**Санкт-Петербургское государственное бюджетное профессиональное   
образовательное учреждение «Колледж метростроя»**

**ОТЧЕТ**

**по учебной практике**

Выполнил студент группы № 39 Михеев Н.С.

(фамилия, инициалы)

(код и наименование профессии (специальности))

Заключение и оценка

Руководителя практики

от организации \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(отлично, хорошо, удовлетворительно)

Руководитель практики от организации \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (должность) (фамилия, инициалы)

М.П. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2024 г.

(роспись)

Оценка руководителя практики от колледжа \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(отлично, хорошо, удовлетворительно)

Руководитель практики от колледжа \_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(роспись) (фамилия, инициалы)

Дата сдачи отчета \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись обучающегося\_\_\_\_\_\_\_\_\_\_\_\_

САНКТ–ПЕТЕРБУРГ

2024

Оглавление

[Введение 3](#_Toc185320495)

Тема №1. Создание клиент-серверной архитектуры в среде виртуализации…4

Тема №2. Установка MS SQL Server и Management Studio…………………..13

Тема №3. Постановка задачи разработки БД. Выделение сущностей предметной области……………………………………………………………..17

Тема №4. Анализ информационных задач и круга пользователей БД. Логическое проектирование и нормализация БД……………………………..18

Тема №5. Физическое проектирование БД……………………………………20

Тема №6. Работа с транзакциями и операторами процедурной логики…….27

Тема №7. Создание хранимых процедур, пользовательских функций и триггеров………………………………………………………………………...3[1](#_Toc185320496)

Заключение……………………………………………………………………….33

Список литературы………………………………………………………………34

### Введение

Современные информационные технологии играют ключевую роль в функционировании различных организаций, обеспечивая хранение, обработку и передачу данных. Базы данных являются основным компонентом любой информационной системы, так как они позволяют эффективно организовать и управлять большими объемами данных. С развитием технологий и ростом требований к обработке информации, важность разработки, администрирования и защиты баз данных становится все более очевидной.

Разработка базы данных включает в себя проектирование структуры хранения данных, создание схемы базы данных, а также настройку механизма взаимодействия с пользователями и другими системами. Администрирование баз данных связано с обеспечением их бесперебойной работы, производительности, а также своевременным выполнением операций по резервному копированию и восстановлению данных.

Не менее важной задачей является защита базы данных от несанкционированного доступа, потерь данных и внешних угроз. В условиях постоянно увеличивающихся объемов информации и роста числа киберугроз, защита данных и систем хранения становится приоритетной задачей для любой организации.

**Цели и задачи**

Изучение процессов разработки, администрирования и защиты баз данных, а также исследование методов и технологий, применяемых для их реализации. В ходе работы будут рассмотрены основные аспекты проектирования баз данных, их администрирования, а также внедрения эффективных мер безопасности для защиты данных от различных угроз.

**Разработка структуры базы данных**: проектирование схемы данных, создание таблиц, индексов и других объектов базы данных, соответствующих требованиям системы.

**Администрирование базы данных**: обеспечение эффективного функционирования базы данных, мониторинг ее производительности, настройка параметров и оптимизация запросов.

**Защита базы данных**: внедрение методов защиты от несанкционированного доступа, резервное копирование данных, настройка прав доступа, шифрование данных.

**Обеспечение отказоустойчивости**: внедрение механизмов для защиты от потерь данных при сбоях системы, таких как регулярные резервные копии и системы репликации.

**Тема №1. Создание клиент-серверной архитектуры в среде виртуализации**

В качестве серверной операционной системы для БД SQL Server будим использовать.

Windows Server 2019, в качестве операционных систем клиентов БД – Windows 10.

Установка WindowsServer 2019.

После того, как мы загрузились с виртуальной «флешки», приступаем к началу установки.

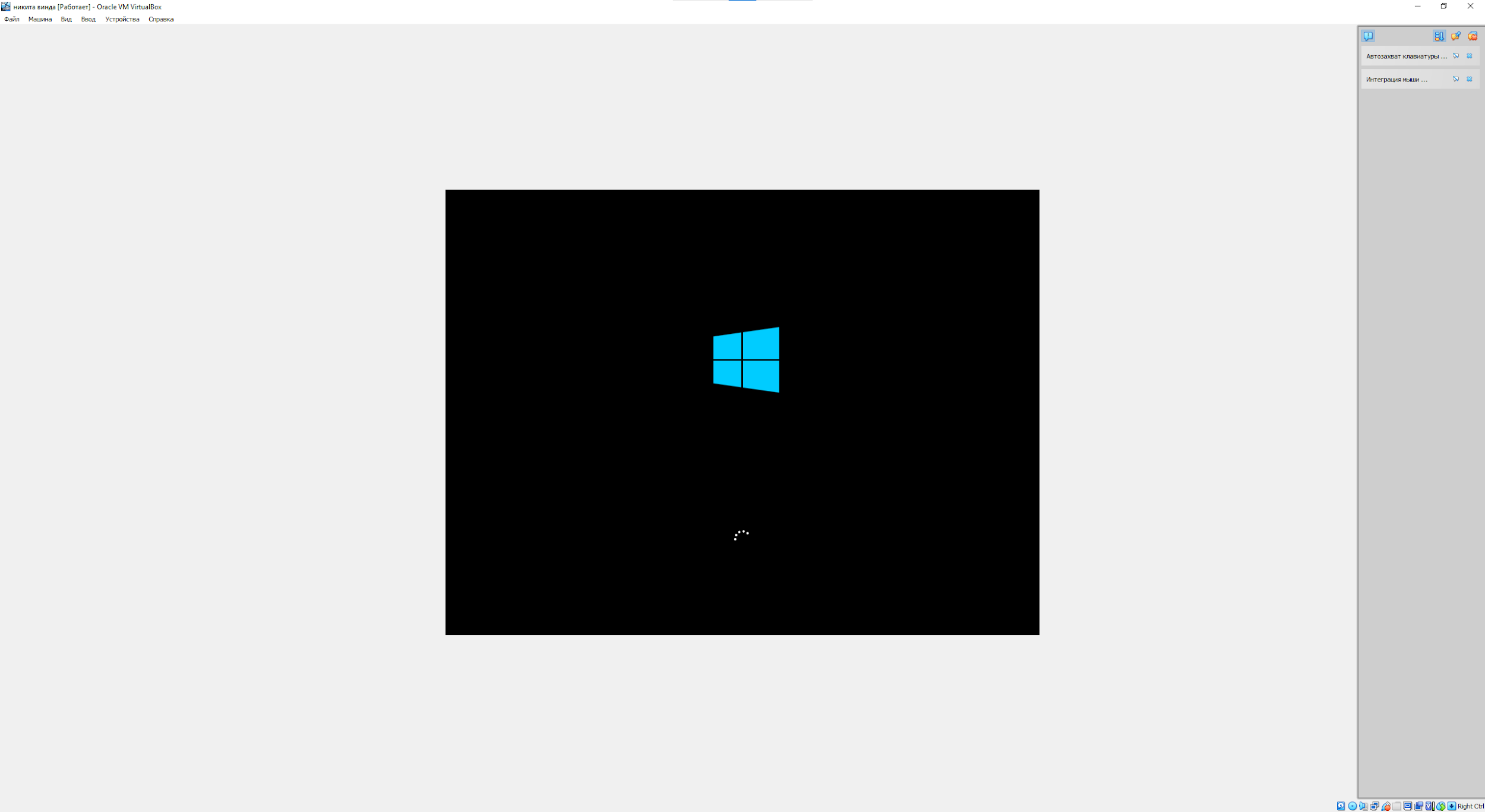


Рис.1 Начало установки

Выбираем язык операционной системы, выберем русский. Начинаем установку Windows Server 2019

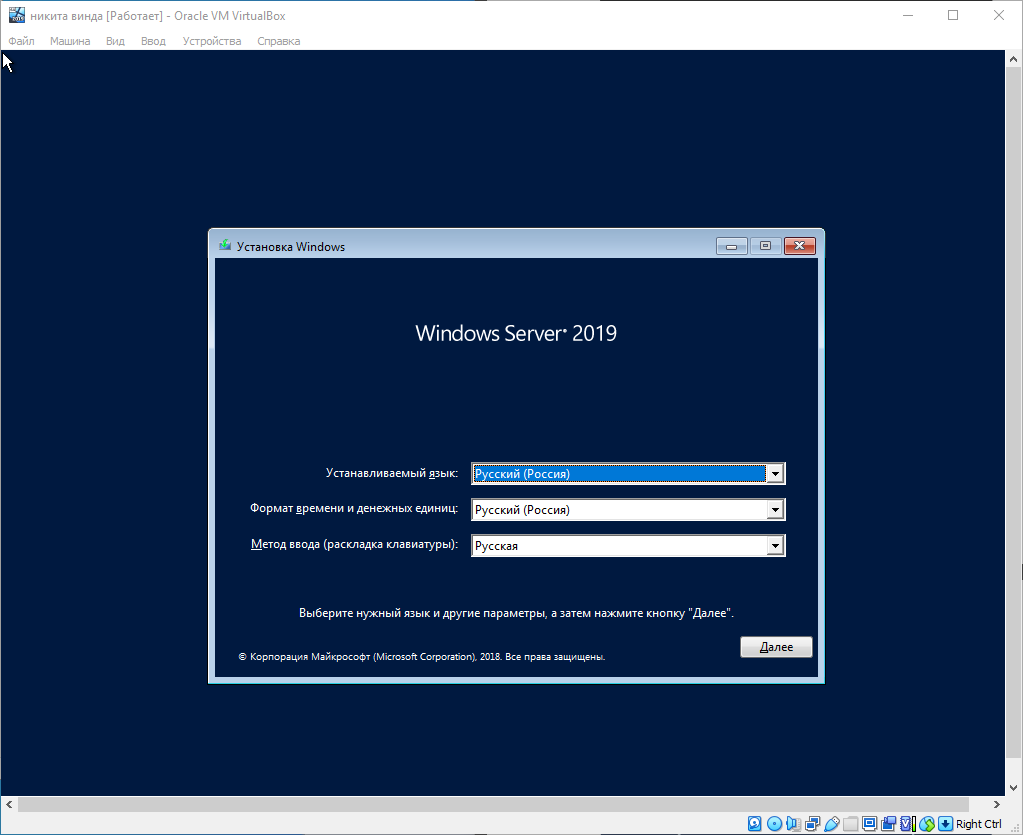


Рис.2 Выбор языка

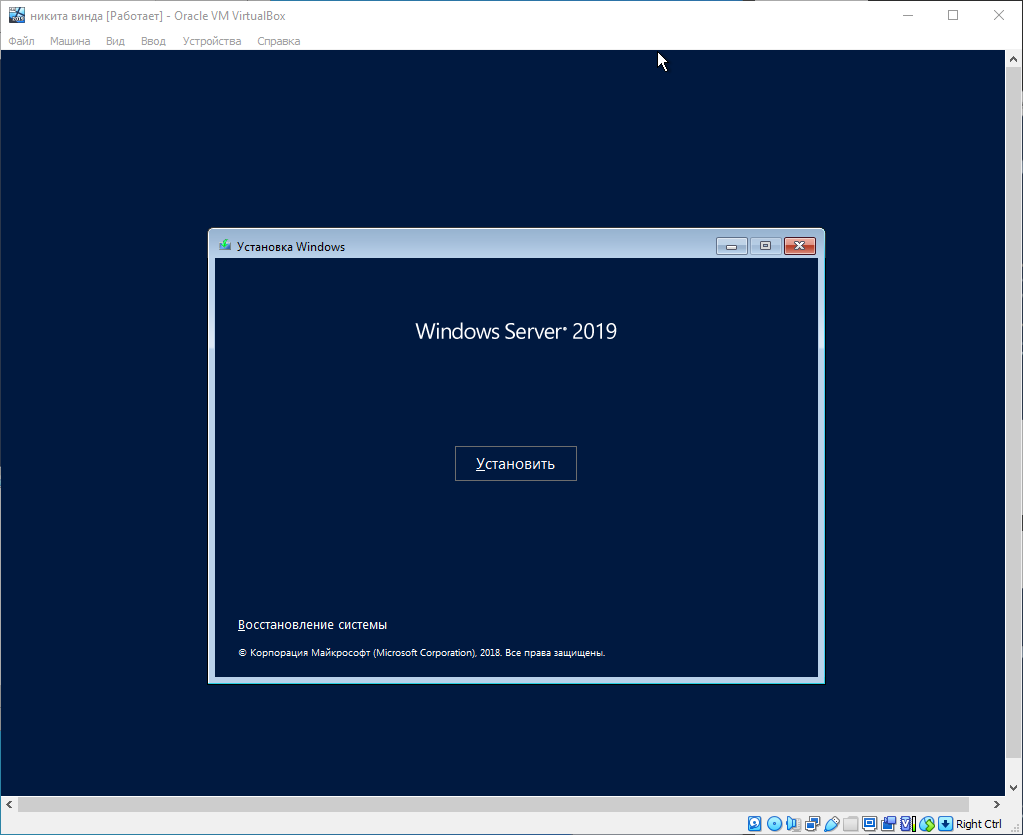


Рис.3 Установка Windows Server

Далее выбираем желаемую редакцию Windows Server 2019 для установки. Мы установим версию Standard, но Вы можете выбрать другую версию, к примеру Datacenter.

Порядок установки в ней аналогичный.

А так же, обязательно выбираете дистрибутив «Возможности рабочего стола», иначе у нас система установится без графической оболочки и управление операционной системой будет только из под консоли.

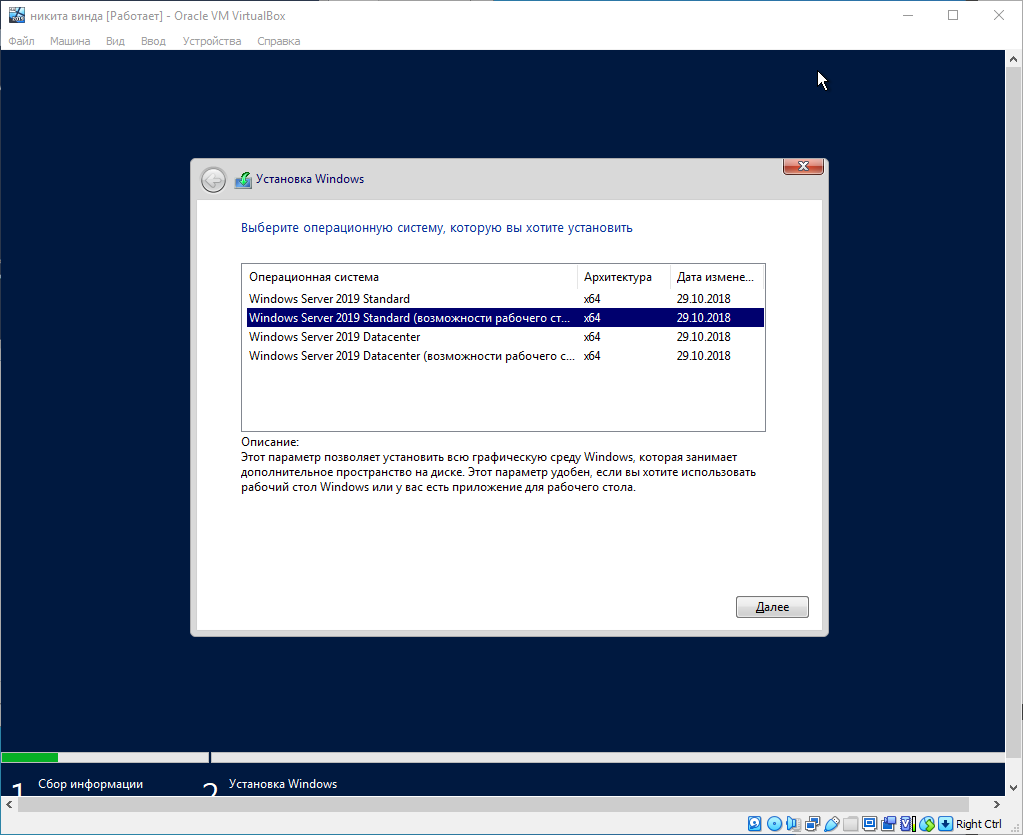


Рис.4 Выбор операционной системы

После соглашаемся со всеми условиями, задаем пароль и потом заходим под учетной записью администратора в систему.

В конечном итоге зашли в свой Windows Server 2019 (рис.5)

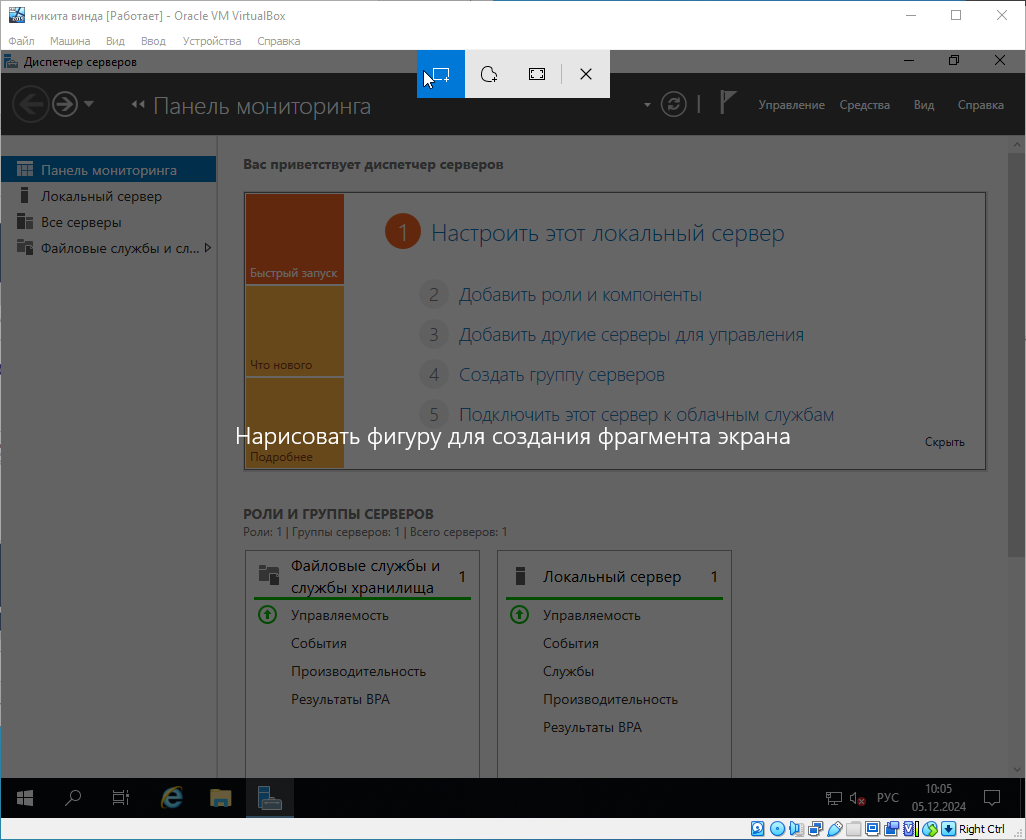


Рис.5 Главный экран при входе в Windows Server

Установим Windows 10 на предварительно подготовленную виртуальную машину.

Переходим к самой установке. Выбор языка и раскладки - будет первым окном, мы оставим как есть, на русском.

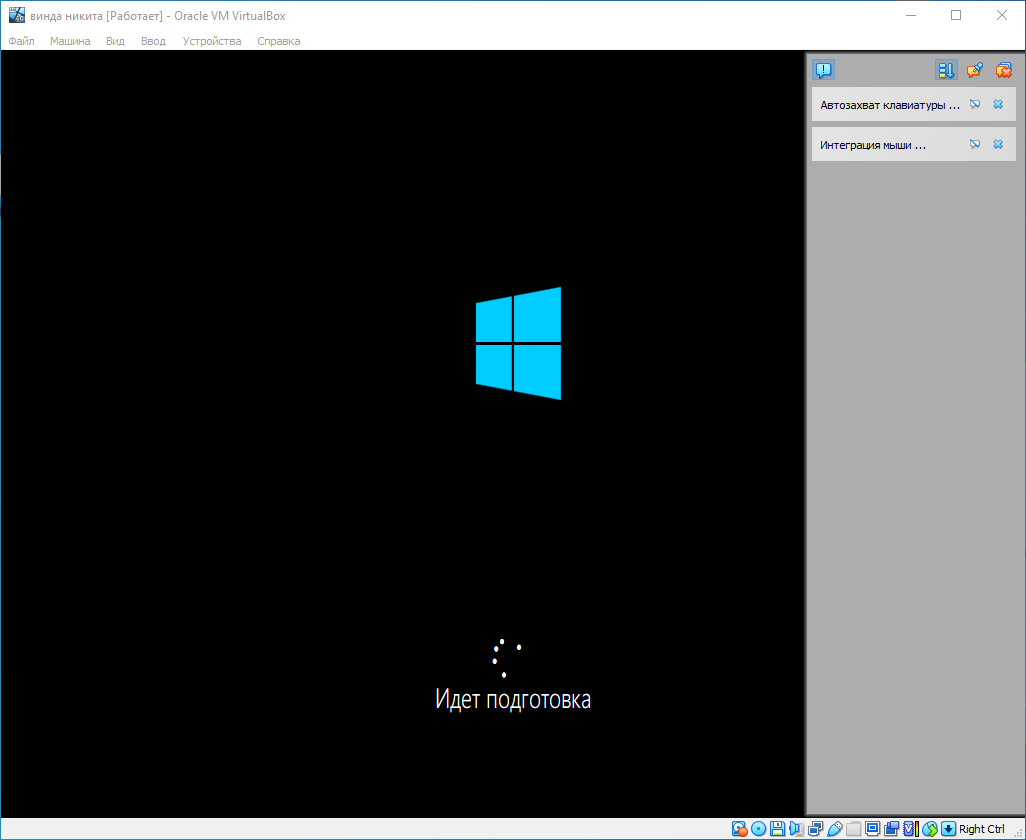


Рис.6 Подготовка к установке

Далее выбираю нужную мне версию Windows 10. Можете выбрать любую, по своему желанию. В моем примере буду устанавливать Windows 10 Pro, порядок установки у всех версий одинаковый.

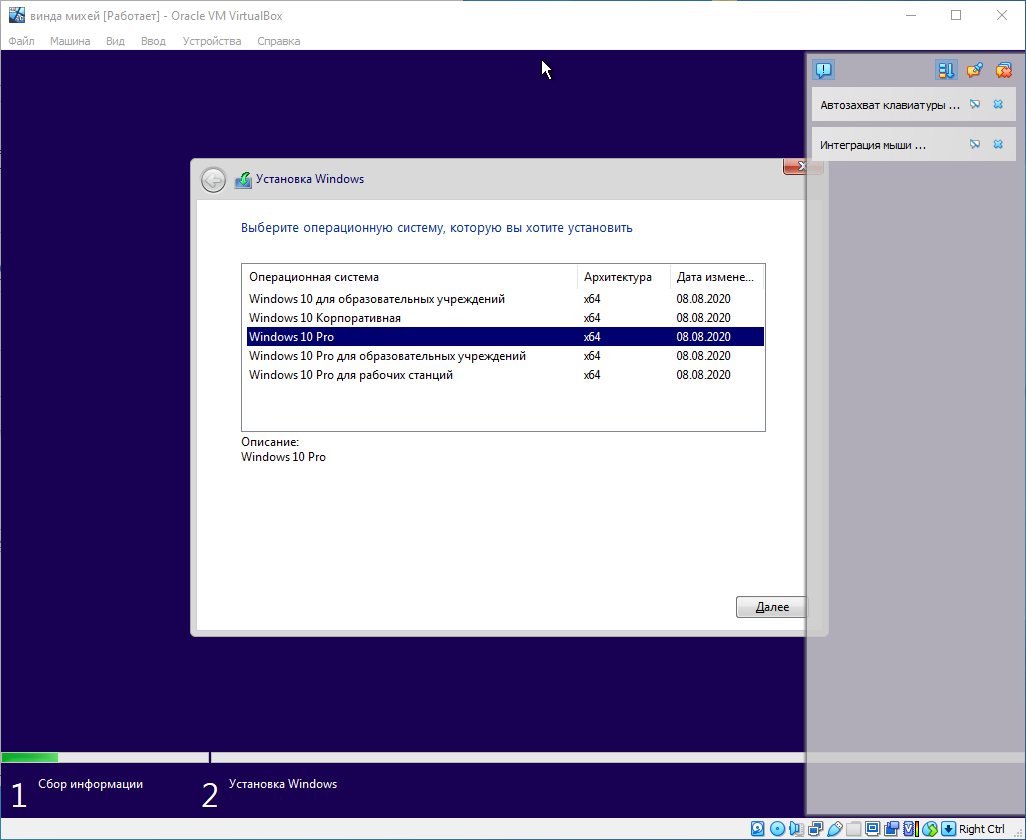


Рис.7 Выбор операционной системы

Когда сам процесс установки закончен, мне нужно сделать первоначальную настройку. У меня выйдет следующее окно, нужно выбрать регион, мы на примере выберем «Россия».

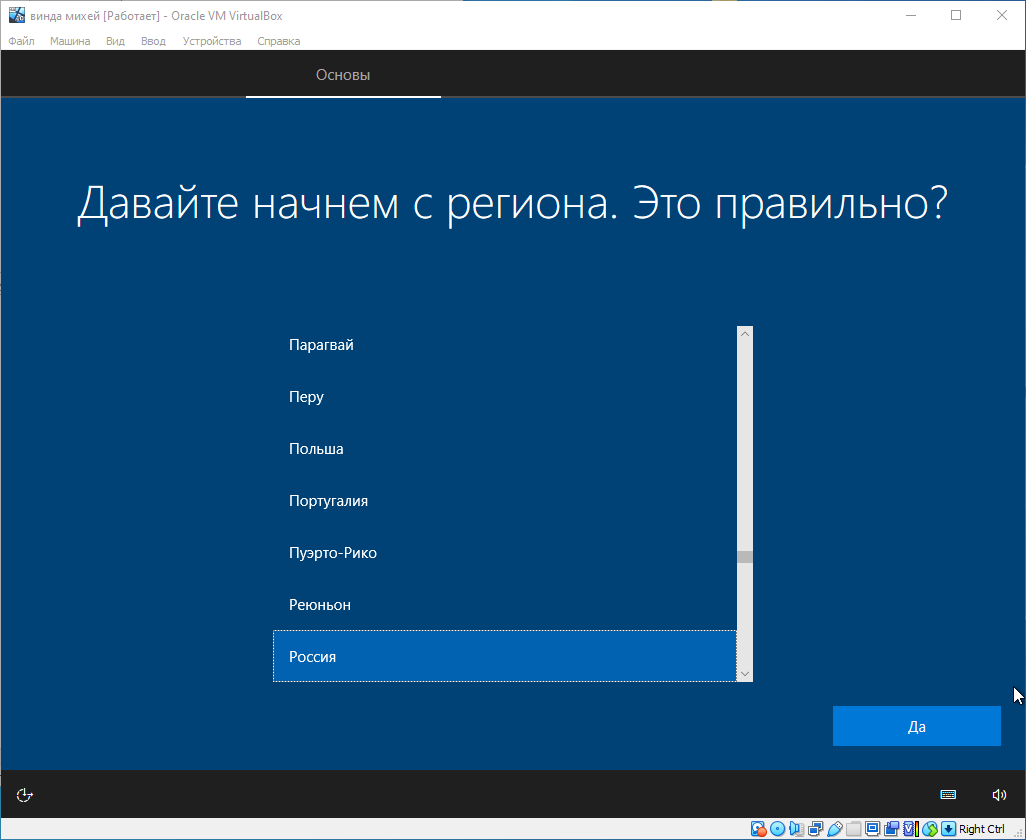


Рис.8 Выбор региона

Теперь выберем основную раскладку, в моем случае «Русская».

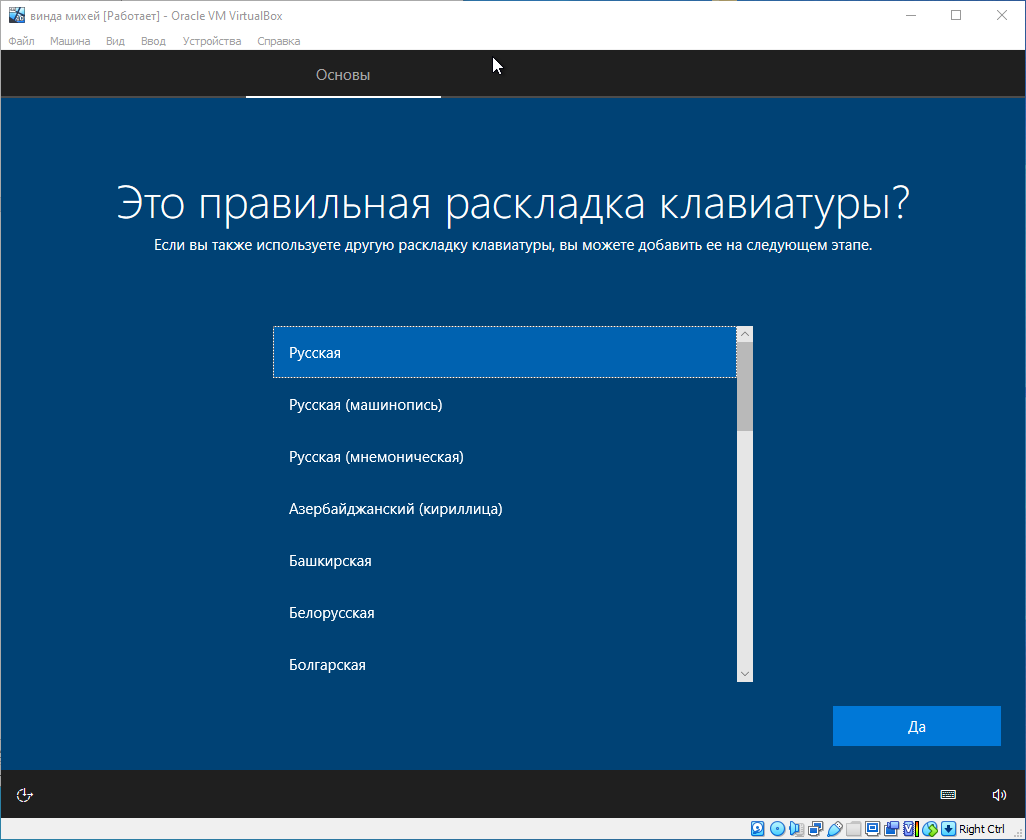


Рис.9 Выбор раскладки клавиатуры

После я создал пользователя и задал пароль. Настроил все параметры и разрешения и после всех шагов высветился главный экран.

**Тема №2. Установка MS SQL Server и Management Studio**

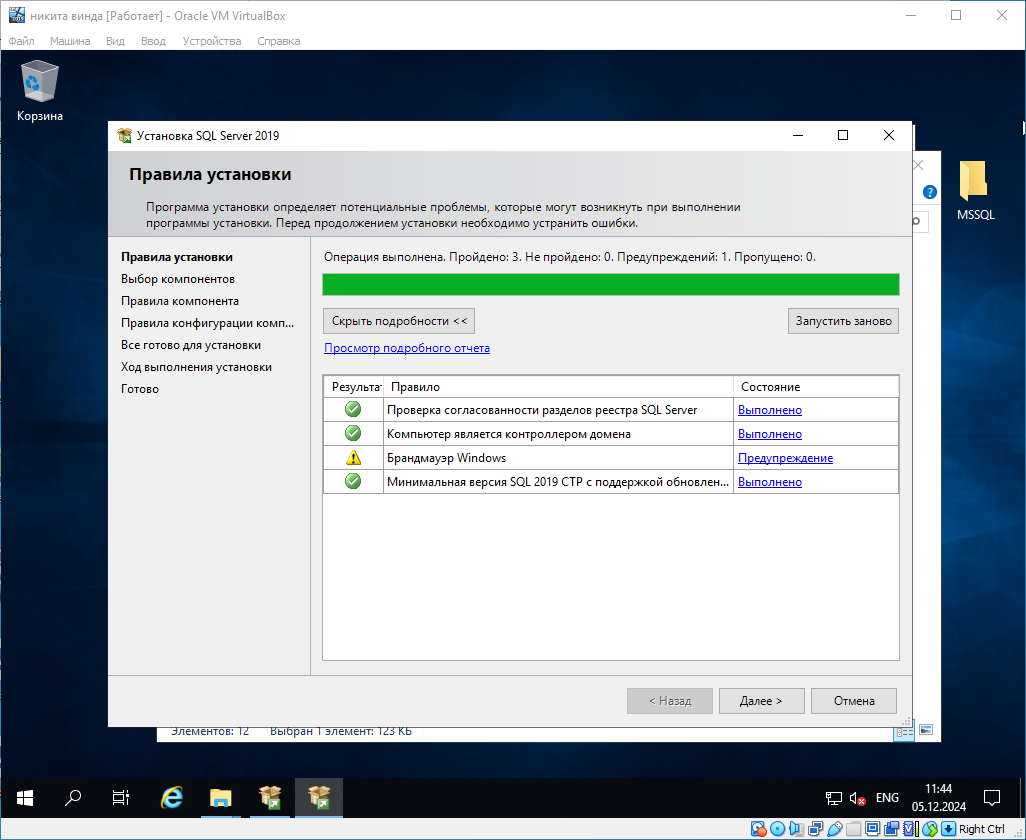


Рис.10 Начало установки SQL Server 2019

Когда загрузка будет завершена, произойдет извлечение файлов установки и запустится программа установки SQL Server. В нашем случае для новой установки Microsoft SQL Server выбираем первый пункт «Новая установка изолированного экземпляра SQL Server или добавление компонентов к существующей установке».

Далее нам необходимо принять условия лицензионного соглашения. Для этого прочитайте их и отметьте галочкой пункт «Я принимаю условия лицензии».

Нажимаем «Далее».

На данном шаге программа установки определит возможные глобальные проблемы, которые могут возникнуть в ходе установки, это так называемые «Глобальные правила».

Например, если моя система не соблюдает какое-нибудь из представленных правил, во время установки могут возникнуть проблемы, поэтому рекомендуется все ошибки и предупреждения устранить перед продолжением. Программа автоматически, если все хорошо, перейдет к следующему шагу, в остальных случаях устраняем ошибки и нажимаем «Далее».

Здесь я могу поставить галочку, чтобы Microsoft SQL Server обновлялся, используя «Центр обновления Microsoft» операционной системы Windows.

Только следует помнить о том, что «Центр обновления Windows» должен быть включен.

Нажимаем «Далее».

Программа установки установит необходимые для ее работы файлы, а также определит потенциальные проблемы, которые могут возникнуть при работе программы установки.

Если никаких проблем нет, то программа установки автоматически перейдет к следующему шагу, в противном случае все ошибки необходимо устранить, а предупреждения проанализировать, так как, например, включенный брандмауэр Windows может затруднить доступ к SQL серверу, поэтому программа установки предупредит меня о том, что необходимо добавить в брандмауэр соответствующие правила для SQL Server.

В следующем окне мы можем задать: режим проверки подлинности, администраторов SQL сервера, изменить каталоги данных и внести дополнительные параметры.

Режим проверки подлинности Windows – предполагает интеграцию с учетными записями.

Windows, т.е. пользователь, который аутентифицировался в Windows, сможет подключиться к SQL серверу;

Смешанный режим – позволяет использовать и проверку подлинности Windows и встроенную проверку SQL сервера, при которой администратор SQL Server сам создает учетные записи непосредственно в SQL сервере.

По умолчанию в администраторы будет добавлен пользователь, от имени которого я произвожу установку SQL Server, но я могу добавить и дополнительных администраторов.

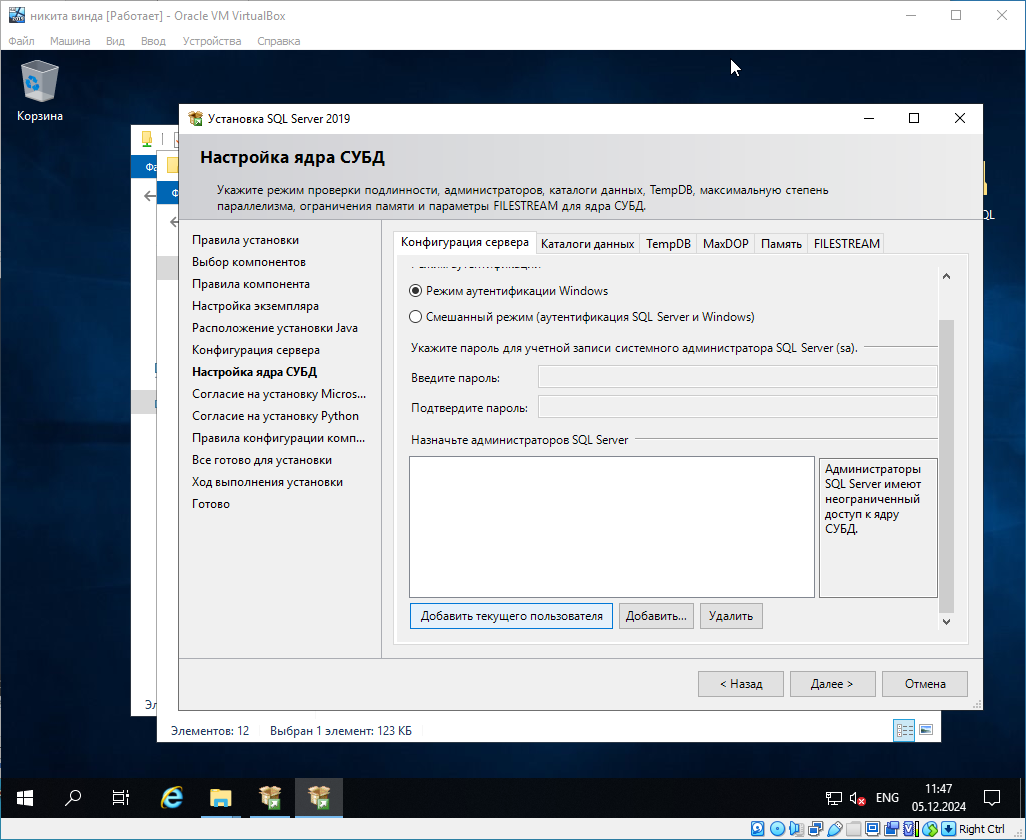


Рис.11 Настройка ядра СУБД

Я изменил эти каталоги (в случае тестовой установки это делать, конечно же, необязательно, но «боевые» базы данных всегда должны храниться в отдельном месте на отдельном хранилище). Для этого перешел на вкладку «Каталоги данных» и указал пути к новым каталогам для хранения пользовательских баз данных.

Я, для примера, указал для баз данных каталог Database на диске D, а для резервных копий каталог BACKUP\_DB.

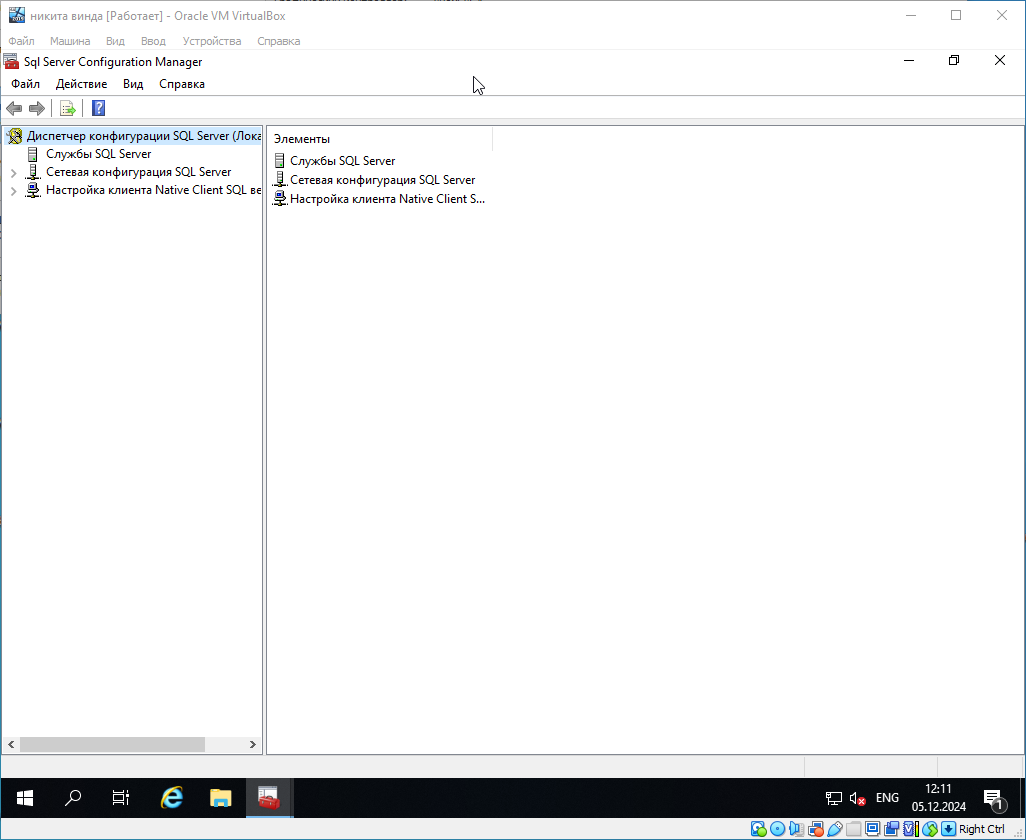


Рис.12 Конфигурация SQL Manager

Теперь, чтобы проверить работу Microsoft SQL Server 2019, давайте подключимся к серверу и напишем какой-нибудь запрос.

Для этого запускаем среду SQL Server Management Studio, например, из меню, ярлык называется «Microsoft SQL Server Management Studio 18».

В окне «Соединение с сервером» указываем: тип сервера (в нашем случае ip-адрес\тип сервера), к которому мы хотим подключиться, и учетные данные пользователя.

По умолчанию все уже заполнено так, как нам нужно, нажимаем «Соединить».

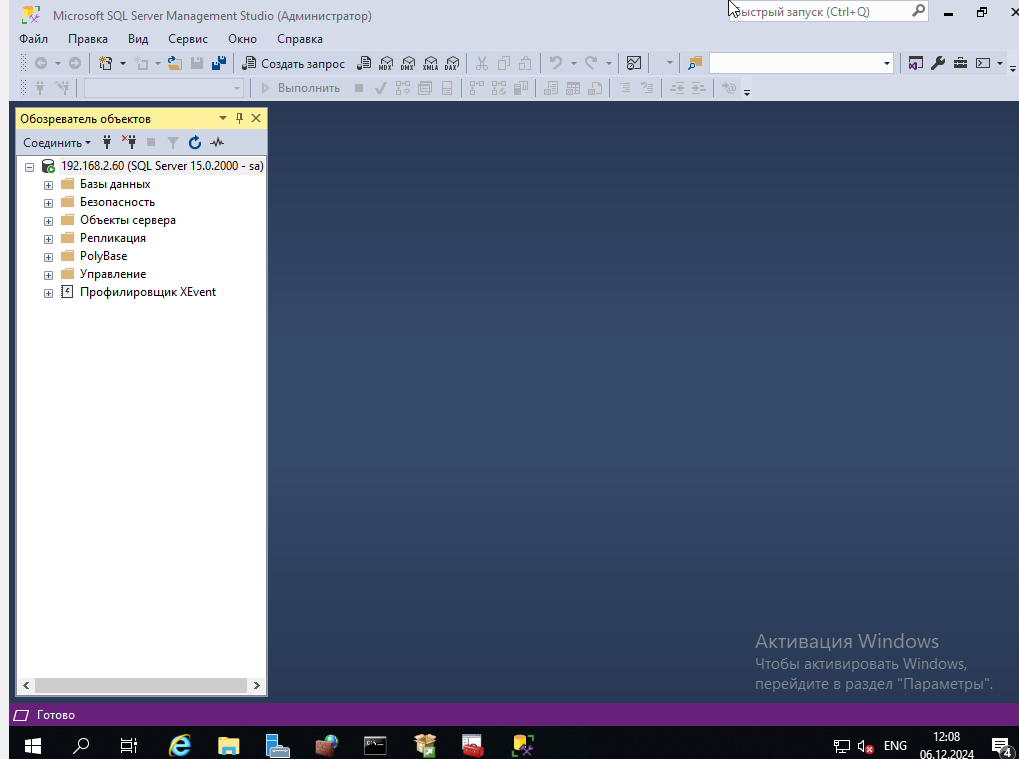


Рис.13 SQL Server

Далее открываем окно редактирования и выполнения SQL запросов, и пишем запрос, например, вот такой.

**Тема №3. Постановка задачи разработки БД. Выделение сущностей предметной области**

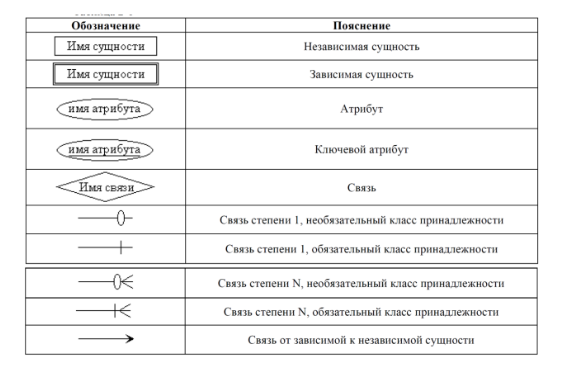
****

Рис.14 Степени бинарных связей



Рис.15 Пример для БД проектной организации

**Тема №4. Анализ информационных задач и круга пользователей БД. Логическое проектирование и нормализация БД**

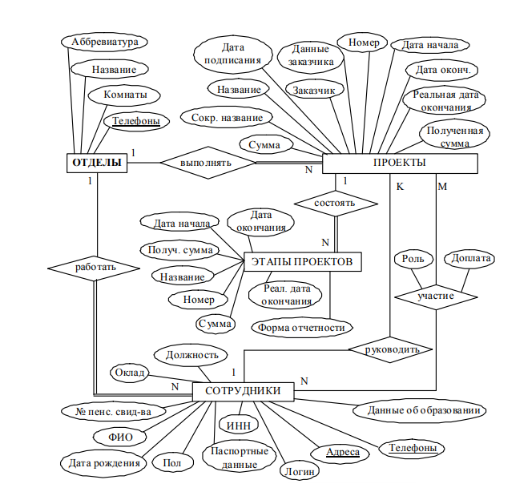
****

Рис.16 Логическое проектирование БД

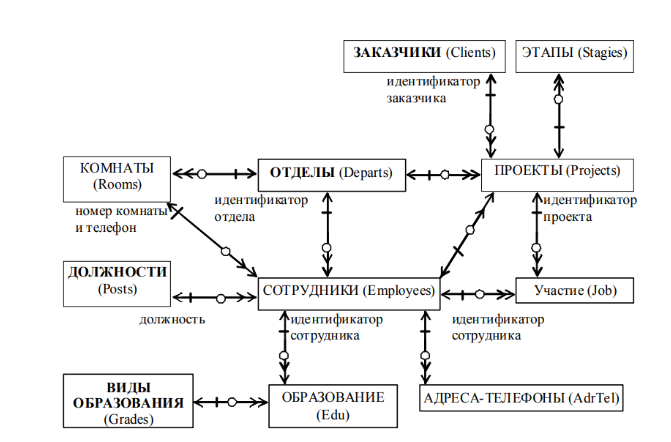


Рис.17 Ограничение Целостности



Рис.18 Схема для разрабатываемой БД

**Тема №5. Физическое проектирование БД**

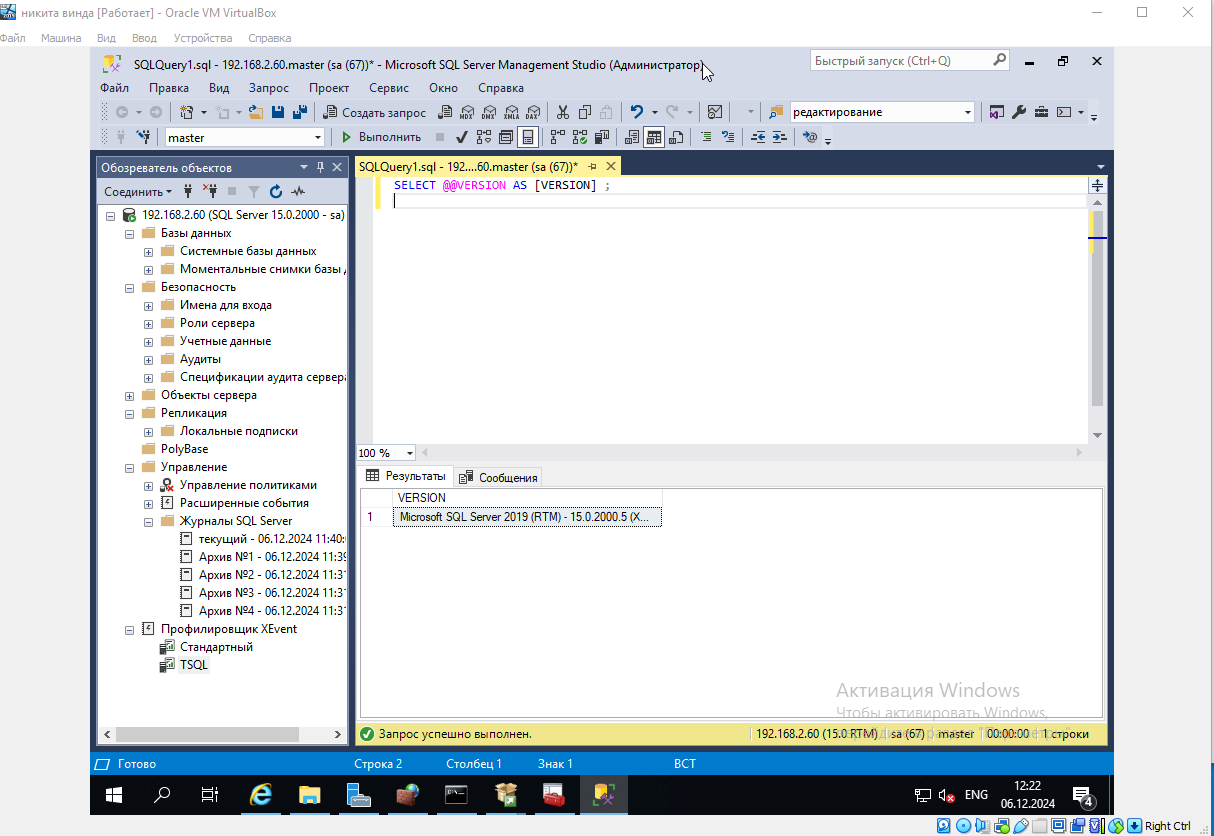


Рис.19 Простой код для проверки работы

Мы написали простой SQL запрос, в нем использовали системную функцию

@@VERSION, которая показывает текущую версию Microsoft SQL Server.Как результат, мы

видим, что Microsoft SQL Server 2019 Express у нас успешно установлен и функционирует.

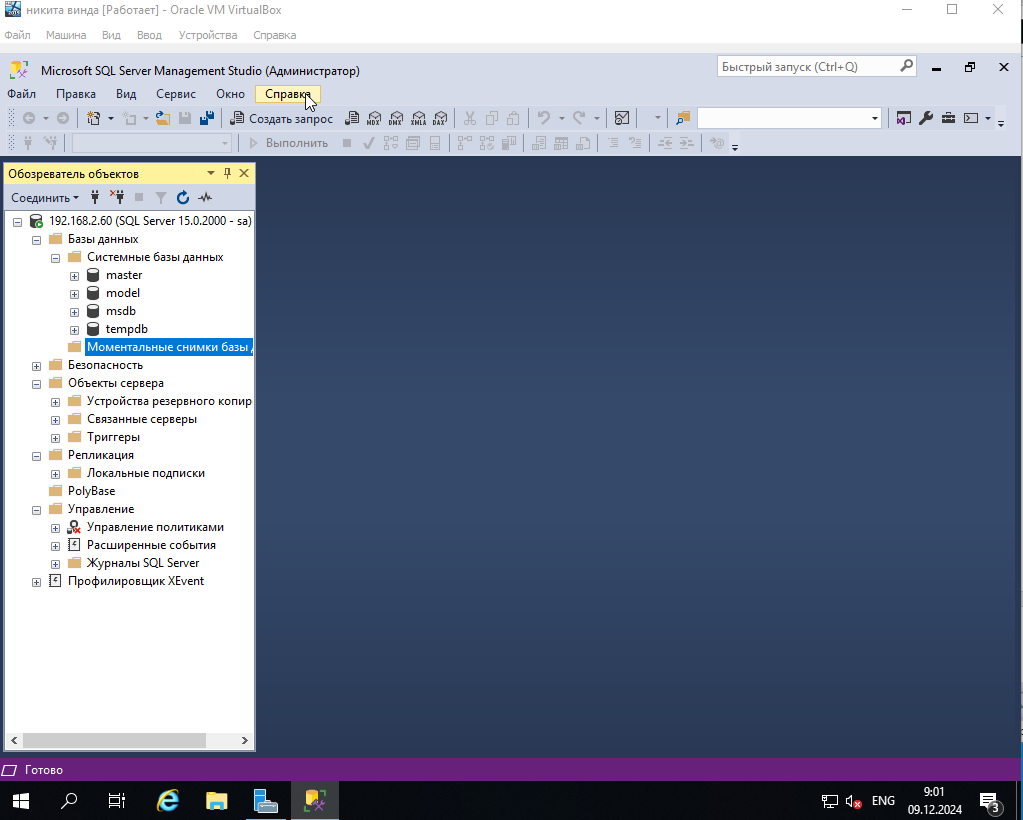


Рис.20 Развертывание базы данных

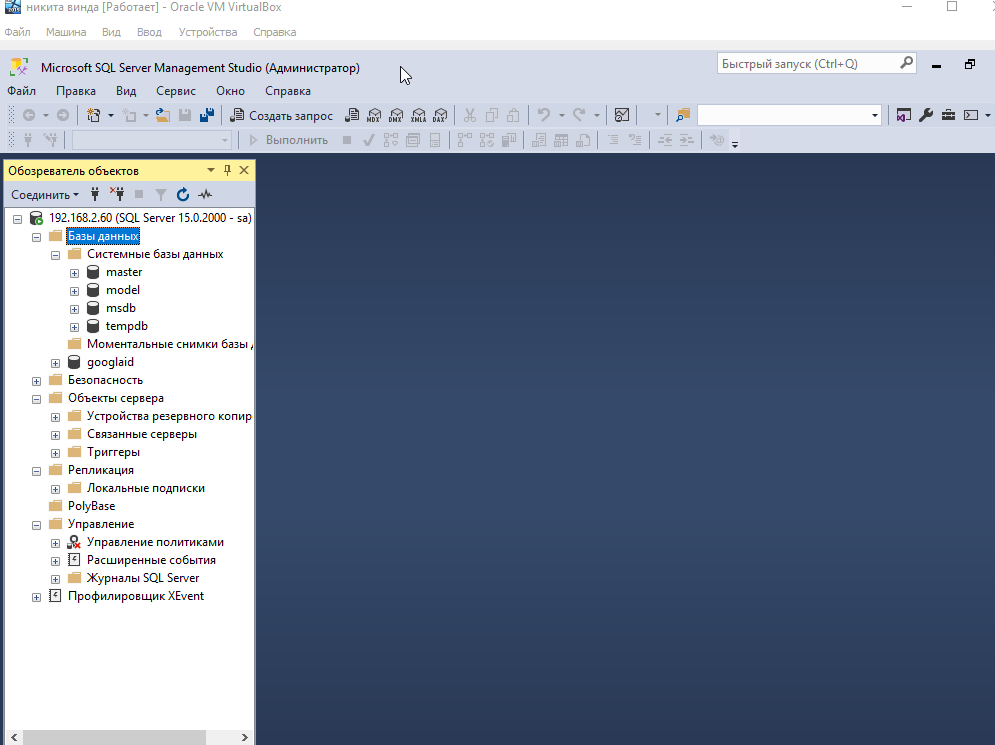


Рис.20 Создание новой базы данных

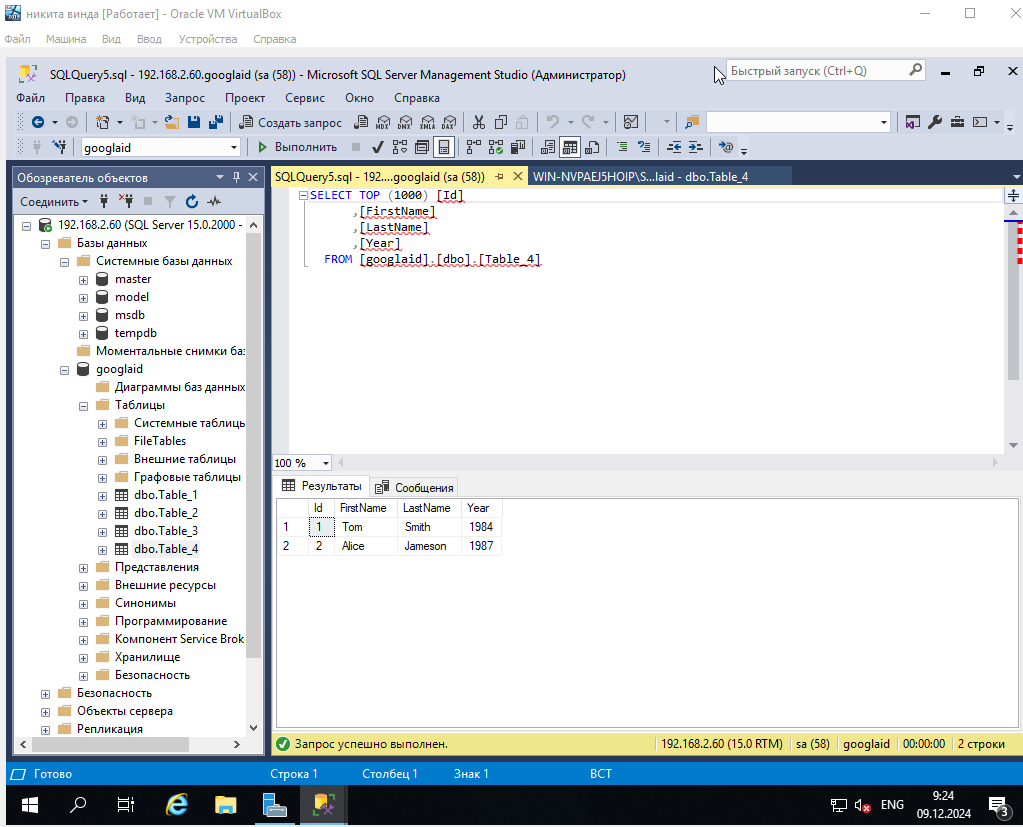


Рис.21 Вывод таблицы

1. **Таблица сотрудников (Employees)**:

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY AUTO\_INCREMENT, -- уникальный идентификатор сотрудника

first\_name VARCHAR(100) NOT NULL, -- имя сотрудника

last\_name VARCHAR(100) NOT NULL, -- фамилия сотрудника

position VARCHAR(100), -- должность

department\_id INT, -- идентификатор отдела

hire\_date DATE, -- дата приема на работу

salary DECIMAL(10, 2), -- зарплата

email VARCHAR(255) UNIQUE, -- email сотрудника

phone\_number VARCHAR(15) -- телефонный номер

);

1. **Таблица департаментов (Departments)**:

CREATE TABLE Departments (

department\_id INT PRIMARY KEY AUTO\_INCREMENT, -- уникальный идентификатор департамента

department\_name VARCHAR(255) NOT NULL, -- название департамента

manager\_id INT, -- идентификатор менеджера департамента

FOREIGN KEY (manager\_id) REFERENCES Employees(employee\_id) -- связь с менеджером

);

1. **Таблица проектов (Projects)**:

CREATE TABLE Projects (

project\_id INT PRIMARY KEY AUTO\_INCREMENT, -- уникальный идентификатор проекта

project\_name VARCHAR(255) NOT NULL, -- название проекта

start\_date DATE, -- дата начала проекта

end\_date DATE, -- дата завершения проекта

budget DECIMAL(15, 2), -- бюджет проекта

department\_id INT, -- идентификатор департамента

FOREIGN KEY (department\_id) REFERENCES Departments(department\_id) -- связь с департаментом

);

1. **Таблица участия сотрудников в проектах (Employee\_Projects)**:

CREATE TABLE Employee\_Projects (

employee\_id INT, -- идентификатор сотрудника

project\_id INT, -- идентификатор проекта

role VARCHAR(100), -- роль сотрудника в проекте

start\_date DATE, -- дата начала участия

end\_date DATE, -- дата окончания участия

PRIMARY KEY (employee\_id, project\_id), -- составной первичный ключ

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id), -- связь с сотрудником

FOREIGN KEY (project\_id) REFERENCES Projects(project\_id) -- связь с проектом

);

1. **Таблица задач (Tasks)**:

CREATE TABLE Tasks (

task\_id INT PRIMARY KEY AUTO\_INCREMENT, -- уникальный идентификатор задачи

task\_name VARCHAR(255) NOT NULL, -- название задачи

project\_id INT, -- идентификатор проекта

assigned\_to INT, -- сотрудник, которому назначена задача

status VARCHAR(50), -- статус задачи (например, "выполнено", "в процессе")

due\_date DATE, -- дата завершения задачи

FOREIGN KEY (project\_id) REFERENCES Projects(project\_id), -- связь с проектом

FOREIGN KEY (assigned\_to) REFERENCES Employees(employee\_id) -- связь с сотрудником

);

1. **Таблица ресурсов (Resources)**:

CREATE TABLE Resources (

resource\_id INT PRIMARY KEY AUTO\_INCREMENT, -- уникальный идентификатор ресурса

resource\_name VARCHAR(255) NOT NULL, -- название ресурса

project\_id INT, -- идентификатор проекта

quantity INT, -- количество ресурса

cost DECIMAL(10, 2), -- стоимость ресурса

FOREIGN KEY (project\_id) REFERENCES Projects(project\_id) -- связь с проектом

);

Эти таблицы создают основу для базы данных проектной организации. В них хранится информация о сотрудниках, департаментах, проектах, задачах и ресурсах, которые необходимы для выполнения проектов.

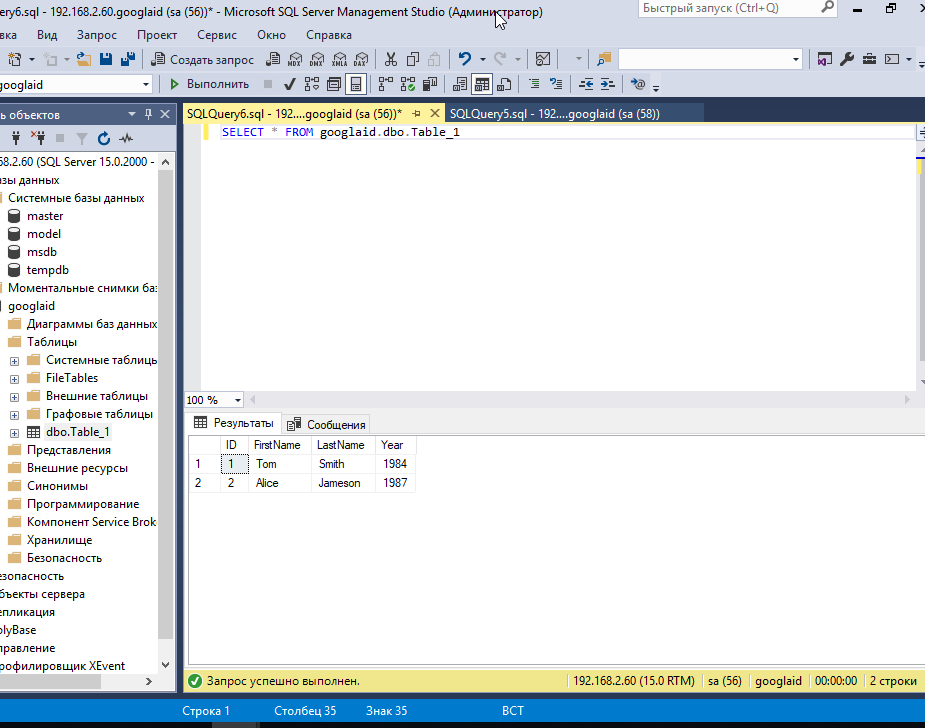


Рис.22 Результат запроса

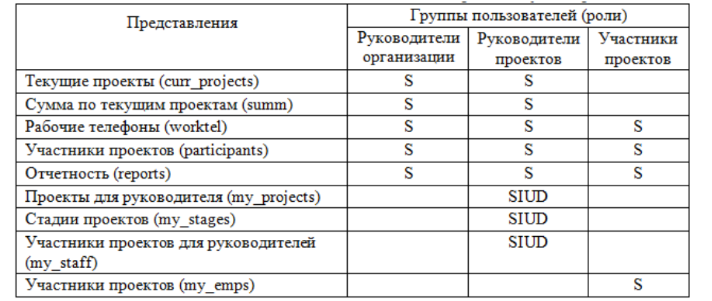


Рис.23 Назначение прав доступа к представлениям.

**Тема №6. Работа с транзакциями и операторами процедурной логики**

Общие принципы использования транзакций:

1. BEGIN TRANSACTION — начало транзакции.
2. COMMIT — коммитит все изменения, сделанные в рамках транзакции (сохраняет изменения в базе данных).
3. ROLLBACK — откатывает все изменения, сделанные в рамках транзакции, если возникла ошибка.
4. Транзакции следует использовать для операций, где несколько действий должны быть выполнены как единое целое, чтобы избежать частичных обновлений данных, которые могут нарушить целостность базы данных.

Транзакции обеспечивают целостность данных и позволяют справляться с ошибками, гарантируя, что база данных останется в консистентном состоянии.

- Начало транзакции BEGIN TRANSACTION; -- Добавление нового сотрудника INSERT INTO Employees (first\_name, last\_name, position, department\_id, hire\_date, salary, email, phone\_number) VALUES ('Иван', 'Иванов', 'Инженер', 1, '2024-12-12', 50000, 'ivanov@example.com', '1234567890'); -- Добавление нового проекта INSERT INTO Projects (project\_name, start\_date, end\_date, budget, department\_id) VALUES ('Проект A', '2024-12-15', '2025-12-15', 1000000, 1); -- Если все операции успешны, подтверждаем транзакцию COMMIT;

-- Начало транзакции

BEGIN TRANSACTION;

PRINT 'Транзакция началась. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- Точка сохранения

SAVE TRANSACTION SavePoint1;

PRINT 'Точка сохранения создана. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- Уменьшение оклада первого сотрудника в два раза

UPDATE Employees

SET salary = salary / 2

WHERE employee\_id = 1;

PRINT 'Оклад сотрудника 1 уменьшен. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- Увеличение оклада второго сотрудника на 50%

UPDATE Employees

SET salary = salary \* 1.5

WHERE employee\_id = 2;

PRINT 'Оклад сотрудника 2 увеличен. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- Пример возможной ошибки или условия для отката (если нужно)

-- В реальном приложении можно вставить условие для проверки, например:

-- IF (условие\_ошибки)

-- BEGIN

-- PRINT 'Ошибка. Откат транзакции.';

-- ROLLBACK TRANSACTION;

-- PRINT 'Транзакция откатана. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- END

-- Если все прошло успешно, зафиксируем изменения

COMMIT;

PRINT 'Транзакция зафиксирована. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

-- Если возникла ошибка, откатим к точке сохранения

-- В случае ошибки отката:

-- ROLLBACK TRANSACTION SavePoint1;

-- PRINT 'Откат к точке сохранения. Счетчик транзакций: ' + CAST(@@TRANCOUNT AS VARCHAR);

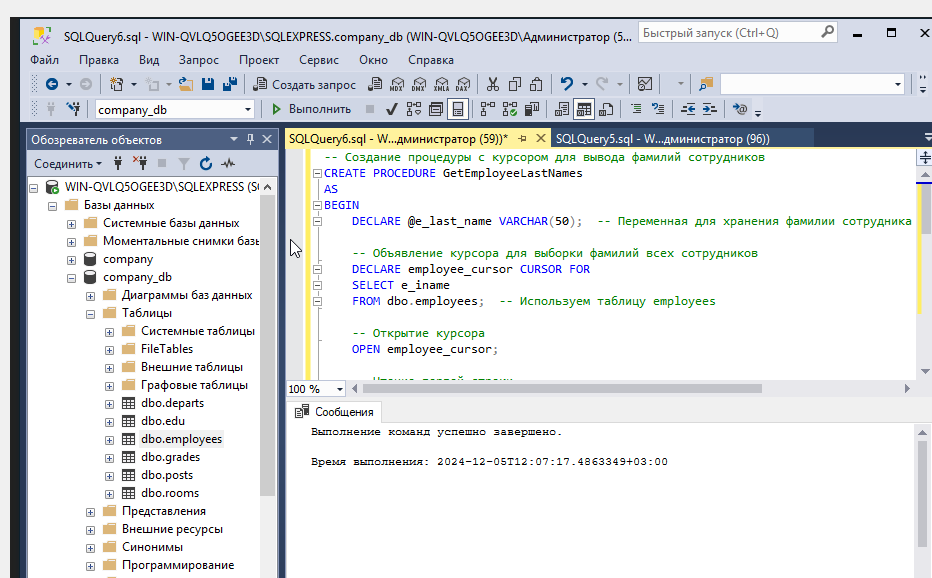


Рис.24 Курсор для вывода фамилии и оклада

**Тема №7. Создание хранимых процедур, пользовательских функций и триггеров**

1. Создание хранимых процедур

2. Создание пользовательских функций

3. Создание триггеров

Материально-техническое обеспечение: инструкционная карта, персональный компьютер.

Задание. Написать и запустить процедуру поиска сотрудников с наименьшим и наибольшим окладом (рис.25).

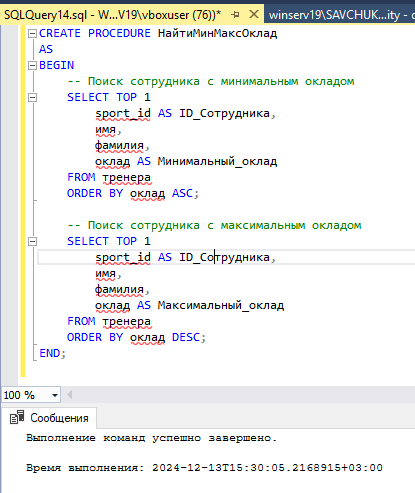


Рис.25 Поиск сотрудника с окладом наименьшим и наибольшим

Задание: Создать и применить функцию (типа multi-statement), которая выводит фамилии принятых на работу сотрудников за последние 30 дней (рис 26).

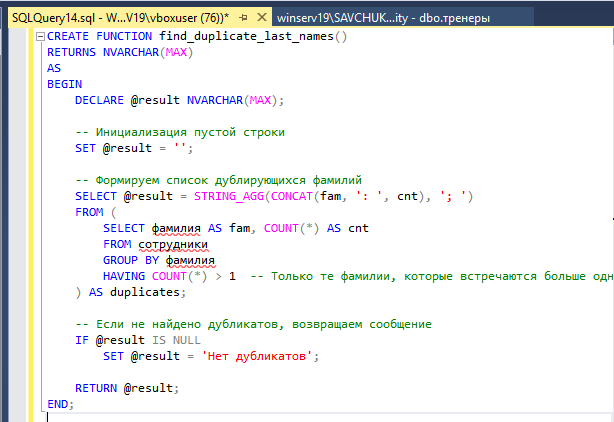


Рис.26 Функция, которая выводит фамилии принятых на работу людей за 30 дней

# Тема №8. Информационная безопасность БД. Организация резервного копирования и восстановления. Информационная безопасность.

1. Изучение основных положений теории и практики обеспечения информационной

безопасности;

1. Организация резервного копирования БД и восстановление из резервной копии.

Создание копии БД (рис.27)

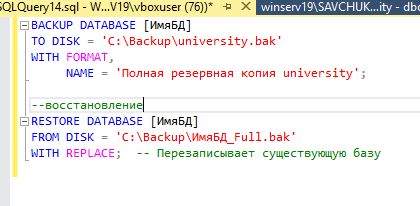


Рис.27 Копия БД

Создание резервной копии журнала транзакций (рис.28)

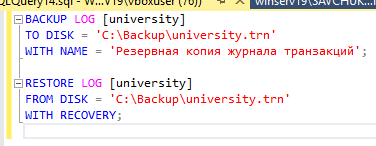


Рис.28 Резервная копия журнала

Создание частичной резервной копии (рис.29)

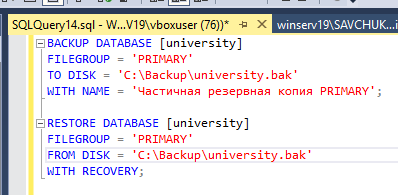


Рис.29 Частичная резервная копия

**Заключение**

Разработка, администрирование и защита баз данных являются важными аспектами информационных технологий, обеспечивающими эффективное хранение, обработку и защиту данных. Правильное проектирование и администрирование базы данных способствует не только повышению производительности и эффективности работы системы, но и уменьшению рисков утечек или потерь данных. Важность защиты базы данных трудно переоценить, так как это не только обеспечивает безопасность и конфиденциальность информации, но и поддерживает доверие пользователей к системе. Таким образом, успешная реализация этих задач является ключом к эффективному и безопасному использованию современных информационных технологий.

В процессе разработки баз данных особое внимание следует уделить правильному проектированию структуры данных, их организации и взаимосвязи. Грамотно спроектированная база данных обеспечивает эффективное хранение информации, высокую производительность запросов и оптимизацию работы всей системы. Не менее важным этапом является администрирование базы данных, которое включает в себя постоянный мониторинг, настройку параметров работы и оптимизацию производительности. Хорошо организованный процесс администрирования способствует бесперебойной работе системы и снижению нагрузки, а также позволяет оперативно реагировать на изменения в запросах и требованиях пользователей.

Однако важнейшим аспектом, который нельзя игнорировать, является защита базы данных. С учетом того, что информационные системы становятся мишенью для киберпреступников, защита данных от несанкционированного доступа, утечек и потерь требует применения многоуровневых мер безопасности. Современные методы защиты баз данных включают шифрование, многофакторную аутентификацию, системы контроля доступа и регулярные обновления безопасности. Резервное копирование и создание планов восстановления данных после сбоев системы также играют ключевую роль в обеспечении надежности и доступности информации. Важно понимать, что безопасность баз данных не ограничивается лишь техническими мерами — она включает в себя также организационные процедуры, такие как обучение пользователей, разработка политик безопасности и соблюдение стандартов защиты информации.

**Список литературы**

1. **TechNet Microsoft**. Управление SQL Server: эффективное проектирование баз данных. – *интернет-ресурс* – <https://technet.microsoft.com>. – Дата обращения: 10 декабря 2024 года.
2. **METANIT**.**COM**. Cайт о программировании – *интернет-ресурс* [MS SQL Server 2022 и T-SQL](https://metanit.com/sql/sqlserver/). – Дата обращения: 10 декабря 2024 года.
3. **SQL Server Tutorial**. Туториал SQL Server – *интернет-ресурс* – [SQL Server Tutorial](https://www.sqlservertutorial.net/). – Дата обращения: 10 декабря 2024 года.
4. **Proselyte**. Учебное пособие – *интернет-ресурс* – [Руководство по SQL (полная версия) – PROSELYTE](https://proselyte.net/tutorials/sql/). – Дата обращения: 10 декабря 2024 года.