Abylkas Saginov Karaganda Technical University

Information Technologies and Security Department

**Laboratory work №1**

**Discipline**: Database management systems

**Topic**: Creating a Database with MS SQL Server

**Prepared by:**

Student of SIB-22-5 group

Batyrkhanov Arman

**Approved by:**

Senior Lecturer

Klyuyeva Ye.G.

**Karaganda 2024**

1. **Exploring the SQL Server Configuration Utility**

1.1 Run the SQL Server Configuration Manager utility and use it to determine the list of services running on the server. Write this list to the report.

1.2 On the server with MS SQL Server 2008 installed, use the Services utility to determine the parameters for starting the MS SQL Server services and write them to the report. (If you do not have access to the Services utility, use SQL Server Configuration Manager).

1.3 Determine which network libraries can be used to establish a connection to MS SQL Server (see example in Fig. 1). Which libraries are active at startup? Write this information to the report.

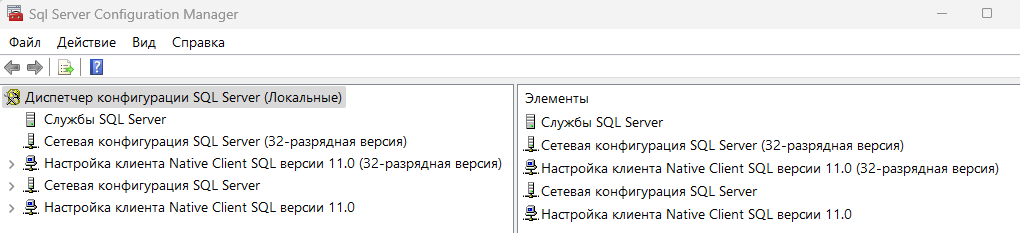


Fig. 1 – Server-side protocols through which it can be connected

1.4 Using SQL Server Configuration Manager, determine which network libraries the client can use to connect to MS SQL Server (see example in Fig. 2). Write the list of libraries to the report.

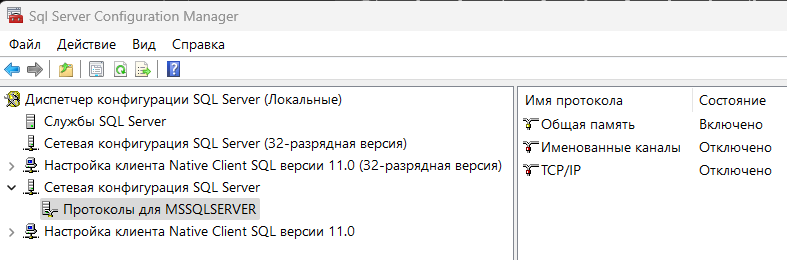


Fig. 2 – The client-side protocols through which it can connect to the server

1. **Establish a connection to the SQL server.**

2.1 On your workstation, launch SQL Server Management Studio and select the logical name of the server running on your computer from the list. If the required server is not in the list, you can select <Browse for more…> and find the required server in the list of servers to which a connection can be made.

2.2 Connect to the server using MS SQL Server authentication tools.

2.3 In order to write a new query, you must execute the New Query command located on the SQL Server Management Studio toolbar. This will open a new tab that provides the following options:

- a header that specifies the logical name of the server, the current database, and the name of the user who established the connection;

- a query area used to enter queries transmitted to MS SQL Server;

- a results area that displays the results of the query, and the display method is set by the Messages (as text) and Results (as a table) buttons, respectively.

2.4 Using the SELECT @@version command, determine and write to the report information about the version of MS SQL Server and the operating system used (the query result should be displayed in text form).

Note: To execute a query, you must execute the Query – Execute (F5) command, and to analyze the correctness of its syntax, you can use the Query – Parse (Ctrl+F5) command.

SQL Server Management Studio allows you to open several query windows and work with several databases simultaneously. Each window establishes its own connection to MS SQL Server based on different user accounts and their passwords. To create a new connection, use the File – New – Database Engine Query command.

The contents of the current connection's query area can be saved to a file on an external drive using the File – Save command.

2.5 Using the Object Explorer panel, determine the names of the supported databases and which server databases are system databases (to do this, expand the Databases node in the Object Explorer panel). Write this information to the report.

1. **Study the MS SQL Server configuration parameters.**

The MSSQLServer service can be configured either by a special stored procedure executed in the SQL Server Management Studio utility, or graphically using the same utility. The choice of method does not matter, since the graphic method provides access to system data using the same stored procedure, only in a more visual form. 3.1 To change the service parameters using SQL Server Management Studio, select the desired server in Object Explorer and select the Properties command in the context menu. In the dialog box that appears, you can configure all the necessary parameters. Display the list of server parameters (example Fig. 3).

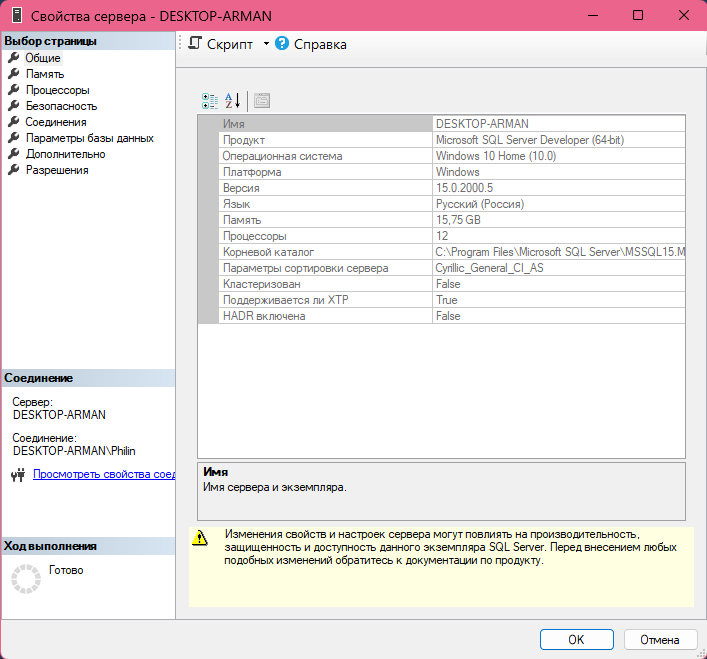


Fig. 3 – Properties of MS SQL Server 2016

The General tab displays basic information about the system: the operating system version, memory size, number of processors, etc., as well as server service startup parameters.

The Memory tab allows you to manage memory allocation for MS SQL Server actions: either dynamic memory management or set a fixed size.

The Security tab is used to define the user authentication type and server access audit parameters. You can configure the server to use a specific account under which the MSSQLServer service will be launched.

The Connections tab allows you to configure client connections to the server. The maximum number of users that can simultaneously connect to the server. If zero is specified, their number is 32767.

The Advanced tab contains some general server settings. For example, it defines the default language for server messages or regulates support for the year 2000, which determines how the last two digits of the year will be interpreted.

The Database Settings tab is used to specify settings for newly created databases: index parameters and parameters for working with backup devices, and the database recovery time.

3.2 Determine and record in the report the server root directory, the number of processors in the system, the user authentication type and the maximum number of users supported by the server.

3.3 Explore the other MS SQL Server properties available in this dialog.

1. **Create a database**

Create a database named Stud\_Batyrkhanov\_Arman\_Kanatovich\_1 using MS SQL Server 2016 DBMS with a log using SQL Server Management Studio and named Stud\_Batyrkhanov\_Arman\_Kanatovich\_2 using Query Editor and write the results of executing the sp\_helpdb procedure to the report…. For the databases you created

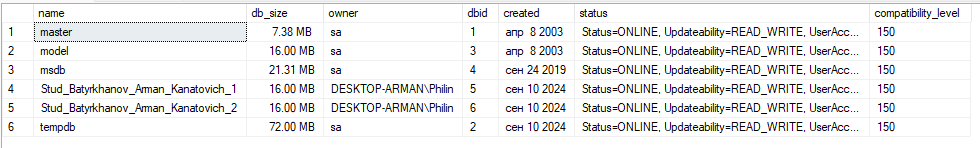


Fig. 4 – Databases

1. **Backup**

Create a backup copy of one of the databases you created and display the result of the BACKUP statement in the report.

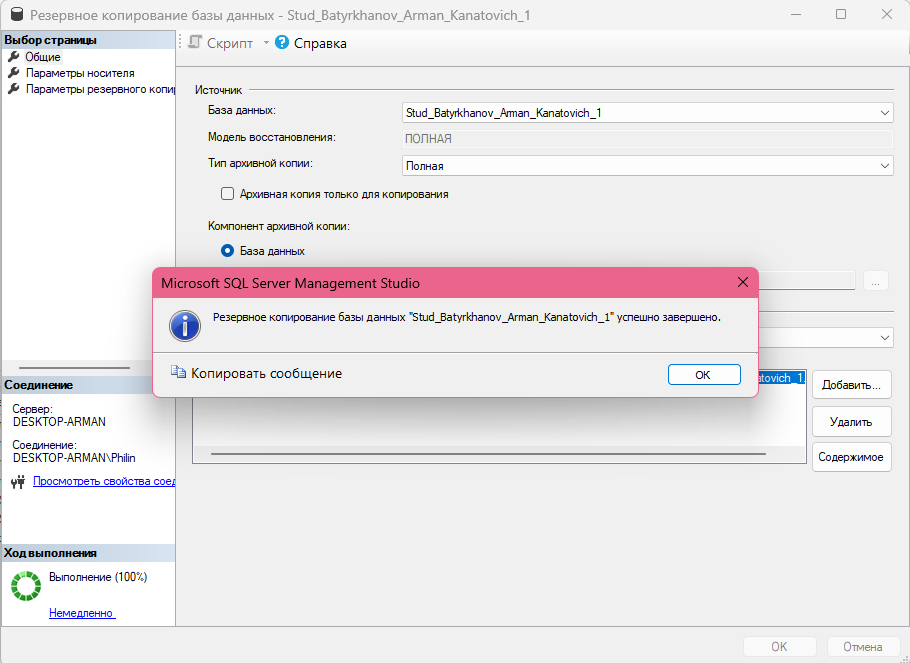


Fig. 5 – Creating a backup copy

1. **Renaming the database**

Rename the database you created to Stud\_Batyrkhanov\_Arman\_Kanatovich and display the result of the renaming operator in the report.

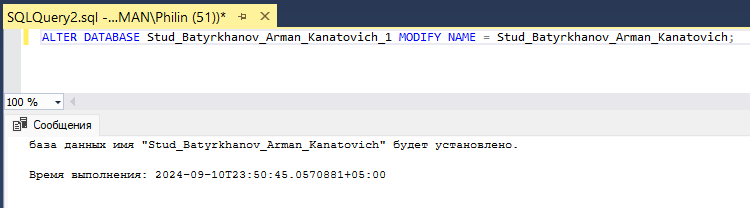


Рис. 6 – Renaming the database

1. **Database compression**

Determine the disk space information occupied by the database you have created. Compress the database so that it contains only 25% of the space currently available to it.

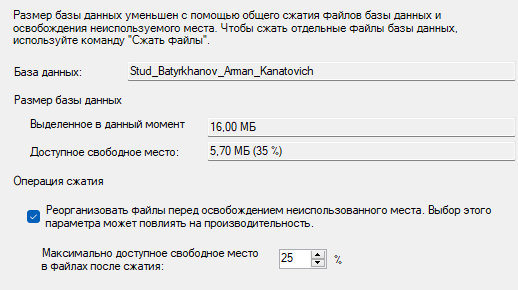


Fig. 7 – Database compression

1. **Deleting the database**

Delete the database you created named Stud\_Batyrkhanov\_Arman\_Kanatovich\_2 and display the result of the delete statement in the report.

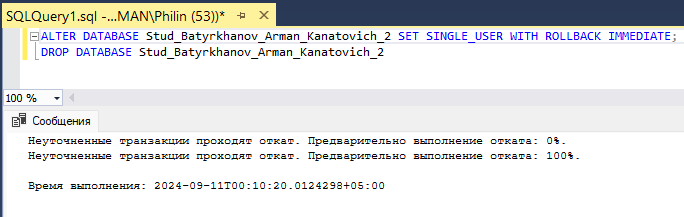


Fig. 8 – Deleting a database

1. **Disconnecting/connecting the DB**

Disconnect/connect the Stud\_Batyrkhanov\_Arman\_Kanatovich DB you created from the server. If the DB was created on a hard drive, then move it to a backup drive and display the operator execution result in the report.

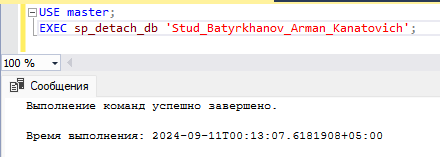


Fig. 9 – Disconnecting the DB from the server

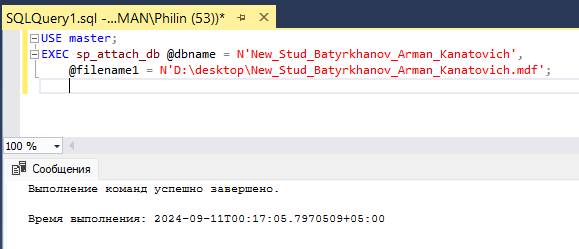


Fig. 10 – Connecting the DB to the server

**Control questions**

1. *Characteristics of MS SQL Server 2016 DBMS versions*

Enterprise: Full functionality for large enterprises.

Standard: Basic data management functions.

Express: Free version with limitations.

Developer: Full functionality for development and testing.

Web: Optimized for web applications.

1. *MS SQL Server 2016 Toolkit Features*

SSMS: Database Management.

SSDT: Database Development and ETL Processes.

Profiler: Query Monitoring.

SQL Server Agent: Task Automation.

SSRS: Report Creation.

SSIS: Data Integration.

SSAS: Data Analysis.

1. *What is SQL Server 2016 Database?*

A structured data store with support for transactions, indexes, keys, constraints, and security mechanisms.

1. *SQL Server 2016 Database Objects*

Tables, views, stored procedures, triggers, functions, indexes.

1. *Methods for creating databases in MS SQL Server 2016 DBMS?*

SSMS: Graphical interface.

T-SQL: CREATE DATABASE command.

SQL Server Management Objects (SMO): Automation via scripts or programs.