**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ «СИНЕРГИЯ»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | Информационные системы и технологии |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

**Лабораторный практикум № 9**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **по дисциплине** |  | Администрирование информационных систем | | |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Обучающийся** |  | Пахомов Антон Юрьевич |  |  |
|  |  | (ФИО) |  | (подпись) |
| **Группа** |  | **ВБИо-307рсоб** |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  | Сибирев Иван Валерьевич |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва 2024 г**

**Задание**

1. Создать пользовательские роли и прописать их доступ к объектам баз данных.
2. Определить объекты базы данных только для чтения (SELECT).
3. Определить изменяемые объекты базы данных (INSERT, UPDATE, DELETE) и частоту их корректировки.
4. Создать пользователей и сделать их членами указанных ролей.

**Ход работы**

Выполним создание базы данных, а уже затем настроим к ней доступ для пользователей. В обозревателе решений выберем ПКМ Базы данных и нажмем создать новую БД, рисунок 1.

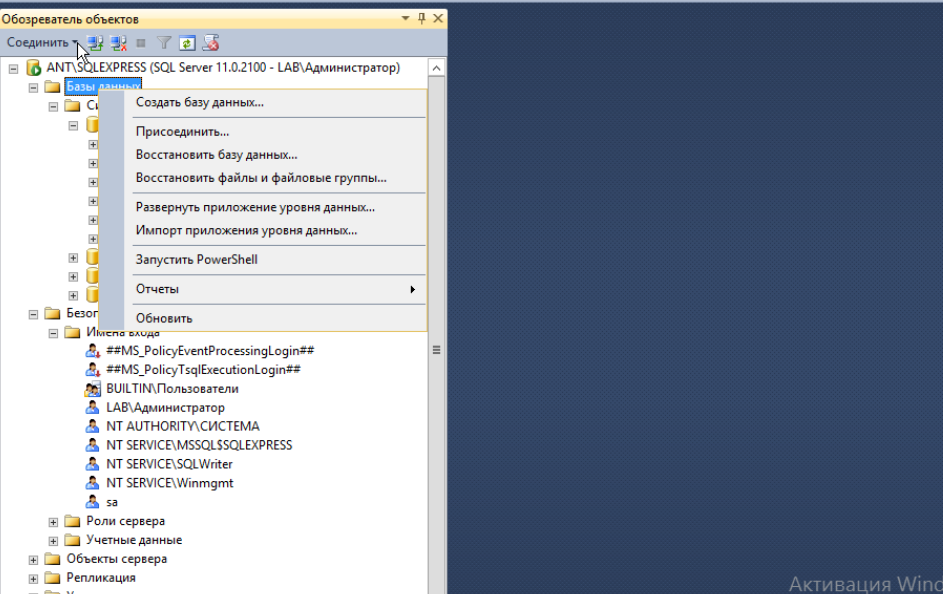


Рисунок 1 − Создание новой базы данных.

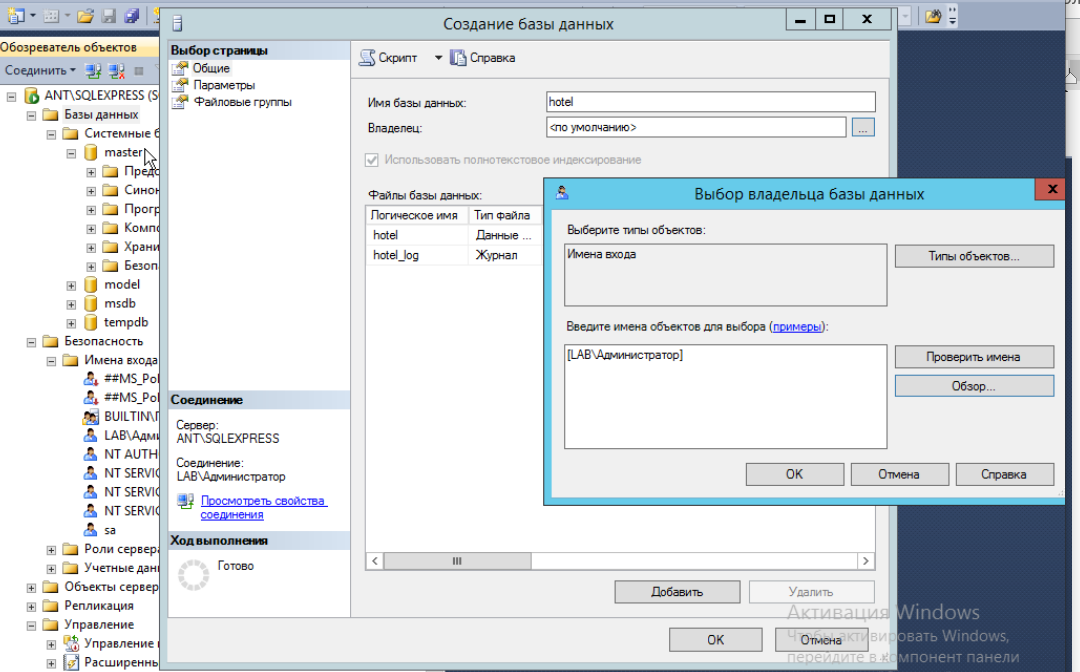
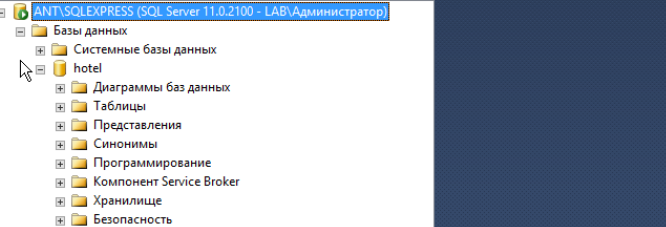
Зададим имя базе данных и владельца, тем самым установим первую роль, рисунок 2. 

Рисунок 2 − Создание новой базы данных.

Файловая группа — это логическое объединение файлов данных, позволяющее администраторам управлять всеми файлами группы как единым целым. Возможность контроля физического размещения отдельных объектов в базе данных дает целый ряд преимуществ в отношении управляемости и производительности. Например, создав несколько файловых групп, можно контролировать физическое сохранение содержимого базы данных на запоминающих устройствах и отделять перезаписываемые данные от доступных только для чтения.

Теперь новая база данных отображается на панеле, причем видны папки с ее таблицами, представлениями и т.д, рисунок 3.

 Рисунок 3 − Новая база данных.

**Анализ предметной области Гостиница.**

Описание предметной области:

База данных отелей предназначена для управления информацией о клиентах, бронированиях, номерах, сотрудниках и других аспектах работы отеля. Основные сущности предметной области включают:

* Клиенты: Информация о клиентах, которые бронируют номера.
* Бронирования: Информация о бронированиях, включая даты заезда и отъезда, выбранные номера и стоимость.
* Номера: Информация о доступных номерах в отеле, их типе, стоимости и статусе.
* Сотрудники: Информация о сотрудниках отеля, их должностях и контактах.
* Типы номеров: Категории номеров (например, стандартный, люкс, апартаменты).
* Услуги: Дополнительные услуги, предоставляемые отелем (например, завтрак, трансфер, экскурсии).
* Платежи: Информация о платежах, связанных с бронированием.

Выявление связей:

* Клиент может сделать несколько бронирований.
* Бронирование связано с одним или несколькими номерами.
* Номер может быть связан с несколькими бронированиями (если он забронирован на разные даты).
* Сотрудник может быть связан с несколькими бронированиями (например, обслуживание клиентов).
* Тип номера определяет характеристики номера.
* Услуги могут быть связаны с бронированием или клиентом.
* Платежи связаны с бронированием.

Согласно схеме, на рисунке 4, создадим таблицы БД.

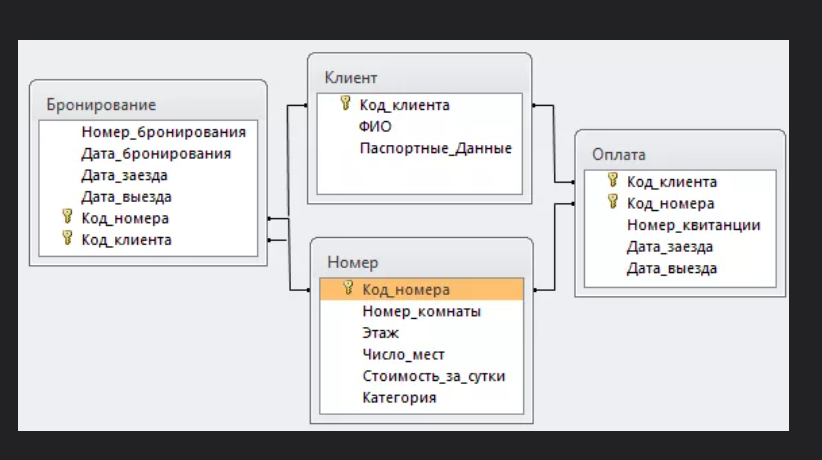


Рисунок 4 − Схема БД.

Создадим таблицу «Клиенты» рисунок 5.

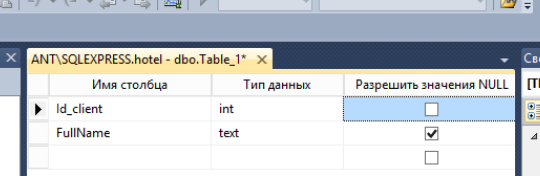


Рисунок 5 − Таблица «Клиенты».

Создадим таблицу «Бронирования», рисунок 6.

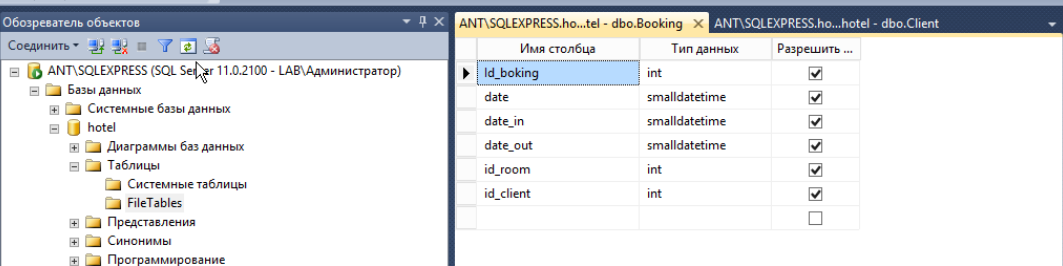


Рисунок 6 − Таблица «Бронирования».

Создадим таблицу «Комнаты», рисунок 7.

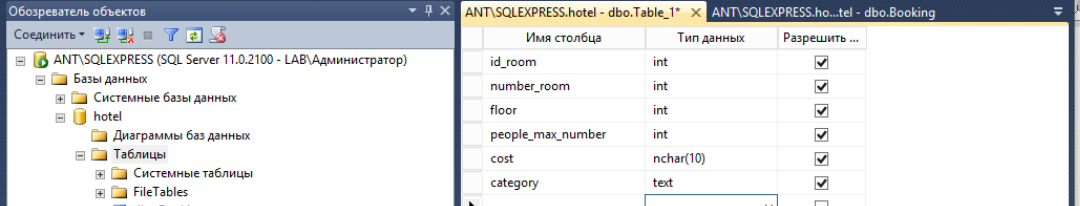


Рисунок 7 − Таблица «Комнаты».

Создадим последнюю таблицу «Оплата», рисунок 8.

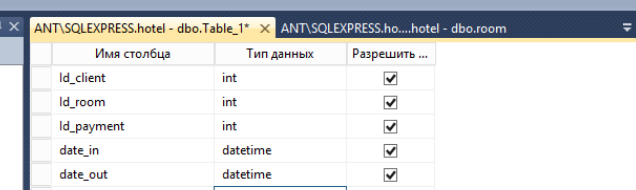


Рисунок 8 − Таблица «Оплата».

Все созданные таблицы теперь отображаются, рисунок 9.

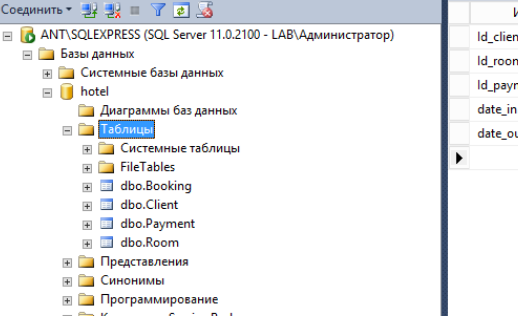


Рисунок 9 − Созданные таблицы.

**Установка серверных ролей**

Для того чтобы установить роли необходимо перейти в раздел безопасность – роли, в окне будут показаны все роли, которые могут быть использованы на сервере, рисунок 10.

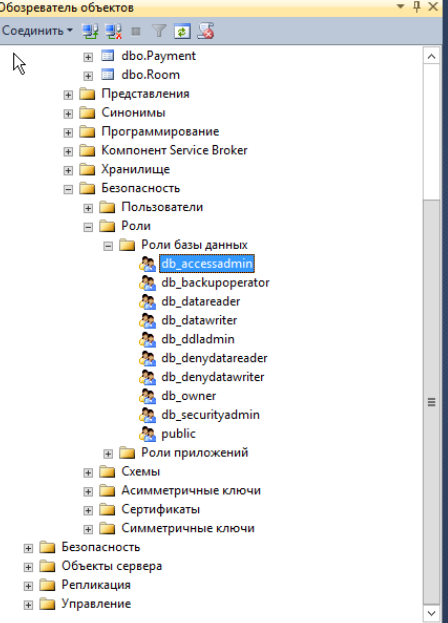


Рисунок 10 − Просмотр существующих ролей сервера.

Роль может быть присвоена нескольким пользователям, рисунок 11.

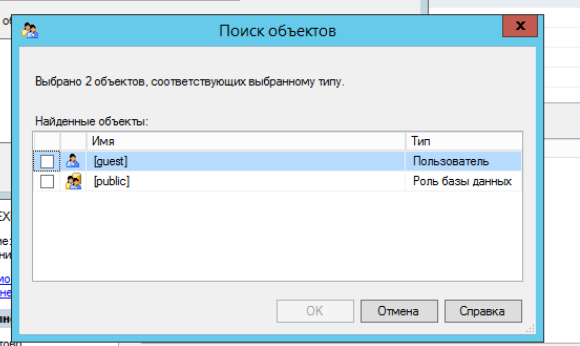


Рисунок 11 − Роли пользователей.

Нажав на роль ПКМ, выберем свойства. На вкладке «защищаемые объекты», предоставляется разршения на совершение некоторых действий с базой данных для опредлнных ролей, рисунок 12.

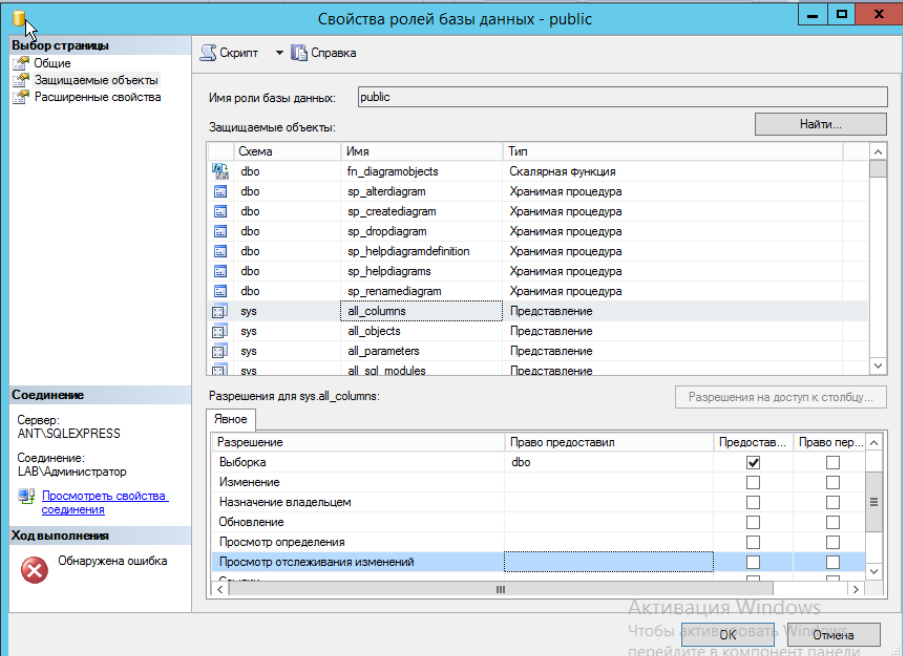


Рисунок 12 − Свойства ролей базы данных.

Серверные роли также можно добавлять и удалять. Создадим свою роль, определим кто ей будет наделен, рисунок 13.

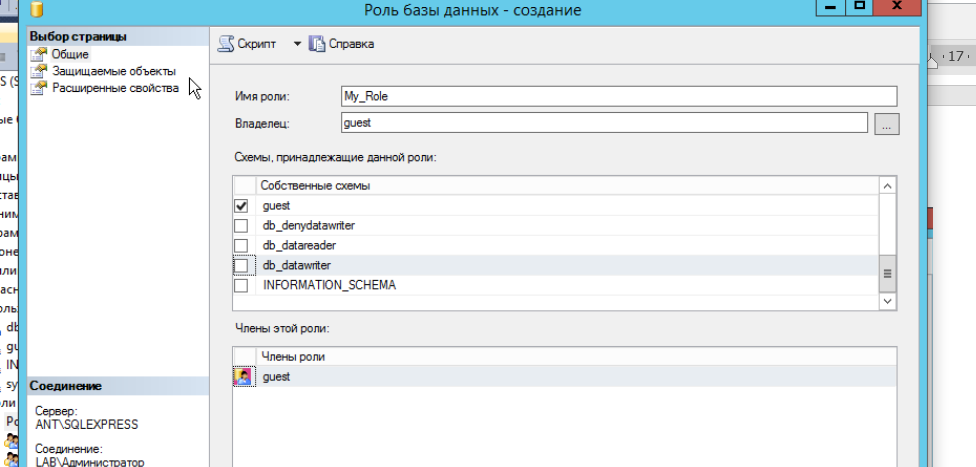


Рисунок 13 − Создание своей роли.

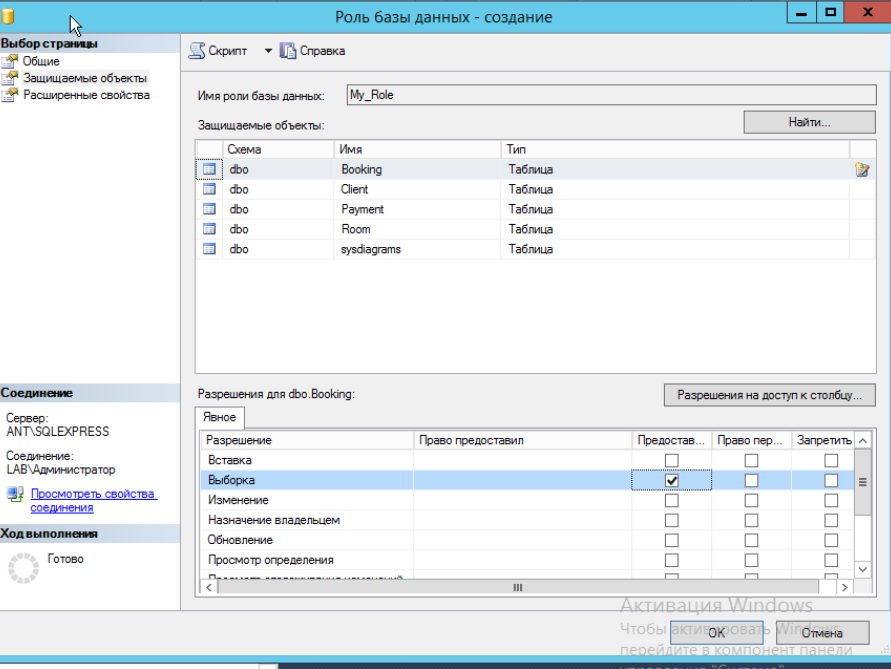


Рисунок 14 − Определение прав роли.

Определим защищаемые объекты (таблицы), а также определим, что данной роли можно делать с этими объектами, а что нельзя (создадим разрешение для пользователя (гость) делать выборку из БД), рисунок 14. Роль была успешно создана, рисунок 15.

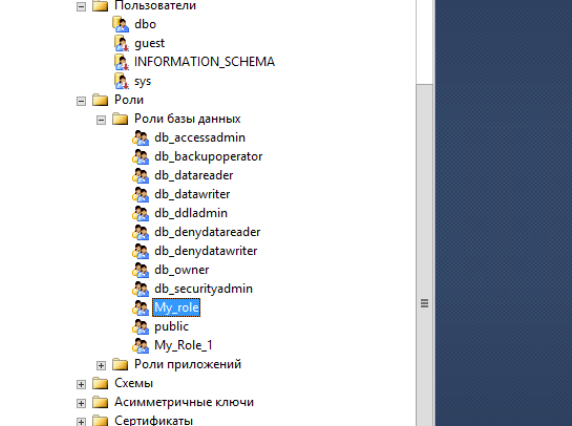


Рисунок 15 − Созданные роли.

**Определим объекты базы данных только для чтения (SELECT).**

**Для предоставления доступа только для чтения** в **Обозревателе объектов**необходимо раскрыть таблицы или представления, перейти в свойства, разрешения и выбрать роль или пользователя и предоставить SELECT, выберем таблицу Booking, рисунок 16.

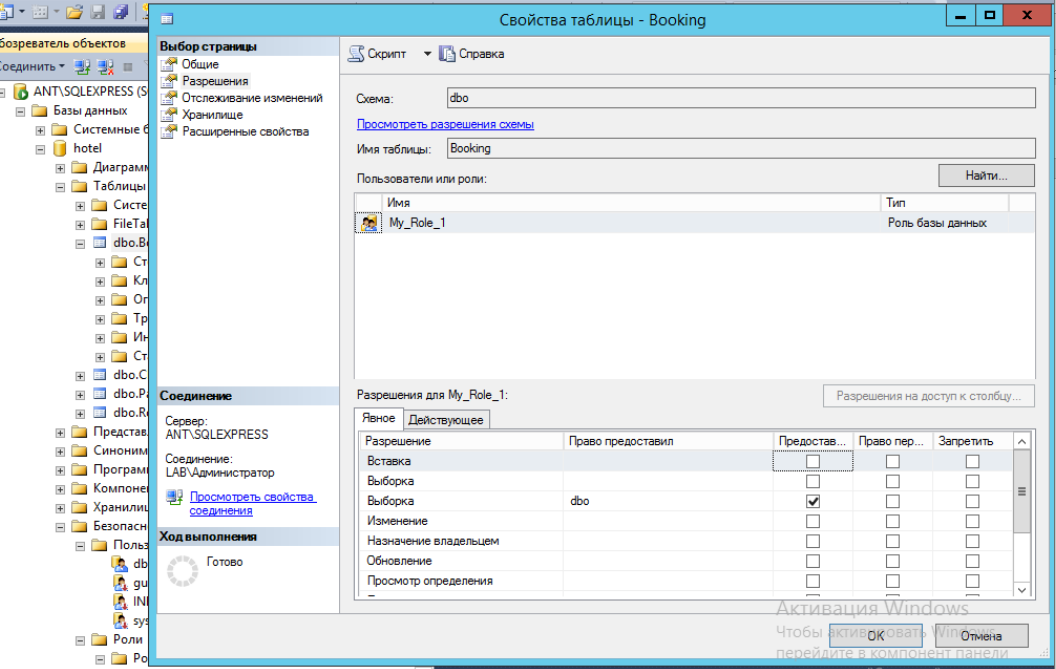


Рисунок 16 − Определение параметров только для чтения для таблицы Booking.

Выберем для роли My\_Role\_1 все свойства которые разрешают только просмотр и выборку SELECT, рисунок 17

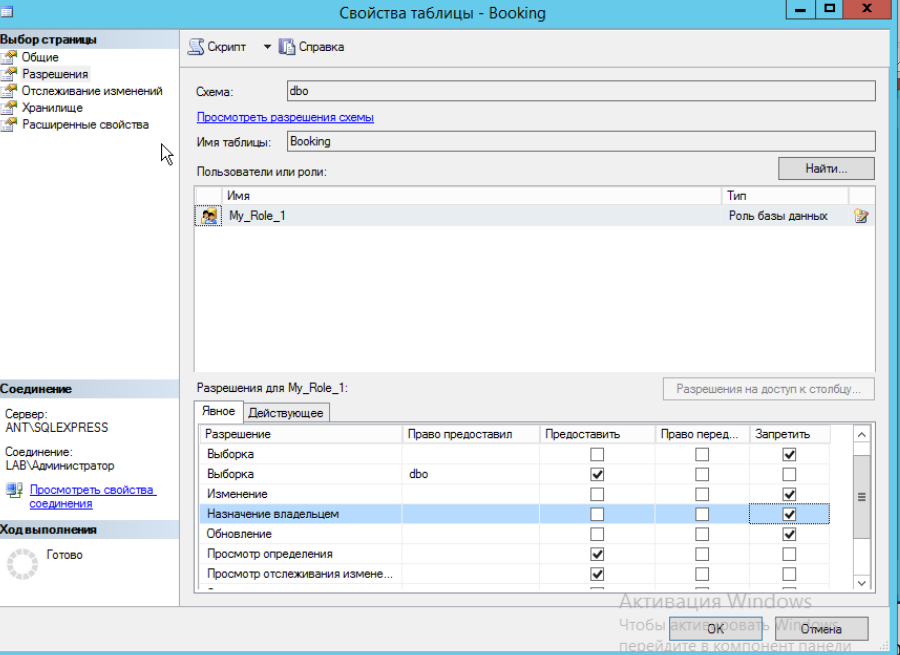


Рисунок 17 − SELECT свойства.

**Для таблицы Payment определим свойства для группы My\_Role\_1 UPDATE, DELETE, INSERT.**

Частоту корректировки можно задавать на анализе системных представления, таких как sys.dm\_db\_index\_usage\_stats и sys.dm\_db\_index\_operational\_stats, чтобы получить информацию о количестве операций INSERT, UPDATE и DELETE для таблиц и индексов, например, выполнив следующий запрос:

SELECT

OBJECT\_NAME(s.object\_id) AS TableName,

i.name AS IndexName,

s.user\_updates AS TotalUpdates,

s.user\_seeks + s.user\_scans + s.user\_lookups AS TotalReads,

s.user\_seeks,

s.user\_scans,

s.user\_lookups,

s.user\_updates

FROM

sys.dm\_db\_index\_usage\_stats s

JOIN

sys.indexes i ON s.object\_id = i.object\_id AND s.index\_id = i.index\_id

WHERE

s.database\_id = DB\_ID()

ORDER BY

s.user\_updates DESC;

Благодаря запросу, можно определить сколько раз были выполнены операции UPDATE (и другие операции) для каждой таблицы. Зададим роли на удаление и обновление, рисунок 18.

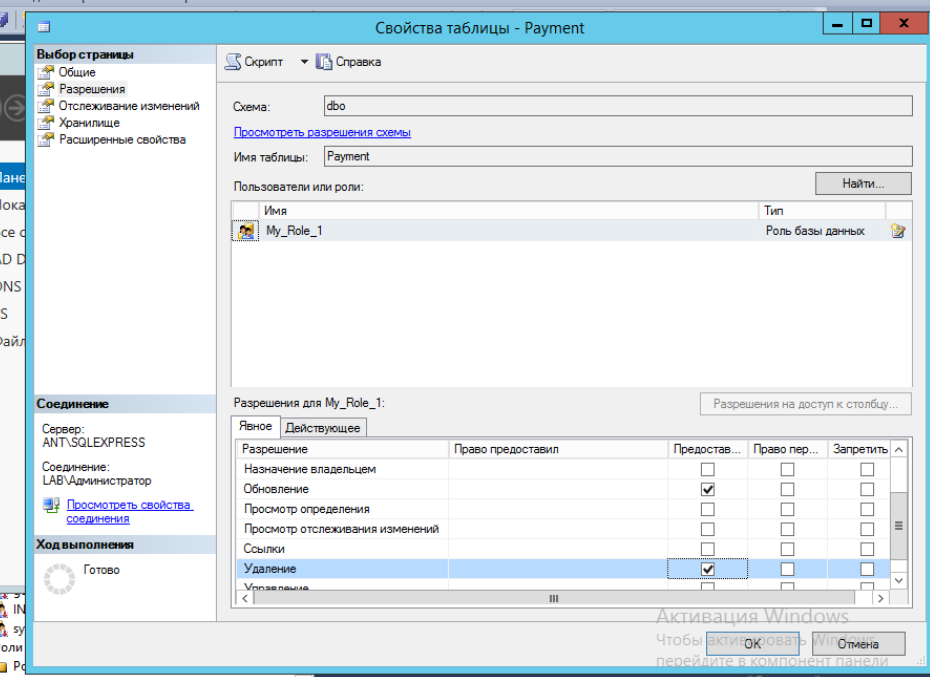


Рисунок 18 − Разрешение на обновление и удаление.

**Создадим пользователей и присвоим им роли, рисунок 19 (а,б).**

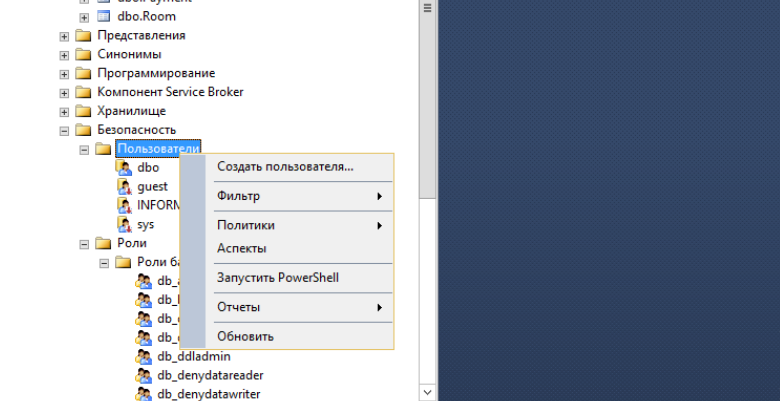


Рисунок 19 (а) − Создание нового пользователя.

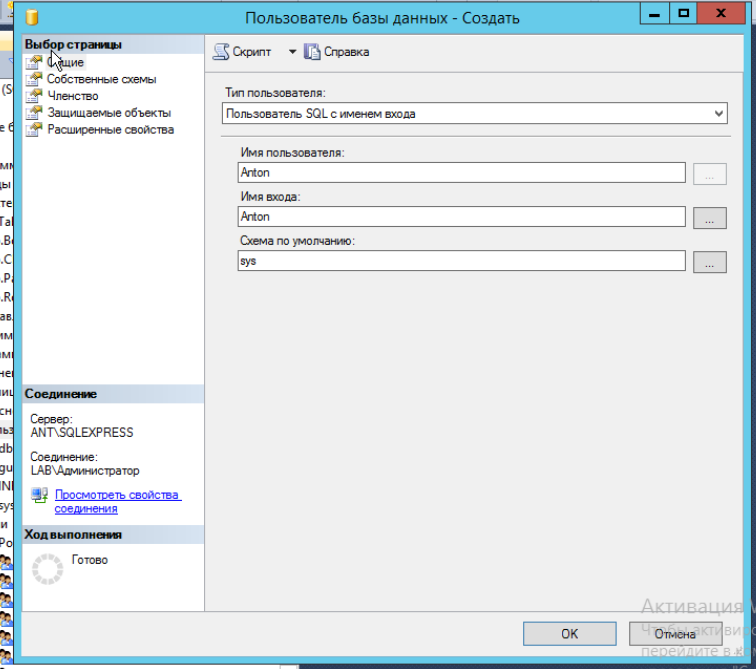


Рисунок 19 (б) − Создание пользователя (продолжение).

И аналогично предыдущему шагу, добавляем пользователя в роли , рисунок 20

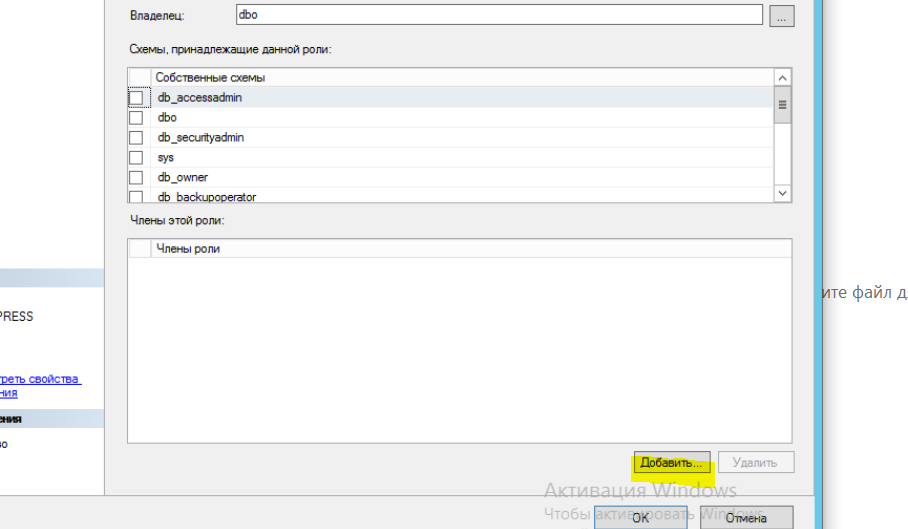
.

Рисунок 20 − Присвоение роли новому пользователю.

**Вывод:** в результате работы мы научились добавлять таблицы в базу данных, определять объекты базы данных только для чтения, обновления, удаления и вставки, научились создавать пользователей и присваивать им роли.