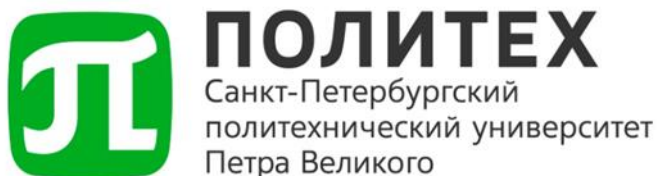


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»

Институт компьютерных наук и технологий

Высшая школа программной инженерии



КУРСОВАЯ РАБОТА

Разработка оконного приложения для работы с PostgreSQL

по дисциплине «Системы управления базами данных»

Выполнил

Студент гр. 3530202/80201

И.А.Шмелёв

Руководитель

Ст. преподаватель

О.В. Прокофьев

«25» мая 2021 г.

Санкт-Петербург

2021 г

Оглавление

Задание на курсовую работу	3
Авторизация.....	4
Роли	4
Описание работы интерфейса	8
Выборка данных.....	8
Вставка данных	10
Удаление данных.....	12
Модификация данных.....	12
Представления.....	12
Хранимые процедуры.....	13
Триггера	16
Курсоры.....	18
Выводы.....	19
Список литературы	20

Задание на курсовую работу

Вариант №3 «Автоматизация работы оптовой фирмы»

В рамках данного цикла лабораторных работ необходимо автоматизировать работу оптовой фирмы. Для этого в рамках базы данных PostgreSQL необходимо создать объекты в схеме вашего пользователя (логин и пароль пользователя для доступа к базе данных студент должен получить у преподавателя) и написать клиентское приложение на базе компонентов ADO.NET или JDBC.

Оптовая фирма (далее организация) занимается оптовыми и мелкооптовыми поставками различных товаров в магазины Санкт-Петербурга. Требуется автоматизировать рабочее место менеджера по распределению товаров между двумя складами организации. Два склада несут функционально различную нагрузку в целях уменьшения арендной платы организации. Первый склад находится на территории Санкт-Петербурга и сильно меньше второго, который находится в значительном удалении. Первый склад выполняет функции КЭШа множества самых популярных товаров. Задачей менеджера по распределению является изучение спроса и выделения множества приоритетных товаров, которые будут завезены на первый склад. Менеджер должен иметь доступ к состоянию обоих складов, справочнику товаров и журналу реализации. При появлении заявки на определенный товар, менеджер заносит заявку в журнал и списывает необходимое количество товара с первого склада. В случае если на первом складе товара не достаточно – остаток списывается со второго. Такой вариант приносит лишние расходы фирме, так как в этом случае она занимается доставкой товара на территорию заказчика, чтобы не потерять клиента. Завоз товаров на первый склад производится ночью каждого дня в соответствии с приоритетами выставленными менеджером. Второй склад содержит достаточное число товаров, и его работа выходит за рамки данного проекта.

База данных должна удовлетворять следующим требованиям:

1. Контроль целостности данных, используя механизм связей
2. Операции модификации групп данных и данных в связанных таблицах должны быть выполнены в рамках транзакций.
3. Логика работы приложения должна контролироваться триггерами. В частности:
 - Триггер должен не позволять списать товар со второго склада при наличии товара на первом
 - Триггер должен контролировать, чтобы вводимая заявка не превышала суммарное количество товара на первом и втором складах
4. Все операции вычисления различных показателей (из требований к клиентскому приложению) должны реализовываться хранимыми процедурами.

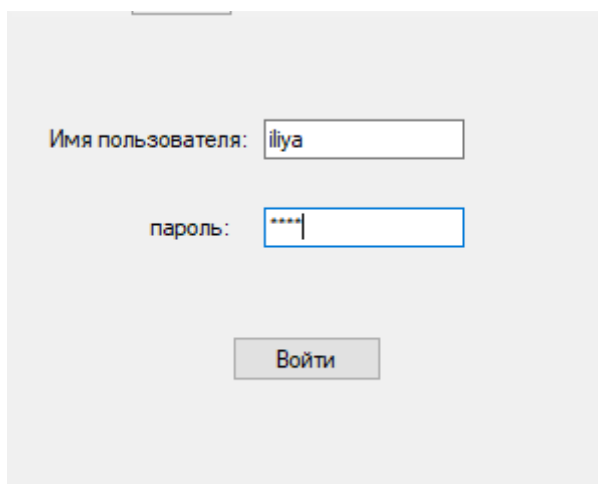
Требования к клиентскому приложению:

1. Необходимо реализовать интерфейсы для ввода, модификации и удаления
 - Товаров, включая задание приоритета
 - Заявки, с автоматическим списанием товара со складов
2. В главном окне приложения должен быть реализован журнал менеджера с просмотром количества товаров на складах.

3. Необходимо реализовать возможность просмотра менеджером следующих показателей:
- Пять самых популярных товаров.
 - Изменение спроса данного товара за некоторый промежуток времени.
 - Графическое отображение изменения спроса заданного товара

Авторизация

В данном интерфейсе реализован вход в систему по двум пользователям: Администратор и Менеджер. Пароль зашифрован алгоритмом md5.



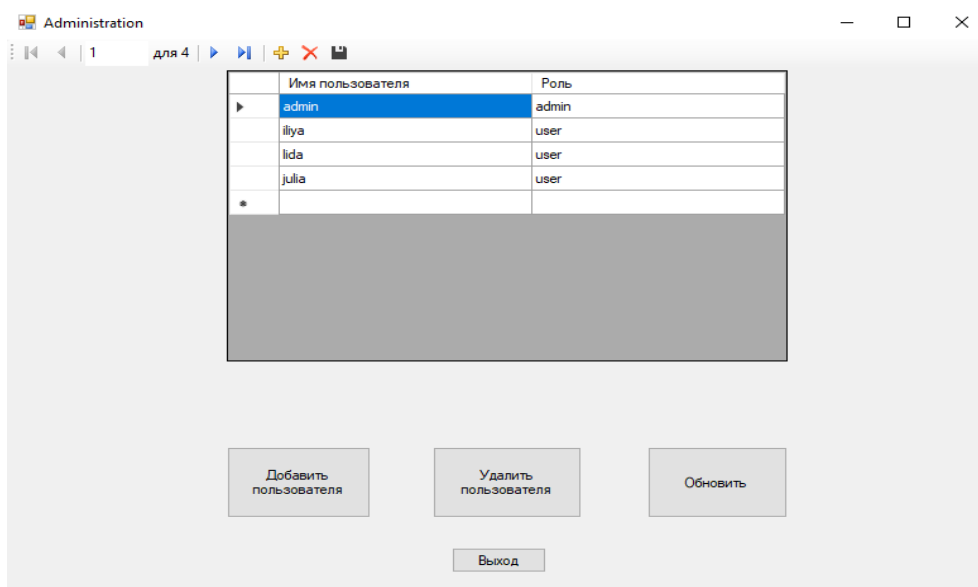
Имя пользователя:

пароль:

Роли

Существуют две активные роли в приложении: Администратор и Менеджер. Администратор осуществляет добавление и удаление сотрудников через свой интерфейс, в то время как Менеджер непосредственно пользуется самой базой данных и её возможностями.

Панель администратора:



Administration

Имя пользователя	Роль
admin	admin
iliya	user
lida	user
julia	user

Добавить пользователя Удалить пользователя Обновить

Выход

Adding_user

0 для 0

Имя пользователя:

пароль:

роль:

Добавить Отмена

Как уже говорилось ранее, администратор может видеть всех пользователей системы а также удалять или добавлять их.

Менеджер:

User

Заявки

Товары

Склад №1

Склад №2

Выход

С помощью данного меню менеджер получает доступ к основным таблицам БД, также внутри них находится весь необходимый функционал.

Sales

0

для 0

+

×

📄

	Наименование	Id заявки	Id товара	Кол-во	Дата создания
▶	Bread	1	1	4	20.03.2020 0:00:..
	Potato	3	4	5	21.03.2020 0:00:..
	Milk	2	2	10	21.03.2020 0:00:..
	Butter	9	3	5	15.04.2020 0:00:..
	Milk	10	2	6	18.04.2020 0:00:..
	Milk	7	2	3	08.04.2020 0:00:..
	Milk	5	2	4	19.04.2020 0:00:..
	Potato	14	4	7	18.04.2020 0:00:..
	Shampoo	21	6	5	15.02.2020 0:00:..
	Sour_Cream	22	10	5	16.02.2020 0:00:..
	Yogurt	23	12	5	16.02.2020 0:00:..

Кол-во товаров за дату

Кол-во свободного товара

Товар на удаление

Его замена

Кол-во товаров на замену

Дата: 25 мая 2021 г. 📅

Кол-во товаров за дату

Выбор по приоритету

Все товары

Кол-во свободного товара

Добавить заявку

Удалить товар

Минимальный спрос

Максимальный спрос

Большой спрос между товарами

Прирост спроса на товар

Удалить заявки по дате

Изменить кол-во товаров

Обновить

Отмена

Goods

1

для 11

▶

▶▶

+

×

📄

	id	Наименование	Приоритет
▶	6	Shampoo	0,01
	52	Bananas	0
	1	Bread	0,45
	2	Milk	0,12
	3	Butter	0,04
	4	Potato	0,02
	10	Sour_Cream	0,03
	12	Yogurt	0,03
	13	Soap	0,03
	14	Beer	0,03
	19	Wiskey	0,03

Общее кол-во товара

Новый приоритет

По наименованию

Отсутствующие товары

Общее кол-во товара

Топ 5 товаров за время

Добавить новый товар

Добавление на склады

Удалить товар

Товары на складе 1 < 30

Топ 5 в месяце

Удалить товары не имеющие заявок

Обновить приоритет

Обновить

Отмена

Описание работы интерфейса

Выборка данных

- однотабличная выборка
 - Вывести товары, упорядочив в алфавитном порядке по наименованию и в обратном порядке по приоритету

	id	Наименование	Приоритет
▶	52	Bananas	0
	14	Beer	0,03
	1	Bread	0,45
	3	Butter	0,04
	2	Milk	0,12
	4	Potato	0,02
	6	Shampoo	0,01
	13	Soap	0,03
	10	Sour_Cream	0,03
	19	Wiskey	0,03
	12	Yogurt	0,03
*			

По наименованию

Отсутствующие товары

Общее кол-во товара

Топ 5 товаров за время

Добавить новый товар

Добавление на склады

Удалить товар

Товары на складе 1 < 30

Топ 5 в месяце

Удалить товары

- Посчитать количество товаров в заявках за заданную дату

Кол-во товаров за дату

Дата:

- выборка с подзапросами
 - Вывести все заявки на товары с приоритетом меньше заданного

Priority — □ ×

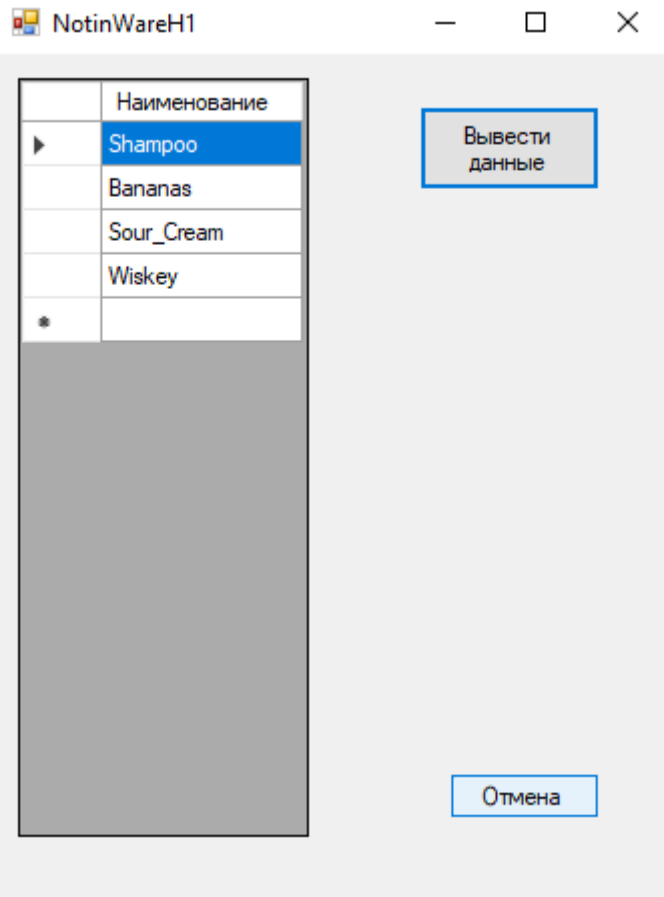
	Наименование	Id заявки	Id товара	Кол-во	Дата создания
▶	Potato	3	4	5	21.03.2020 0:00
	Milk	2	2	10	21.03.2020 0:00
	Butter	9	3	5	15.04.2020 0:00
	Milk	10	2	6	18.04.2020 0:00
	Milk	7	2	3	08.04.2020 0:00
	Milk	5	2	4	19.04.2020 0:00
	Potato	14	4	7	18.04.2020 0:00

Приоритет

Показать результаты

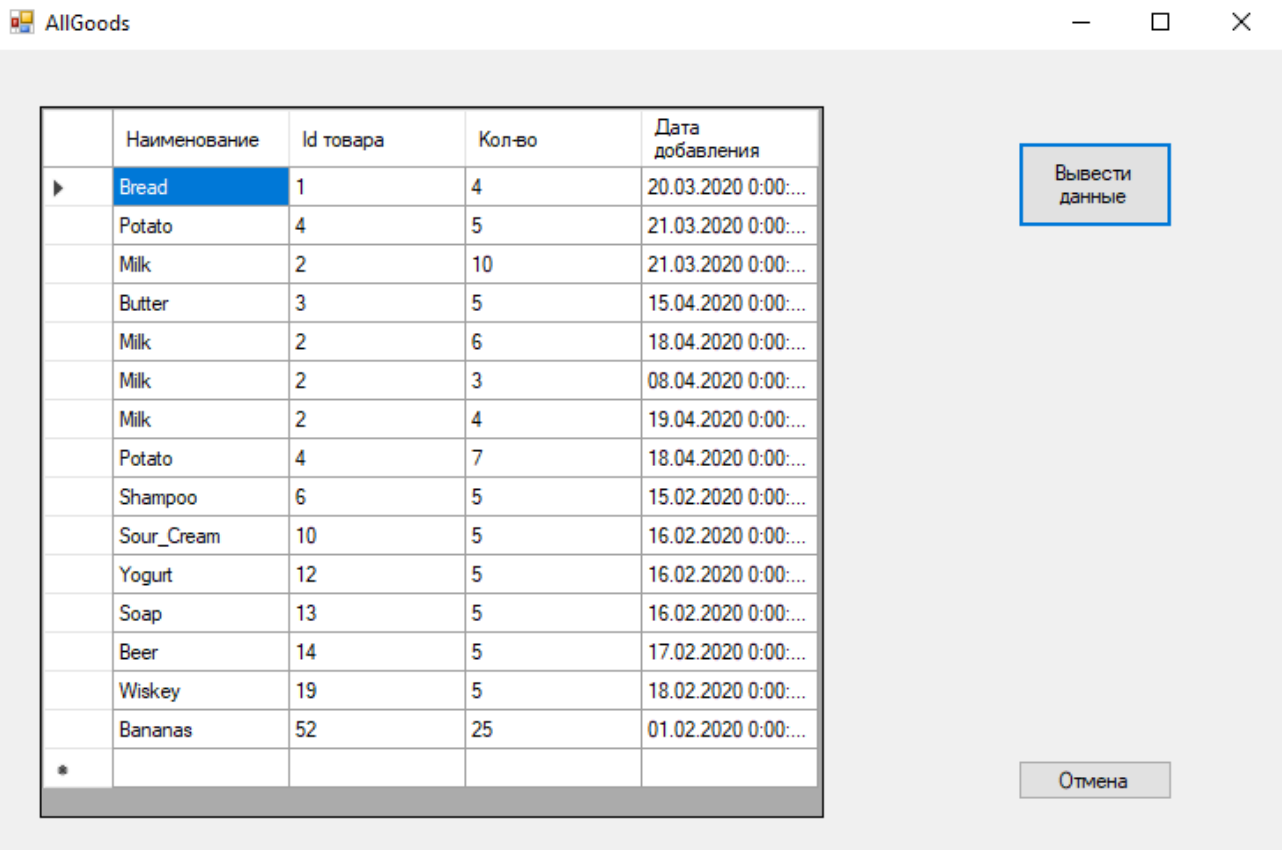
Отмена

2. Вывести наименования на товары, отсутствующие на «складе 1»



- соединение таблиц (join)

1. Вывести заявки и наименования товаров, включая товары, на которые не было заявок



2. Вывести количество товаров, на которые не было заявок за заданную дату

Кол-во свободного товара

- для реализации проекта

1. Вывести суммарное количество заданного товара на первом и втором складах.

Общее кол-во товара

2. Пять самых популярных товаров за заданный промежуток времени, упорядочив в порядке уменьшения спроса.

The screenshot shows a window titled "TopGoods". On the left, there is a table with two columns: "Наименование" (Name) and "Приоритет" (Priority). The table contains the following data:

Наименование	Приоритет
Bread	0.45
Milk	0.12
Butter	0.04
Potato	0.02
*	

Below the table is a large grey rectangular area. To the right of the table, there are two date pickers: "Начальная дата" (Start date) set to "1 марта 2020 г." and "Конечная дата" (End date) set to "30 апреля 2020 г.". Below these are two buttons: "Вывести данные" (Output data) and "Отмена" (Cancel).

Вставка данных

- однотабличная вставка

1. Добавить новый товар

The screenshot shows a window titled "InsertGood". At the top, there is a toolbar with navigation icons and a text input field containing "0" followed by "для 0". Below the toolbar, there are two labels: "Наименование:" (Name) and "Приоритет:" (Priority), each followed by an empty text input field. At the bottom, there are two buttons: "Добавить" (Add) and "Отмена" (Cancel).

2. Добавить заявку на товар из п1

InsertSale

0 для 0

Id товара:

Кол-во:

Дата добавления: 25 мая 2021 г.

Добавить Отмена

- многотабличная вставка в рамках транзакции
 1. Добавить в рамках транзакции новый товар и некоторое количество его завести на «склад 2» и «склад 1»
 2. Добавить в рамках транзакции новый товар и некоторое количество его завести на «склад 2» и «склад 1». В случае, если суммарное число товаров на складе 1 превышает некоторую заданную величину – транзакцию откатить.

InsertWarehouses

Наименование

Приоритет

В склад 1

В склад 2

Добавить Отмена

Удаление данных

- удаление по фильтру и удаление из связанных таблиц
 - Удалить заявки, у которых дата меньше заданной

Дата:	<input data-bbox="323 331 603 365" type="text" value="21 марта 2020 г."/>	<input data-bbox="643 331 922 365" type="button" value="Удалить заявки по дате"/>
-------	---------------------------------------------------------------------------	-----------------------------------------------------------------------------------

- Удалить товары, не имеющие заявок

<input data-bbox="244 448 416 526" type="button" value="Удалить товары
не имеющие
заявок"/>

- Удалить товар и заявки на него

<input data-bbox="260 616 410 645" type="button" value="Удалить товар"/>

- удаление в рамках транзакции
 - Удалить в рамках транзакции товар со «склада 1» с наименьшим приоритетом
 - то же, что и п1, но транзакцию откатить

<input data-bbox="244 864 448 943" type="button" value="Удаление
наименьшего
приоритета"/>

Модификация данных

- модификация по фильтру
 - Изменить количество товаров с заданным наименованием

Кол-во товаров на замену	<input data-bbox="528 1176 727 1216" type="text"/>	<input data-bbox="751 1149 1031 1216" type="button" value="Изменить кол-во
товаров"/>
--------------------------	----------------------------------------------------	-------------------------------------------------------------------------------------------------

- Списать 5 единиц заданного товара со «склада 1»
- Списать 5 единиц заданного товара со «склада 1». Если количество товара на этом складе меньше 5, не достаемое число списать со второго склада.

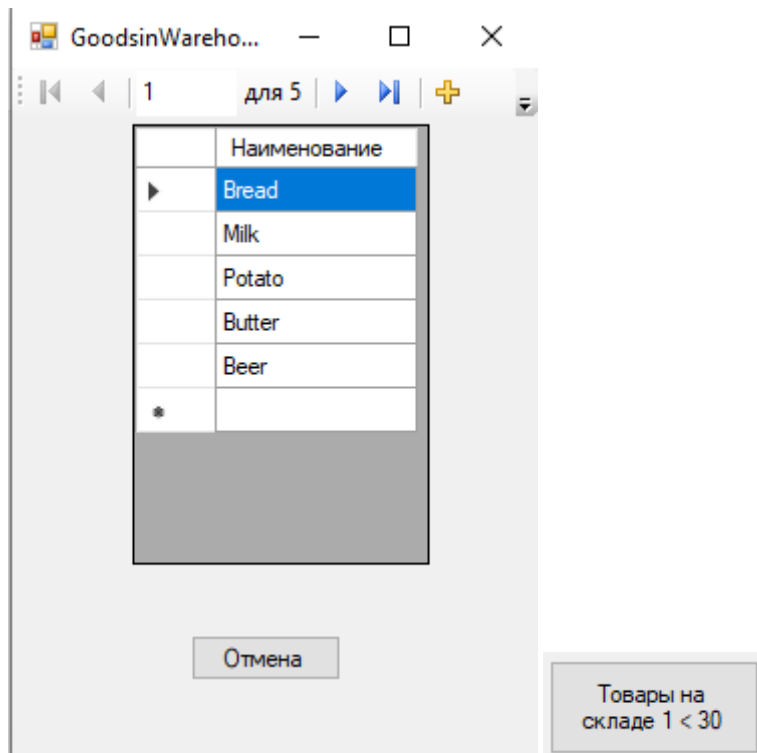
<input data-bbox="260 1379 459 1473" type="button" value="Списать товар"/>

- модификация в рамках транзакции
 - В рамках транзакции поменять заданный товар во всех заявках на другой и удалить его.
 - то же, что и п1, но транзакцию откатить

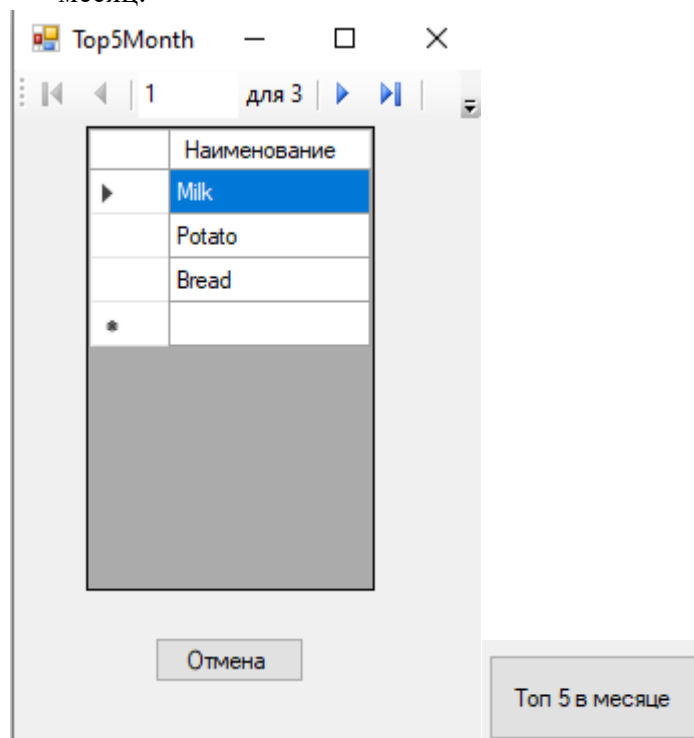
Товар на удаление	<input data-bbox="517 1659 719 1704" type="text"/>	<input data-bbox="743 1682 1007 1749" type="button" value="Удалить товар"/>
Его замена	<input data-bbox="517 1715 719 1760" type="text"/>	

Представления

- Создать представление, отображающее все товары, число которых на складе1 менее некоторого числа



2. Создать представление, отображающее 5 самых популярных товаров за заданный месяц.



Хранимые процедуры

- без параметров
 1. Создать хранимую процедуру, выводящую список товаров для перевоза и его количество согласно текущему состоянию приоритетов.

List

1

для 11

▶▶

+

×

📄

	Наименование	Транспортировка
▶	Bananas	0
	Shampoo	1
	Potato	2
	Beer	3
	Soap	3
	Sour_Cream	3
	Wiskey	3
	Yogurt	3
	Butter	4
	Milk	12
	Bread	45
*		

Отмена

Товары для перевоза

- с входными параметрами
 - Создать хранимую процедуру с параметром количество перевозимого товара за ближайший рейс и выводящую все товары, которые необходимо привезти, и их количество.

Transporting

1

для 2

▶▶

+

×

📄

Кол-во для перевоза: 30

Показать

	Наименование	Кол-во
▶	Shampoo	30
	Bananas	30
*		

Отмена

Перевоз

2. Создать хранимую процедуру, имеющую два параметра «товар1» и «товар2». Она должна возвращать даты, спрос в которые на «товар1» был больше чем на «товар2». Если в какой-либо день один из товаров не продавался, такой день не рассматривается.

Max_priority

Товар 1: Potato Товар 2: Milk Показать

Дата
18.04.2020
*

Отмена

Большой спрос между товарами

- с выходными параметрами

1. Создать хранимую процедуру с входными параметрами, задающими интервал времени, и выходным – идентификатором товара. Процедура должна возвращать товар с максимальным приростом спроса.

Max_Stonk

min_date: 2020-03-01 max_date: 2020-04-30 Fill

maxstonk
Milk

Отмена

Максимальный спрос

Таблица успешно сохранена в: Max_stonk_out.txt

OK

Max_stonk_out.txt – Блокнот

Файл Правка Формат Вид Справка

Максимальный спрос на товар:

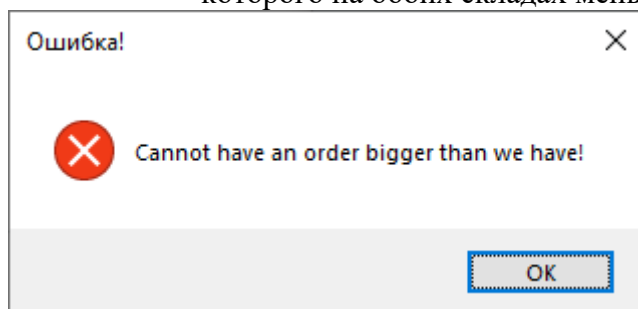
Наименование товара: Milk

2. Создать хранимую процедуру с входными параметрами, задающими интервал времени, и выходным параметром – товар, с минимальным числом продаж за заданный период времени.

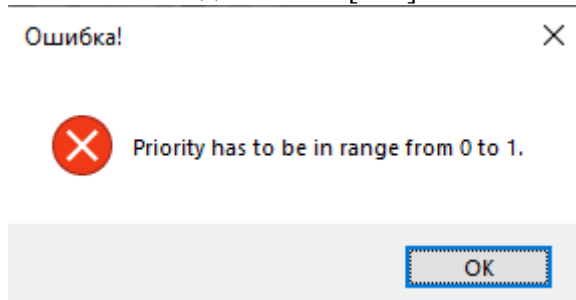
The screenshot shows the 'MinStonk' application window. At the top, there are navigation buttons and a label 'для 1'. Below this, the date range is set from 'Начальная дата: 2020-03-01' to 'Конечная дата: 2020-04-30', with a 'Показать' button. A list box displays 'Наименование' with 'Bread' selected. An 'Отмена' button is at the bottom right. A separate box at the bottom left is labeled 'Минимальный спрос'.

Триггера

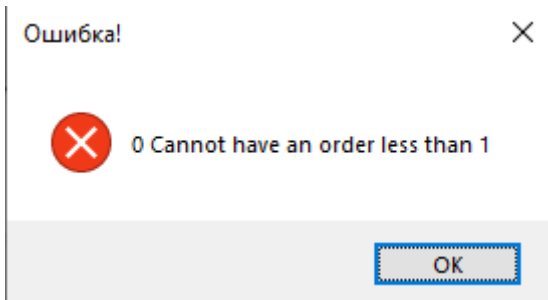
- Триггера на вставку
 1. Создать триггер, который не позволяет добавить заявку на товар, число которого на обоих складах меньше указанного в заявке



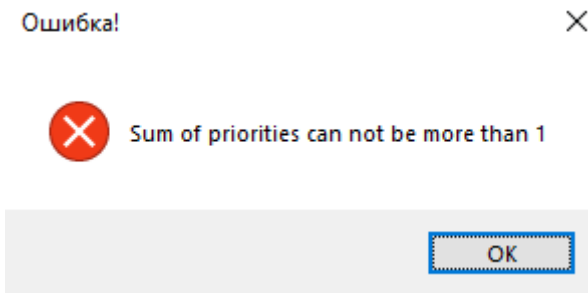
2. Создать триггер, который не позволяет задать приоритет товару вне диапазона [0..1]



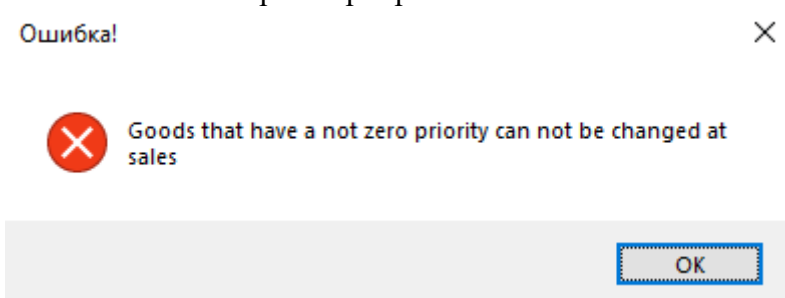
3. Создать триггер, который не позволяет добавить заявку с числом товара меньше 1.



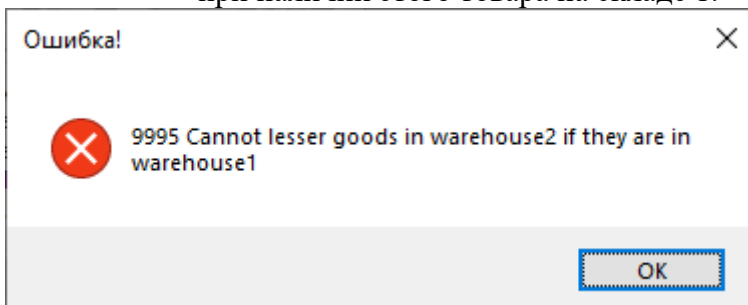
- Триггера на модификацию
 1. Создать триггер, не позволяющий откорректировать приоритет, таким образом, чтобы сумма приоритетов была больше 1.



2. Создать триггер, который не позволяет корректировать заявки на товар, у которого приоритет больше 0.



3. Создать триггер, который не позволяет уменьшить число товара на складе 2 при наличии этого товара на складе 1.

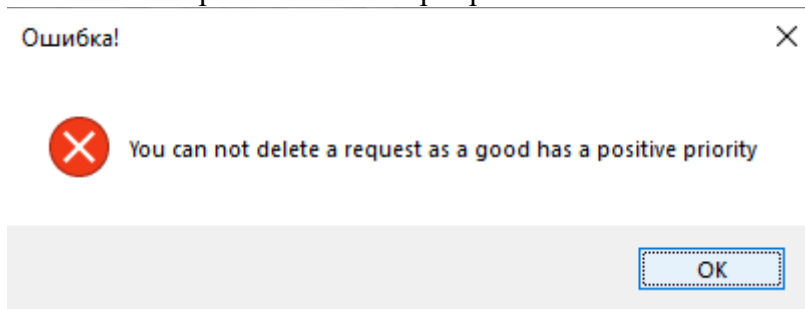


4. Создать триггер, который при обнулении числа товара на складе удаляет строку.

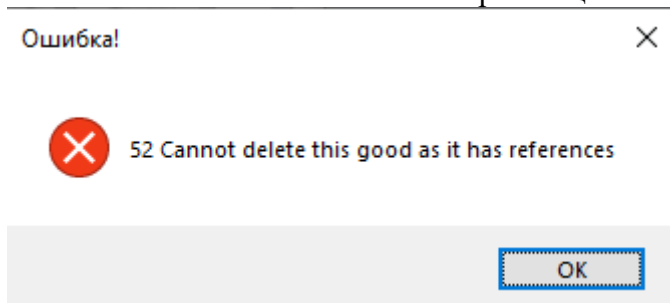
	Наименование	Id товара	Кол-во
▶	Bread	1	25
	Milk	2	25
	Potato	4	25
	Yogurt	12	30
	Soap	13	30
	Butter	3	20
	Beer	14	15
*			

Нулевые строки автоматически удаляются

- Триггера на удаление
 1. Создать триггер, который не позволяет удалить заявку на товар с рассчитанным приоритетом.



2. Создать триггер, который при удалении товара в случае наличия на него ссылок откатывает транзакцию



Курсоры

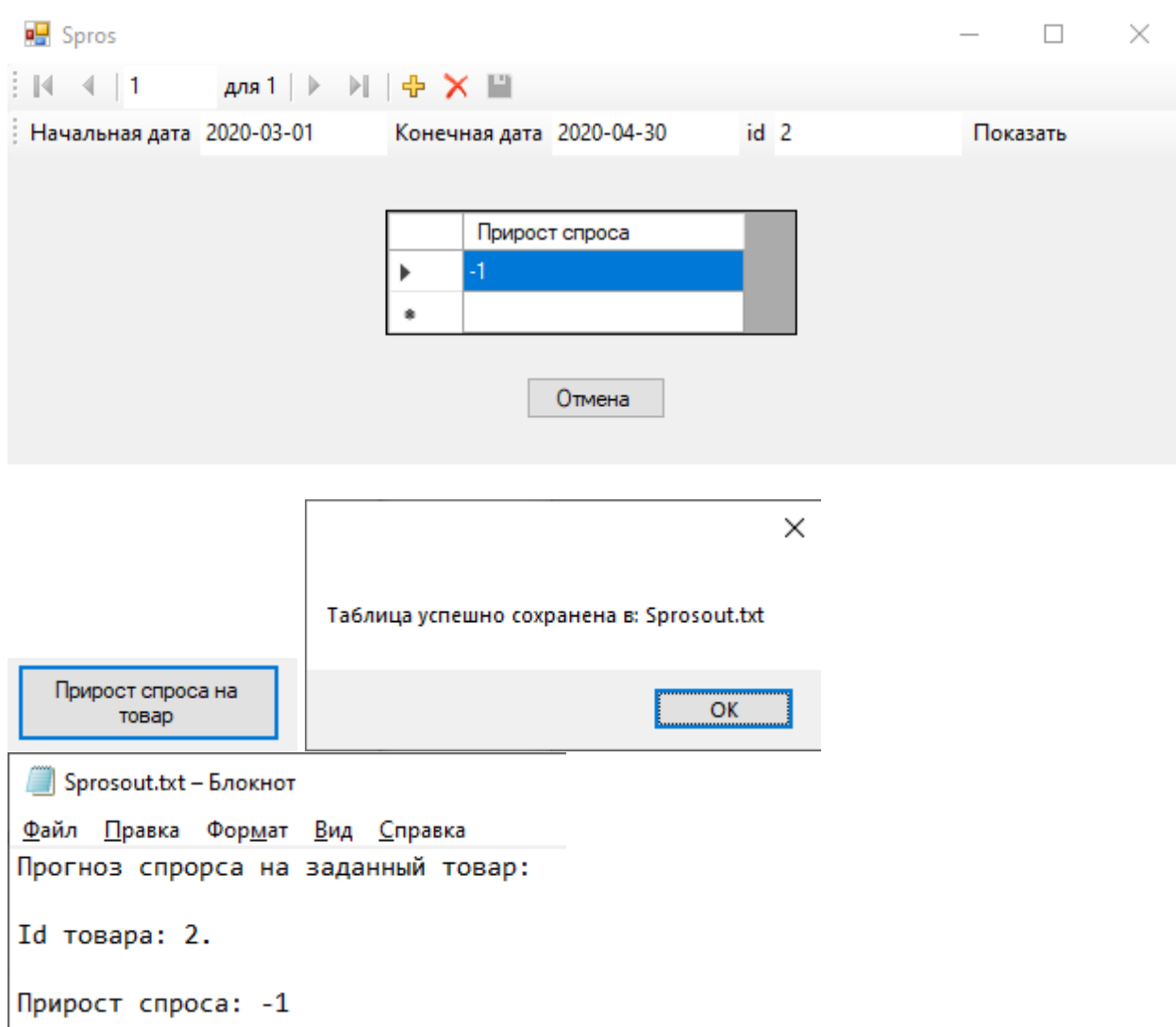
- Хранимая процедура для прогноза спроса на заданный товар

Необходимо реализовать хранимую процедуру, рассчитывающую прогнозируемый спрос на 7 дней на некоторый товар. Хранимая процедура должна иметь два входных параметра задающие интервал времени для анализа изменения спроса и параметр, задающий анализируемый товар.

Предлагаемый алгоритм: с помощью курсора формируем временную таблицу, содержащую номер дня в рассматриваемом интервале и число хранящегося товара. Максимальный номер сохраняем в переменной.

Организуем курсор, перебирающий последовательно строки временной таблицы, упорядоченные по номеру точки, и усредняющий попарно соседние точки (за каждую итерацию точек становится на 1 меньше). Полученное среднее значение каждой пары, заменяет значение минимальной по номеру точки. Последняя точка удаляется.

Работа курсора повторяется до тех пор, пока в таблице не останется 2 точки. Будем считать, что разница спроса в этих двух точках равна разнице спроса между прогнозируемым на следующий день спросом и спросом на заданный товар последнего дня. Таким образом, мы можем получить величину прогнозируемого спроса. (экстраполяция методом скользящего среднего)



Выводы

В результате данной работы было реализовано приложение на платформе Windows Forms на языке C# с использованием данных из PostgreSQL. В данном приложении реализованы все функции которые требовались в лабораторных работах, также был реализован вывод некоторых процедур в файл.

Единственной трудностью с которой пришлось столкнуться в ходе выполнения курсовой работы, было подключение PostgreSQL к Visual Studio 2012, для этого пришлось устанавливать дополнительное ODBC.

Также в ходе работы были изучены особенности работы с базой данных PostgreSQL, особенности написания запросов, триггеров и функций.

Список литературы

1. PostgreSQL Документация / Эл.ресурс, дата – 05.05.21 режим доступа – свободный: <https://postgrespro.ru/docs/postgresql>
2. Руководство по классическим приложениям Windows Forms / Эл.ресурс, дата – 06.05.21 режим доступа – свободный: <https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/>
3. Infragistics Windows Forms / Эл.ресурс, дата – 06.05.21 режим доступа – свободный: <https://www.infragistics.com/help/winforms/infragistics.documents.reports>