Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Саратовский государственный технический университет

имени Гагарина Ю.А.»

Институт электронной техники и приборостроения

Кафедра Информационная безопасность автоматизированных систем

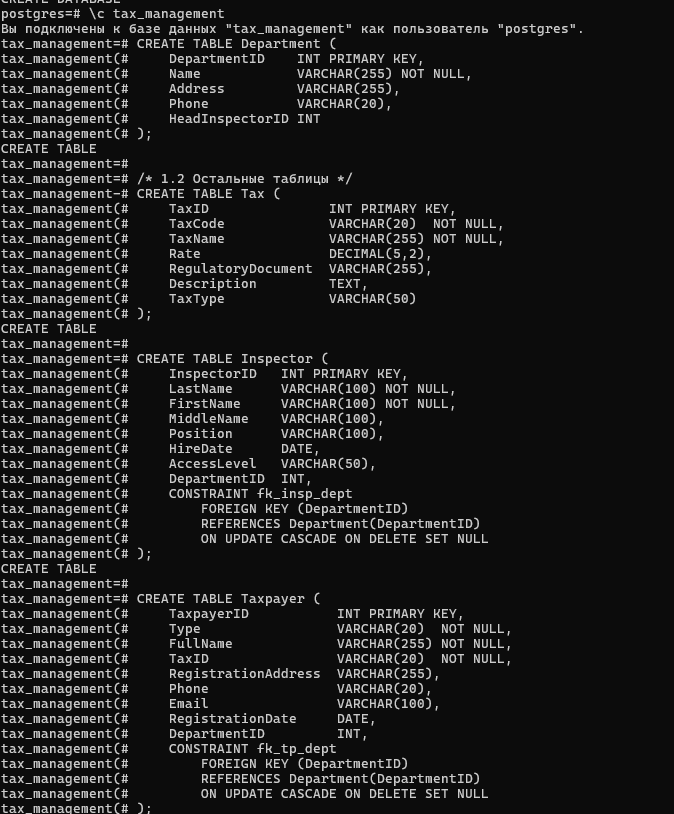
Специальность 10.05.03 Информационная безопасность автоматизированных систем

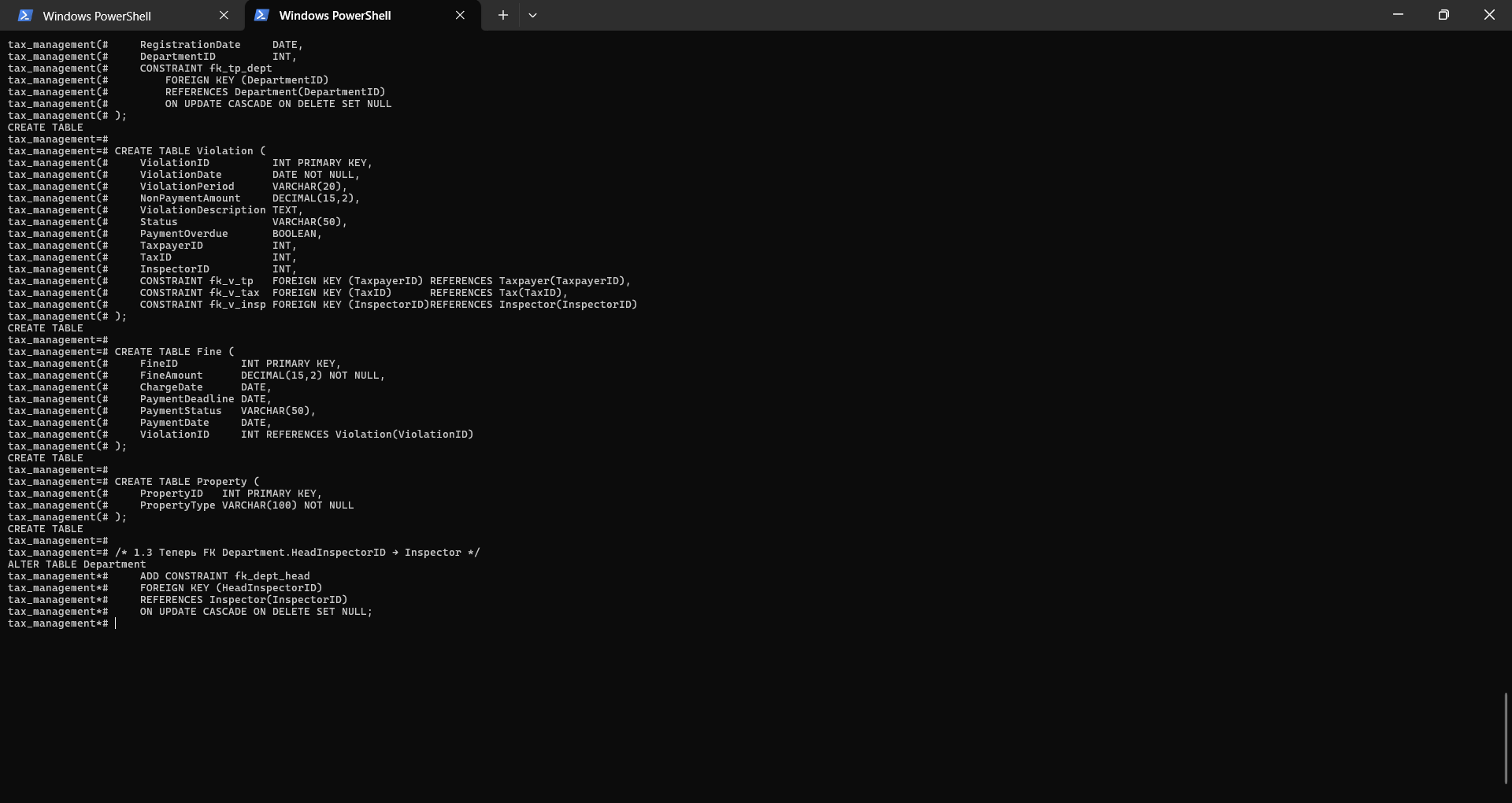
Лабораторная работа № 1

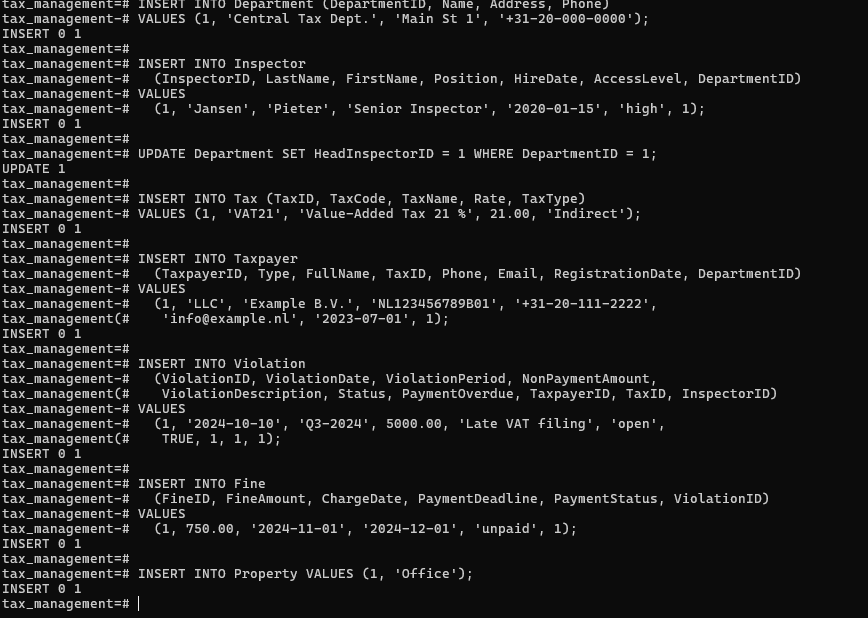
|  |  |
| --- | --- |
|  | Выполнил: студент 3 курса  учебной группы с-ИБС 31  очной формы обучения  Шлипакова А. М.  Проверил: доцент кафедры ИБС Ершов АС |

Саратов 2025

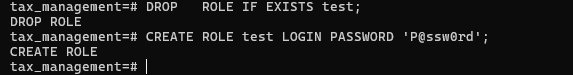
1. Создание базы данных. Создание и вставка данных.
2. Создание пользователя test с правами

****

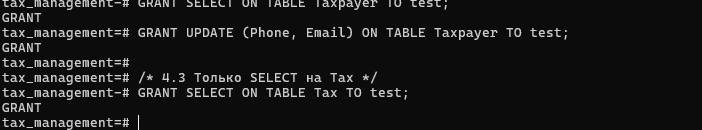




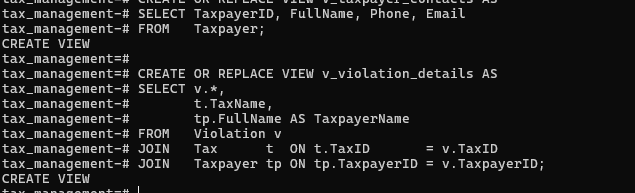
1. Полные права (SELECT, INSERT, UPDATE) на одну таблицу:



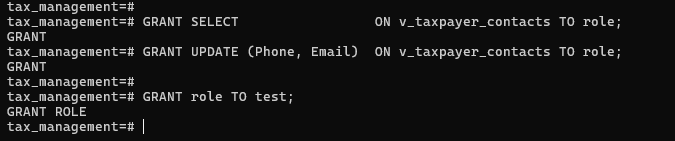
1. Ограниченные права (SELECT, UPDATE только избранных столбцов)



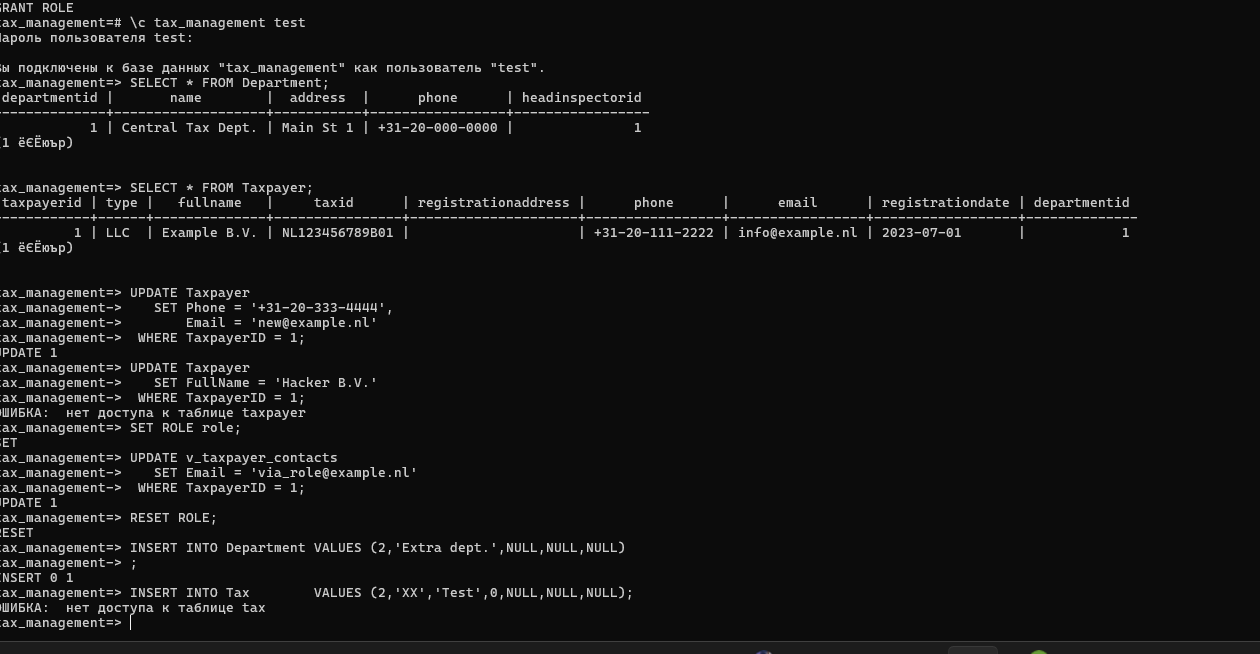
1. Только право SELECT на одну таблицу



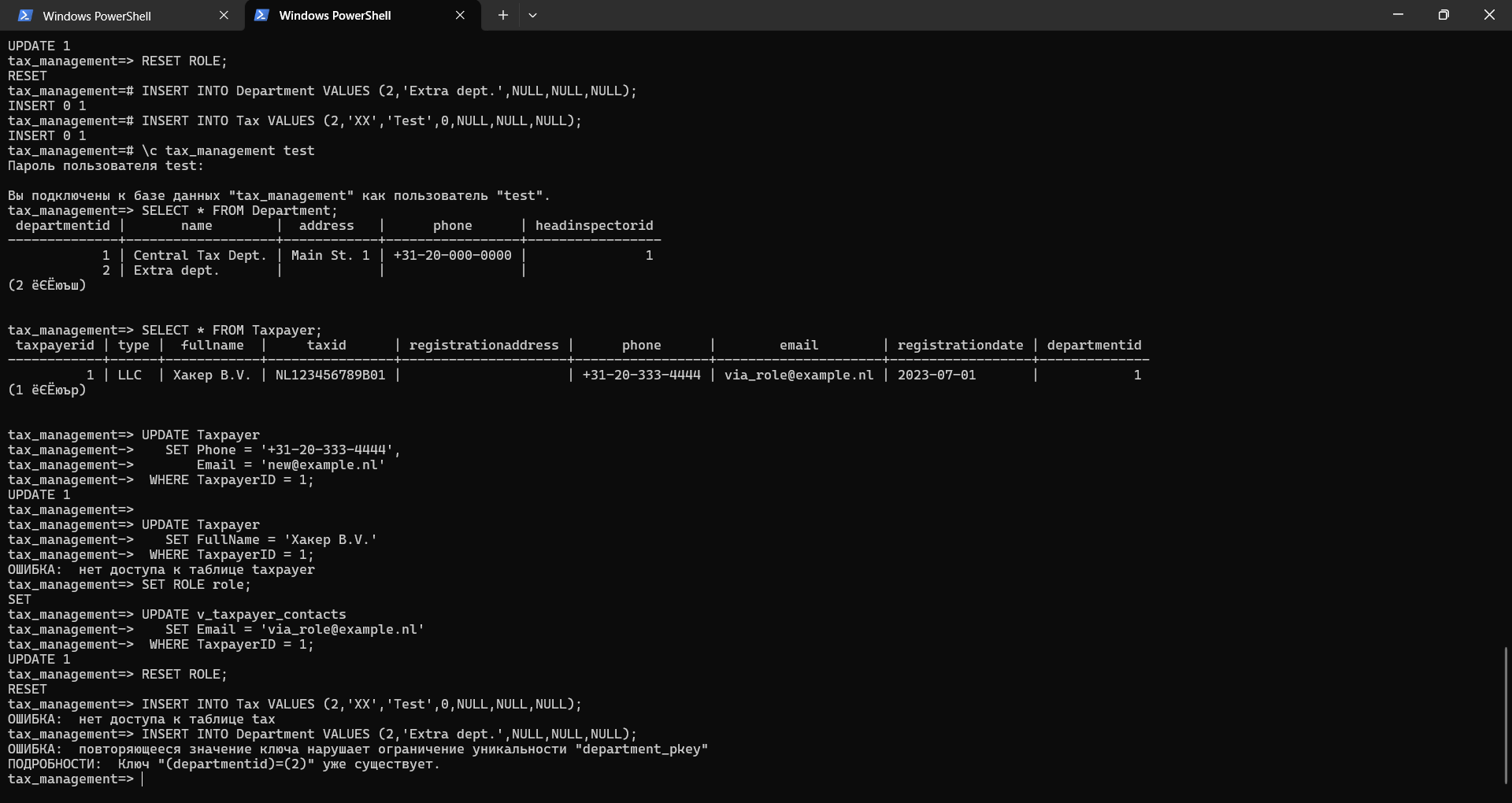
1. Создание представлений для таблиц



7. Создадим роль role и назначим ей права на обновление определенных столбцов одного из представлений.



8. Проверка прав доступа



**Ответы на вопросы**

• Для чего нужны роли?

Роли позволяют управлять правами доступа пользователей к объектам базы данных. Обеспечивают контроль над тем, кто может выполнять определенные действия в базе данных.

• Что такое схема?

Схема — это контейнер для объектов базы данных, таких как таблицы, представления и функции.  Схемы могут использоваться для изоляции объектов между различными приложениями или пользователями.

• Рассказать про директивы GRANT и REVOKE.

GRANT: используется для предоставления прав доступа к объектам базы данных. Например, можно предоставить пользователю право на выборку данных из таблицы.

REVOKE: используется для отзыва ранее предоставленных прав. Например, можно отозвать право на выборку данных у пользователя.

• Для чего нужна роль PUBLIC?

Роль PUBLIC автоматически присваивается всем пользователям базы данных.  Позволяет назначать права, которые будут доступны всем пользователям, что упрощает управление доступом к общим объектам.

• Как добавить нового пользователя в текущую базу данных?

CREATE USER для создания нового пользователя.

CREATE USER new\_user WITH PASSWORD 'user\_password';

• Как позволить пользователю заходить на сервер?

Для этого необходимо создать пользователя и предоставить ему необходимые права доступа. Можно использовать команду GRANT для предоставления прав на подключение к базе данных:

GRANT CONNECT ON DATABASE database\_name TO new\_user;

• Какие существуют права?

SELECT: Право на выборку данных из таблиц.

INSERT: Право на вставку новых данных в таблицы.

UPDATE: Право на изменение существующих данных.

DELETE: Право на удаление данных из таблиц.

• Сменить владельца базы данных.

Для смены владельца базы данных используется команда ALTER DATABASE.

ALTER DATABASE database\_name OWNER TO new\_owner;

• Сменить пароль для пользователя.

Используйте команду ALTER USER для изменения пароля пользователя.

ALTER USER user\_name WITH PASSWORD 'new\_password';

• Определить роль с заданными правами.

Для создания роли с определенными правами используется команда CREATE ROLE.

CREATE ROLE role\_name WITH LOGIN PASSWORD 'role\_password';

GRANT SELECT, INSERT ON table\_name TO role\_name;

• Рассказать о CHECK OPTION.

CHECK OPTION используется в представлениях для обеспечения того, чтобы все изменения данных через представление соответствовали условиям, определенным в представлении. Если представление фильтрует данные, то любые вставки или обновления через это представление должны соответствовать фильтру.

• Рассказать о модификации данных через представления.

Данные в базах данных могут быть изменены через представления, если они определены с возможностью обновления. Не все представления поддерживают модификацию данных, особенно если они содержат агрегатные функции или соединения.

• Рассказать о вставке данных через представления.

Данные могут быть вставлены в базу данных через представления, если представление поддерживает операцию вставки.

Пример: Если представление основано на одной таблице и не содержит ограничений, можно использовать команду INSERT:

INSERT INTO view\_name (column1, column2) VALUES (value1, value2);