

БУ ВО Ханты-Мансийского округа – Югры
«Сургутский государственный университет»
Политехнический институт
Кафедра информатики и вычислительной техники

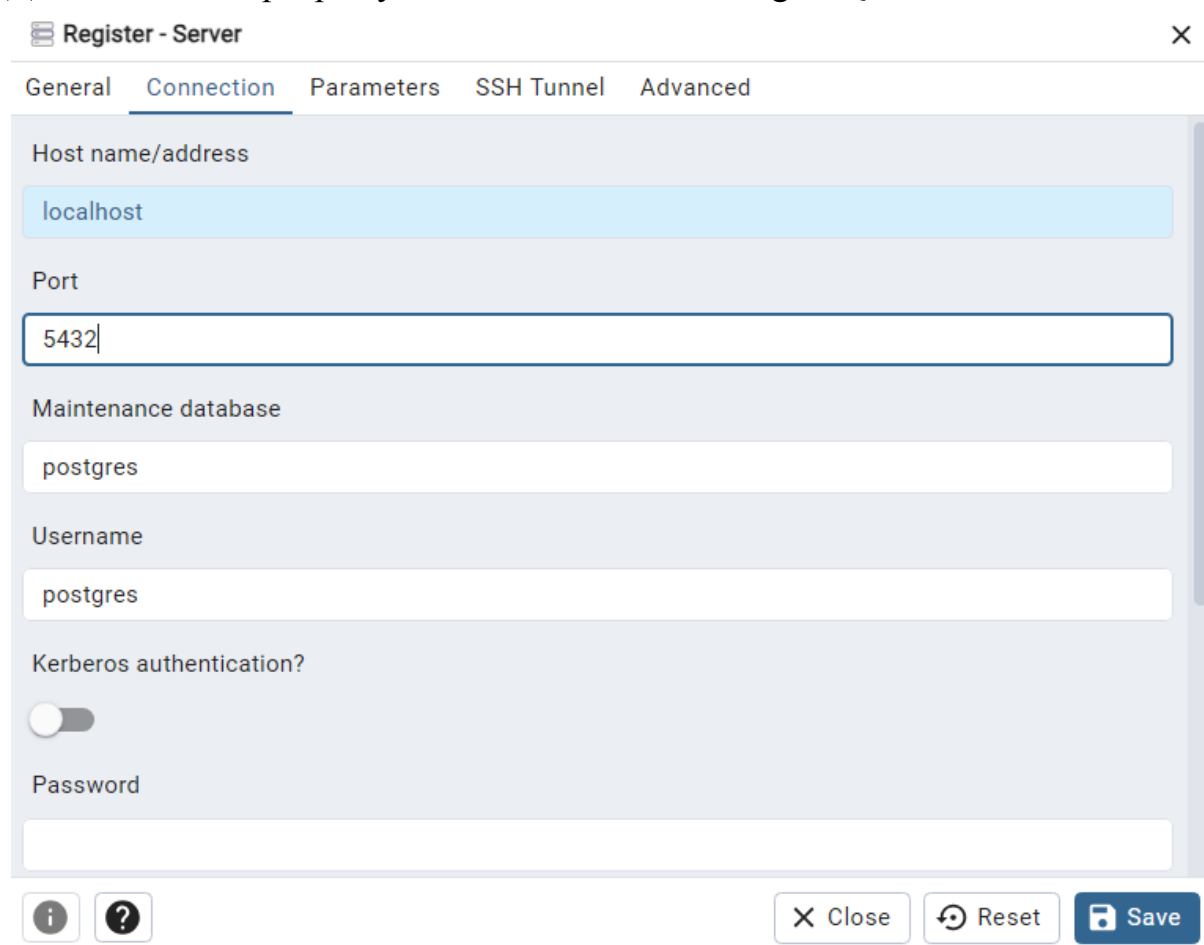
Отчет
Лабораторная работа №1
Базы данных. Управление базами данных.

Проверил:
Гавриленко А.В
Выполнил: студент
группы 606-12
Речук Д.М

Сургут
2023 г

Задание 0. Подключитесь к серверу WS0481.

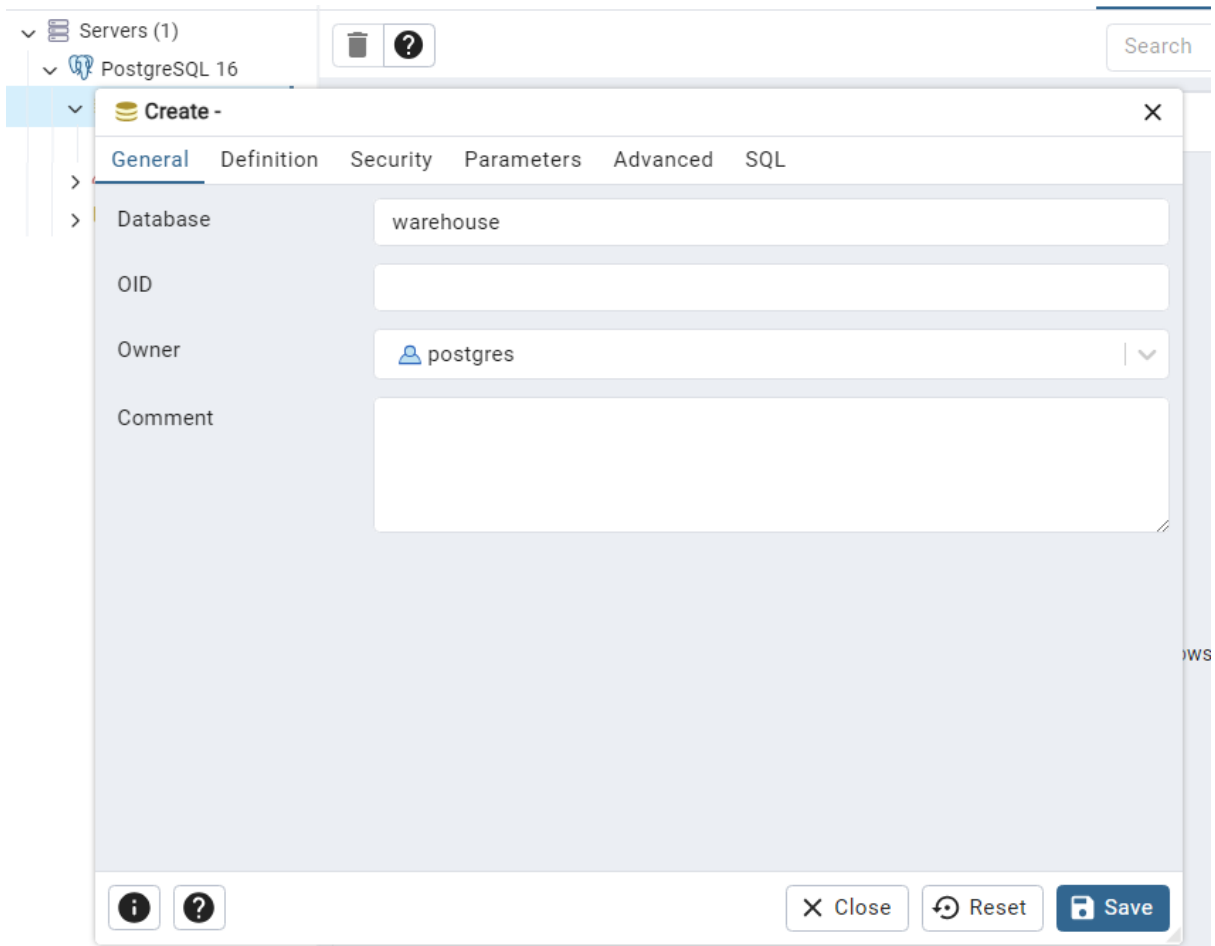
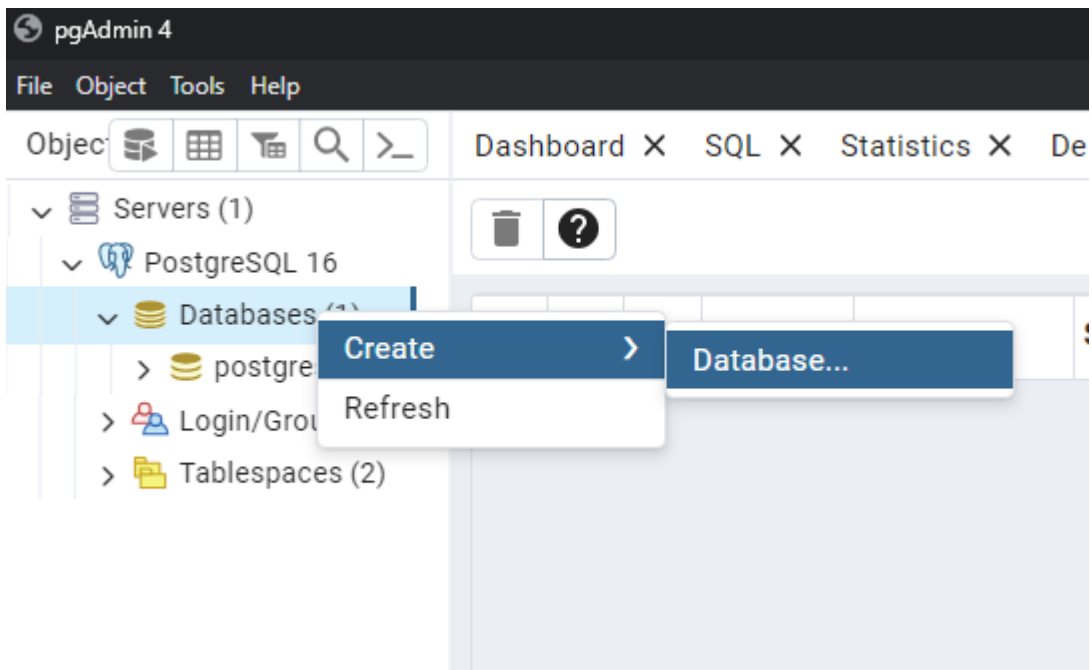
Для создания сервера будет использоваться PostgreSQL 16.



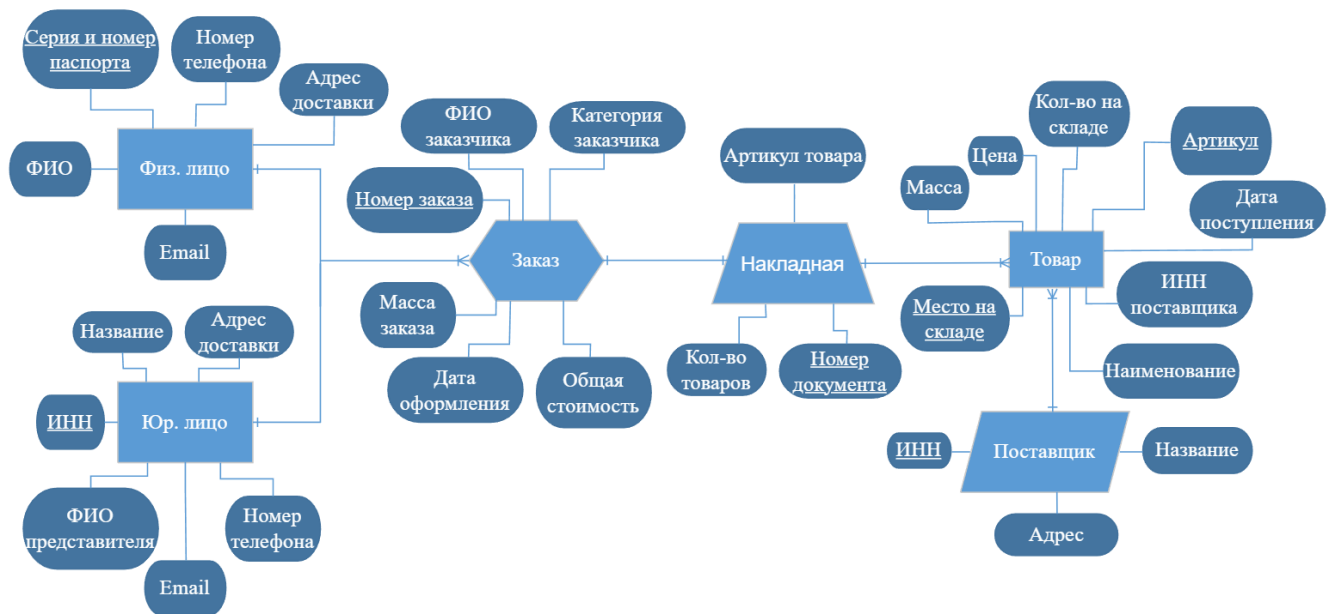
The screenshot shows a 'Register - Server' dialog box with a close button (X) in the top right corner. The 'Connection' tab is selected, showing fields for 'Host name/address' (localhost), 'Port' (5432), 'Maintenance database' (postgres), 'Username' (postgres), 'Kerberos authentication?' (disabled), and 'Password' (empty). At the bottom are buttons for 'Close', 'Reset', and 'Save', along with information and help icons.

Field	Value
Host name/address	localhost
Port	5432
Maintenance database	postgres
Username	postgres
Kerberos authentication?	<input type="checkbox"/>
Password	

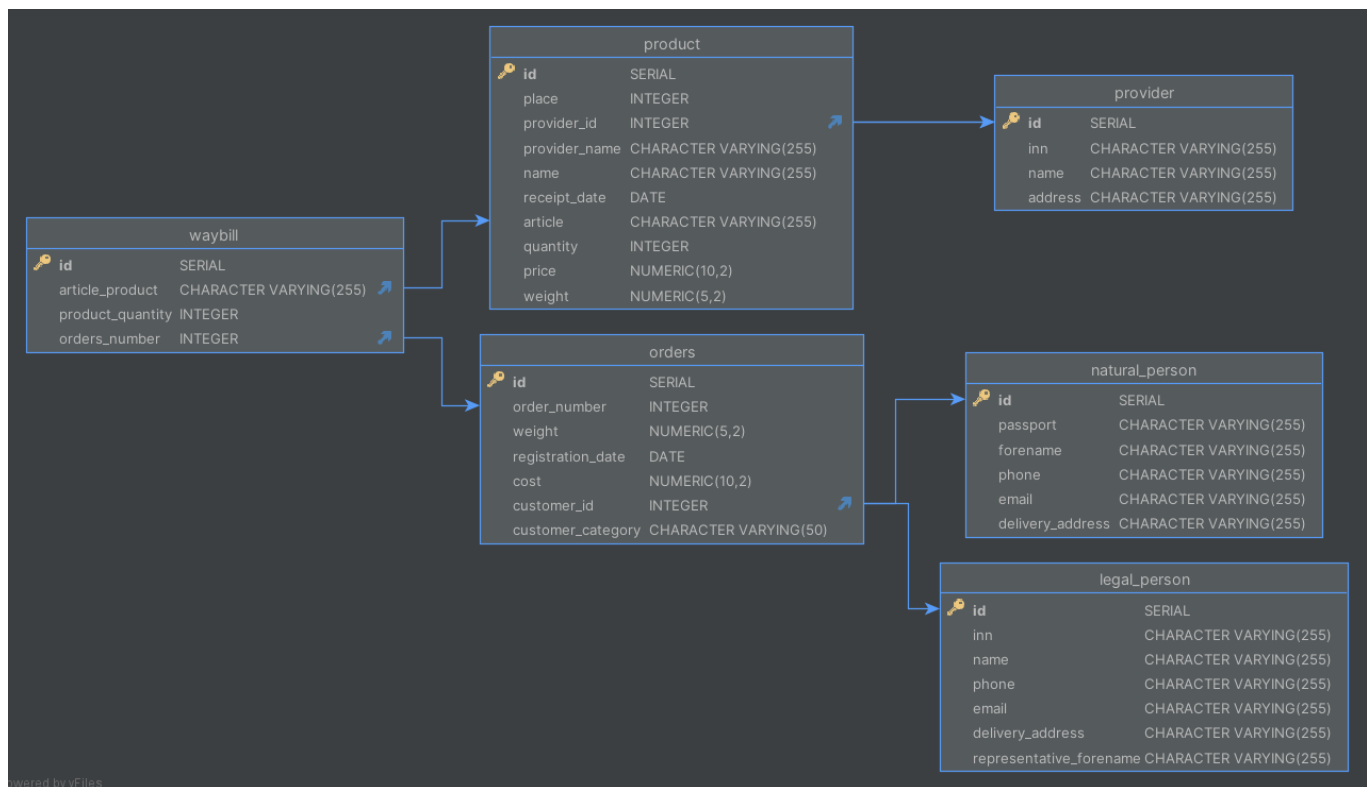
После этого создали базу данных склада под названием “warehouse”



Задание 1. Спроектировать логическую модель



Далее спроектировали физическую модель базы данных










Самостоятельно написали и выполнили sql-скрипт для создания необходимых таблиц

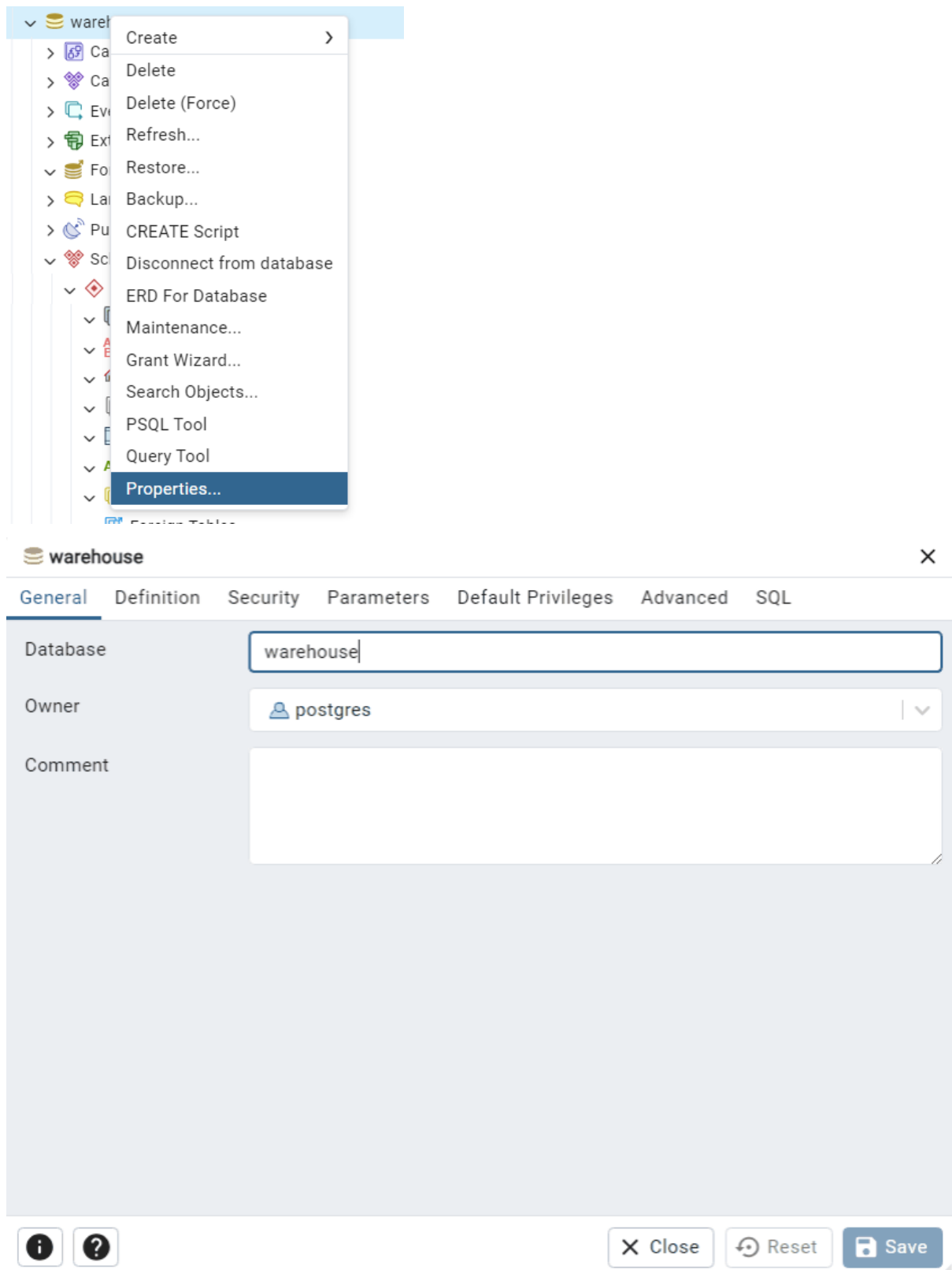
Query	Query History
10	
11	-- Создание таблицы legal_person
12	CREATE TABLE legal_person (
13	id SERIAL PRIMARY KEY,
14	inn VARCHAR(255) UNIQUE,
15	name VARCHAR(255),
16	phone VARCHAR(255),
17	email VARCHAR(255),
18	delivery_address VARCHAR(255),
19	representative_forename VARCHAR(255)
20);
21	
22	-- Создание таблицы provider
23	CREATE TABLE provider (
24	id SERIAL PRIMARY KEY,
25	inn VARCHAR(255) UNIQUE,
26	name VARCHAR(255),
27	address VARCHAR(255)
28);
29	
30	-- Создание таблицы product
31	CREATE TABLE product (

Data Output	Messages	Notifications
CREATE TABLE		
Query returned successfully in 54 msec.		

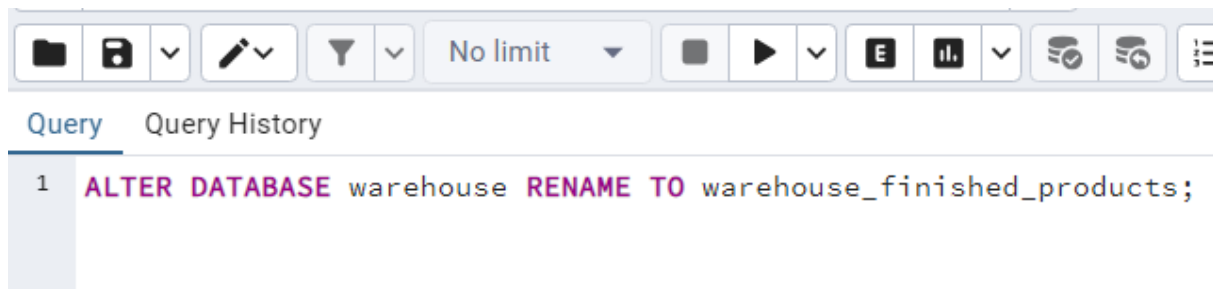
Убедились в успешном создании таблиц

- ▼  Tables (6)
 - >  legal_person
 - >  natural_person
 - >  orders
 - >  product
 - >  provider
 - >  waybill

Задание 2. Просмотрите список параметров базы данных, которые могут быть установлены; Просмотрите список установленных параметров созданной Вами базы данных.



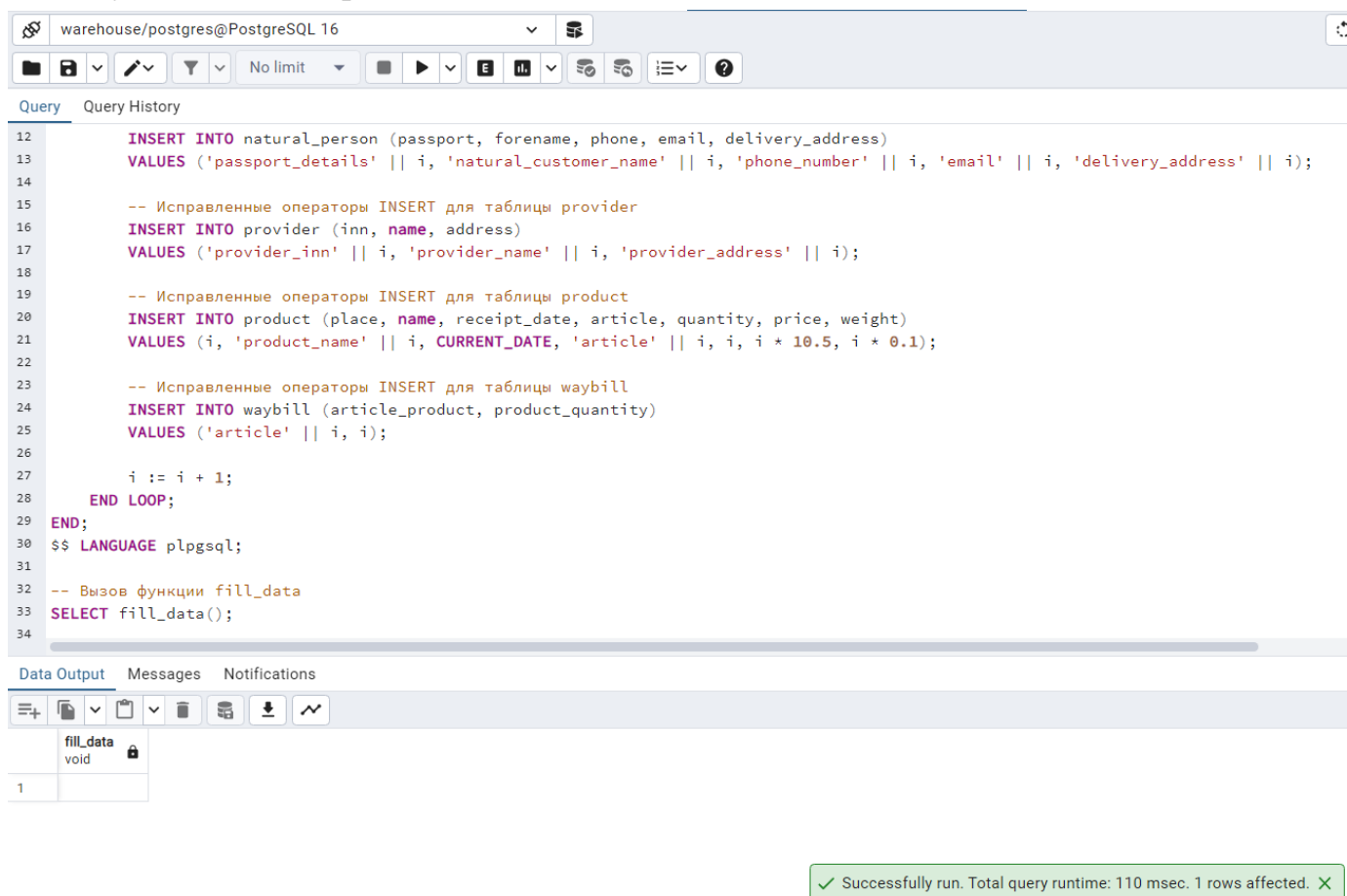
Задание 3. Переименуйте созданную вами базу данных



```
1 ALTER DATABASE warehouse RENAME TO warehouse_finished_products;
```

Задание 4. Для своей базы данных установите возможность автоматического сжатия данных.

Перед установкой возможности автоматического сжатия данных заполним таблицу “provider” тестовыми данными



```
12 INSERT INTO natural_person (passport, forename, phone, email, delivery_address)
13 VALUES ('passport_details' || i, 'natural_customer_name' || i, 'phone_number' || i, 'email' || i, 'delivery_address' || i);
14
15 -- Исправленные операторы INSERT для таблицы provider
16 INSERT INTO provider (inn, name, address)
17 VALUES ('provider_inn' || i, 'provider_name' || i, 'provider_address' || i);
18
19 -- Исправленные операторы INSERT для таблицы product
20 INSERT INTO product (place, name, receipt_date, article, quantity, price, weight)
21 VALUES (i, 'product_name' || i, CURRENT_DATE, 'article' || i, i, i * 10.5, i * 0.1);
22
23 -- Исправленные операторы INSERT для таблицы waybill
24 INSERT INTO waybill (article_product, product_quantity)
25 VALUES ('article' || i, i);
26
27 i := i + 1;
28 END LOOP;
29 END;
30 $$ LANGUAGE plpgsql;
31
32 -- Вызов функции fill_data
33 SELECT fill_data();
34
```

Data Output Messages Notifications

fill_data	void
1	

✓ Successfully run. Total query runtime: 110 msec. 1 rows affected. ✕

Далее для своей базы данных установили возможность автоматического сжатия данных.

provider

×

GeneralColumnsAdvancedConstraintsParametersSecuritySQL

TableTOAST table

Custom auto-vacuum?

Autovacuum Enabled?

Not set

✓ Yes

No

Label	Value	Default
ANALYZE scale factor	<input type="text"/>	0.1
ANALYZE base threshold	<input type="text"/>	50
FREEZE maximum age	<input type="text"/>	200000000
VACUUM cost delay	<input type="text"/>	2
VACUUM cost limit	<input type="text"/>	-1
VACUUM scale factor	<input type="text"/>	0.2
VACUUM base threshold	<input type="text"/>	50

i?

×

 Close

↺

 Reset

💾

 Save

Задание 5. Определите сведения о дисковом пространстве, занимаемом базой данных

Query

Query History

1 `SELECT pg_size_pretty(pg_total_relation_size('provider'))`

Data Output

Messages

Notifications

≡+

▼

▼

	pg_size_pretty text
1	184 kB

После чего произвели сжатие базы данных так, чтобы она содержала только 25% пространства, доступного ей на текущий момент

provider

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Table

TOAST table

Custom auto-vacuum?

Autovacuum Enabled?

Not set

✓ Yes

No

Label	Value	Default
ANALYZE scale factor	<input type="text"/>	0.1
ANALYZE base threshold	<input type="text"/>	50
FREEZE maximum age	<input type="text"/>	200000000
VACUUM cost delay	<input type="text"/>	2
VACUUM cost limit	<input type="text"/>	-1
VACUUM scale factor	<input type="text" value="0.75"/>	0.2
VACUUM base threshold	<input type="text"/>	50

i

?

✕ Close

↺ Reset

💾 Save

Проверили размер дискового пространства, занимаемое базой данной еще раз

Query

Query History










1

`SELECT pg_size_pretty(pg_total_relation_size('provider'));`

Data Output

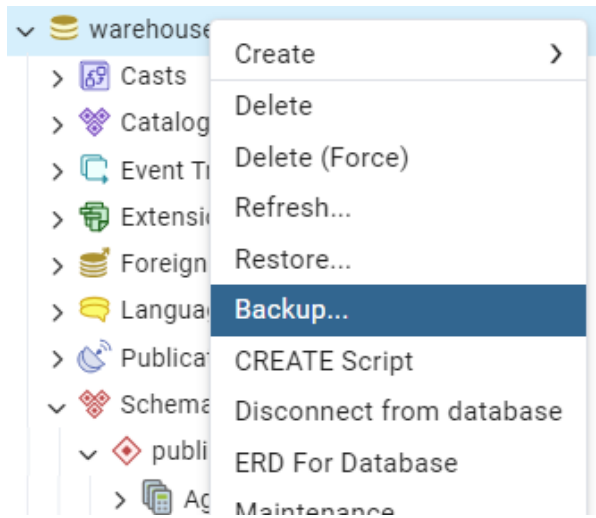
Messages

Notifications

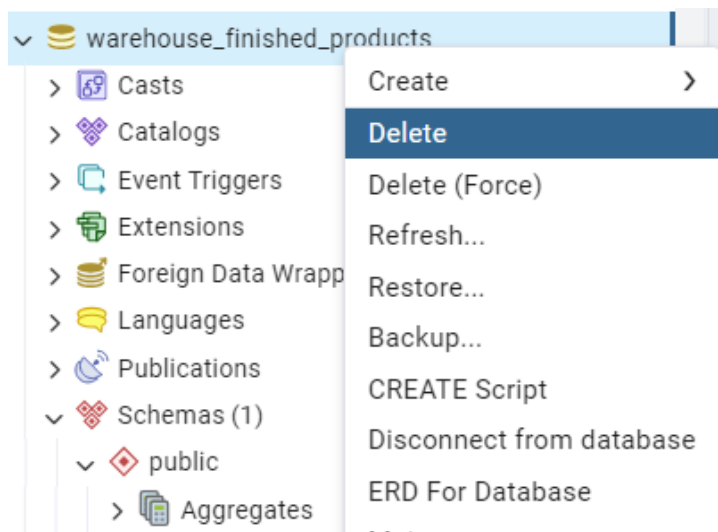


	pg_size_pretty text
1	136 kB

Задание 6. Создайте резервную копию базы данных, задав физическое имя устройства резервного копирования

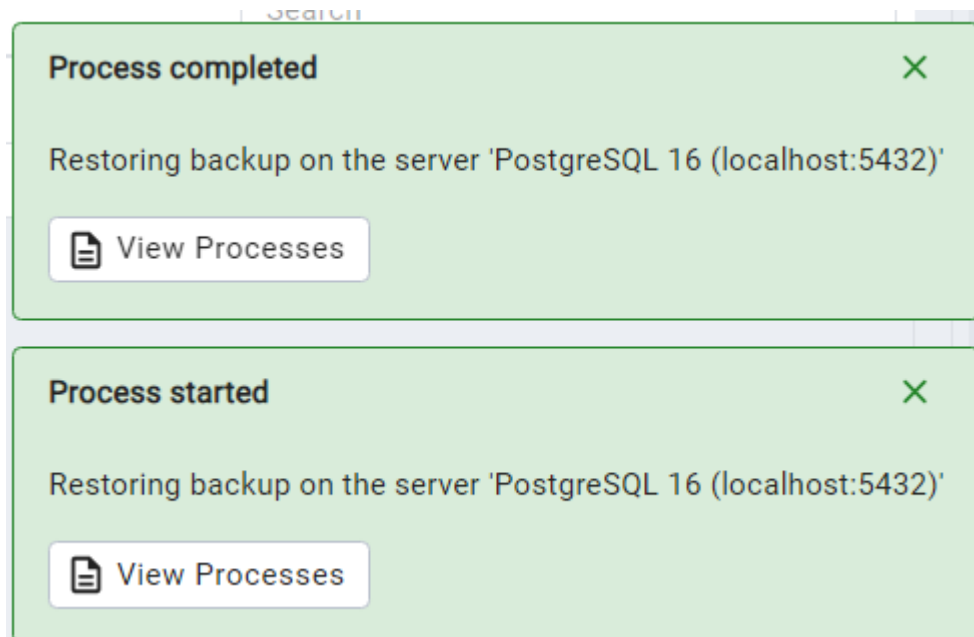
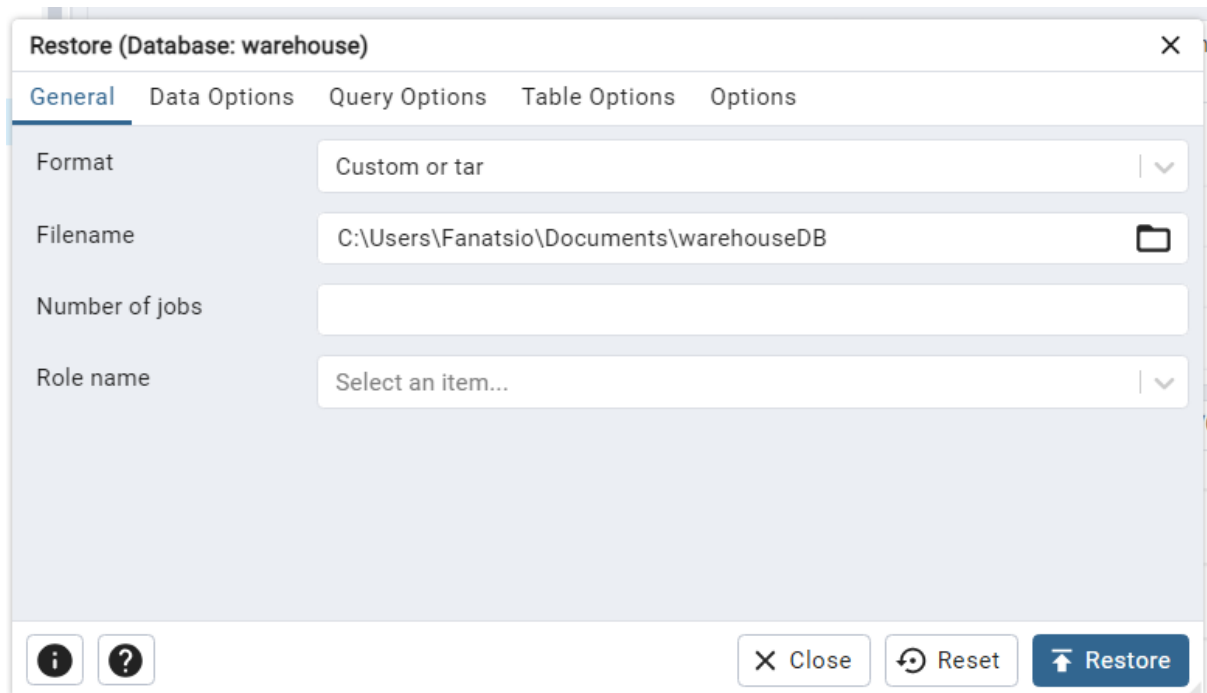
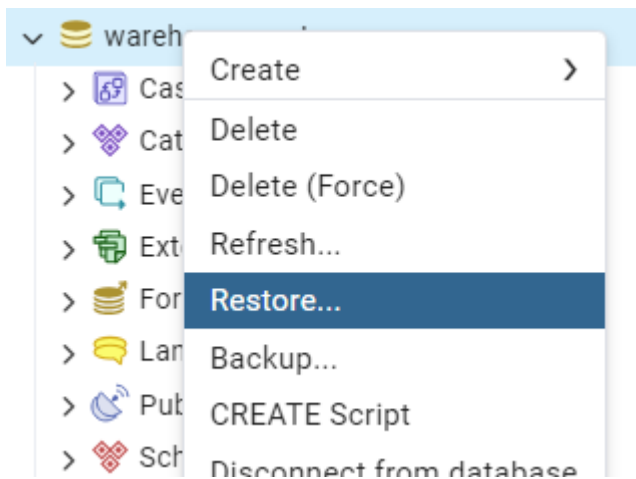


Задание 7. Удалите базу данных.

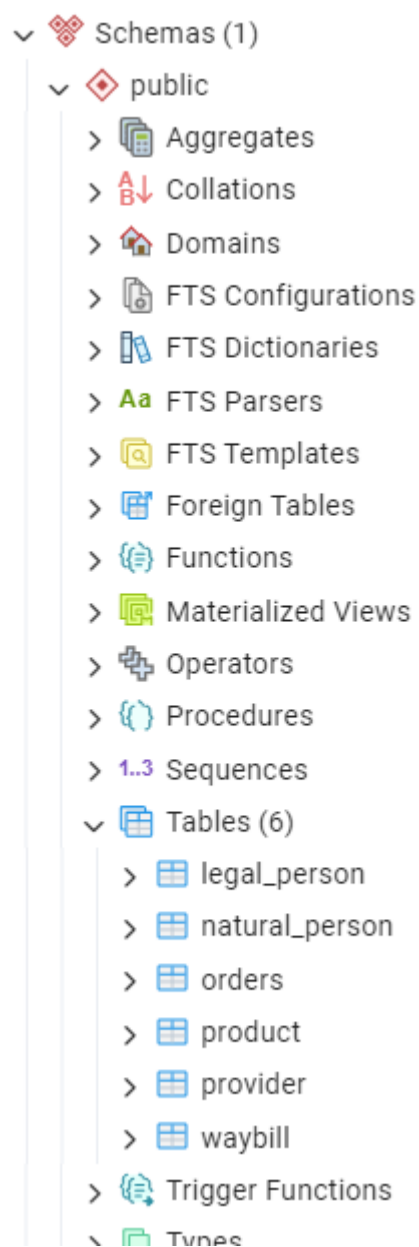


Далее создадим базу данных “warehouse_restore”, чтобы в неё ВОССТАНОВИТЬ удаленную базу данных

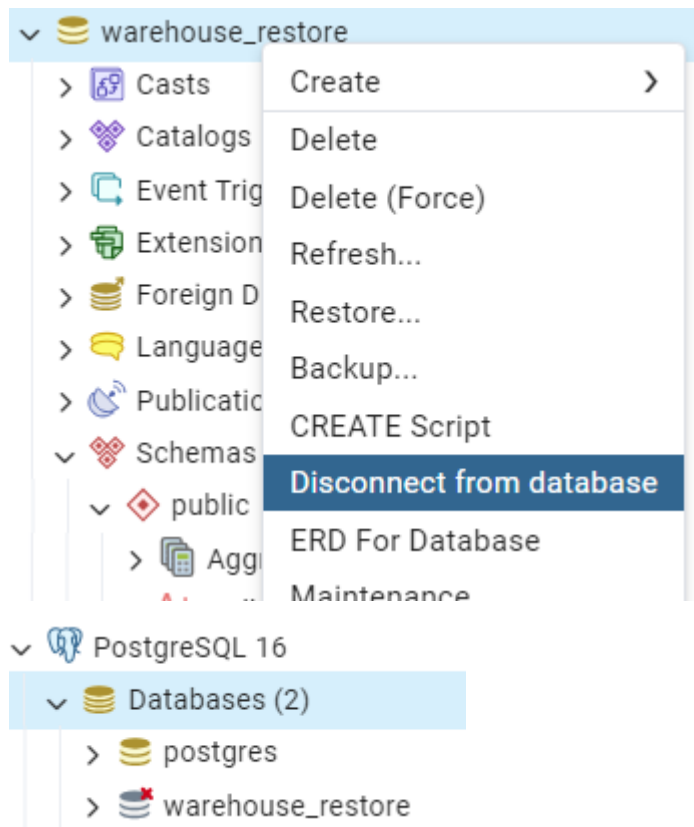
A screenshot of a 'Create -' dialog box in a database management tool. The dialog has a title bar with a close button. Below the title bar are tabs: 'General' (selected), 'Definition', 'Security', 'Parameters', 'Advanced', and 'SQL'. The 'General' tab contains four fields: 'Database' with the text 'warehouse_restore', 'OID' (empty), 'Owner' with a dropdown menu showing 'postgres', and 'Comment' (empty text area). At the bottom of the dialog are three buttons: 'Close', 'Reset', and 'Save'.



Проверяем наличие таблиц после восстановления



Задание 8. Открепите базу данных от сервера.



Далее прикрепили базу данных обратно к серверу

