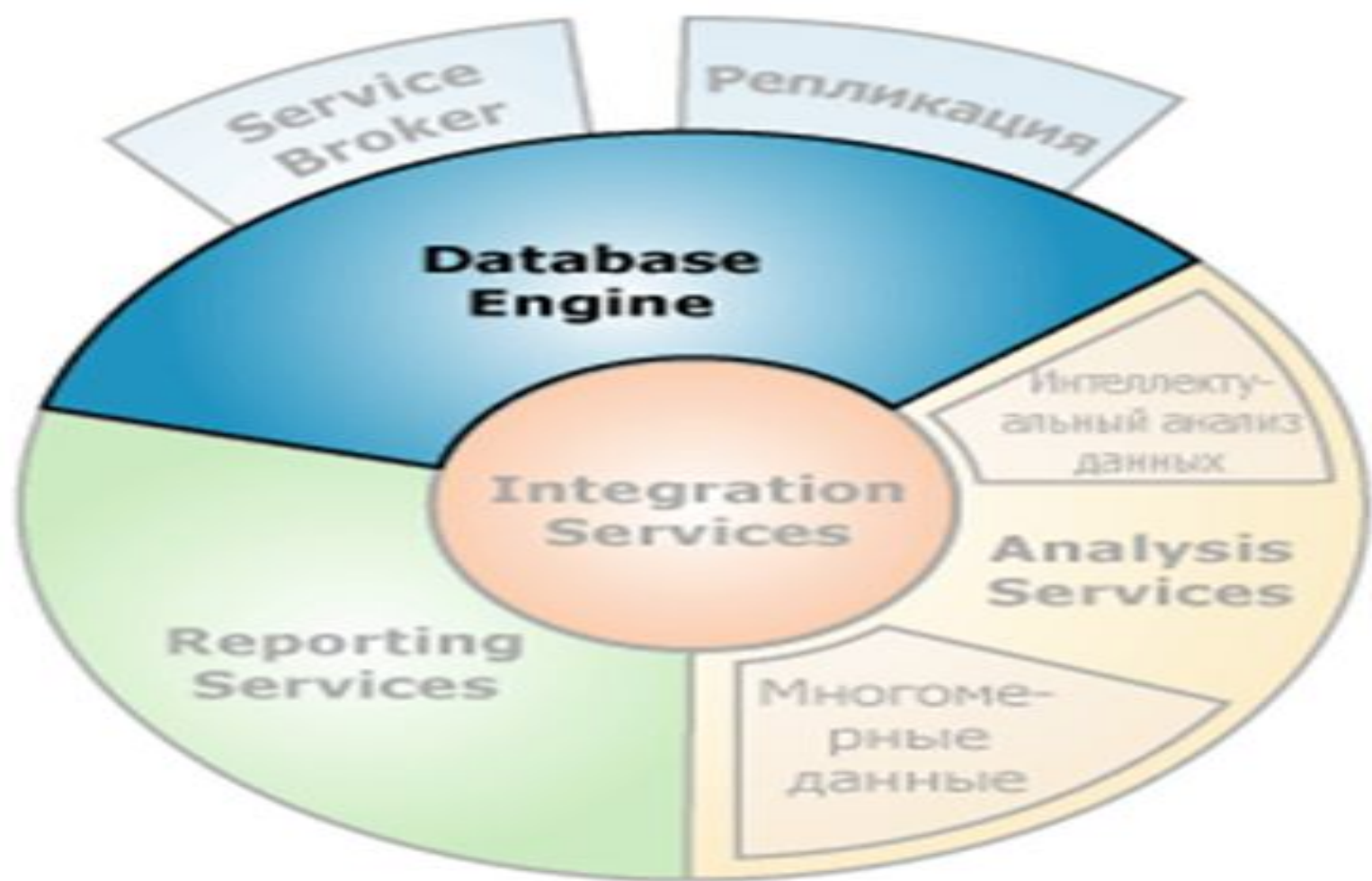


# **Створення та технологія роботи з БД в середовищі реляційної СКБД Microsoft SQL Server**

- 1. Характеристика СКБД Microsoft SQL Server**
- 2. Створення БД в середовищі SQL Server Management Studio.**
- 3. Створення таблиць та побудова діаграми (схеми) БД в середовищі SQL Server Management Studio.**
- 4. Характеристика типів даних**

Microsoft SQL Server – система керування реляційними БД, розроблена корпорацією Microsoft. Основна мова, що використовується – мова запитів – **Transact-SQL**, створена Microsoft разом з Sybase. Transact-SQL є реалізацією стандарта ANSI/ISO по



# Database Engine

Основна служба для зберігання, оброблення та захисту даних.

використовується для створення реляційних баз даних для оперативної обробки транзакцій та оперативної аналітичної обробки даних. Тобто для створення таблиць, для зберігання даних і об'єктів баз даних, таких як індекси, представлення і процедури, що зберігаються, для перегляду і захисту даних і для управління ними

## **Analysis Services**

- Підтримується багатовимірний оперативний аналіз даних (OLAP)
- Підтримують інтелектуальний багатовимірний аналіз даних (Data Mining)

# Integration Services

компонента для створення рішень з інтеграції даних, в тому числі пакетів для зберігання даних, які забезпечують видобування, перетворення і завантаження даних

# Reporting Services

Служби Reporting Services пропонують засоби створення корпоративних звітів з підтримкою веб-інтерфейсу, які дозволяють включати в звіти дані з різних джерел, публікувати звіти в різних форматах, а також централізовано управляти безпекою і підписками

# SQL Server Management Studio

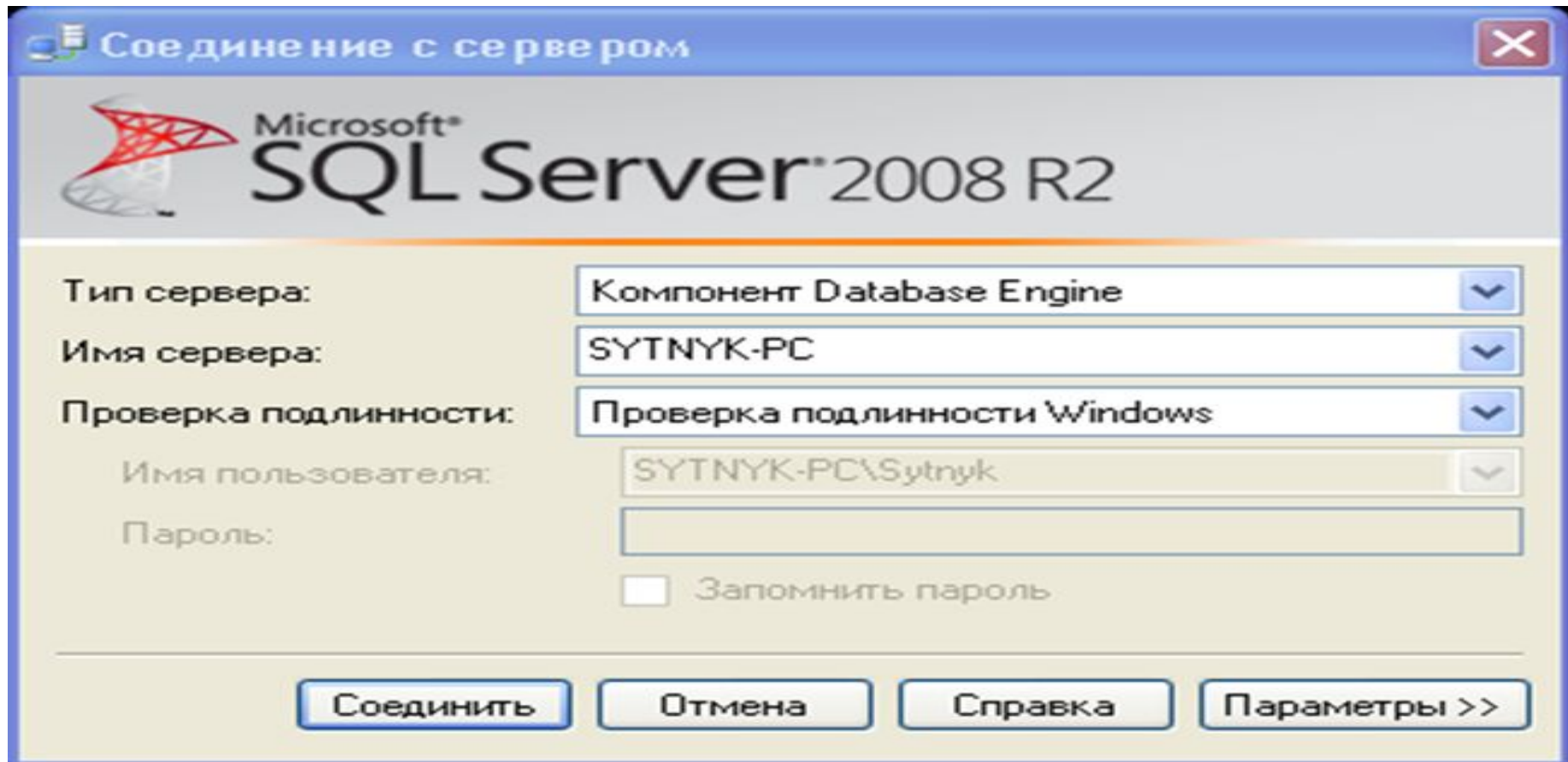
інструмент для доступу, налагодження, управління, адміністрування і розробки всіх компонент SQL Server. SQL Server Management Studio має графічні засоби та набір редакторів - сценаріїв для доступу до SQL Server розробників і адміністраторів.

працює зі всіма компонентами SQL Server, наприклад зі службами Reporting Services, Integration Services



# Запуск SQL Server Management Studio

1. Пуск/Все программы/Microsoft SQL Server 2008/SQL Server Management Studio
2. З'явиться вікно з'єднання з сервером (Connect to Server)



Соединение с сервером

Microsoft®  
SQL Server® 2008 R2

Тип сервера:

Имя сервера:

Проверка подлинности:

Имя пользователя:

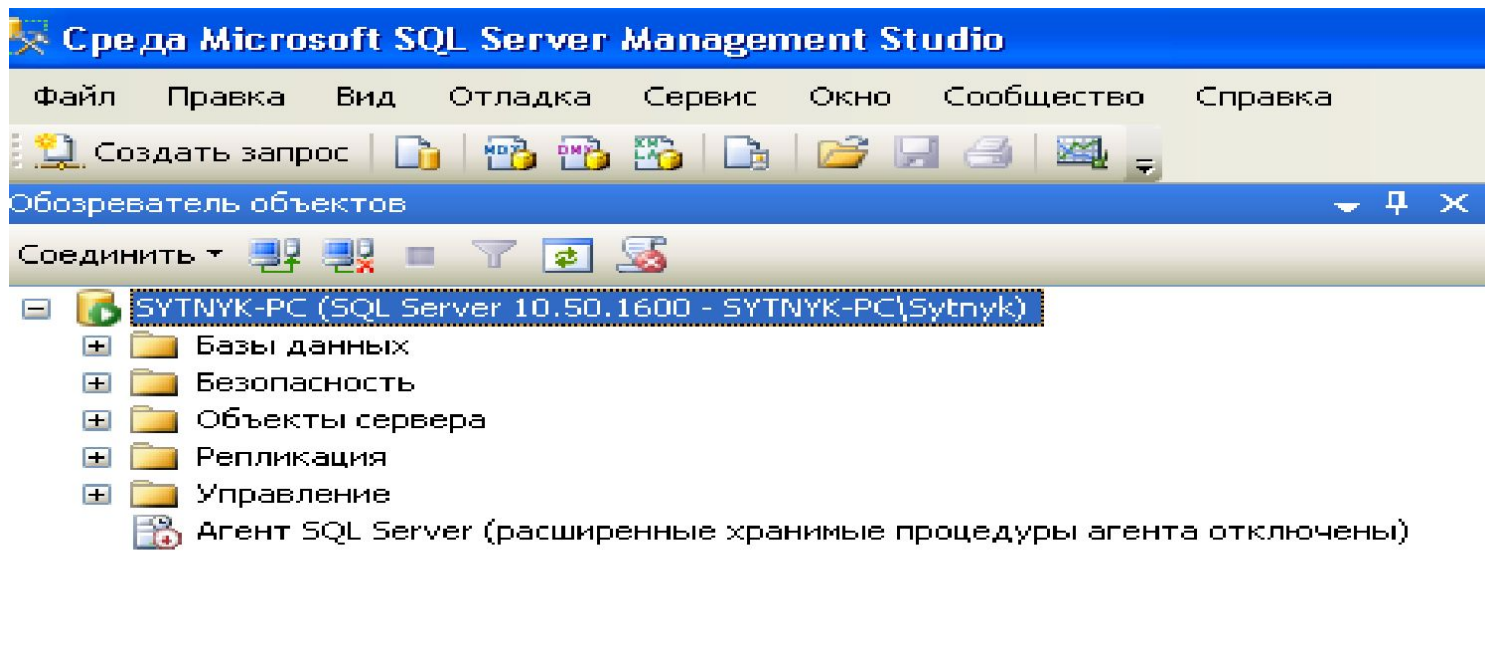
Пароль:

Запомнить пароль

# З'єднання з сервером

- ▣ Тип сервера (Server type) – Database Engine
- ▣ Ім'я сервера (Server name) – місце розташування Microsoft SQL Server.
- ▣ Аутентифікація (Authentication) вказується метод авторизації. Є два варіанти : **Database Authentication** і **Windows Authentication**.
- ▣ Якщо був обраний метод авторизації **Database Authentication**, то в полях **User Name** і **Password** необхідно ввести відповідно ім'я користувача та пароль.

- При використанні методу **Windows Authentication** з'єднання із СКБД Microsoft SQL Server буде відбуватися з користувачем, який увійшов до ОС Windows.
- В обох випадках користувач повинен мати права на створення БД.
- Після заповнення всіх полів натисніть **Connect**. Якщо з'єднання пройшло успішно, то зліва у вікні **Обозреватель объектов (Object Explorer)** з'явиться структура **SQL Server**



**Вікно Microsoft SQL Server Management Studio після з'єднання із Microsoft SQL Server**

- ▣ Є два способи: конструктор Object Explorer, оператори Transact-SQL
- ▣ 1-й спосіб - натиснути праву кнопку мишки на рядку **Базы данных (Databases)** і обрати опцію **Создать базу данных (New Database)**.
- ▣ У наступному вікні в полі **Database name** ввести ім'я нової БД та натиснути **ОК**.

**Максимальна кількість БД -32767.**

Створення БД та таблиць виконується мовою визначення даних **DDL** (Data Definition Language). Основні команди:

***Create Database*** - створення БД.

***Drop Database*** - вилучення бази даних.

***Alter Database*** - зміна параметрів БД.

***Create Table*** - створення таблиці.

***Drop Table*** - вилучення таблиці.

***Alter Table*** - зміна параметрів таблиці (наприклад, додавлення чи вилучення стовпчика, чи зміна його типів даних).

***Create Index*** - створення індекса.

***Drop Index*** - вилучення індекса.

***Alter Index*** - зміна параметрів індекса.

## ▣ 2-й спосіб

**CREATE DATABASE db\_name**

**[ON [PRIMARY] {file\_spec1}, ....]**

**[LOG ON {file\_spec2}, ....]**

**[COLLATE collaction\_name]**

**[FOR {ATTACH | ATTACH\_REBUILD\_LOG} ]**

**Опція ON** – задає всі файли БД явно.

**file\_spec** – задає додаткові опції такі як логічне ім'я файла, фізичне ім'я та розмір

**PRIMARY** – задає перший найважливіший файл, який містить системні таблиці та іншу внутрішню інформацію про БД. Якщо ця опція відсутня, то перший файл у списку

- ▣ **LOG ON** - файл для розміщення протоколу транзакцій. Якщо опція LOG ON відсутня, то протокол транзакцій створюється за змовчанням.
- ▣ **COLLATE** – задає порядок сортування БД.
- ▣ **FOR {ATTACH | ATTACH\_REBUILD\_LOG}** – вказує, що БД створена шляхом приєднання набору файлів операційної системи.

Приклад створення БД без явних специфікацій:

**USE master**

**CREATE DATABASE sample**

Система за змовчанням створить два файли: логічний з іменем – sample, розмір 2 Мбайта і протокол транзакцій з іменем sample\_log – розмір 1Мбайт.



# Створення таблиць та побудова діаграми (схеми) БД

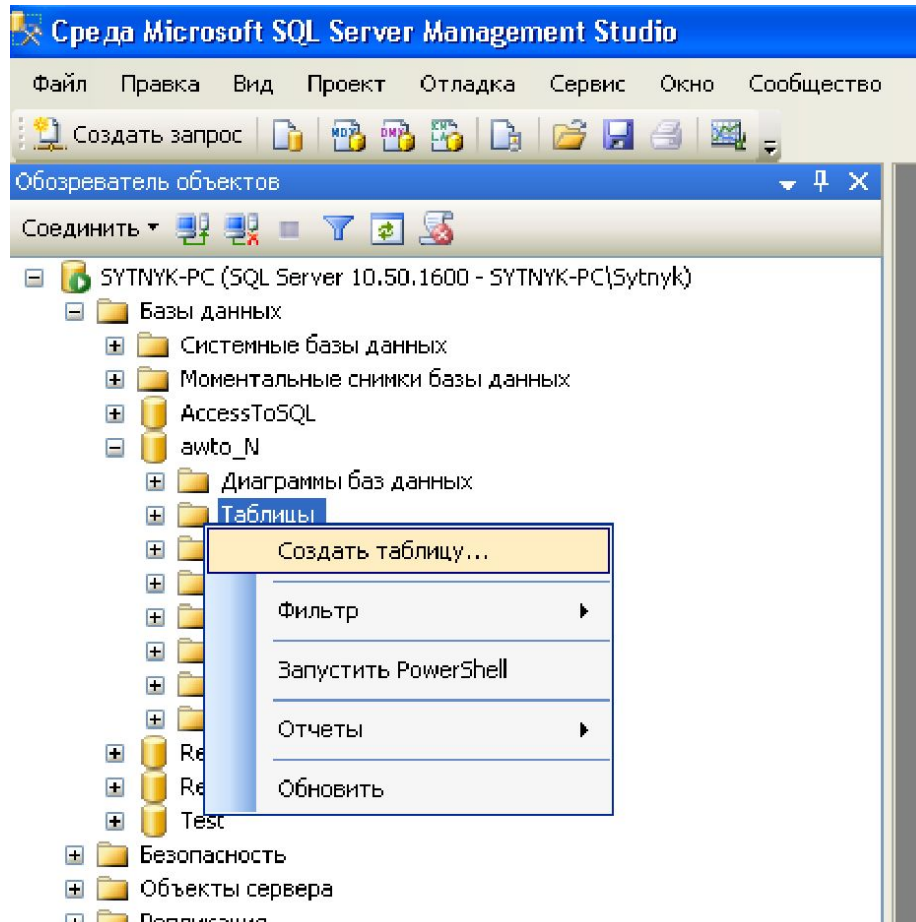
SQL Server 2008 підтримує в базі даних до двох **мільярдів** таблиць.

Кожна таблиця може вміщувати до **1024** стовпчиків. Імена таблиць і стовпчиків повинні бути унікальними в межах таблиці. Максимальне число байтів в рядку має складе до **8 060**.

Таблиці також можна створювати двома способами: **за допомогою конструктора та з використанням оператора CREATE TABLE.**

# Створення таблиці за допомогою конструктора

Необхідно натиснути ліву клавiшу мишки на імені бази даних і у меню, що з'явиться в якому вибрати опцію **Создать таблицу** активізувавши правою кнопкою мишки опцію **Таблицы**.



# Среда Microsoft SQL Server Management Studio

Файл Плавка Вид Проект Отладка Конструктор таблиц Сервис Окно Сообщество Справка

Создать запрос



Обозреватель объектов

Соединить

SYTNYK-PC (SQL Server 10.50.1600 - SYTNYK-PC\Sytnyk)

Базы данных

Системные базы данных

Моментальные снимки базы данных

AccessToSQL

awto\_N

Диаграммы баз данных

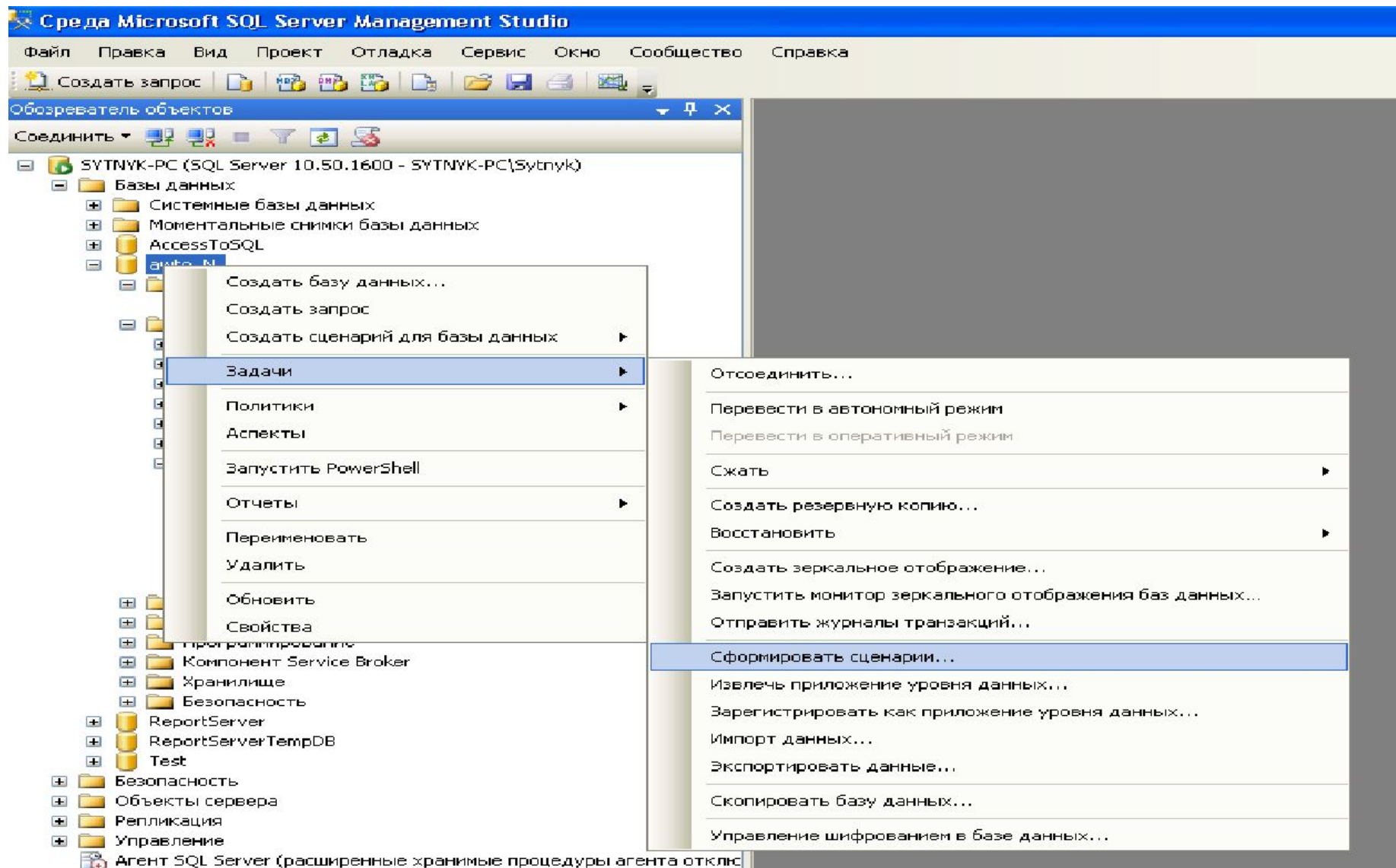
Таблицы

SYTNYK-PC.awt...- dbo.Table\_1\*

Имя столбца	Тип данных	Разрешит...
Kod_avto	numeric(18, 0)	<input type="checkbox"/>
Name_avto	char(100)	<input type="checkbox"/>
		<input type="checkbox"/>

**Вікно конструктора для створення таблиць**

# Перегляд сценарію створення таблиці



# Створення таблиці на мові SQL– базовий варіант

```
CREATE TABLE   table_name  
(column_name1 <data_type1> [  
NULL | NOT NULL ]  
[{,column_name2 <data_type2> [  
NULL | NOT NULL ]}.... ]);
```

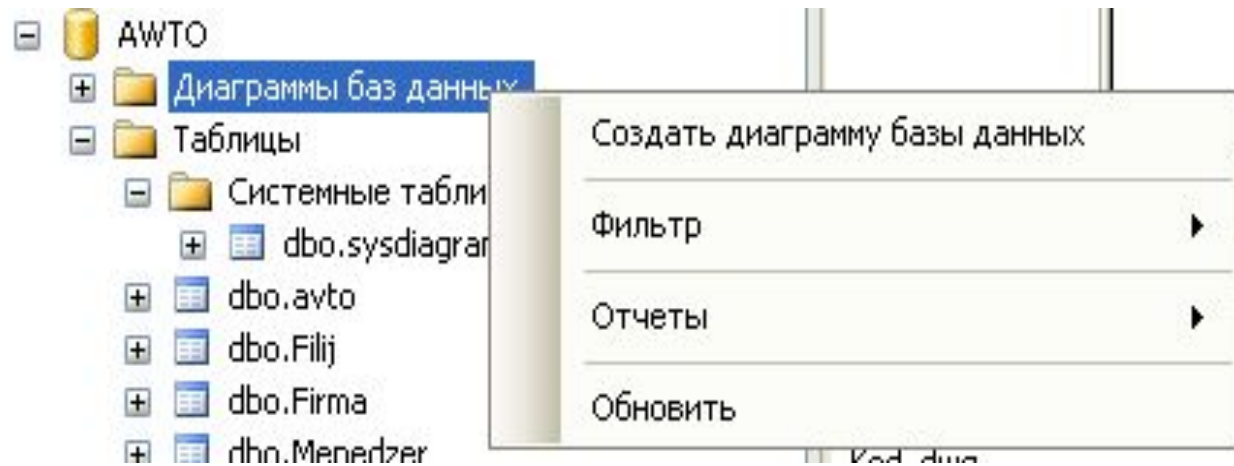
SQL Server підтримує зв'язки між таблицями 1:1 та 1:Б.

Необхідною умовою зв'язування двох таблиць є наявність в них спільних атрибута, які представлені однаковою типом і форматом.

Батьківською таблицею є таблиця з первинними ключами, дочірньою – таблиця з відповідними вторинними (зовнішніми) ключами.

Для створення діаграми необхідно активізувати опцію

**Создать диаграмму базы данных** , в меню **Диаграммы баз данных** .



Відмітивши мишкою дугу і активізувавши в меню **Вид /Окно свойства** можна задати правила для операцій INSERT і UPDATE.

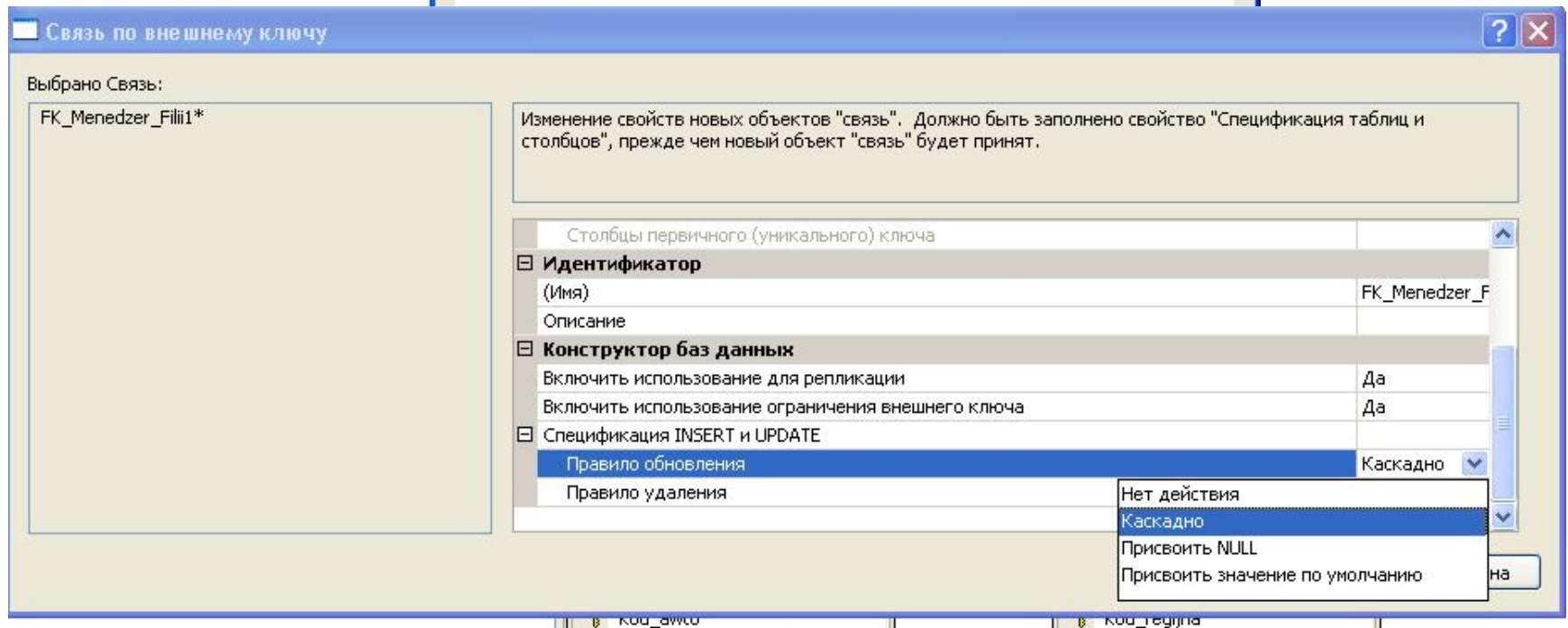
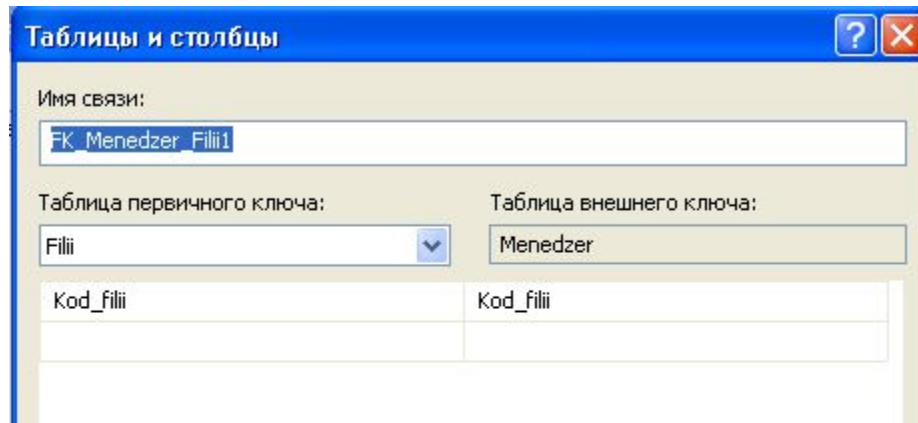


**ЗВ'ЯЗОК 1:5**



**ЗВ'ЯЗОК 1:1**





**Вікна для встановлення правил для операцій INSERT і UPDATE.**

# **4. ХАРАКТЕРИСТИКА ТИПІВ ДАНИХ SQL SERVER 2008**

- ▣ Числові дані**
- ▣ Символьні рядки**
- ▣ Дані часу (Дата і час)**
- ▣ Двійкові дані**
- ▣ Типи даних великих об'єктів**
- ▣ Дані спеціальних типів**

## Числові дані (цілі)

- **Tinyint** - займає 1 байт. Невід'ємне ціле число від 0 до 255.
- **Smallint** - займає 2 байта, цілі числа від -32767 до +32676.
- **Inteteger (Int)** - займає 4 байта, цілі числа від -2147483647 до +2147483647
- **Bigint** - займає 8 байт, ціле число від -9223372036854775808 до

# Числові дані (дробові)

- **Decimal (p[,s]) (Dec)**- займає від 5 до 17 байт. Число з фіксованою крапкою,  $p$  – загальна кількість цифр, з врахуванням дробової частини,  $s$  – кількість цифр після десяткової крапки.
- **Numeric (p[,s])** – синонім **Decimal**.
- **Real** – числа з плаваючою крапкою.  
Діапазон позитивних значень – від  
Діапазон від'ємних значень – від

## Числові дані (дробові)

**Float [(p)]** – числа з плаваючою крапкою, як і **Real**, **p** – визначає точність. Якщо  $p < 25$  – одинарна точність (4 байта), якщо  $p \geq 25$  – подвоєна точність (8 байтів).

## Грошові дані (дробові)

**Money** – грошові дані, відповідають 8-байтовим значенням **Decimal** і округляються до 4-х знаків після десяткової коми.

# Символьні дані

**CHAR[(n)]** - рядок фіксованої довжини із одnobайтних символів, де **n** – кількість символів у рядку. Максимальне **n = 8000**, якщо **n** пропущено, то довжина рядка дорівнює – 1.

**VARCHAR[(n)]** – рядок змінної довжини із одnobайтних символів, де **(0 <n<=8000)**. На відміну від **CHAR** значення для даного типу зберігаються в їх фактичній довжині. Цей тип має два синоніми: **CHAR VARYING** і **CHARACTER VARYING**.

**NCHAR[(n)]** - рядки фіксованої довжини символів Unicode. Основна відмінність між **CHAR** і **NCHAR** полягає в обсягах пам'яті, кожен символ **NCHAR** – займає 2 байта, а кожен символ **CHAR** – 1байт, тому максимальна кількість символів з типом **NCHAR** - рівна 4000.

**NVARCHAR[(n)]** - рядок змінної довжини символів Unicode. Основна відмінність між **VARCHAR** і **NVARCHAR** полягає в обсягах пам'яті, кожен символ **NVARCHAR**– займає 2 байта, а кожен символ **VARCHAR** – 1байт, тому максимальна кількість символів з типом **NVARCHAR**– рівна 4000.

# Типи даних дати і часу

**DATETIME** і **SMALLDATETIME** - описують дату і час в 4 і 2 байтах відповідно

**DATE** і **TIME** – окремо описують дату і час в 3 байтах і 3-5 байтах відповідно.

**DATETIME2** – тип даних для зберігання дати і часу з високою точністю, може бути заданий з різними розмірами в залежності від вимог. Розмір 6-8 байт. Точність часу складає – 100 наносекунд.

**DATETIMEOFFSET** – на відміну від інших цей тип підтримує часові пояси, все інше так як у **DATETIME2**.

За змовчанням підтримується формат: *mmm dd yy*.  
Порядок чередування місяця, дня і року управляється оператором *SET DATEFORMAT*

## **Двійкові типи даних**

**BINARY [(n)]** – бітовий рядок,  
довжиною  $n$  ( $0 < n \leq 8000$ ).

**VARBINARY [(n)]** - бітовий рядок,  
довжиною не більше  $n$   
( $0 < n \leq 8000$ ).

**BIT** – логічний тип даних з трьома  
допустимими значеннями:  
**FALSE, TRUE і NULL**



# Типи даних великих об'єктів

- **Великі об'єкти** (Large objects – LOB) – об'єкти з максимальним розміром 2 Гбайти.
- **VARCHAR (MAX), NVARCHAR (MAX), VARBINARY (MAX)** – специфікатор **MAX** визначає змінну довжину стовпчиків для полів типу VARCHAR, NVARCHAR, VARBINARY.
- **TEXT, NTEXT, IMAGE** – поля типу «текст/зображення». **TEXT, NTEXT**- великі довгі тексти, **IMAGE** – модулі завантаження, аудіо, відео.

*\*\* Обмеження на використання даних типу «текст/зображення»: не можуть бути локальними змінними в процедурах, що зберігаються, не можуть бути індексами.*