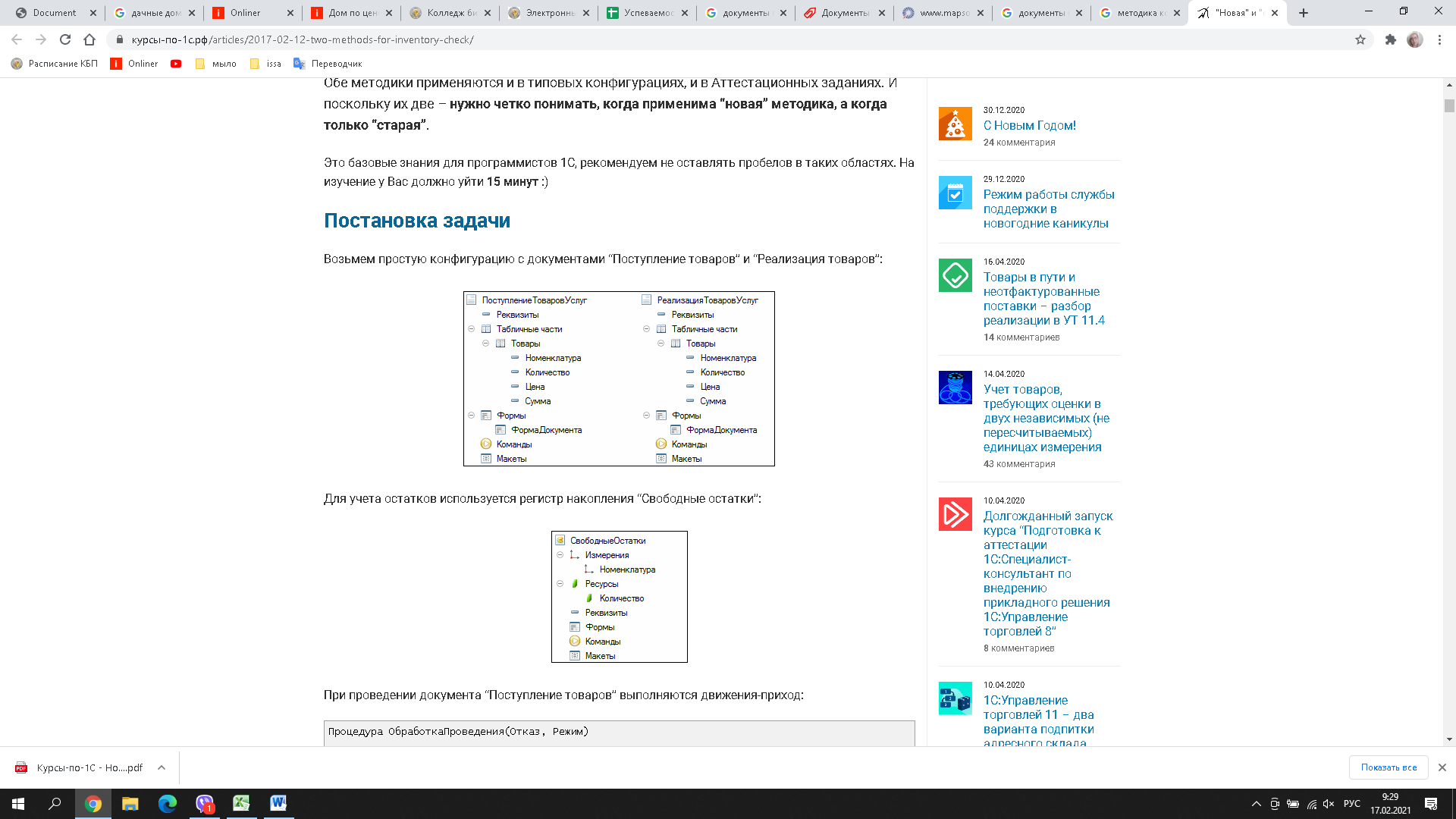
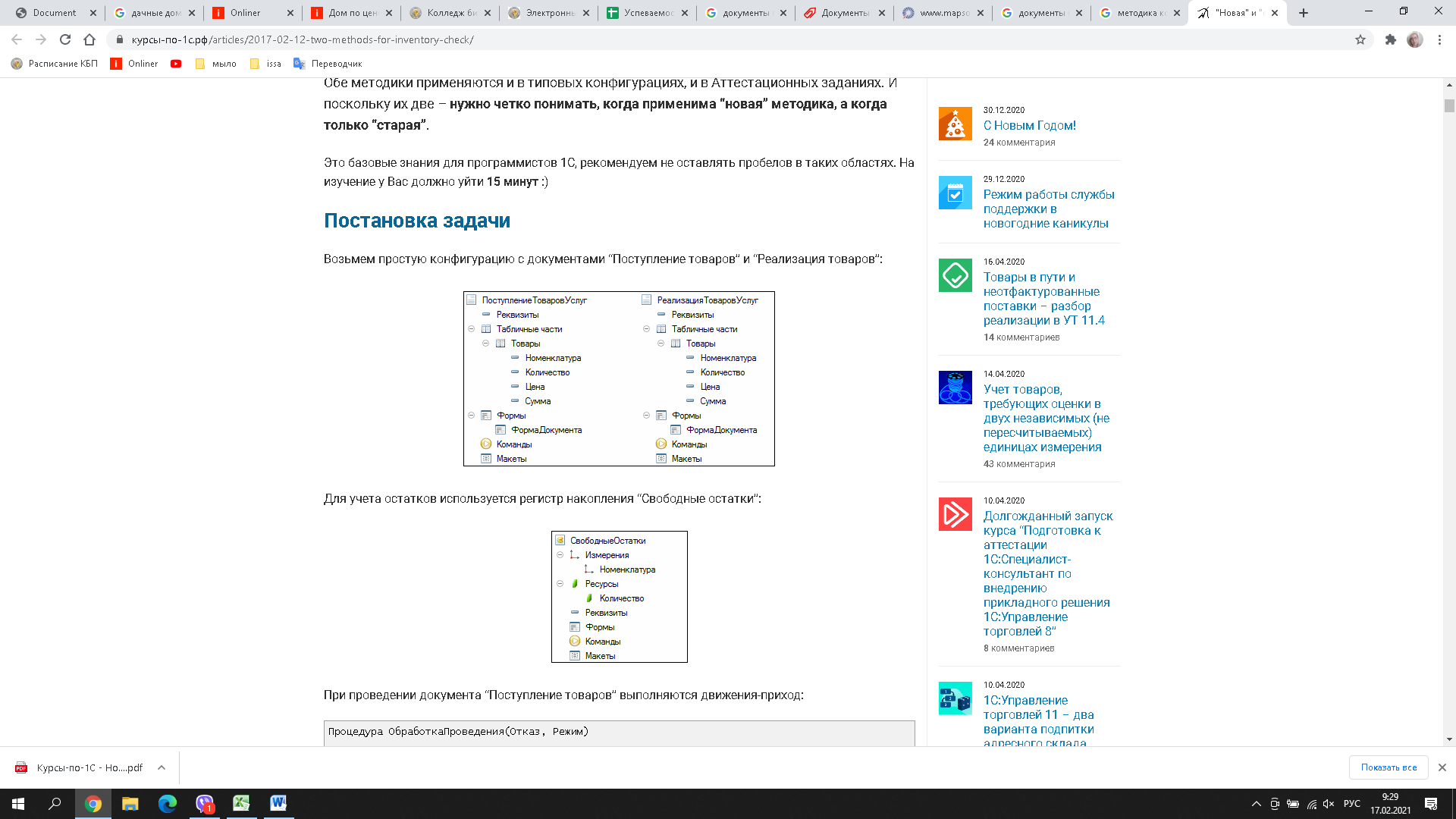
**Методика контроля отрицательных остатков при проведении документов в системе 1С:Предприятие 8.3**

Постановка задачи

Возьмем простую конфигурацию с документами “Поступление товаров” и “Реализация товаров”:



Для учета остатков используется регистр накопления “Свободные остатки”:



При проведении документа “Поступление товаров” выполняются движения-приход:

Процедура ОбработкаПроведения(Отказ, Режим)

    Движения.СвободныеОстатки.Записывать = Истина;

    Для Каждого ТекСтрокаТовары Из Товары Цикл

        Движение = Движения.СвободныеОстатки.Добавить();

        Движение.ВидДвижения = ВидДвиженияНакопления.Приход;

        Движение.Период = Дата;

        Движение.Номенклатура = ТекСтрокаТовары.Номенклатура;

        Движение.Количество = ТекСтрокаТовары.Количество;

    КонецЦикла;

КонецПроцедуры

Обработка проведения документа «Поступление товаров» выполнена с помощью конструктора движений и интереса не представляет, так как при поступлении на склад контроль остатков не нужен.

Иногда контроль остатков реализуют и для документа «Поступление товаров» – чтобы при отмене проведения или перепроведении документа не образовался отрицательный остаток.

Например, на склад поступили 10 новых телевизоров LG, 6 из них было продано. Если в документе поступления 10 шт. исправить на 5 шт. – образуется отрицательный остаток «минус 1 шт.».

В типовой УТ 11 подобный контроль включается с помощью функциональной опции «Контролировать товары организаций при отмене приходов».

При проведении документа «Реализация товаров» необходимо организовать контроль остатков. Если товара на остатках недостаточно, документ не проводится и выдается диагностическое сообщение. В этом и состоит решаемая задача.

Мы намеренно работаем над простой задачей, когда себестоимость при списании не рассчитывается. Это позволит нам сосредоточиться именно на нюансах контроля остатков.

Решение задачи может быть выполнено двумя способами. Начнем с методики, которая применялась ещё со времен «1С:Предприятие 8.0».

**Методика контроля остатков**

В новой методике используется принцип: списываем необходимые товары, далее проверяем – образовались ли отрицательные остатки по товарам документа. Если да, то нужно откатить проведение документа.

Как видите, принципиальная разница в моменте контроля остатков:

Старая методика – сначала проверяем остаток, потом списываем

Новая методика – сначала списываем, потом проверяем остаток.

В результате программный код будет выглядеть следующим образом:

Процедура ОбработкаПроведения(Отказ, РежимПроведения)

    //  1. Получение запросом данных документа

    Запрос = Новый Запрос;

    Запрос.МенеджерВременныхТаблиц = Новый МенеджерВременныхТаблиц;

    Запрос.Текст =

        "ВЫБРАТЬ

        |   Товары.Номенклатура КАК Номенклатура,

        |   СУММА(Товары.Количество) КАК Количество

        |ПОМЕСТИТЬ Товары

        |ИЗ

        |   Документ.РеализацияТоваровУслуг.Товары КАК Товары

        |ГДЕ

        |   Товары.Ссылка = &Ссылка

        |

        |СГРУППИРОВАТЬ ПО

        |   Товары.Номенклатура

        |

        |ИНДЕКСИРОВАТЬ ПО

        |   Номенклатура

        |;

        |

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

        |ВЫБРАТЬ

        |   Товары.Номенклатура КАК Номенклатура,

        |   Товары.Количество КАК Количество

        |ИЗ

        |   Товары КАК Товары";

    Запрос.УстановитьПараметр("Ссылка", Ссылка);

    РезультатЗапроса = Запрос.Выполнить();

    //  2. Формирование движений-расход регистра

    Движения.СвободныеОстатки.Очистить();

    ВыборкаТовары = РезультатЗапроса.Выбрать();

    Пока ВыборкаТовары.Следующий() Цикл

        Движение = Движения.СвободныеОстатки.ДобавитьРасход();

        Движение.Период = Дата;

        Движение.Номенклатура = ВыборкаТовары.Номенклатура;

        Движение.Количество = ВыборкаТовары.Количество;

    КонецЦикла;

    //  3. Запись движений в БД

    Движения.СвободныеОстатки.Записывать = Истина;

    Движения.Записать();

    //  4. Запрос, получающий отрицательные остатки из регистра

    Запрос.Текст =

        "ВЫБРАТЬ

        |   Остатки.Номенклатура КАК Номенклатура,

        |   ПРЕДСТАВЛЕНИЕССЫЛКИ(Остатки.Номенклатура) КАК НоменклатураПредставление,

        |   -Остатки.КоличествоОстаток КАК Дефицит

        |ИЗ

        |   РегистрНакопления.СвободныеОстатки.Остатки(

        |           &МоментВремени,

        |           Номенклатура В

        |               (ВЫБРАТЬ

        |                   Товары.Номенклатура КАК Номенклатура

        |               ИЗ

        |                   Товары КАК Товары)) КАК Остатки

        |ГДЕ

        |   Остатки.КоличествоОстаток < 0";

    ГраницаКонтроля = Новый Граница(МоментВремени(), ВидГраницы.Включая);

    Запрос.УстановитьПараметр("МоментВремени", ГраницаКонтроля);

    РезультатЗапроса = Запрос.Выполнить();

    //  5. Вывод сообщений о недостатке товаров

    Если Не РезультатЗапроса.Пустой() Тогда

        Отказ = Истина;

        ВыборкаОшибки = РезультатЗапроса.Выбрать();

        Пока ВыборкаОшибки.Следующий() Цикл

            Сообщение = Новый СообщениеПользователю;

            Сообщение.Текст = "Товара "+ВыборкаОшибки.НоменклатураПредставление+" недостаточно в количестве "+ВыборкаОшибки.Дефицит+" шт.";

            Сообщение.Сообщить();

        КонецЦикла;

    КонецЕсли;

КонецПроцедуры

Разберем ключевые точки алгоритма.

1. Получение запросом данных документа

2. Формирование движений-расход регистра

3. Запись движений в БД

4. Запрос, получающий отрицательные остатки из регистра

5. Вывод сообщений о недостатке товаров

Преимущества контроля остатков по новой методике

Итак, оба алгоритма решают одну и ту же задачу.

Разница между алгоритмами видна, но преимущества не очевидны.

Поэтому давайте подчеркнем их:

Нет необходимости очищать старые движения документа. По сути это операция записи в БД пустого набора движений и удаление существующих движений – это довольно ресурсоемкие операции

Запрос, получающий данные по отрицательным остаткам, обращается только к одной таблице – нет необходимости делать левое соединение с данными документа и применять функцию «ЕСТЬNULL()»

Кроме этого, при нормальном течении бизнес-процессов пользователь указывает количество, не превышающее остаток на складе.

В этом случае второй запрос не вернет никаких данных и проведение документа будет максимально быстрым.

А так ли важны эти миллисекунды?

На базах с небольшим количеством данных и пользователей разница будет незаметна. Но в нагруженных системах с десятками пользователей цена каждой миллисекунды высока.

Кроме того, на экзамене 1С:Специалист по платформе нужно обязательно использовать новый способ контроля остатков, если это допускает конкретная задача.

Ok, значит, нужно всегда использовать новую методику, верно?

Нет, это не так!

Новая методика может использоваться только в том случае, если для проведения документа есть все необходимые данные в самом документе.

То есть для получения данных не нужно обращаться к регистрам, по которым производится контроль остатков.

Так, например, если в регистре «Свободные остатки» учитывалась бы и сумма, то пришлось бы использовать старую методику контроля.

Почему?

Да просто, чтобы рассчитать сумму списания себестоимости, придется обратиться к регистру. И раз уж мы делаем этот запрос ДО формирования движений, то будет иметь смысл сразу получить доступный остаток.

К слову сказать, в типовой «1С:Управление торговлей 11» реализован контроль остатков по новой методике, а в «1С:Бухгалтерии 8» – по старой методике.

Представленный выше алгоритм можно использовать лишь в учебных целях. Дело в том, что в них не учитываются управляемые блокировки, которые необходимо применять, если в системе работает более одного пользователя.