

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА – Российский технологический университет»**

**РТУ МИРЭА**

Институт кибербезопасности и цифровых технологий

Кафедра КБ-2 «Информационно-аналитические системы кибербезопасности»

Зубарев Василий Сергеевич

Отчет по практической работе № 1

по дисциплине «Безопасность систем баз данных»

«Среда управления SQL Server Management Studio. ЯзыкTransact-SQL. Компоненты SQL»

студент \_ курса по специальности (направлению подготовки):

10.05.03 «Информационная безопасность автоматизированных ИС»

10.03.01 «Информационная безопасность»

|  |  |
| --- | --- |
|  | Проверила:  старший преподаватель Войтенков Д.В.  Отметка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Москва 2025

**Цель**: изучить среду управления SQL Server Management Studio, функциональности этой среды, связанные с компонентом DatabaseEngine, а также изучить возможности администрирования и управления базами данных.

**Задачи**:

1. Изучение среды управления SQL Server Management Studio.

2. Выполнить примеры из лабораторной работы.

3.Выполнить упражнения из лабораторной работы.

4. Сделать отчет по лабораторной работе.

**Ход работы**:

Пример 1.

При запуске SQL Server Management Studio открывается окно Connect to Server. Окно “Соединение с сервером” показано на рис. 1.

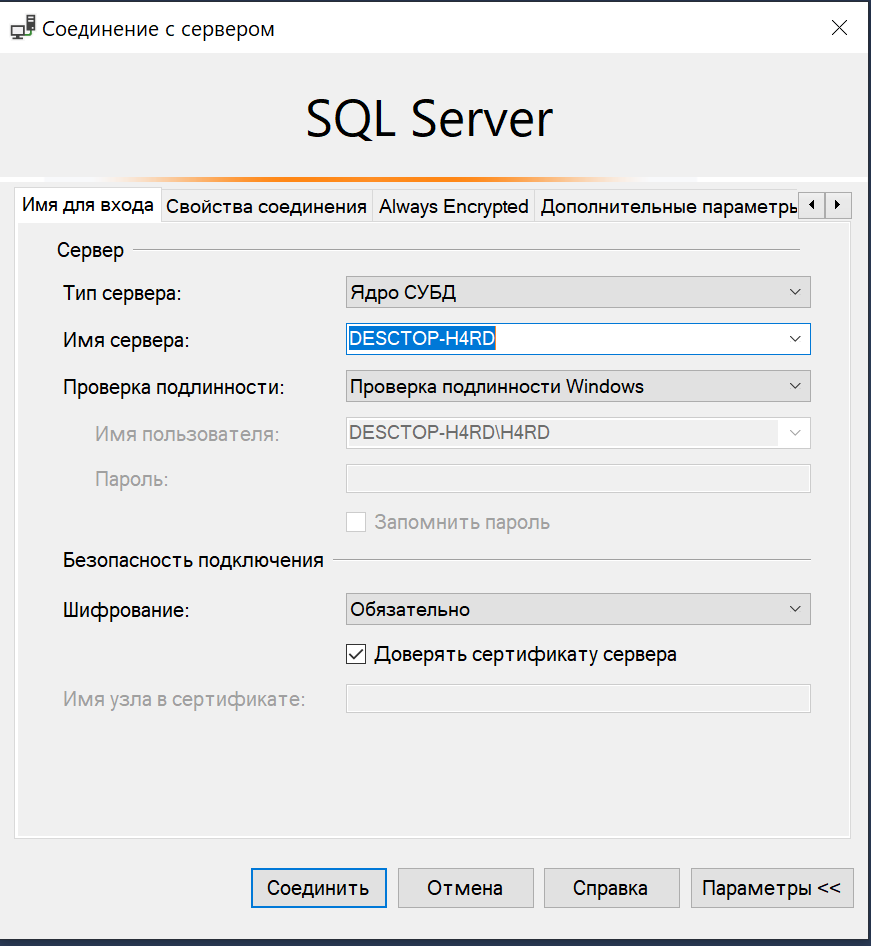


Рис. 1. Диалоговое окно Connect to Server.

После подключения к серверу базы данных открывается главное окно среды SQL Server Management, а также окна “Зарегистрированные серверы”, “Обозреватель объектов”.

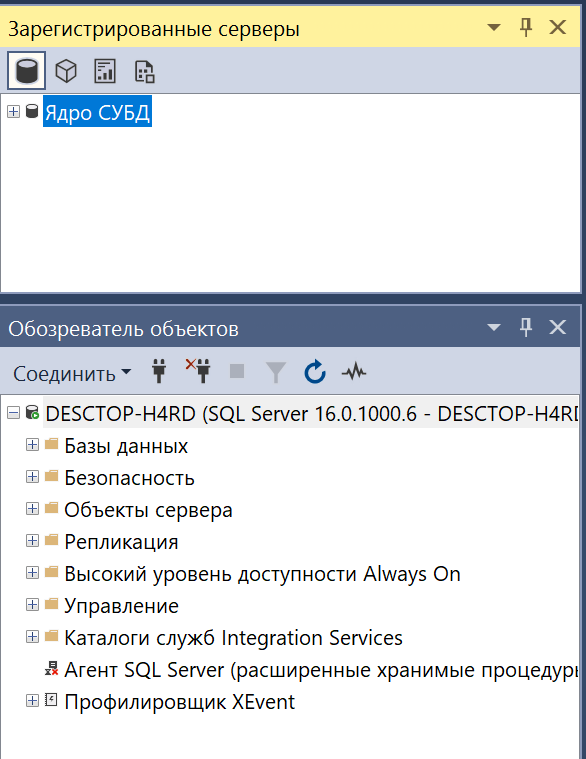


Рис. 2. Компоненты Registered Servers и Object Explorer

Можно зарегистрировать новый сервер базы данных. Откроется диалоговое окно New Server Registration (Регистрация нового сервера), которое показано на рис. 3.

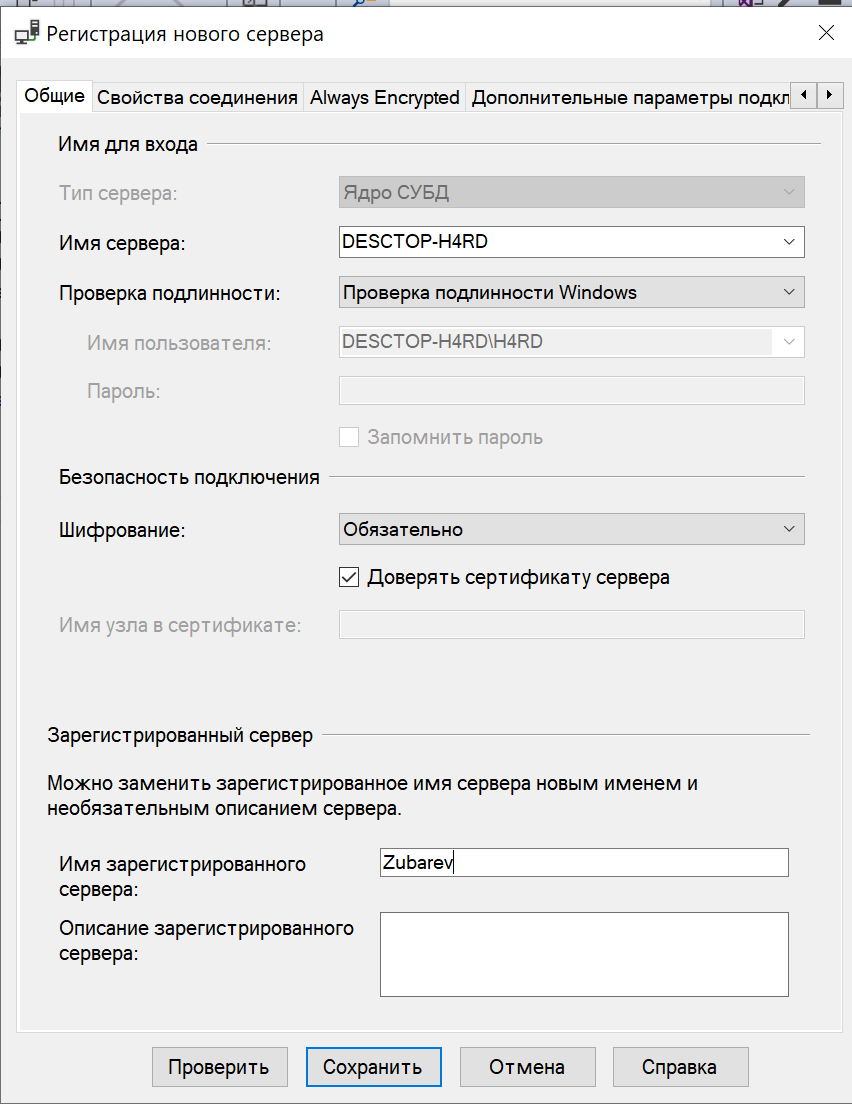


Рис. 3. Регистрация сервера.

Подключение к серверу на рис. 4.

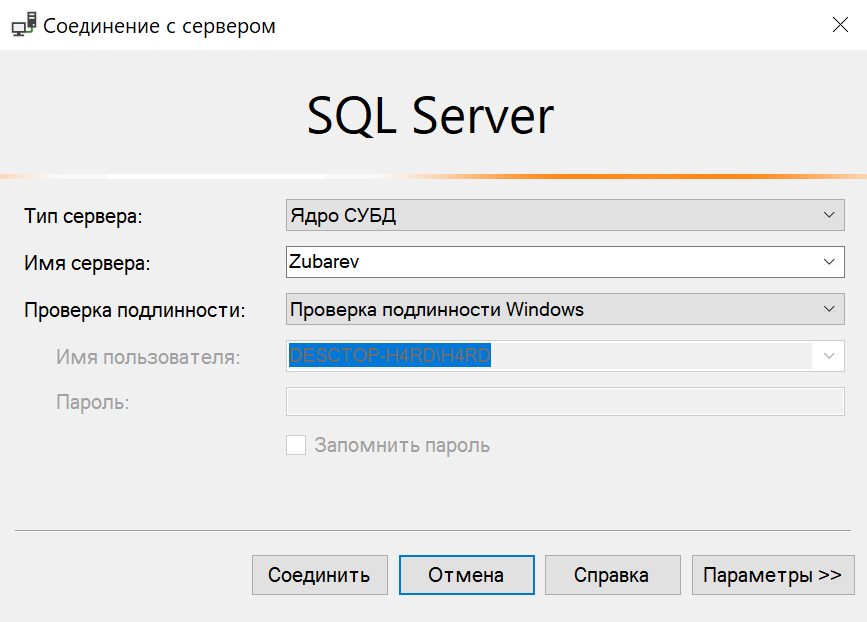


Рис. 4. Подключение к серверу.

Создание новой группы серверов показано на рис. 5.

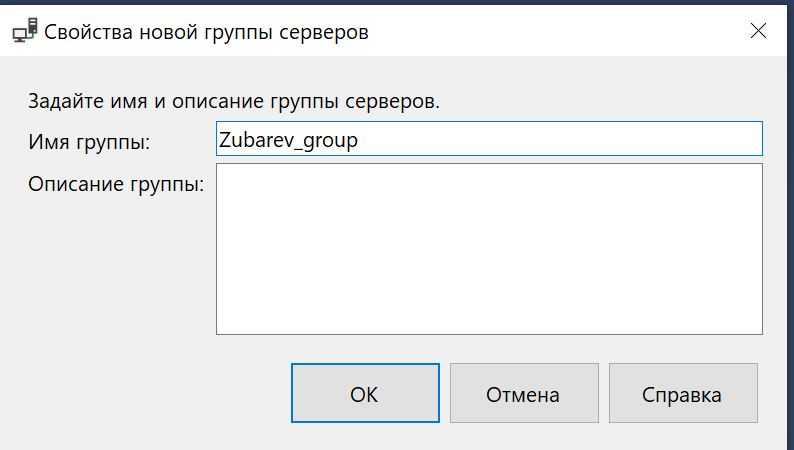


Рис. 5. Создание новой группы серверов.

Окно свойств нового сервера показано на рис. 7.

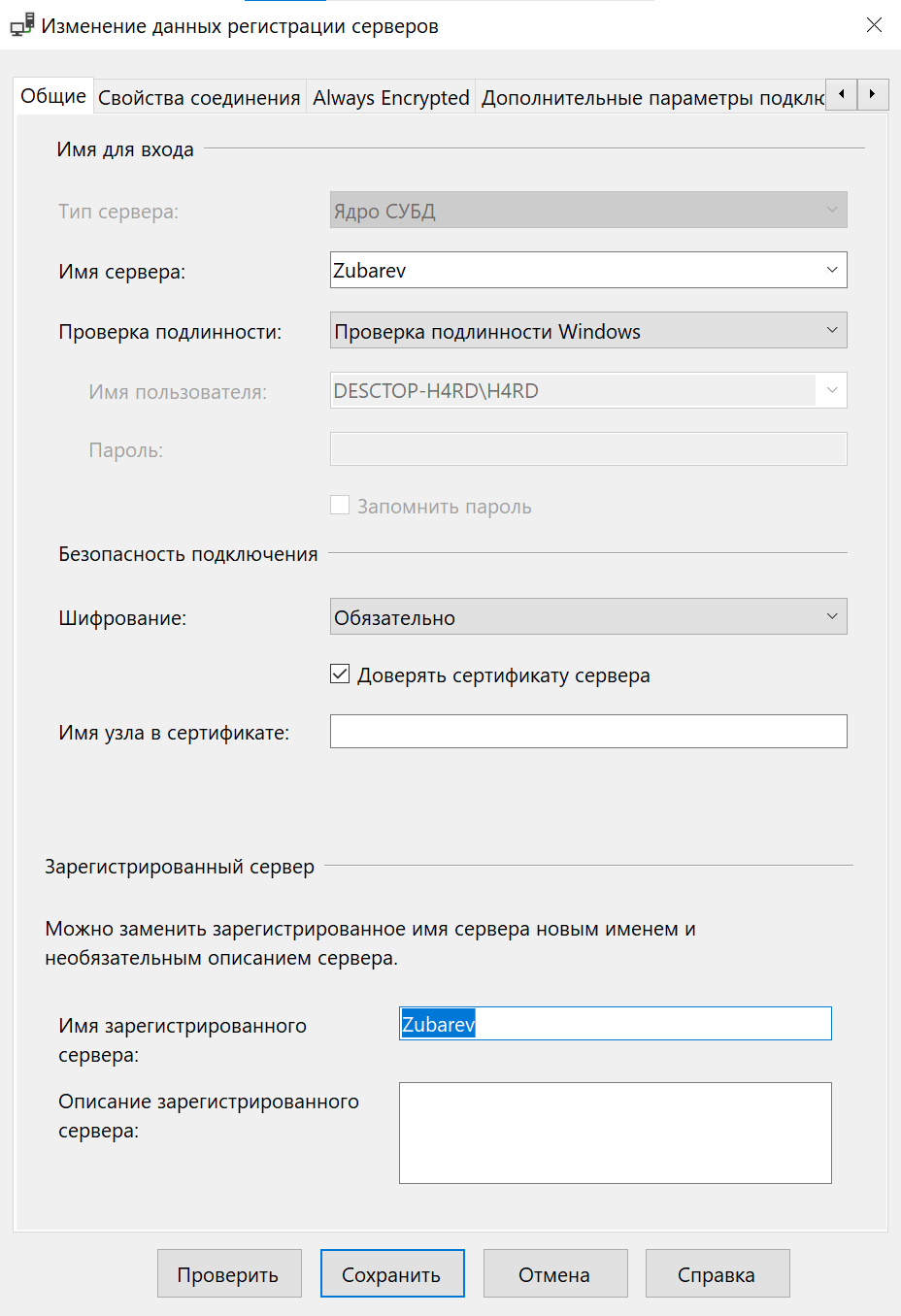


Рис. 7. Диалоговое окно “Свойства сервера”.

Пример 6. Создание баз данных, не прибегая к использованию языка Transact-SQL.

Создание баз данных показано на рис.8

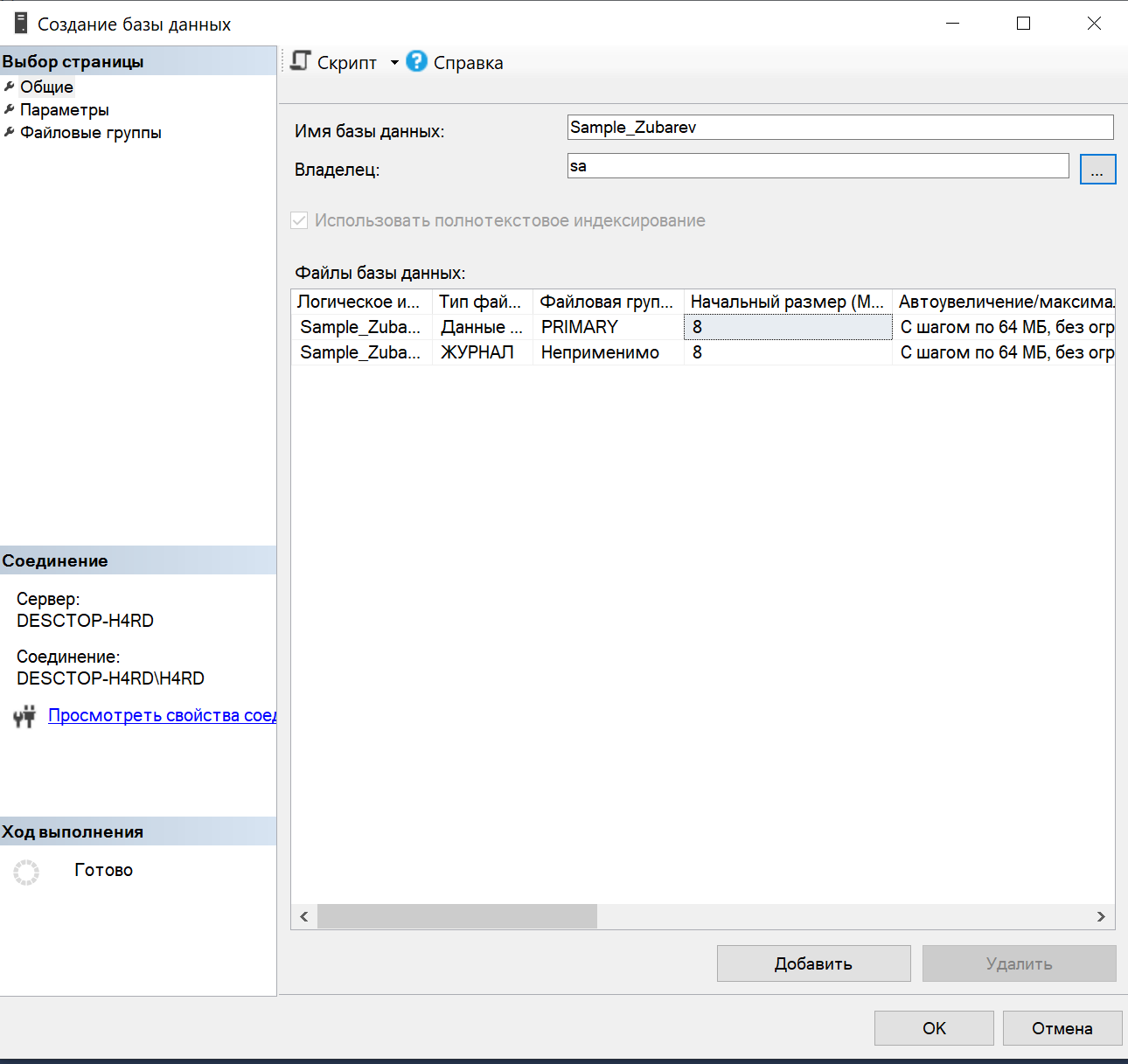


Рис. 8. Диалоговое окно “Создание базы данных”.

Страница “Общие”, представленная на рис. 9, содержит информацию об имени владельца, параметры сортировки БД.

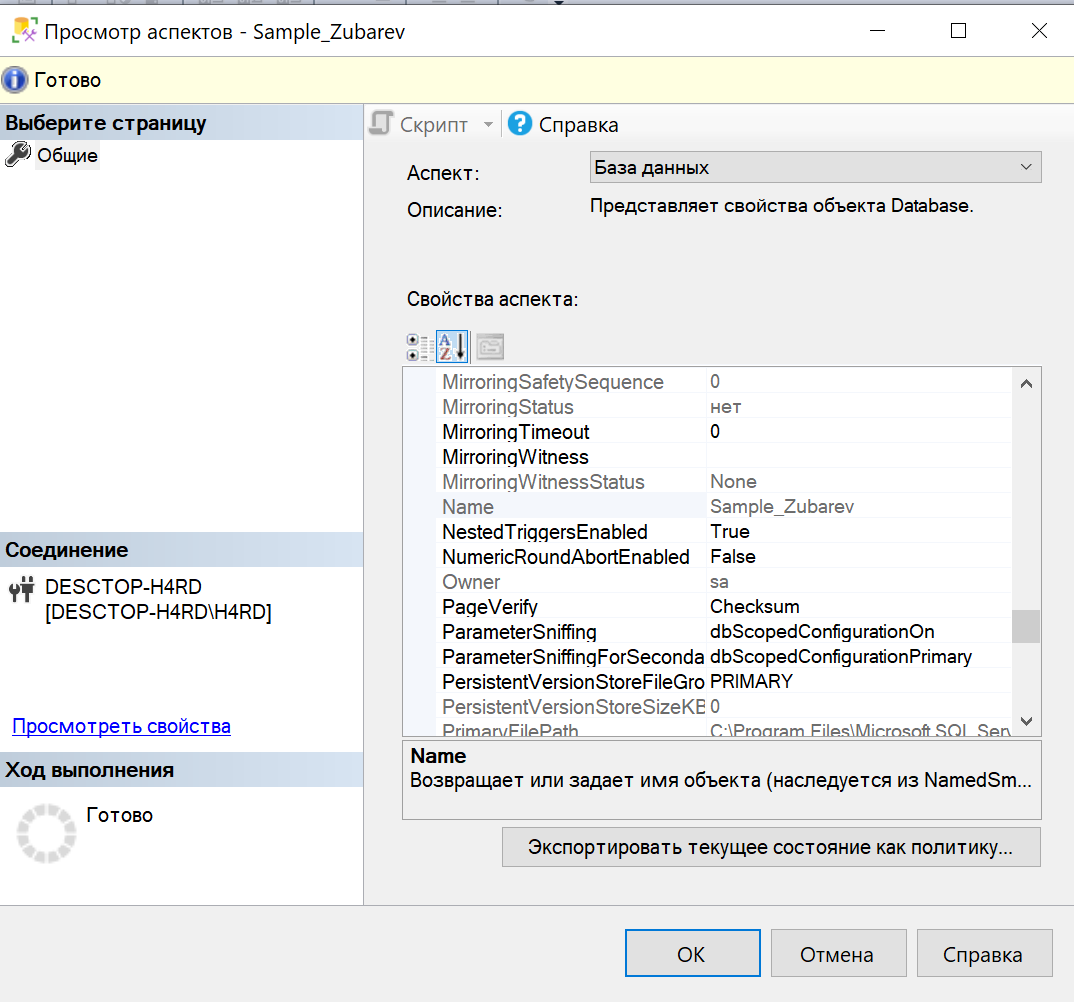


Рис. 9. Страница “Общие” диалогового окна “Свойства базы данных”.

Пример 7. Создание четырех таблиц. Последовательность команд показана на рис. 10.

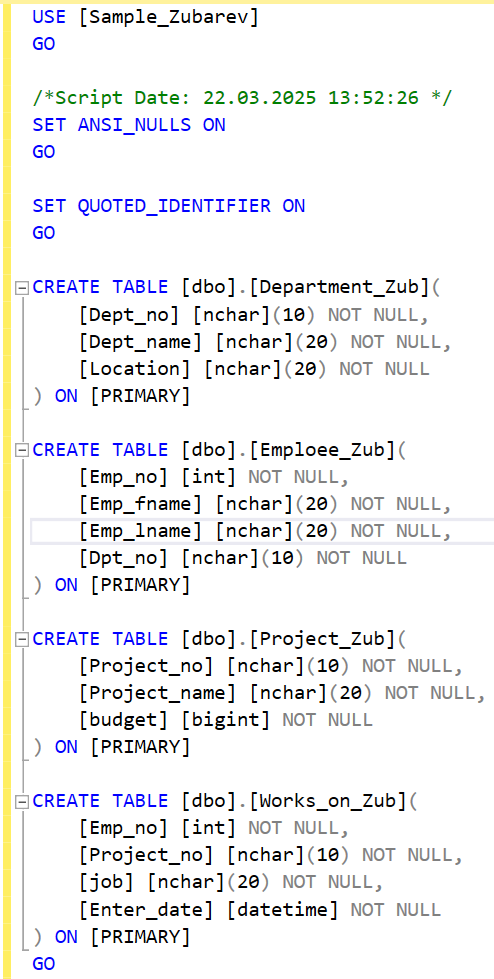


Рис. 10. Создание таблицы.

Отношение ch\_department содержит 3 атрибута: Dept\_no, Dept\_name, Location. Создание таблицы ch\_department показано на рис.11.

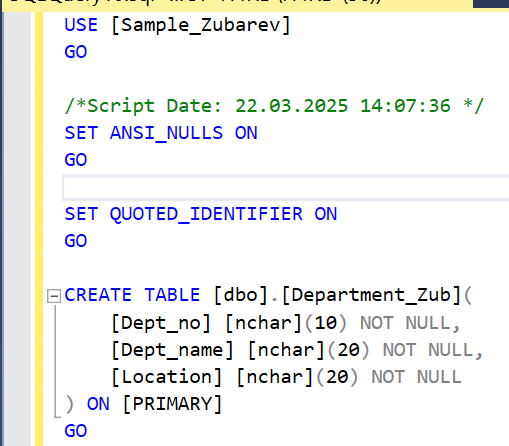
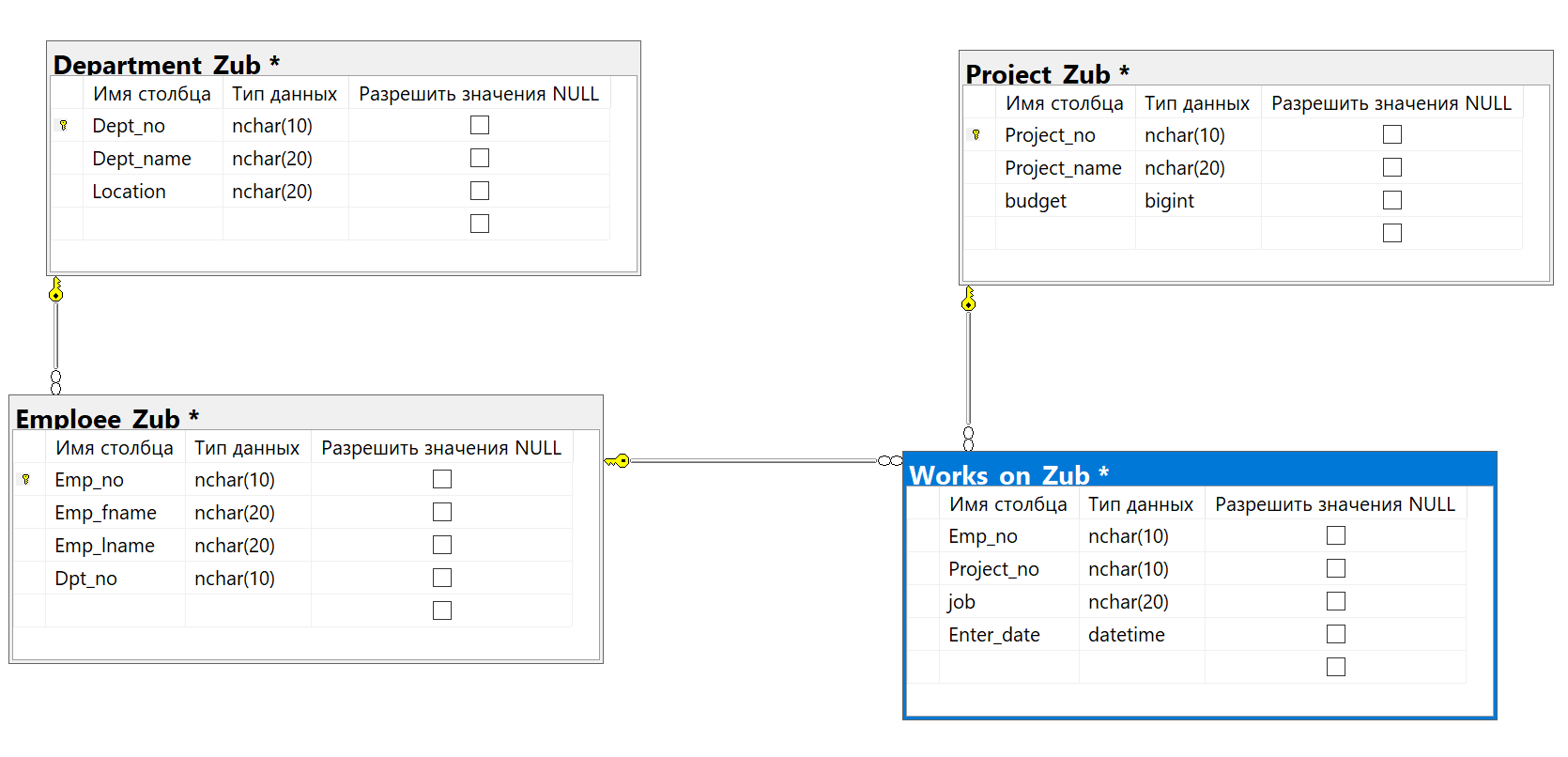


Рис. 11. Создание атрибутов отношения ch\_department.

Пример 8. Создание диаграммы типа "сущность — отношение". Диаграмма представлена на рис. 12.

Рис. 11. ER-диаграмма.

Пример 9. Разработка запросов, используя среду SQL Server Management Studio. Работа с редактором запросов представлена на рис. 12.

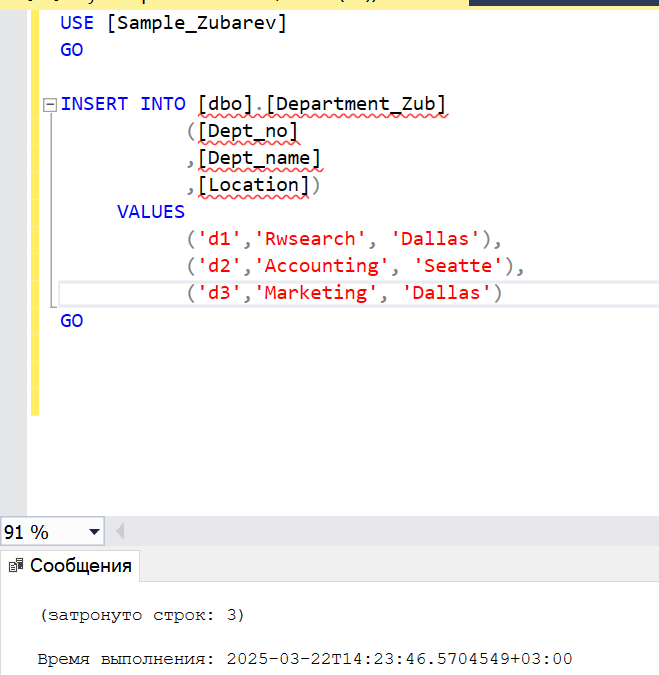


Рис. 12. Редактор запросов и результат выполнения запроса.

Упражнение 1

Используя среду SQL Server Management Studio, создайте базу данных под названием test. Сохраните эту базу данных в файле testdate\_a в папку C:\tmp и выделите для этого файла 10 Мбайт дискового пространства. Установите параметры файла базы данных для автоматического увеличения размера с шагом по 2 Мбайта до максимального размера 20 Мбайт.

На рис.13. Представлено создание базы данных на примере test\_Zub и задание необходимых параметров.

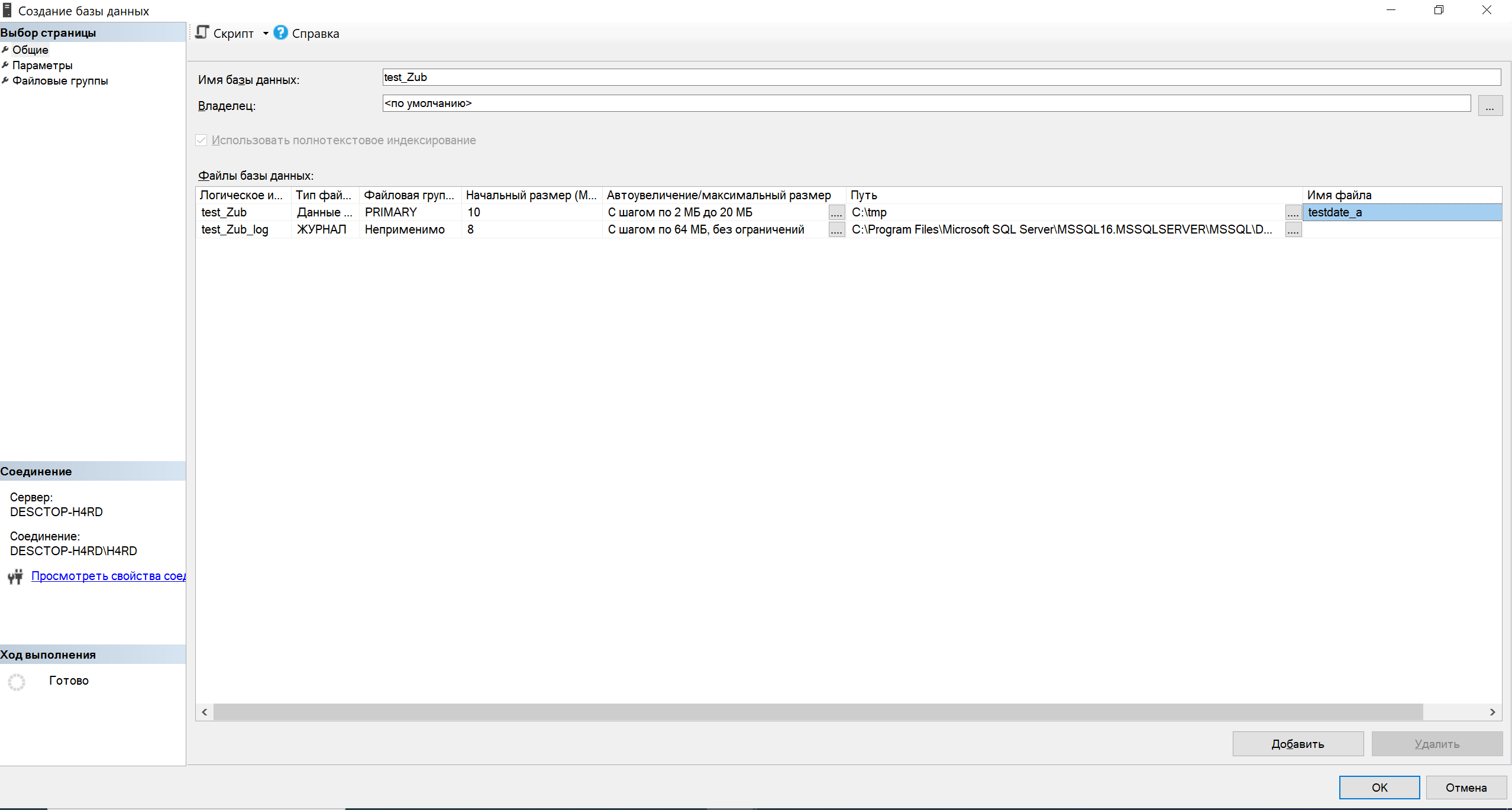
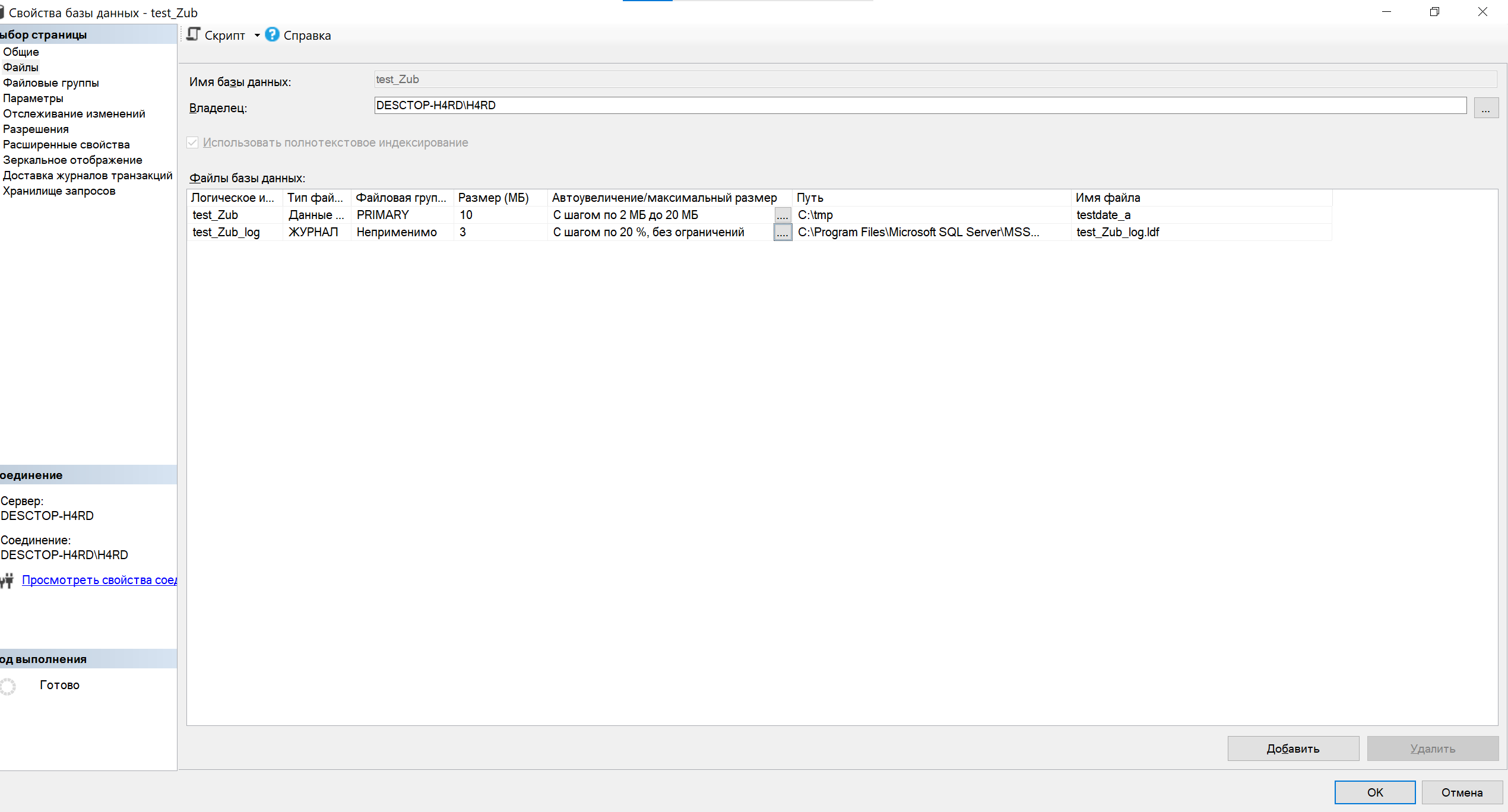


Рис.13. Создание БД test\_Zub

Упражнение 2

С помощью среды SQL Server Management Studio измените параметры журнала транзакций для базы данных test. Разрешите авто расширение файла журнала транзакций и установите его начальный размер в 3 Мбайта и увеличение размера с шагом по 20%.

На рис.14. Представлены измененные параметры журнала транзакций для базы данных test\_Zub.

Рис.14. Измененные параметры файлов БД test\_Zub.

Упражнение 3

Используя среду SQL Server Management Studio, создайте все четыре таблицы базы данных sample со всеми их столбцами. Нарисуйте инфологическую и даталогическую модель.

На рис 15. представлена диаграмма "сущность — отношение", на рис. инфологическая модель, а на рис. даталогическая модель.

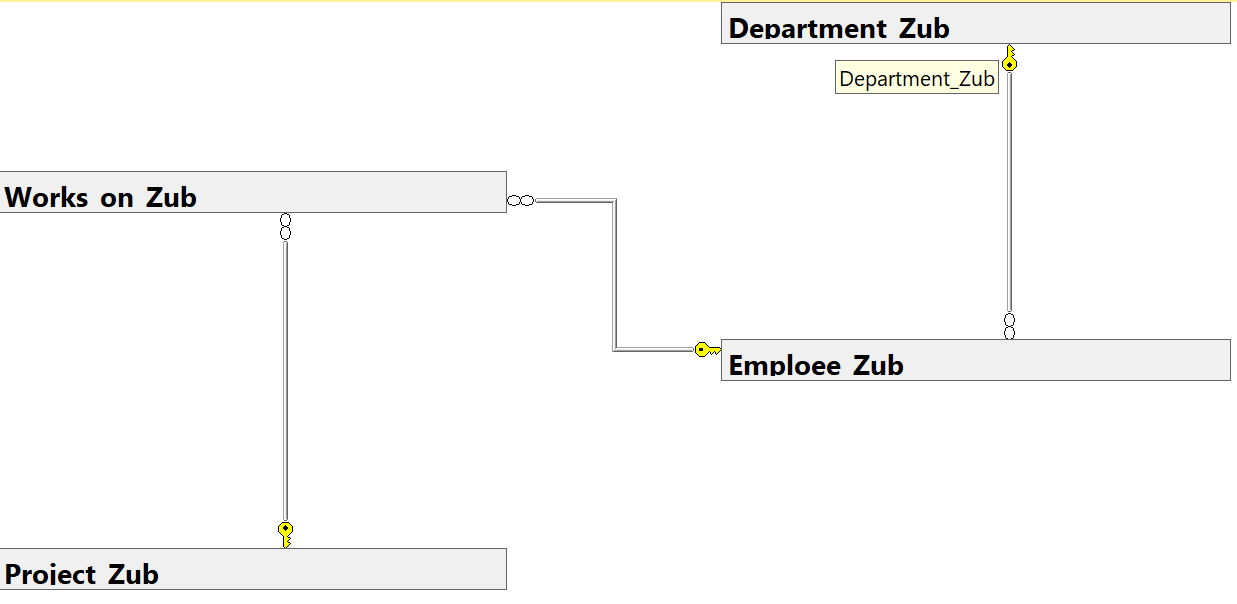


Рис. 15. Диаграмма "сущность — отношение" базы данных sample\_Zubarev.

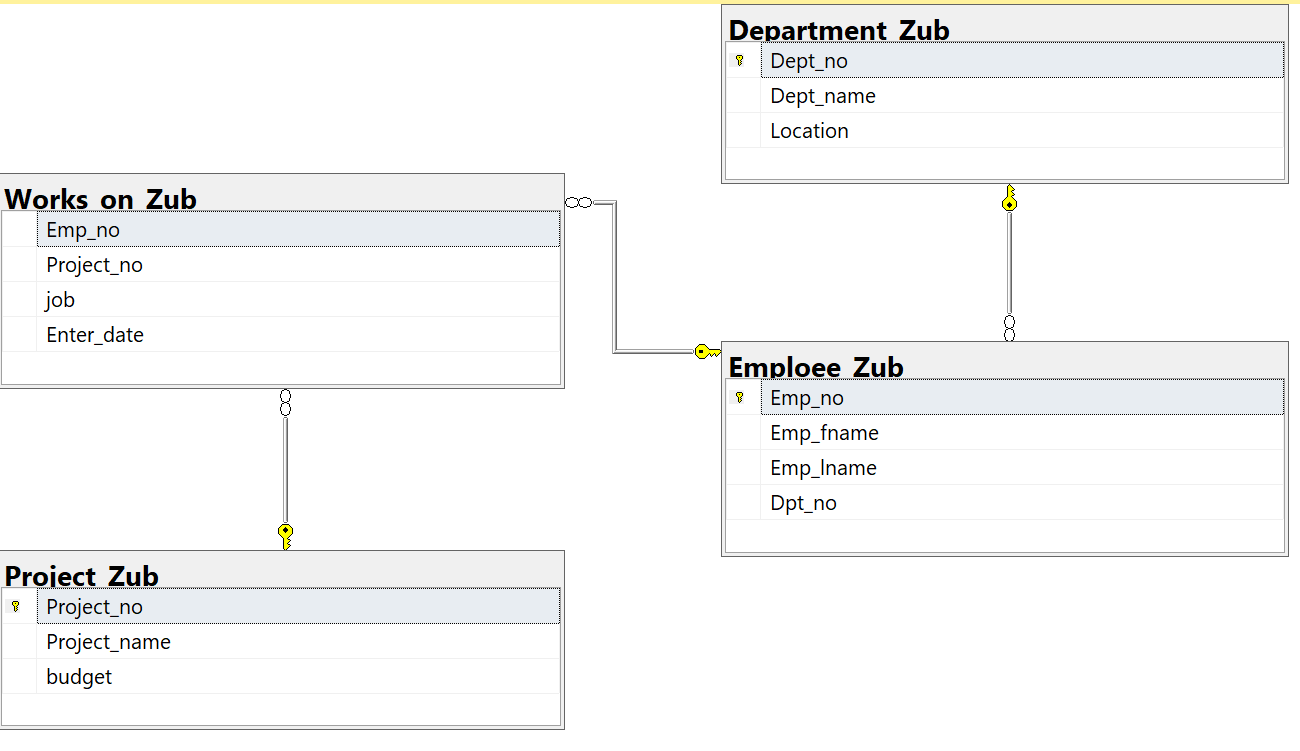


Рис.16. Инфологическая модель.

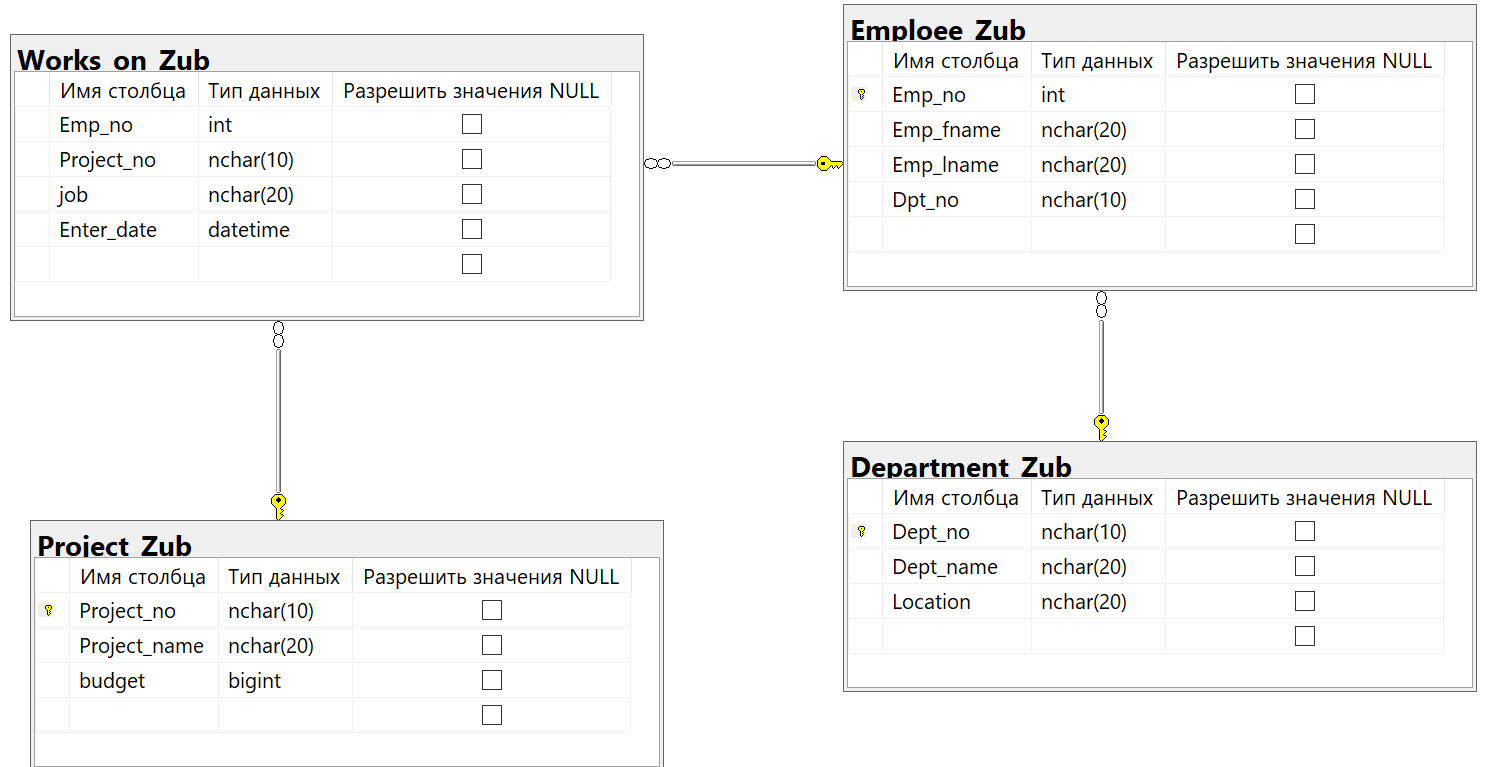


Рис.17. Даталогическая модель.

Упражнение 4

Используя среду SQL Server Management Studio, просмотрите, какие таблицы содержит база данных sample. Затем выберите в этой базе данных таблицу employee и просмотрите ее свойства.

На рис.18 показан обозреватель объектов, где можно увидеть созданные таблицы.

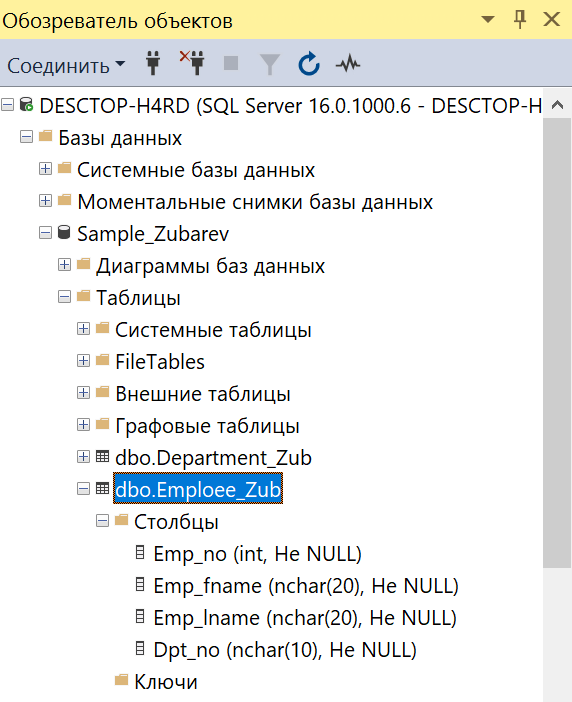


Рис.18. Таблицы, содержащиеся в БД sample.

Свойства таблицы employee показаны на рис.18.

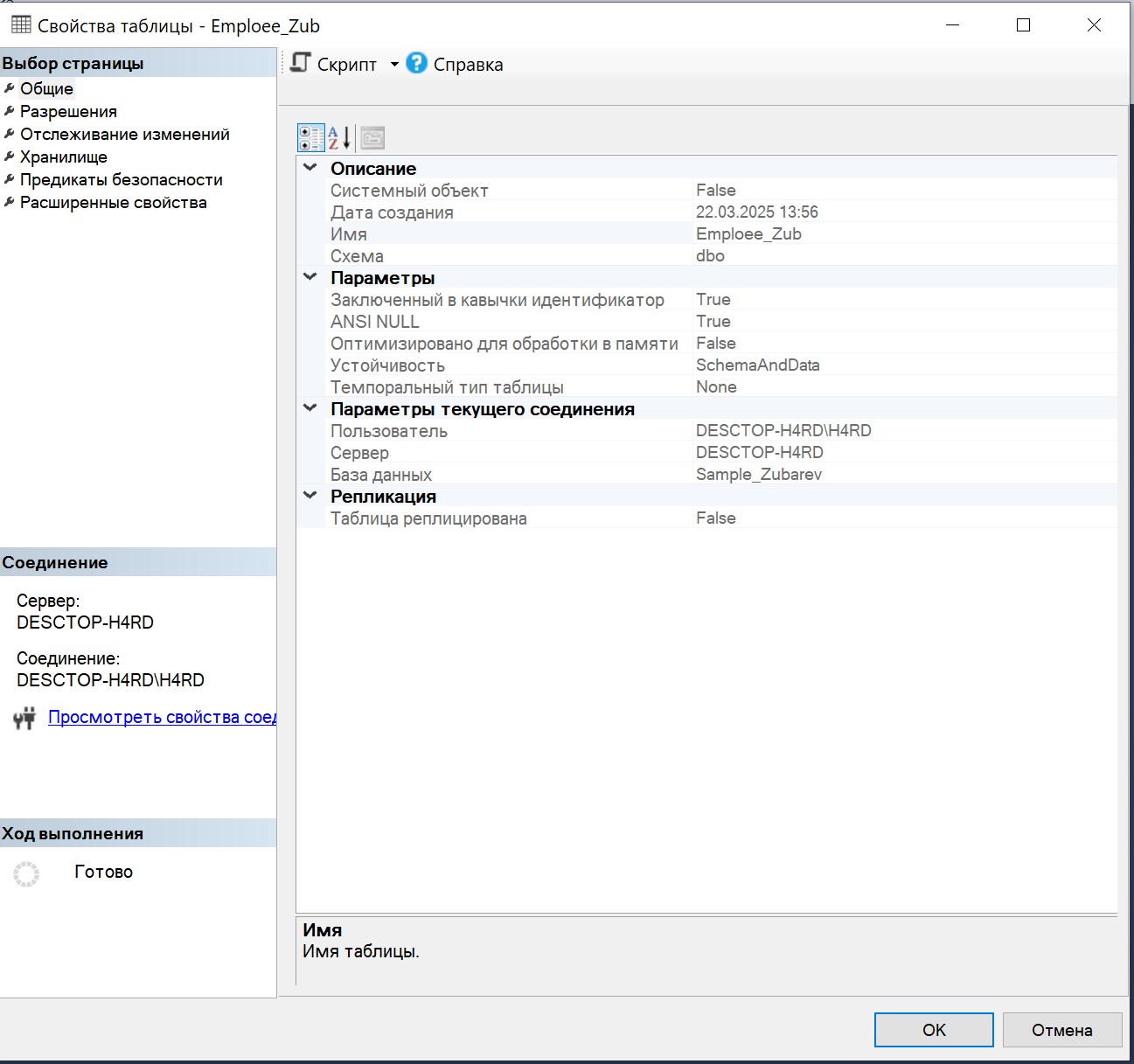


Рис.18. Свойства таблицы ch\_employee.

Упражнение 5

Введите и выполните в редакторе запросов следующую инструкцию Transact- SQL:

CREATE DATABASE test

Объясните причину сообщения об ошибке, выводимого в панели результатов.

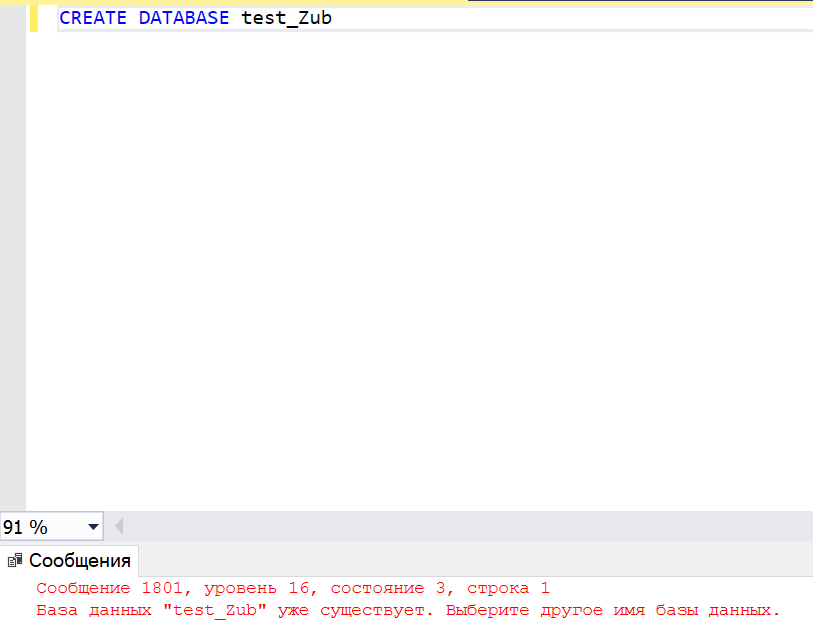


Рис. 18. Сообщение об ошибке.

Запрос не может быть выполнен, так как такая база данных уже существует на сревере.

Упражнение 6

Сохраните инструкцию Transact-SQL из упражнения 3 в файл C:\tmp\ createdb.sql.

Сохранение в нужный файл показано на рис.19.

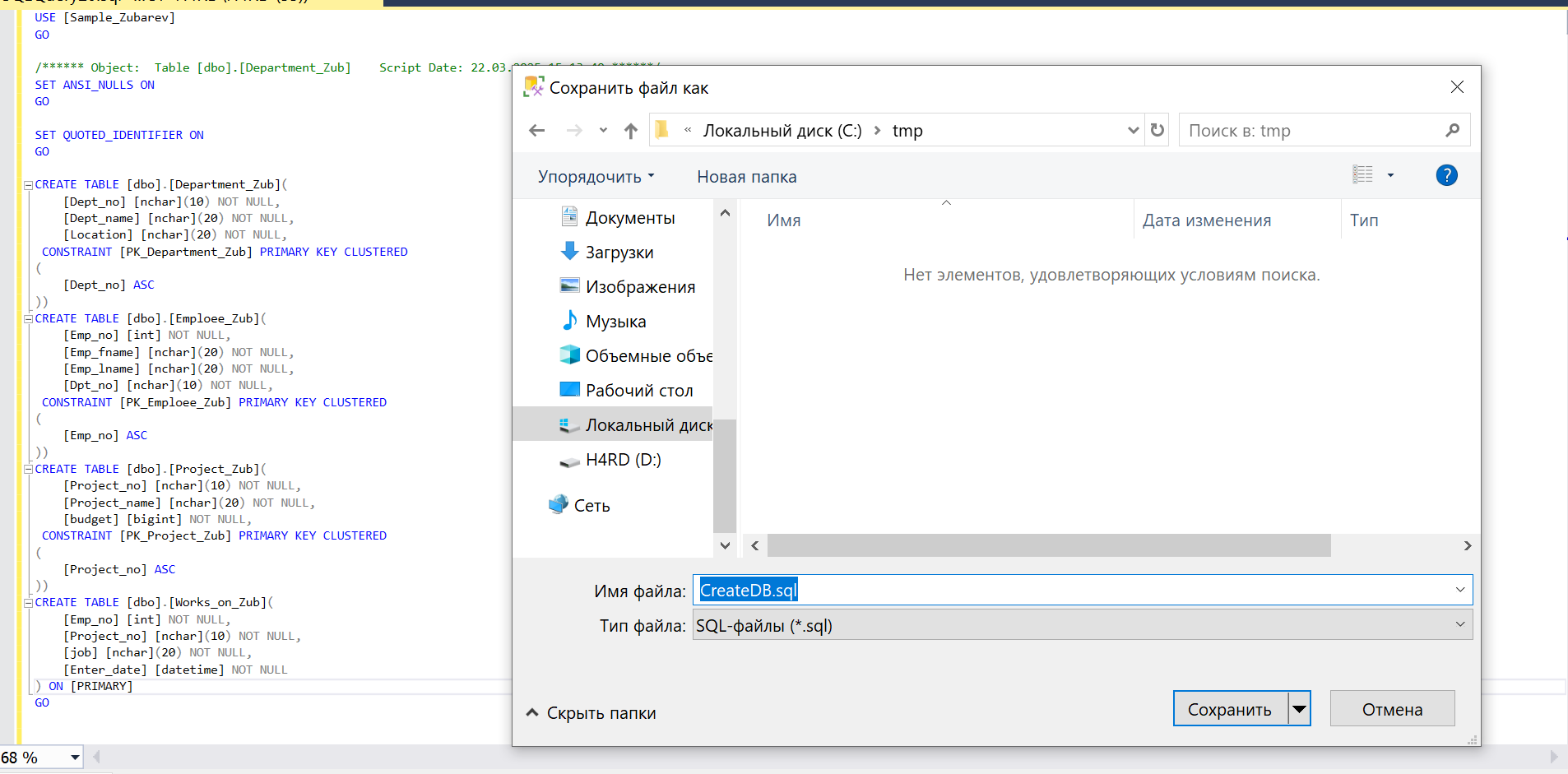


Рис.19. Сохраненная инструкция.

Упражнение 7

Как можно в редакторе запросов сделать базу данных test текущей базой данных?

Упражнение 8

Используя редактор запросов, сделайте базу данных sample текущей и выполните следующую инструкцию Transact-SQL: SELECT \* FROM employee.

На рис.19 показано выполнение данной инструкции.

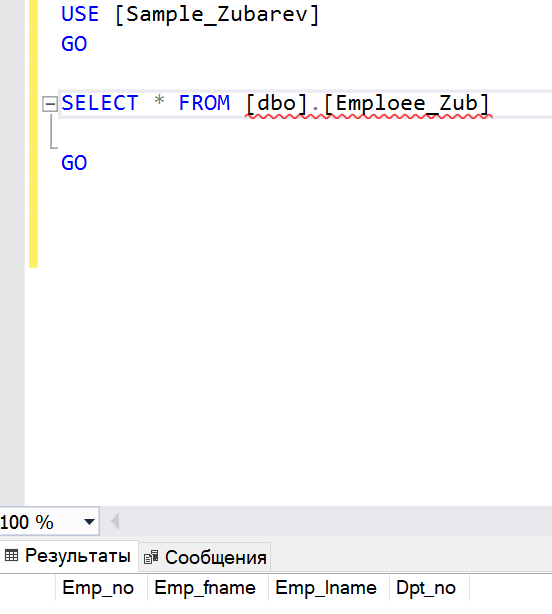


Рис. 19. Выполнение SELECT \* FROM employee.

Для остановки выполнения запроса необходимо в среде SSMS необходимо нажать на кнопку «остановить запрос».

Упражнение 9

Используя редактор запросов, измените вывод инструкции SELECT из упражнения 8 так, чтобы результаты выводились в виде текста, а не в виде таблицы.

Для вывода данных в виде текста существует кнопка “в виде текста”. Выполнение показано на рис.20.

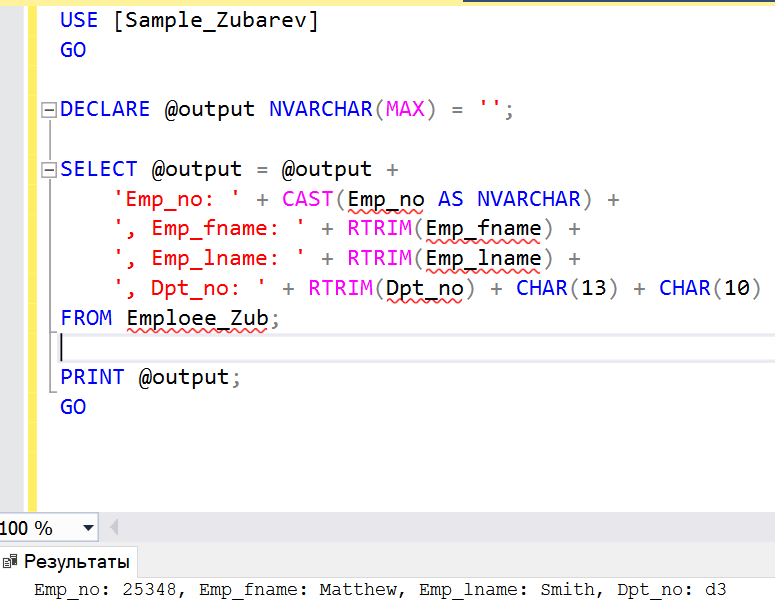


Рис. 20. Вывод в виде текста.

Упражнение 10

Какая разница между числовыми типами данных INT, SMALLINT и TINYINT?

Различие между этими типами данных заключаются в хранимых значениях

TINYINT:

Минимальное значение: 0

Максимальное значение: 255

SMALLINT:

Минимальное значение: -32,768

Максимальное значение: 32,767

INT:

Минимальное значение: -2,147,483,648

Максимальное значение: 2,147,483,647

Упражнение 11

Какая разница между типами данных CHAR и VARCHAR? Когда следует использовать первый, а не второй, и наоборот? Какая разница между типами данных CHAR и NCHAR? Сколько байт будет занимать строка ’Transact SQL’ для типов CHAR (15), VARCHAR(15), NCHAR(15), NVARCHAR(15)

Между типами данных char и varchar существуют следующие различия

CHAR:

1. Фиксированная длина: всегда занимает указанное количество символов, даже если строка короче.
2. Дополнение пробелами: если строка короче указанной длины, она дополняется пробелами до конца.

VARCHAR:

1. Переменная длина: занимает только столько места, сколько нужно для хранения строки (плюс 2 байта для хранения длины строки).
2. Экономия памяти: не дополняется пробелами, поэтому экономит место.

Использовать CHAR следует, если длина данных всегда одинакова (например, коды стран, номера телефонов фиксированной длины).

Использовать VARCHAR следует, если длина данных может меняться (например, имена, адреса, описания).

Различия между CHAR и NCHAR

CHAR:

1. Хранит символы в однобайтовой кодировке (например, ASCII).
2. Подходит для текста, который не содержит символов за пределами стандартной латиницы.
3. Занимает 1 байт на символ.

NCHAR:

1. Хранит символы в Unicode (двухбайтовая кодировка, например, UTF-16).
2. Подходит для текста, который может содержать символы различных языков (например, кириллица, иероглифы).
3. Занимает 2 байта на символ.

Различия между VARCHAR и NVARCHAR

VARCHAR:

1. Хранит символы в однобайтовой кодировке.
2. Подходит для текста, который не содержит символов за пределами стандартной латиницы.
3. Занимает 1 байт на символ (плюс 2 байта для хранения длины строки).

NVARCHAR:

Хранит символы в Unicode.

1. Подходит для текста, который может содержать символы различных языков.
2. Занимает 2 байта на символ (плюс 2 байта для хранения длины строки).

5. Сколько байт будет занимать строка 'Transact SQL'?

Строка 'Transact SQL' состоит из 12 символов (включая пробел).

CHAR (15):

Фиксированная длина: 15 байт.

Строка дополняется пробелами до 15 символов.

Занимает: 15 байт.

VARCHAR (15):

Переменная длина: 12 байт (для символов) + 2 байта (для хранения длины строки).

Занимает: 14 байт.

NCHAR (15):

Фиксированная длина: 15 символов × 2 байта = 30 байт.

Строка дополняется пробелами до 15 символов.

Занимает: 30 байт.

NVARCHAR (15):

Переменная длина: 12 символов × 2 байта = 24 байта + 2 байта (для хранения длины строки).

Занимает: 26 байт.

Упражнение 12

Как настроить столбец типа данных DATE для ввода значений в формате 'гггг/мм/дд'?

Порядок составляющих месяца, дня и года можно изменять с помощью инструкции setdateformat.

Для ввода данных типа DATE в формате ‘ггг/мм/дд’ необходимо выполнить скрипт   
SET DATEFORMAT ymd; -- 'год/месяц/день'

Упражнение 13

Используя системные функции, узнайте идентификационный номер базы данных test, которую создали на прошлой лабораторной работе. Результат представлен на рис.21.

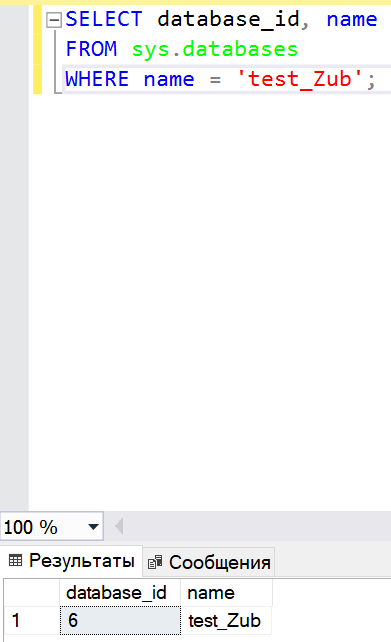


Рис.21. Идентификационный номер БД.

Упражнение 14

Используя системные переменные, узнайте текущую версию программного обеспечения системы базы данных и используемый в программном обеспечении язык. Выполнение показано на рис. 22.

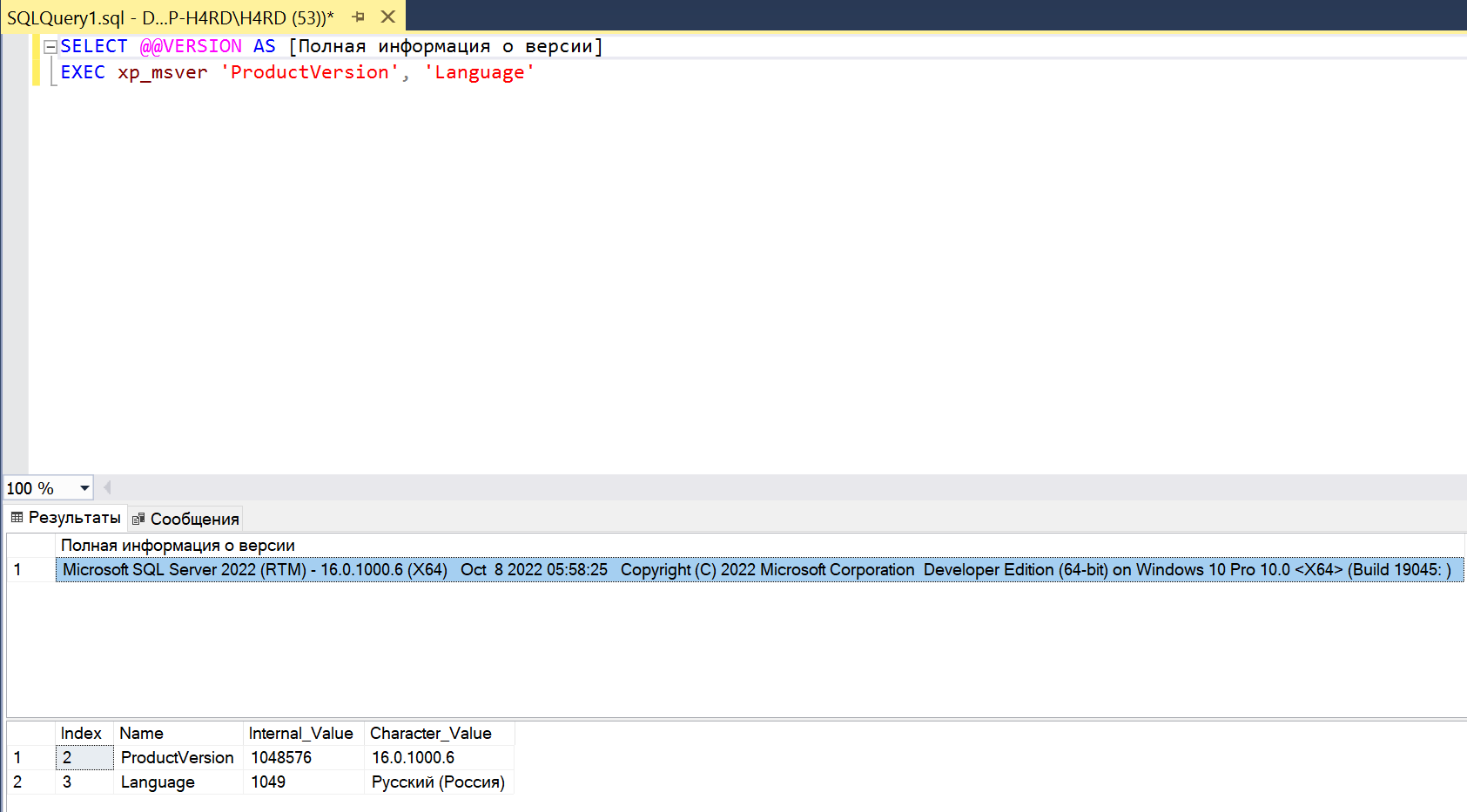


Рис.22. Текущая версия БД и используемый язык.

Упражнение 15 (выполняется вручную)

Используя битовые операторы &, | и ^, выполните следующие операции над битовыми строками:

(11100101) & (01010111)

**11100101**

**01010111**

**--------**

**01000101**

* 1 & 0 = 0
* 1 & 1 = 1
* 1 & 0 = 0
* 0 & 1 = 0
* 0 & 0 = 0
* 1 & 1 = 1
* 0 & 1 = 0
* 1 & 1 = 1

Ответ: 01000101

(10011011) | (11001001)

**10011011**

**11001001**

**--------------**

**11011011**

* **1 | 1 = 1**
* **0 | 1 = 1**
* **0 | 0 = 0**
* **1 | 0 = 1**
* **1 | 1 = 1**
* **0 | 0 = 0**
* **1 | 0 = 1**
* **1 | 1 = 1**

Ответ: 11011011

(10110111) ^ (10110001)

**10110111**

**10110001**

**--------------**

**00000110**

* **1 ^ 1 = 0**
* **0 ^ 0 = 0**
* **1 ^ 1 = 0**
* **1 ^ 1 = 0**
* **0 ^ 0 = 0**
* **1 ^ 0 = 1**
* **1 ^ 0 = 1**
* **1 ^ 1 = 0**

Ответ: 00000110

Упражнение 16

Какими будут результаты следующих выражений? (Выражение А — числовое, а В — логическое.)

**1. A + NULL → NULL**

**2. NULL = NULL → NULL (не TRUE!)**

**3. Если B = TRUE, то B OR NULL → TRUE**

**4. Если B = FALSE, то B OR NULL → NULL**

**5. Если B = TRUE, то B AND NULL → NULL**

**6. Если B = FALSE, то B AND NULL → FALSE**

Упражнение 17

В каких случаях можно использовать как одинарные, так и двойные кавычки для определения строковых и временных констант?

1. Только при отключенном QUOTED\_IDENTIFIER (не рекомендуется).
2. В динамическом SQL.

Упражнение 18

Что такое идентификатор с ограничителями и когда требуется использовать идентификаторы этого типа?

Идентификатор с ограничителями— это имя объекта базы данных (таблицы, столбца, индекса и т.д.), заключенное в специальные символы для корректной обработки сервером. В SQL Server используются два типа ограничителей:

1. Квадратные скобки: [Order]
2. Двойные кавычки: "Order" (требует настройки QUOTED\_IDENTIFIER)

Когда требуется использовать ограничители?

1. Когда имя содержит пробелы или спецсимволы
2. Совпадение с зарезервированными словами SQL
3. Использование Unicode-символов или нестандартных символов
4. Совпадение с системными функциями
5. При динамическом SQL или миграции из других СУБД