МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ

МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

**ОСНОВЫ ПРОГРАММИРОВАНИЯ В 1С:ПРЕДПРИЯТИЕ**

ДОМАШНЯЯ КОНТРОЛЬНАЯ РАБОТА

Выполнил Учащаяся группы ПО-455 О.Е.Листопадова

Шифр 21

# 202327 Охарактеризуйте функции преобразования (из числа в строку, из строки в число, в дату). Приведите примеры реализации данных функций.

Разберем основные вопросы преобразования типов в 1С 8.3: из строки в число, из числа в строку, из даты в число и прочие варианты.

### Строка в число в 1С

Пусть, у нас есть числа в таком виде.

СтрокаЧ1 = "120";

СтрокаЧ2 = "12.24";

СтрокаЧ3 = "12,24";

СтрокаЧ4 = "0,24";

СтрокаЧ5 = "0000001";

Для того, чтобы выполнить преобразование этих строк в число, необходимо воспользоваться методом Число. Этот метод преобразует параметр в число.

Ч1 = Число(СтрокаЧ1);

Ч2 = Число(СтрокаЧ2);

Ч3 = Число(СтрокаЧ3);

Ч4 = Число(СтрокаЧ4);

Ч5 = Число(СтрокаЧ5);

Обратите внимание, что разделять дробную часть можно в строке как при помощи символа точка «.», так и при помощи символа запятая «,». Если перед каким-то числом в строке стоят нули, то они отсекаются. У нас будет следующий результат (Рисунок 1).

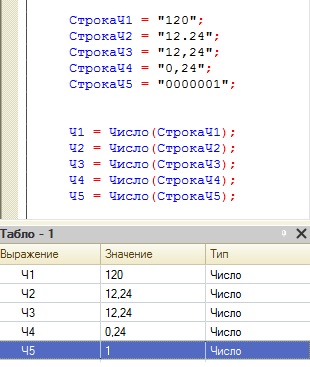


Рисунок 1 Вывод результата.

### Число в строку в 1С

Рассмотрим обратную задачу, когда в 1С 8.3 нужно преобразовать число в строку. Самый простой вариант использовать метод Строка, который преобразует собственный параметр в строку.

Например:

Ч1 = 120;

Ч2 = 12.24;

Ч3 = 10000000000;

СтрокаЧ1 = Строка(Ч1);

СтрокаЧ2 = Строка(Ч2);

СтрокаЧ3 = Строка(Ч3);

Сообщить(СтрокаЧ1);

Сообщить(СтрокаЧ2);

Сообщить(СтрокаЧ3);

И какой результат возвращает этот код (Рисунок 2):



Рисунок 2. Результат преобразования.

У этого способа имеется недостаток: он ставит пробелы между разрядами у длинных чисел. Для того, чтобы в 1С преобразовать число в строку без пробелов нужно воспользоваться функцией Формат.  Данная функция имеет два параметра: преобразуемое значение и форматную строку. Если нам нужно указать, что число преобразуется в строку без пробелов в разряде, то необходимо указать в форматной строке, что порядок разделения группировки разрядов числа или равен 0, или пустой.

Ч1 = 10000000000;

СтрокаЧ1 = Формат(Ч1,"ЧГ=");

Сообщить(СтрокаЧ1);

Результат будет без разрядов (Рисунок 3):

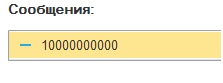


Рисунок 3. Вывод без разделителей групп разрядов.

### Строка в дату 1С

Узнаем, как преобразовать в 1С 8.3 строку в дату.  Для этого необходимо использовать метод Дата, который преобразует параметр (в том числе строку ) в дату . Причем, замечу, что строка должна быть задана в таком формате «ГГГГММДДЧЧММСС».

Рассмотрим пример:

СтрокаД1 = "20120910";

СтрокаД2 = "20120910121559";

СтрокаД3 = "00010101121559";

Дата1 = Дата(СтрокаД1);

Дата2 = Дата(СтрокаД2);

Дата3 = Дата(СтрокаД3);

И результат представлен на рисунке 4.

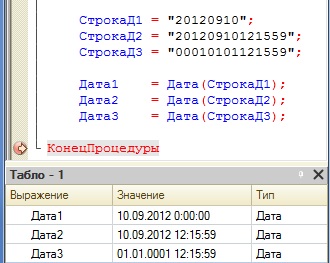


Рисунок 4. Преобразование строки в дату.

Если мы зададим дату в не верном формате, то преобразование не произойдет и возникнет ошибка «Преобразование значения к типу Дата не может быть выполнено».

Например, этот код:

СтрокаД1 = "121559";

Дата1 = Дата(СтрокаД1)

Приведет к ошибке (Рисунок 5):

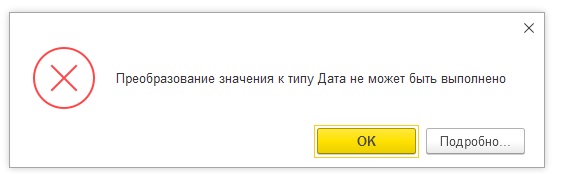


Рисунок 5. Ошибка преобразования.

### Дата в строку в 1С

Преобразовать дату в строку можно несколькими способами. Первый способ: воспользоваться уже знакомым нам методом Строка.

Дата1 = Дата(2012,10,12);

СтрокаД = Строка(Дата1);

Сообщить(СтрокаД);

В этом случае у нас выйдет строка в обычном «полном» формате даты (Рисунок 6).

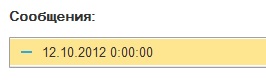


Рисунок 6, Дата в строку.

А чтобы получить дату в том формате, в каком нам нужно, необходимо воспользоваться методом Формат.

Форматов дат может быть великое множество ,все я разбирать в этой статье не буду, покажу только как можно быстро воспользоваться этой функцией.

Мы изменим предыдущий код, написав вместо метода Строка метод Формат, а в качестве второго параметра у метода Формат напишем просто две кавычки.

Дата1 = Дата(2012,10,12);

СтрокаД = Формат(Дата1,"");

Сообщить(СтрокаД);

После нужно между кавычками поставить курсор, вызвать контекстное меню, и применить в нем команду [Конструктор форматной строки](https://www.1s-up.ru/konstruktor-formatnoj-stroki/) (Рисунок 7).

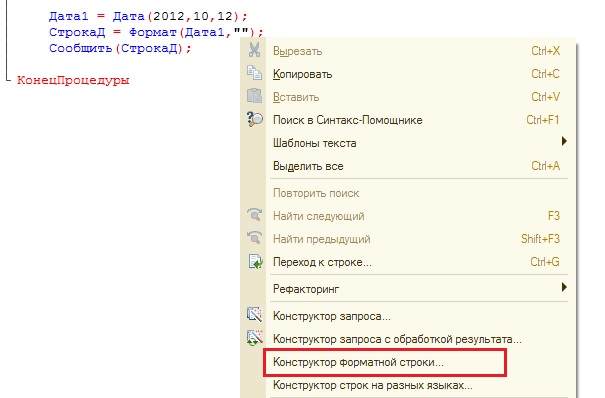


Рисунок 7. Конструктор форматной строки.

В этом конструкторе на закладке Дата можно выбрать удобный формат даты (Рисунок 8).

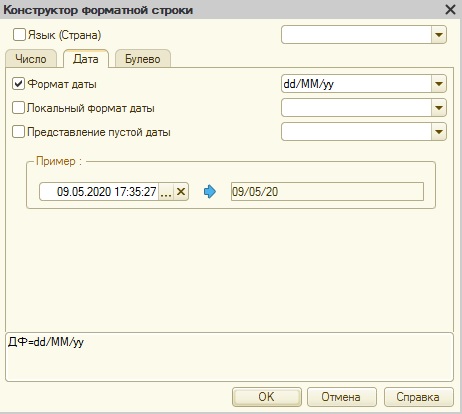


Рисунок 8. Формат даты в конструкторе.

И после нажатия кнопки ОК этого конструктора, нужный формат появится в виде строки.

Дата1 = Дата(2012,10,12);

СтрокаД = Формат(Дата1,"ДФ=dd/MM/yy");

Сообщить(СтрокаД);

В результате метода Формат, дата будет преобразована в строку в том виде, в каком нам необходимо (Рисунок 9).



Рисунок 9. Результат работы форматной строки.

# 49 Опишите процесс проведения документов.

**Документы** — это прикладные объекты конфигурации. Они позволяют хранить в прикладном решении информацию о совершенных хозяйственных операциях или о событиях, произошедших в «жизни» предприятия вообще. Это могут быть, например, приходные накладные, приказы о приеме на работу, счета, платежные поручения и т. д.

Структура документа

Каждый документ характеризуется номером, датой и временем. Система поддерживает режим автоматической нумерации документов, при котором она самостоятельно может генерировать номер для нового документа. Кроме этого, система позволяет осуществлять контроль уникальности номеров документов, не разрешая создавать документы с одинаковыми номерами (Рисунок 10).

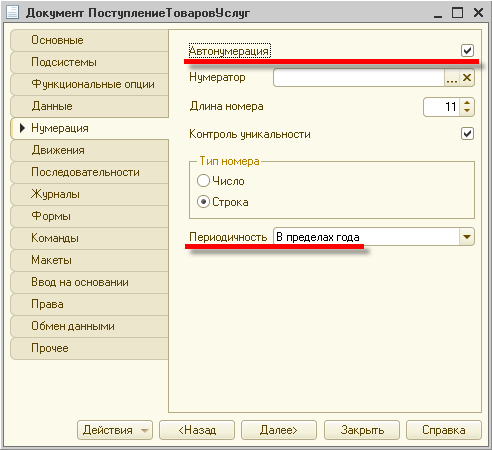


Рисунок 10. Автонумерация документа.

Система автоматически поддерживает режим, при котором уникальность номеров и автоматическая нумерация могут выполняться в пределах определенного периода (день, месяц, квартал, год). Например, если периодичность установлена год, то с нового года система опять начнет нумерацию указанных документов с 1.

Важными характеристиками документа являются дата и время. Они позволяют установить строгую временную последовательность совершения операций. Таким образом, документы могут отличаться друг от друга не только номером, но и своим положением на временной оси. В результате всегда можно сказать, какая из двух операций была совершена раньше.

Помимо номера, даты и времени, каждый документ, как правило, содержит некоторую дополнительную информацию, которая подробно описывает этот документ. Например, для документа **Поступление товаров и услуг** это может быть информация о поставщике товаров (контрагенте), складе, на который приходуются товары и т. п. Набор такой информации является одинаковым для всех документов конкретного вида, и для ее хранения служат реквизиты документа (Рисунок 11):

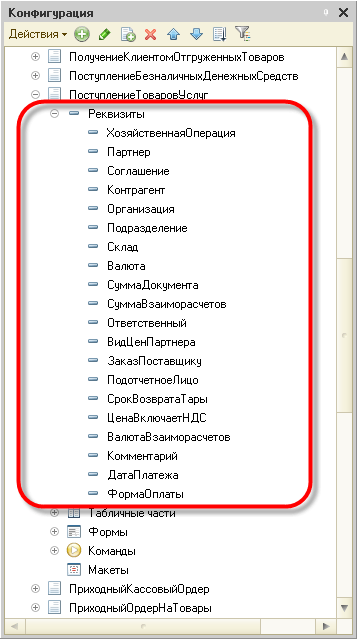


Рисунок 11. Реквизиты документа.

Кроме этого, каждый документ может содержать некоторый набор информации, которая одинакова по своей структуре, но различна по количеству, для разных документов. Например, для документа **Поступление товаров и услуг** это может быть информация о товарах, поступивших на предприятие (наименование, количество, и т. д.), серийных номерах и возвратной таре. Для хранения подобных данных служат табличные части документа (Рисунок 12):

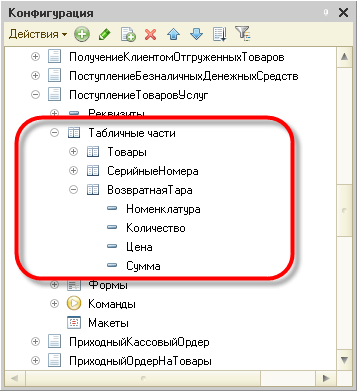


Рисунок 12. Табличная часть документа.

## Проведение документа

Важным свойством документа является возможность его проведения. Если документ проводится, то он может изменить состояние тех или иных учитываемых данных. Если же документ не является «проводимым» это значит, что событие, которое он отражает, не влияет на состояние учета, который ведется в данном прикладном решении.

Например, документ **Поступление товаров и услуг** при своем проведении может вносить изменения в состояние расчетов с поставщиками, в учет остатков товаров, в состояние заказов покупателей и другие учетные данные (Рисунок 13):

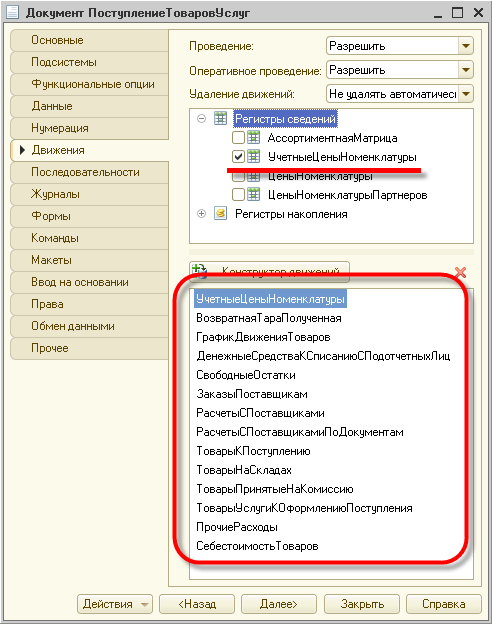


Рисунок 13. Другие учетные данные.

Для документа **Коммерческое предложение контрагенту** напротив, проведение может быть запрещено, потому что этот документ никак не влияет на состояние учета (Рисунок 14):

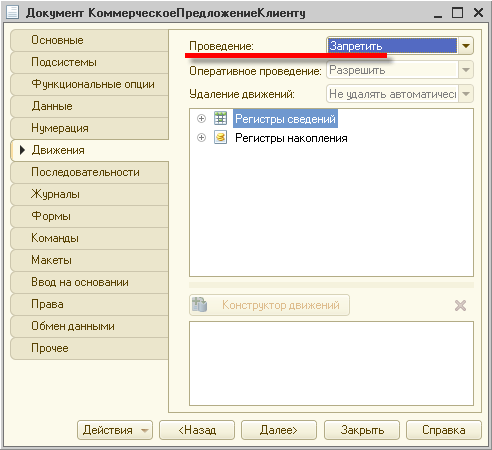


Рисунок 14. Запрет проведения документа.

### Конструктор движений

Алгоритм, на основании которого документ вносит те или иные изменения в состояние учетных данных при своем проведении, описывается средствами встроенного языка на этапе разработки прикладного решения. Система содержит конструктор движений, который помогает разработчику создавать алгоритмы проведения документа.

# 62 Опишите назначение оборотного регистра накопления

Регистры накопления — это прикладные объекты конфигурации. Они составляют основу механизма учета движения средств (финансов, товаров, материалов и т. д.), который позволяет автоматизировать такие направления, как складской учет, взаиморасчеты, планирование.

Регистр накопления образует многомерную систему измерений и позволяет «накапливать» числовые данные в разрезе нескольких измерений. Например, в таком регистре можно накапливать информацию об остатках товаров в разрезе номенклатуры и склада, или информацию об объемах продаж в разрезе номенклатуры и подразделения компании.

## Структура

Информация в регистре накопления хранится в виде записей, каждая из которых содержит значения измерений и соответствующие им значения ресурсов.

Измерения регистра описывают разрезы, в которых хранится информация, а в ресурсах регистра накапливаются нужные числовые данные. Например, для регистра ТоварныеЗапасы, который имеет следующую структуру (Рисунок 15):

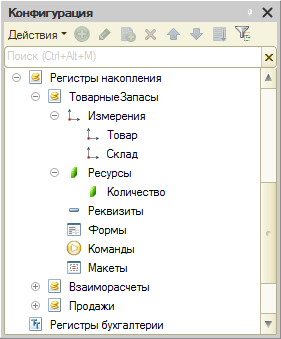


Рисунок 15. Структура регистра.

Записи, производящие изменение ресурсов регистра в базе данных, будут выглядеть следующим образом (Рисунок 16):

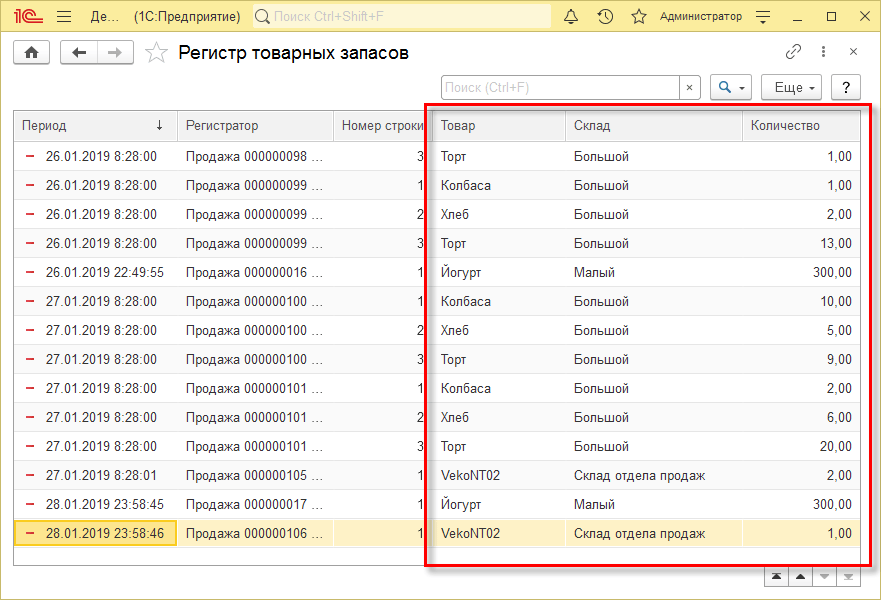


Рисунок 16. Записи производящие изменения.

Поскольку регистр накопления служит для накопления числовых значений, каждая запись выполняет изменение хранимых ресурсов — движение. Движения, в общем случае, могут либо добавлять некоторые приращения к хранимым ресурсам, либо отнимать их. Если должно выполняться увеличение хранимых ресурсов, — такое движение называется движением прихода («+»), если уменьшение хранимых ресурсов — движением расхода («-»).

Вместе с каждой записью, находящейся в регистре накопления, можно хранить дополнительную произвольную информацию. Для этого служат реквизиты регистра накопления представленные на рисунке 17.

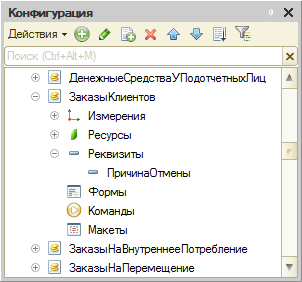


Рисунок 17. Реквизиты регистра.

## Связь с регистратором

Изменение состояния регистра накопления происходит, как правило, при проведении документа. Поэтому каждая запись регистра связана с определенным документом — регистратором, номером строки этого документа, и датой — периодом (Рисунок 18):

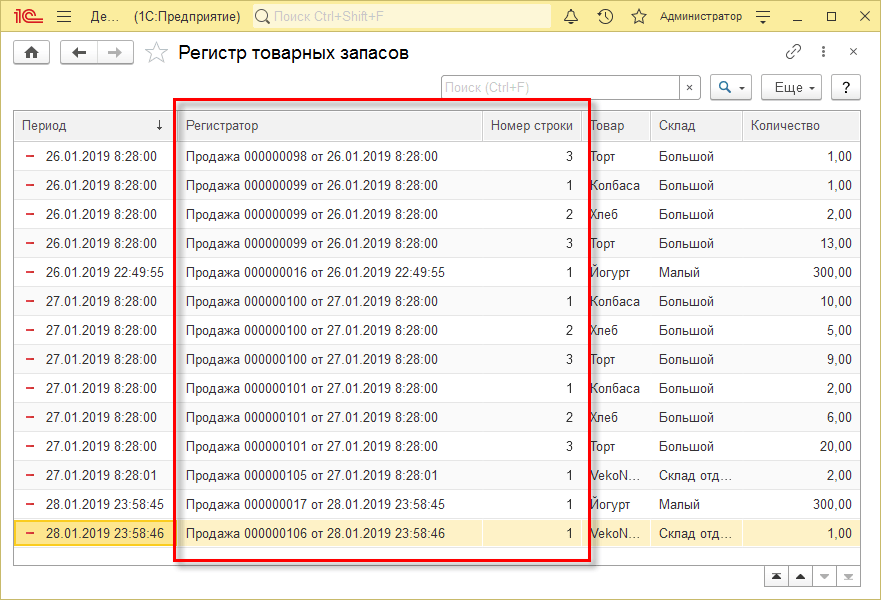


Рисунок 18. Пример связей.

В общем случае значение поле Период может не совпадать с датой документа. Например, документ План продаж может внести в регистр накопления записи о предполагаемых продажах компании несколькими разными датами.

Состав документов, которые могут создавать записи в регистре накопления, задается разработчиком в процессе создания прикладного решения (Рисунок 19):

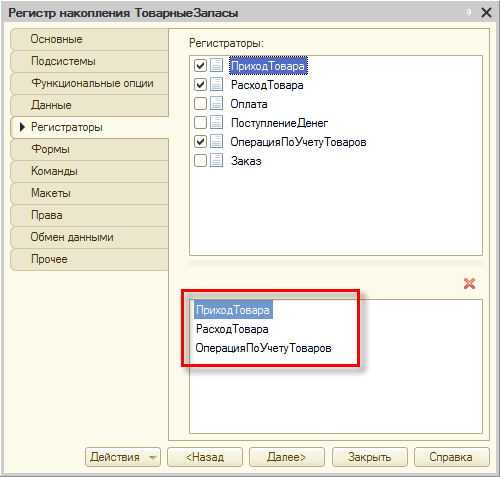


Рисунок 19. Состав документов.

# 84 Охарактеризуйте процесс создания ролей и добавления пользователей.

**Роли** — это общие объекты конфигурации. Они предназначены для реализации ограничения прав доступа в прикладных решениях. Роль в конфигурации может соответствовать должностям или видам деятельности различных групп пользователей, для работы которых предназначена данная конфигурация, рисунок 20.

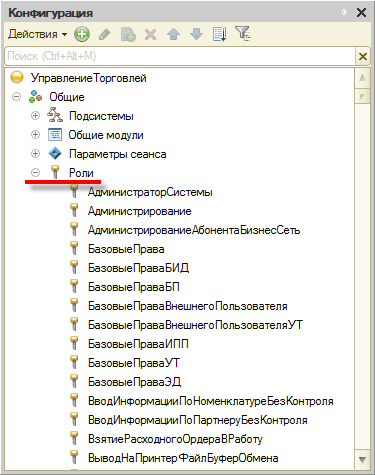


Рисунок 20. Роли пользователей в конфигурации.

Роль определяет, какие действия, над какими объектами метаданных может выполнять пользователь, выступающий в этой роли (Рисунок 21):

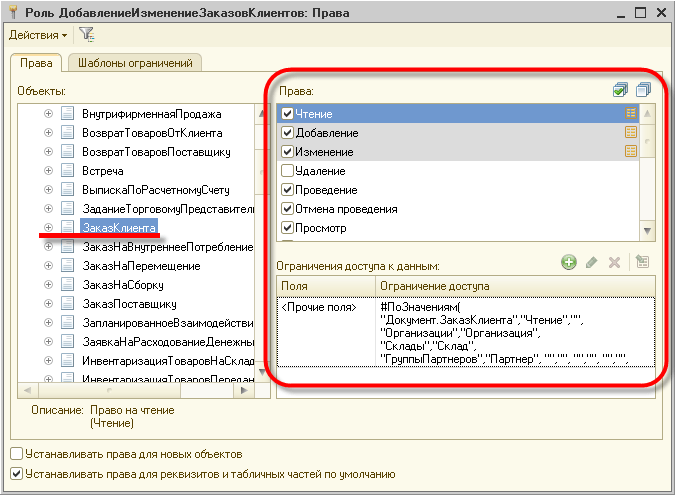


Рисунок 21. Права пользователей на объект.

В процессе ведения списка пользователей прикладного решения каждому пользователю ставится в соответствие одна или несколько ролей, рисунок 22.

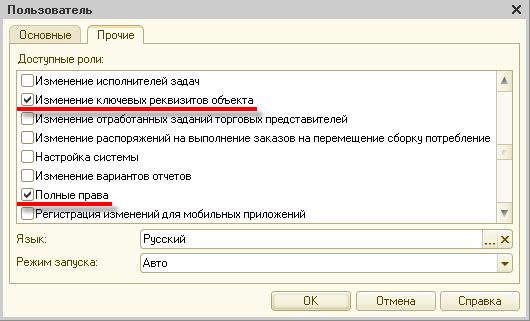


Рисунок 22. Права пользователя.

При попытке пользователя выполнить действие, на которое у него нет разрешения, действие выполнено не будет, а система выдаст окно предупреждения (Рисунок 23):

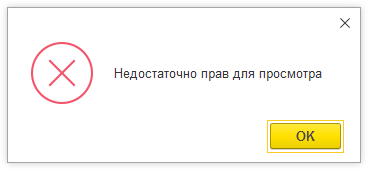


Рисунок 23. Окно предупреждения.

Для редактирования состава ролей платформа содержит два редактора:

* **Редактор роли ­** Этот редактор позволяет состав прав для одной выбранной роли.
* **Редактор «Все роли» ­** Этот редактор позволяет изменять и анализировать состав прав сразу для нескольких или для всех ролей, существующих в прикладном решении.

**Список использованных источников**

1. Быстрова, Н.Ю. Лабораторный практикум по бухгалтерскому учету: практикум / Н.Ю. Быстрова – Ярославль: ЯРГУ, 2018 - 72
2. Кашаев, С.М. Программирование в 1С:Предприятие 8.3 /С.М. Кашаев - СПб.: Питер, 2014. — 304
3. Низамутдинов, И. Программирование в 1С за 11 шагов / И. Низамутдинов. – 2-е издание – Ижевск, 2018 – 711
4. Радченко, М.Г 1С:Преприятие 8.3. Практическое пособие разработчика. Примеры и типовые приемы / М.Г. Радченко, Е.Ю. Хрусталева – Москва : 1С-Паблишинг, 2013. – 943
5. Чистов, П.А. Сборник лабораторных работ для студентов учебных заведений, изучающих программирование в системе 1С:Предприятие 8 (1С:Enterprice 8) / П.А. Чистов, А.А. Мальгинова – Москва: 1С-Паблишинг, 2021 - 492