ЛИФО). УПРАВЛЯЕМЫЕ БЛОКИРОВКИ»**

Цель работы

Изучить и научиться: порядок списания взаиморасчетов, принцип управляемых блокировок.

Используемое программное обеспечение: учебная версия платформы «1С: Предприятие 8.3».

Режим совместимости: «Версия 8.3.9».

Вариант 22

Предметная область: Пассажирские перевозки.

Описание деятельности: Трамвайное депо "Конькобежец" бегом принимает заявки на проведение выпускных в трамвае и ждет оплаты выпускных. В любой момент может измениться цена за выпускной и тип выпускного. Заккрытие оплаты выпускного осуществляется в соответствии с учетной политикой (стратегии ФИФО, ЛИФО). Депо предоставляется отчетность.

Необходимо сделать:

1. Написать блокировку записей регистра ОстаткиЗаявок перед списанием остатков. Желательно (точнее, обязательно) минимизировать количество блокируемых записей.
2. Переписать обработчик проведения документа Оплата по регистру ОстаткиЗаявок, используя стратегию учета ПоСредней.

3. Переписать обработчик проведения документа Оплата по регистру ОстаткиЗаявок, используя стратегию учета ФИФО.

4. Переписать обработчик проведения документа Оплата по регистру ОстаткиЗаявок, используя стратегию учета ЛИФО.

Ход работы

1. Был написан код в обработчике проведения документа Оплата, блокирующий записи регистра ОстаткиЗаявок перед списанием остатков.

```
// регистр ОстаткиЗаявок Расход
Движения.ОстаткиЗаявок.Записывать = Истина;

// Блокировка Группы для списания

ЗапросДляБлокировки = Новый Запрос();
ЗапросДляБлокировки.Текст = "ВЫБРАТЬ
                                |      ОплатаГруппы.Группа      КАК
Группа
                                |
                                | ИЗ
                                |      Документ.Оплата.Группы      КАК
ОплатаГруппы
                                |
                                | ГДЕ
                                |      ОплатаГруппы.Ссылка      =
&Ссылка
                                |
                                | СГРУППИРОВАТЬ ПО
                                |      ОплатаГруппы.Группа";
ЗапросДляБлокировки.УстановитьПараметр("Ссылка", Ссылка);

Блокировка = Новый БлокировкаДанных();
ЭлементБлокировки
Блокировка.Добавить("РегистрНакопления.ОстаткиЗаявок");
ЭлементБлокировки.ИсточникДанных
ЗапросДляБлокировки.Выполнить();
```

```

ЭлементБлокировки.ИспользоватьИзИсточникаДанных ("Группа",
"Группа");
ЭлементБлокировки.Режим
РежимБлокировкиДанных.Исключительный;
Блокировка.Заблокировать();

```

2. Был переписан обработчик проведения документа Оплата по регистру ОстаткиЗаявок, используя стратегию учета ПоСредней.

```

СтратегияСписания
РегистрыСведений.УчетнаяПолитика.ПолучитьПоследнее(Дата).СтратегияУчета;

```

```

Если СтратегияСписания = "ПоСредней" ИЛИ
СтратегияСписания
Перечисления.СтратегияУчета.ПустаяСсылка() Тогда
Запрос = Новый Запрос();
Запрос.Текст = "ВЫБРАТЬ
| ОплатаГруппы.Группа КАК Группа,
| СУММА(ОплатаГруппы.Количество) КАК
Количество
| ПОМЕСТИТЬ ВТГруппы
| ИЗ
| Документ.Оплата.Группы КАК ОплатаГруппы,
| Документ.Оплата КАК ДокументОплаты
| ГДЕ
| ОплатаГруппы.Ссылка = &Ссылка
|
| СГРУППИРОВАТЬ ПО
| ОплатаГруппы.Группа
| ;

```

```

| //////////////////////////////////////
////////////////////////////////////
|
| ВЫБРАТЬ
| ВТГруппы.Группа КАК Группа,

```

```

|
СУММА (ЕстьNULL (ОстаткиЗаявокОстатки.КоличествоОстаток, 0)) КАК
КоличествоОстаток,
| МАКСИМУМ (ВТГруппы.Количество) КАК
Количество,
|
СУММА (ЕстьNULL (ОстаткиЗаявокОстатки.СуммаОстаток, 0)) КАК
СуммаОстаток
| ИЗ
| ВТГруппы КАК ВТГруппы
| ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиЗаявок.Остатки (
| &Период,
| Группа В (
| ВЫБРАТЬ
| ВТГруппы.Группа
КАК Группа
| ИЗ
| ВТГруппы КАК
ВТГруппы) ) КАК ОстаткиЗаявокОстатки
| ПО ВТГруппы.Группа =
ОстаткиЗаявокОстатки.Группа
|
| СГРУППИРОВАТЬ ПО
| ВТГруппы.Группа
| ";

Запрос.УстановитьПараметр ("Период", Новый
Граница (МоментВремени(), ВидГраницы.Исключая));
Запрос.УстановитьПараметр ("Ссылка", Ссылка);

Выборка = Запрос.Выполнить().Выбрать();
Пока Выборка.Следующий() Цикл
Если Выборка.Количество >
Выборка.КоличествоОстаток Тогда
Сообщить ("Группа: <" + Выборка.Группа + ">".
Нехватка: " +
(Выборка.Количество -
Выборка.КоличествоОстаток));

```

```

        Отказ = Истина;
        Продолжить;
    КонечЕсли;

    Если Выборка.Количество =
Выборка.КоличествоОстаток Тогда
        //Списание "последней копейки"
        СуммаСписания = Выборка.СуммаОстаток;
    Иначе
        СуммаСписания = Выборка.Количество *
(Выборка.СуммаОстаток / Выборка.КоличествоОстаток);
    КонечЕсли;

    Движение = Движения.ОстаткиЗаявок.Добавить();
    Движение.ВидДвижения =
ВидДвиженияНакопления.Расход;
    Движение.Период = Дата;
    Движение.Группа = Выборка.Группа;
    Движение.ДокументВзаиморасчетов =
Документы.Оплата.ПустаяСсылка();
    Движение.Количество = Выборка.Количество();
    Движение.Сумма = СуммаСписания;

    КонечЦикла;

    Если Отказ Тогда
        Возврат;
    КонечЕсли;

```

3. Был переписан обработчик проведения документа Оплата по регистру ОстаткиЗаявок, используя стратегию учета ФИФО.

```

ИначеЕсли СтратегияСписания = "ФИФО" ИЛИ
СтратегияСписания = "ЛИФО" Тогда

    УсловиеСортировки = "";

    Если СтратегияСписания = "ЛИФО" Тогда

```

```

        УсловиеСортировки = " УБЫВ";
КонецЕсли;

Запрос = Новый Запрос();
Запрос.Текст = "ВЫБРАТЬ
                | ОплатаГруппы.Группа КАК Группа,
                | СУММА(ОплатаГруппы.Количество) КАК
Количество,
                | СУММА(ОплатаГруппы.Сумма) КАК Сумма
                | ПОМЕСТИТЬ ВТГруппы
                | ИЗ
                | Документ.Оплата.Группы КАК ОплатаГруппы,
                | Документ.Оплата КАК ДокументОплаты
                | ГДЕ
                | ОплатаГруппы.Ссылка = &Ссылка
                |
                | СГРУППИРОВАТЬ ПО
                | ОплатаГруппы.Группа
                | ;

| //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
| //////////////////////////////////////////////////////////////////

                | ВЫБРАТЬ
                | ОстаткиЗаявокОстатки.Группа КАК
Группа,
                |
                | ОстаткиЗаявокОстатки.ДокументВзаиморасчетов КАК
ДокументВзаиморасчетов,
                | ОстаткиЗаявокОстатки.КоличествоОстаток
КАК КоличествоОстаток,
                | ОстаткиЗаявокОстатки.СуммаОстаток КАК
СуммаОстаток,
                | ВТГруппы.Количество КАК Количество,
                | ВТГруппы.Сумма КАК Сумма
                | ИЗ
                | ВТГруппы КАК ВТГруппы
                | ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиЗаявок.Остатки(
                | &Период,

```

```

        |                                     Группа В (
        |                                     ВЫБРАТЬ
        |                                     ВТГруппы.Группа
КАК Группа
        |                                     ИЗ
        |                                     ВТГруппы          КАК
ВТГруппы) ) КАК ОстаткиЗаявокОстатки
        |             ПО             ВТГруппы.Группа           =
ОстаткиЗаявокОстатки.Группа
        |
        | УПОРЯДОЧИТЬ ПО
        |
        | ОстаткиЗаявокОстатки.ДокументВзаиморасчетов.МоментВремени      "    +
УсловиеСортировки + "
        | ИТОГИ
        | СУММА (КоличествоОстаток) ,
        | СУММА (СуммаОстаток) ,
        | МАКСИМУМ (Количество) ,
        | МАКСИМУМ (Сумма)
        | ПО
        | Группа
        | ";
Запрос.УстановитьПараметр ("Период", Новый
Граница (МоментВремени () , ВидГраницы.Исключая) );
Запрос.УстановитьПараметр ("Ссылка", Ссылка);
ВыборкаИтоги =
Запрос.Выполнить ().Выбрать (ОбходРезультатаЗапроса.ПоГруппировкам) ;
Пока ВыборкаИтоги.Следующий() Цикл
    Если ВыборкаИтоги.Количество >
ВыборкаИтоги.КоличествоОстаток Тогда
        Сообщить ("Группа: <" + ВыборкаИтоги.Группа +
">. Нехватка: " +
(ВыборкаИтоги.Количество -
ВыборкаИтоги.КоличествоОстаток) );
Отказ = Истина;
Продолжить;
КонечЕсли;

```

```

КоличествоСписания = ВыборкаИтоги.Количество;
ВыборкаДетали = ВыборкаИтоги.Выбрать();
Пока ВыборкаДетали.Следующий() Цикл
    Если КоличествоСписания = 0 Тогда
        Прервать;
    КонецЕсли;

    Количество = Мин(КоличествоСписания,
ВыборкаДетали.КоличествоОстаток);

    Если Количество =
ВыборкаДетали.КоличествоОстаток Тогда
        //Списание "последней копейки"
        СуммаСписания =
ВыборкаДетали.СуммаОстаток;
    Иначе
        СуммаСписания = Количество *
(ВыборкаДетали.СуммаОстаток / ВыборкаДетали.КоличествоОстаток);
    КонецЕсли;

    Движение =
Движения.ОстаткиЗаявок.Добавить();

    Движение.ВидДвижения =
ВидДвиженияНакопления.Расход;

    Движение.Период = Дата;
    Движение.Группа = ВыборкаДетали.Группа;
    Движение.ДокументВзаиморасчетов =
ВыборкаДетали.ДокументВзаиморасчетов;

    Движение.Количество = Количество;
    Движение.Сумма = СуммаСписания;

    КоличествоСписания = КоличествоСписания -
Количество;

    КонецЦикла;
КонецЦикла;

КонецЕсли;

```


4. Полный код обработчика проведения документа Оплата.

Процедура ОбработкаПроведения(Отказ, Режим)

```
//{{__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
// Данный фрагмент построен конструктором.
// При повторном использовании конструктора, внесенные
вручную изменения будут утеряны!!!

// регистр ОстаткиЗаявок Расход
Движения.ОстаткиЗаявок.Записывать = Истина;

// Блокировка Группы для списания

ЗапросДляБлокировки = Новый Запрос();
ЗапросДляБлокировки.Текст = "ВЫБРАТЬ
                                |      ОплатаГруппы.Группа      КАК
Группа
                                |ИЗ
                                |      Документ.Оплата.Группы      КАК
ОплатаГруппы
                                |ГДЕ
                                |      ОплатаГруппы.Ссылка      =
&Ссылка
                                |
                                |СГРУППИРОВАТЬ ПО
                                |      ОплатаГруппы.Группа";
ЗапросДляБлокировки.УстановитьПараметр("Ссылка", Ссылка);

Блокировка = Новый БлокировкаДанных();
ЭлементБлокировки
Блокировка.Добавить("РегистрНакопления.ОстаткиЗаявок");
ЭлементБлокировки.ИсточникДанных
ЗапросДляБлокировки.Выполнить();
ЭлементБлокировки.ИспользоватьИзИсточникаДанных("Группа",
"Группа");
ЭлементБлокировки.Режим
РежимБлокировкиДанных.Исключительный;
Блокировка.Заблокировать();
```

```

СтратегияСписания
РегистрыСведений.УчетнаяПолитика.ПолучитьПоследнее (Дата) .СтратегияУчет
а;

```

```

Если СтратегияСписания = "ПоСредней" ИЛИ
СтратегияСписания
Перечисления.СтратегияУчета.ПустаяСсылка () Тогда
Запрос = Новый Запрос ();
Запрос.Текст = "ВЫБРАТЬ
| ОплатаГруппы.Группа КАК Группа,
| СУММА (ОплатаГруппы.Количество) КАК
Количество
| ПОМЕСТИТЬ ВТГруппы
| ИЗ
| Документ.Оплата.Группы КАК ОплатаГруппы,
| Документ.Оплата КАК ДокументОплаты
| ГДЕ
| ОплатаГруппы.Ссылка = &Ссылка
|
| СГРУППИРОВАТЬ ПО
| ОплатаГруппы.Группа
| ;

| //////////////////////////////////////
| //////////////////////////////////////

| ВЫБРАТЬ
| ВТГруппы.Группа КАК Группа,
|
| СУММА (ЕстьNULL (ОстаткиЗаявокОстатки.КоличествоОстаток, 0)) КАК
КоличествоОстаток,
| МАКСИМУМ (ВТГруппы.Количество) КАК
Количество,
|
| СУММА (ЕстьNULL (ОстаткиЗаявокОстатки.СуммаОстаток, 0)) КАК
СуммаОстаток
| ИЗ
| ВТГруппы КАК ВТГруппы
| ЛЕВОЕ СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиЗаявок.Остатки (

```



```

        Движение = Движения.ОстаткиЗаявок.Добавить ();
        Движение.ВидДвижения
ВидДвиженияНакопления.Расход;
        Движение.Период = Дата;
        Движение.Группа = Выборка.Группа;
        Движение.ДокументВзаиморасчетов
Документы.Оплата.ПустаяСсылка ();
        Движение.Количество = Выборка.Количество ();
        Движение.Сумма = СуммаСписания;

    КонечЦикла;

    Если Отказ Тогда
        Возврат;
    КонечЕсли;

    ИначеЕсли СтратегияСписания = "ФИФО" ИЛИ
        СтратегияСписания = "ЛИФО" Тогда

        УсловиеСортировки = "";

        Если СтратегияСписания = "ЛИФО" Тогда
            УсловиеСортировки = " УБЫВ";
        КонечЕсли;

        Запрос = Новый Запрос ();
        Запрос.Текст = "ВЫБРАТЬ
            | ОплатаГруппы.Группа КАК Группа,
            | СУММА (ОплатаГруппы.Количество) КАК
Количество,
            | СУММА (ОплатаГруппы.Сумма) КАК Сумма
            | ПОМЕСТИТЬ ВТГруппы
            | ИЗ
            | Документ.Оплата.Группы КАК ОплатаГруппы,
            | Документ.Оплата КАК ДокументОплаты
            | ГДЕ
            | ОплатаГруппы.Ссылка = &Ссылка
            |

```

```

| СГРУППИРОВАТЬ ПО
|   ОплатаГруппы.Группа
| ;

|  //////////////////////////////////////
|  //////////////////////////////////////

| ВЫБРАТЬ
|   ОстаткиЗаявокОстатки.Группа              КАК
Группа,

|   ОстаткиЗаявокОстатки.ДокументВзаиморасчетов              КАК
ДокументВзаиморасчетов,

|   ОстаткиЗаявокОстатки.КоличествоОстаток
КАК  КоличествоОстаток,

|   ОстаткиЗаявокОстатки.СуммаОстаток        КАК
СуммаОстаток,

|   ВТГруппы.Количество КАК  Количество,
|   ВТГруппы.Сумма КАК  Сумма
| ИЗ
|   ВТГруппы КАК  ВТГруппы
|           ЛЕВОЕ                               СОЕДИНЕНИЕ
РегистрНакопления.ОстаткиЗаявок.Остатки (
|           &Период,
|           Группа В (
|                   ВЫБРАТЬ
|                   ВТГруппы.Группа
КАК  Группа
|                   ИЗ
|                   ВТГруппы              КАК
ВТГруппы) ) КАК  ОстаткиЗаявокОстатки
|           ПО           ВТГруппы.Группа              =
ОстаткиЗаявокОстатки.Группа
|
| УПОРЯДОЧИТЬ ПО
|
|   ОстаткиЗаявокОстатки.ДокументВзаиморасчетов.МоментВремени  "  +
УсловиеСортировки + "

| ИТОГИ
|   СУММА (КоличествоОстаток) ,

```

```

| СУММА (СуммаОстаток) ,
| МАКСИМУМ (Количество) ,
| МАКСИМУМ (Сумма)
| ПО
| Группа
| ";

```

```

Запрос.УстановитьПараметр ("Период", Новый
Граница (МоментВремени () , ВидГраницы.Исключая) ) ;
Запрос.УстановитьПараметр ("Ссылка", Ссылка) ;

```

```

ВыборкаИтоги =
Запрос.Выполнить () .Выбрать (ОбходРезультатаЗапроса.ПоГруппировкам) ;
Пока ВыборкаИтоги.Следующий () Цикл
    Если ВыборкаИтоги.Количество >
ВыборкаИтоги.КоличествоОстаток Тогда
        Сообщить ("Группа: <" + ВыборкаИтоги.Группа +
">. Нехватка: " +
(ВыборкаИтоги.Количество -
ВыборкаИтоги.КоличествоОстаток) ) ;
        Отказ = Истина;
        Продолжить;
    КонецЕсли;

```

```

КоличествоСписания = ВыборкаИтоги.Количество;
ВыборкаДетали = ВыборкаИтоги.Выбрать () ;
Пока ВыборкаДетали.Следующий () Цикл
    Если КоличествоСписания = 0 Тогда
        Прервать;
    КонецЕсли;

    Количество = Мин (КоличествоСписания,
ВыборкаДетали.КоличествоОстаток) ;

    Если Количество =
ВыборкаДетали.КоличествоОстаток Тогда
        //Списание "последней копейки"
        СуммаСписания =
ВыборкаДетали.СуммаОстаток;
    Иначе

```

```

                СуммаСписания      =      Количество      *
(ВыборкаДетали.СуммаОстаток / ВыборкаДетали.КоличествоОстаток);
                КонецЕсли;

                Движение      =
Движения.ОстаткиЗаявок.Добавить ();

                Движение.ВидДвижения      =
ВидДвиженияНакопления.Расход;

                Движение.Период = Дата;
                Движение.Группа = ВыборкаДетали.Группа;
                Движение.ДокументВзаиморасчетов      =
ВыборкаДетали.ДокументВзаиморасчетов;

                Движение.Количество = Количество;
                Движение.Сумма = СуммаСписания;

                КоличествоСписания = КоличествоСписания -
Количество;

                КонецЦикла;

                КонецЦикла;

                КонецЕсли;

//}}__КОНСТРУКТОР_ДВИЖЕНИЙ_РЕГИСТРОВ
КонецПроцедуры

```

Выводы

В ходе выполнения лабораторной работы были изучены : порядок списания взаиморасчетов, принцип управляемых блокировок.

Используемое программное обеспечение: учебная версия платформы «1С: Предприятие 8.3».

Режим совместимости: «Версия 8.3.9».