Telerik Academy

Database Modeling

Creating E/R Diagrams with SQL Server Management Studio and MySQL Workbench

Telerik Software Academy

Learning & Development Team

http://academy.telerik.com

	Affiliate	State	Members	Annual Fee
2	Norfolk	VA	205	50
3	Houston	TX	65	35
4	Manhattan	NY	657	75
5	Albany	NY	336	60
6	Washington	DC	453	50
7	Richmond	VA	432	50
8	Memphis	TN	77	25
9	Brooklyn	NY	578	70
10	Boston	MA	153	65
11	Waltham	MA	32	65
12	Schenectady	NY	43	35
13	Newark	NJ	235	85
14	Morristown	NJ	68	75

Telerik Academy

Table of Contents

1. Data Modeling – Principles
2. Data Types in SQL Server
3. Creating Databases in SQL Server
4. Creating Tables
5. Defining a Primary Key and Identity Columns
6. Creating Relationships between the Tables
   - One-to-many, Many-to-many, One-to-one
7. Naming Conventions
8. Data Modeling in MySQL Workbench

Telerik Academy

Relational Data Modeling

Fundamental Concepts

Steps in Database Design

- Steps in the database design process:
  1. Identification of the entities
  2. Identification of the columns in the tables
  3. Defining a primary key for each entity table
  4. Identification and modeling of relationships
     - Multiplicity of relationships
  5. Defining other constraints
  6. Filling test data in the tables

Identification of Entities

- Entity tables represent objects from the real world
  - Most often they are nouns in the specification
  - For example:

```
We need to develop a system that stores information
about students, which are trained in various courses.
The courses are held in different towns. When
registering a new student the following information is
entered: name, faculty number, photo and date.
```

  - Entities: Student, Course, Town

Identification of Columns

- Columns in the tables are characteristics of the entities
  - They have name and type
- For example students have:
  - Name (text)
  - Faculty number (number)
  - Photo (binary block)
  - Date of enlistment (date)

Identification of the Columns

- Columns are clarifications for the entities in the text of the specification, for example:

> We need to develop a system that stores information about students, which are trained in various courses. The courses are held in different towns. When registering a new student the following information is entered: name, faculty number, photo and date.

- Students have the following characteristics:
  - Name, faculty number, photo, date of enlistment and a list of courses they visit

Telerik Academy

How to Choose a Primary Key?

- Always define an additional column for the primary key
  - Don't use an existing column (for example SSN)
  - Must be an integer number
  - Must be declared as a primary key
  - Use `identity` to implement auto-increment
  - Put the primary key as a first column
- Exceptions
  - Entities that have well known ID, e.g. countries (BG, DE, US) and currencies (USD, EUR, BGN)

Telerik Academy

Identification of Relationships

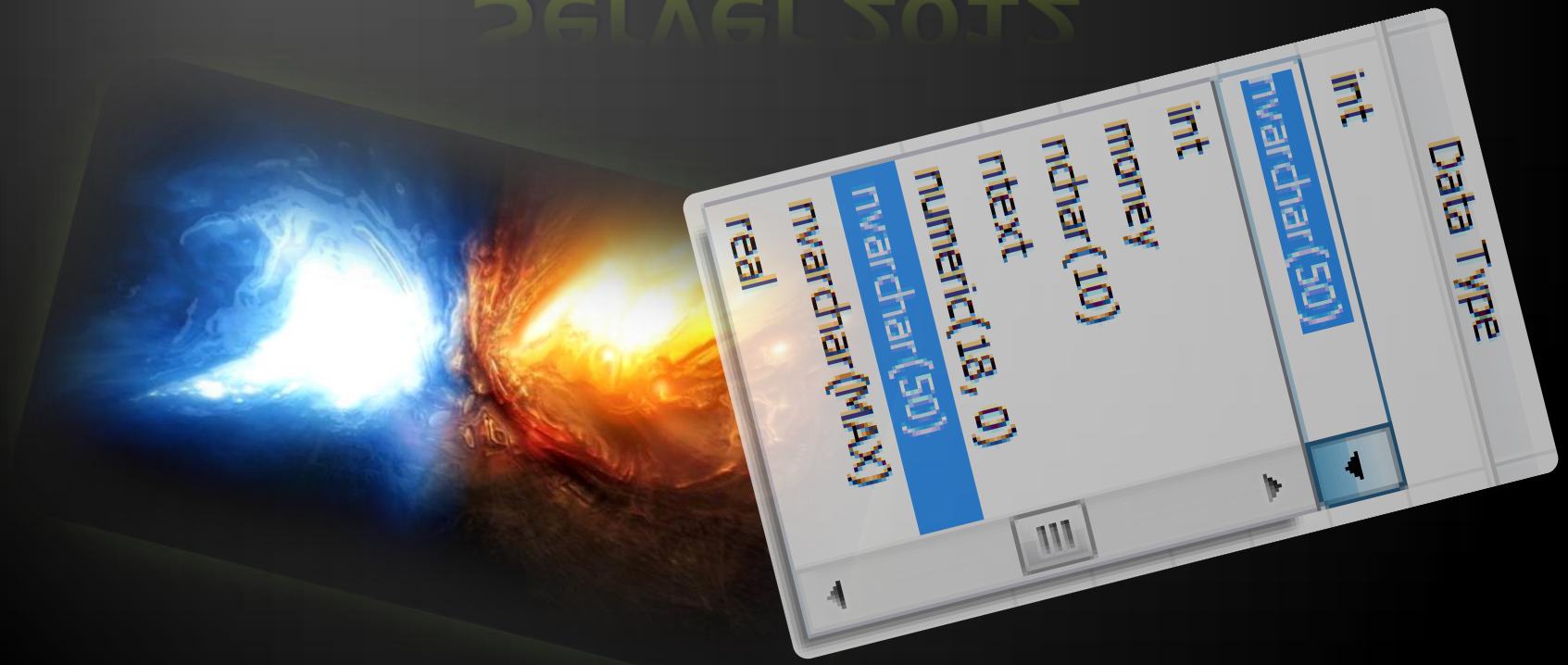
- Relationships are dependencies between the entities:

> We need to develop a system that stores information about students, which are trained in various courses. The courses are held in different towns. When registering a new student the following information is entered: name, faculty number, photo and date.

  - "Students are trained in courses" – many-to-many relationship
  - "Courses are held in towns" – many-to-one (or many-to-many) relationship

Telerik Academy

Data Types in SQL Server 2012



Data Types in SQL Server

- Numeric
  - `bit` (1-bit), `integer` (32-bit), `bigint` (64-bit)
  - `float`, `real`, `numeric(scale, precision)`
  - `money` – for money (precise) operations
- Strings
  - `char(size)` – fixed size string
  - `varchar(size)` – variable size string
  - `nvarchar(size)` – Unicode variable size string
  - `text` / `ntext` – text data block (unlimited size)

Data Types in SQL Server (2)

- Binary data
  - `varbinary(size)` – a sequence of bits
  - `image` – a binary block up to 1 GB
- Date and time
  - `datetime` – date and time starting from 1.1.1753 to 31.12. 9999, a precision of 1/300 sec.
  - `smalldatetime` – date and time (1-minute precision)

Telerik Academy

Data Types in SQL Server (3)

- Other types
  - `timestamp` – automatically generated number whenever a change is made to the data row
  - `uniqueidentifier` – GUID identifier
  - `xml` – data in XML format

Data Types in SQL Server (4)

- `Nullable` and `NOT NULL` types
  - All types in SQL Server may or may not allow `NULL` values

- `Primary key` columns
  - Define the primary key

- `Identity` columns
  - Automatically increased values when a new row is inserted (auto-increment values)
  - Used in combination with `primary key`

Telerik Academy

Database Modeling with SQL Server Management Studio

Creating Database

Connecting to SQL Server

- When starting SSMS a window pops up
- Usually it is enough to just click the "Connect" button without changing anything

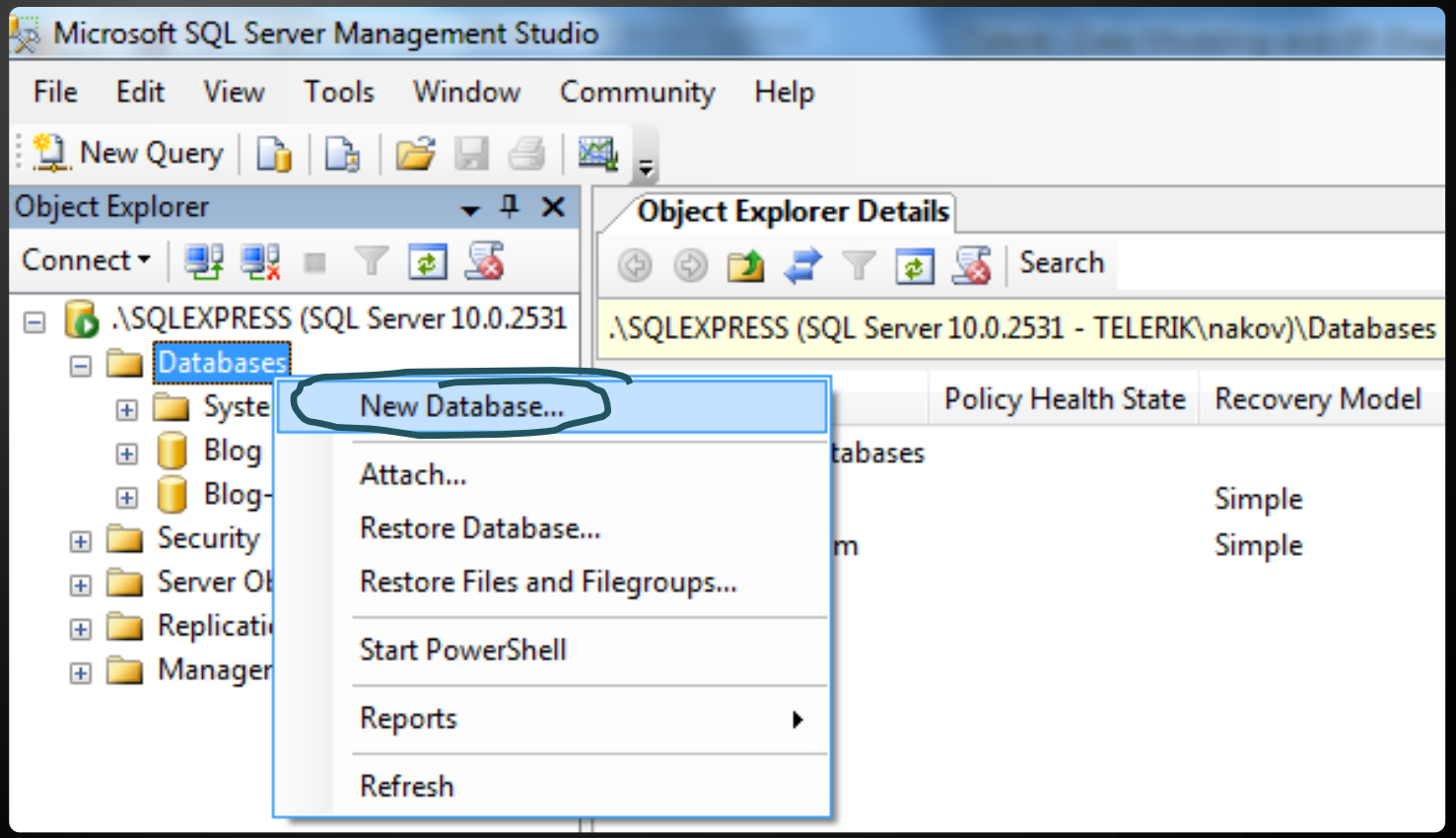
Telerik Academy

Working with Object Explorer

- Object Explorer is the main tool to use when working with the database and its objects
- Enables us:
  - To create a new database
  - To create objects in the database (tables, stored procedures, relationships and others)
  - To change the properties of objects
  - To enter records into the tables

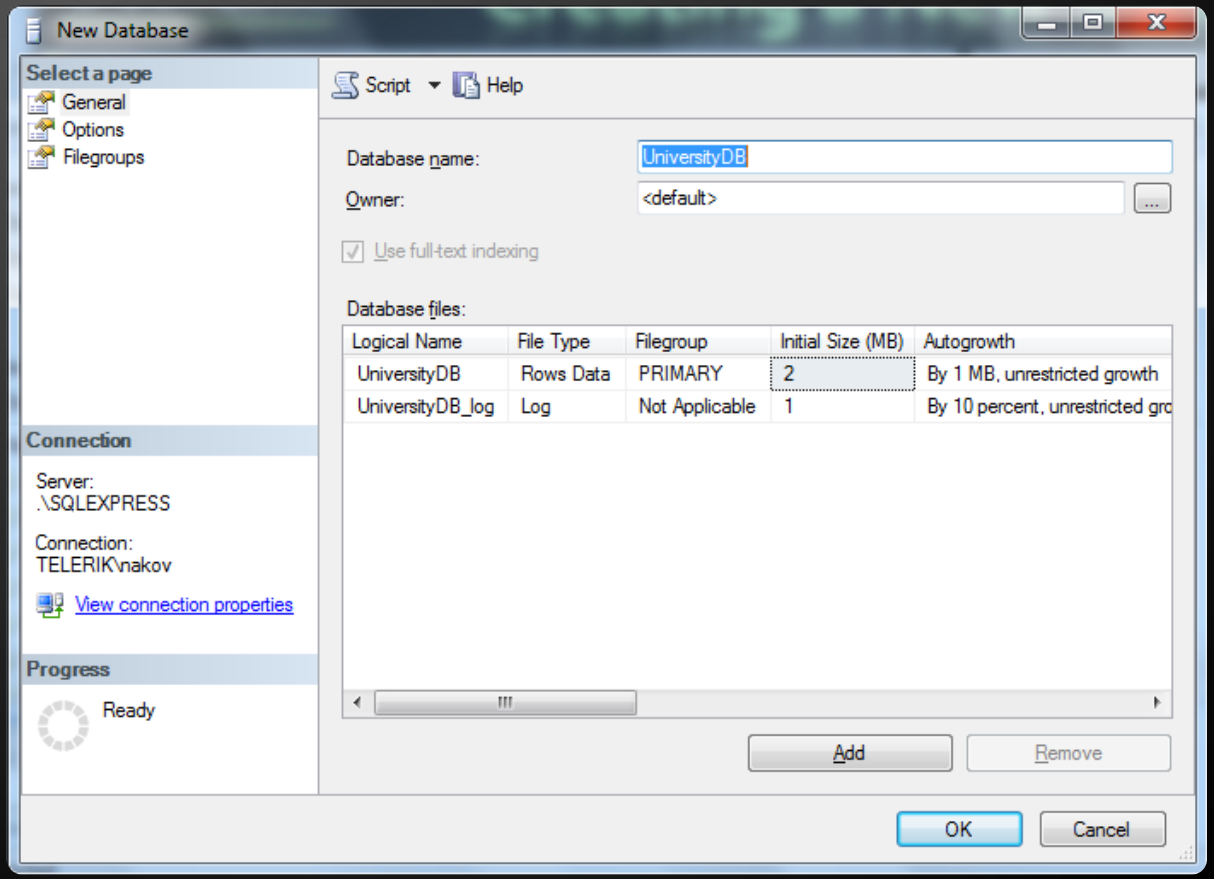
Creating a New Database

- In Object Explorer we go to the "Databases" and choose "New Database..." from the context menu



Creating a New Database (2)

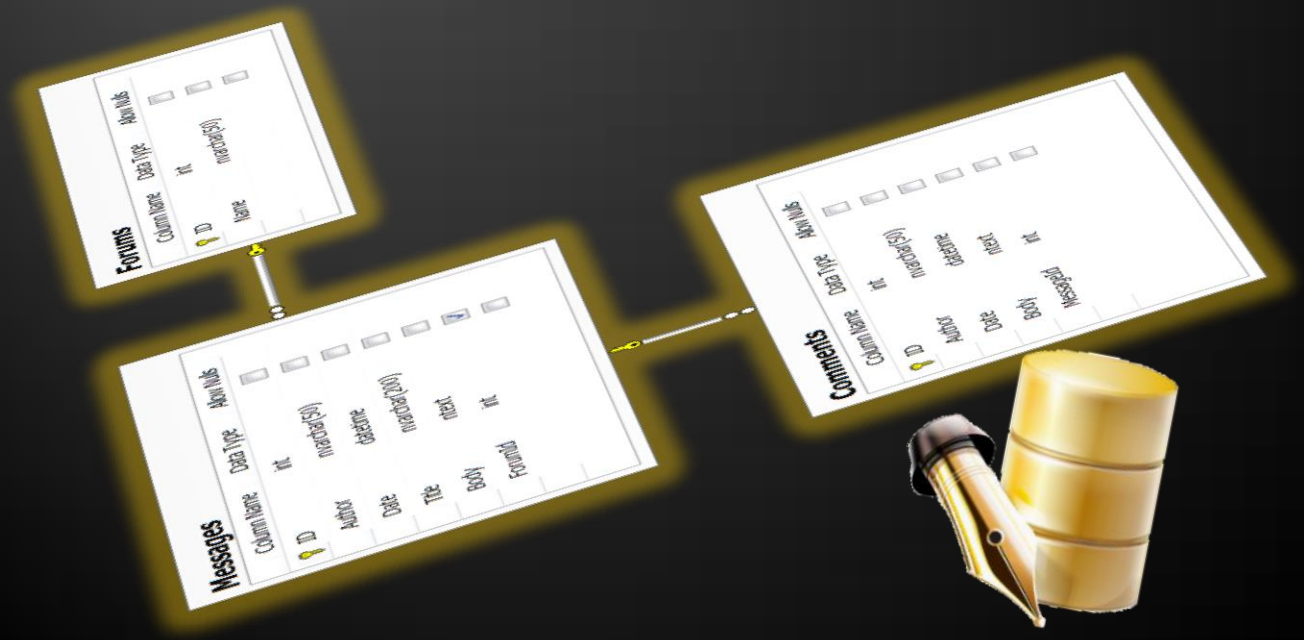
- In the "New Database" window enter the name of the new database and click [OK]



Telerik Academy

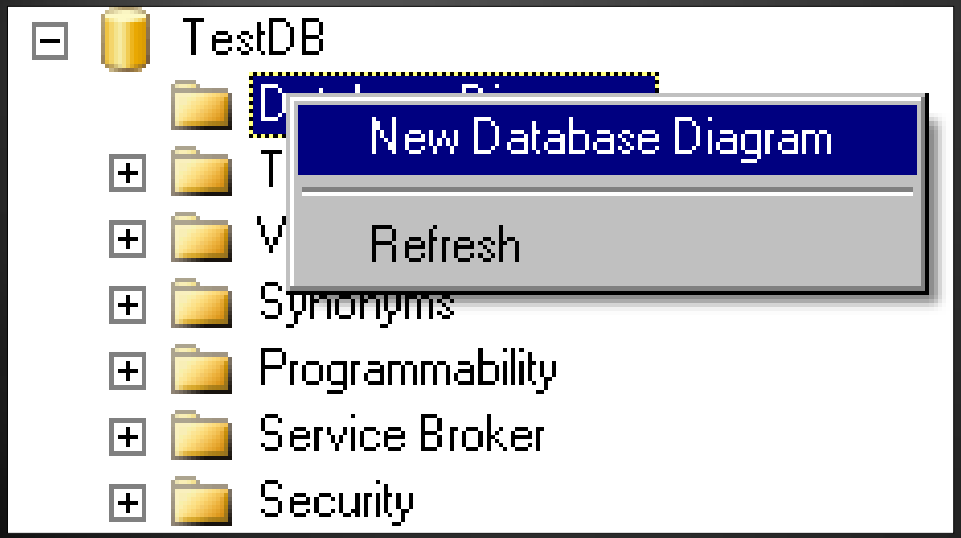
Database Modeling with SQL Server Management Studio

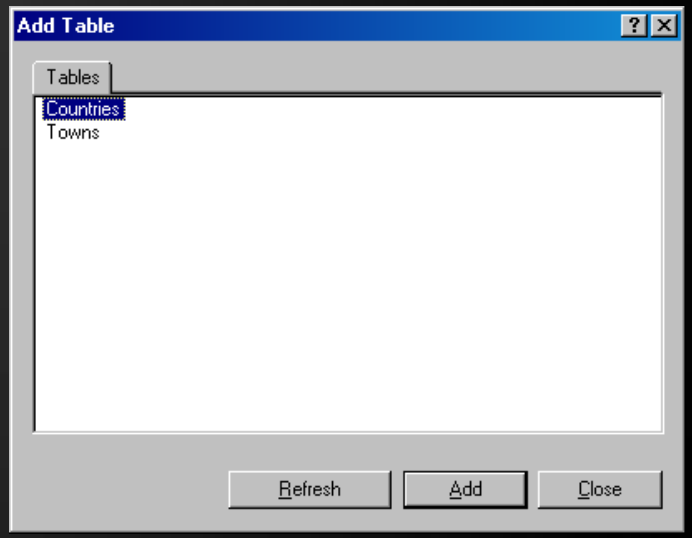
Creating E/R Diagrams



Creating an E/R diagram

- In the "Database Diagrams" menu choose the "New Database Diagram"



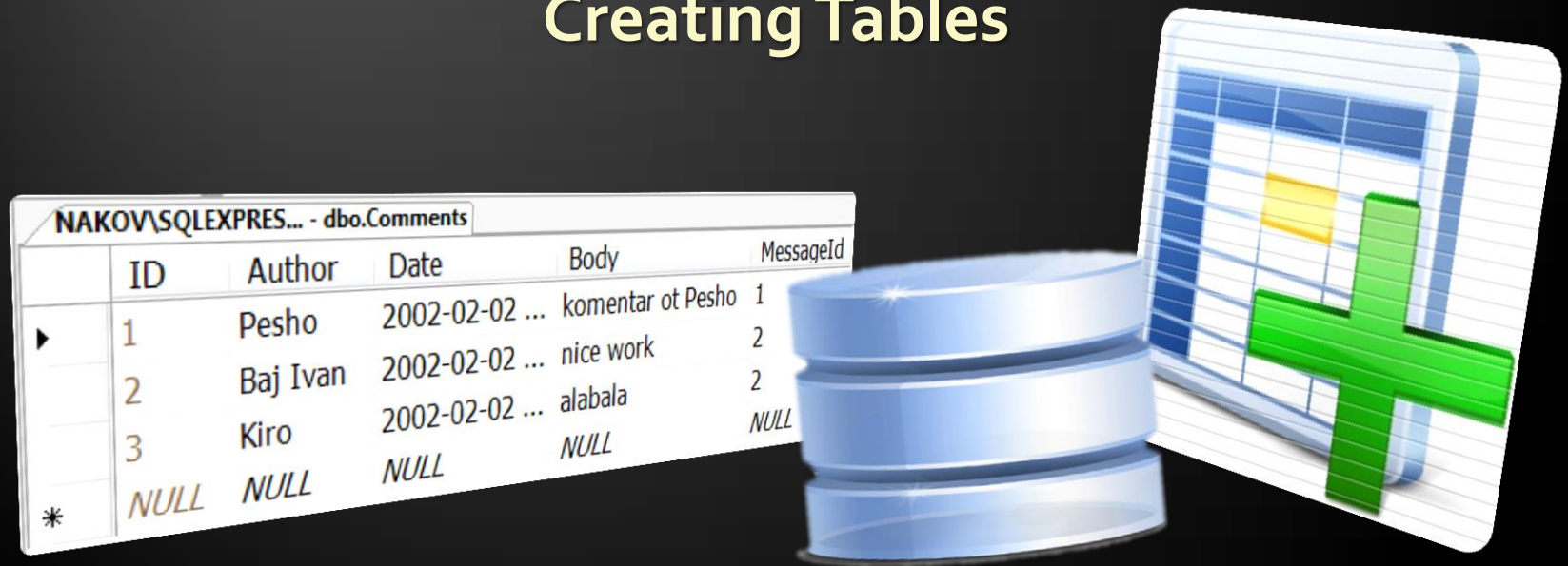


- We can choose from the existing tables, which we want to add to the diagram

Telerik Academy

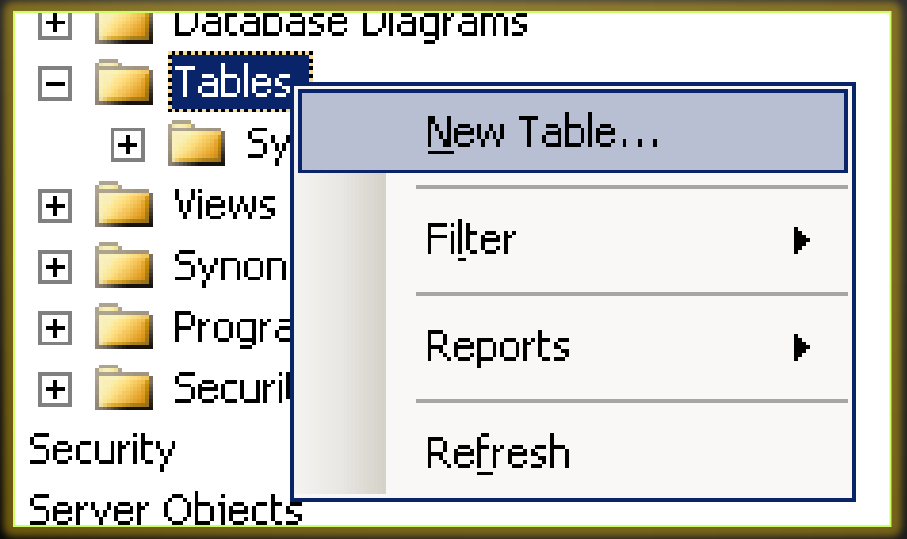
Database Modeling with SQL Server Management Studio

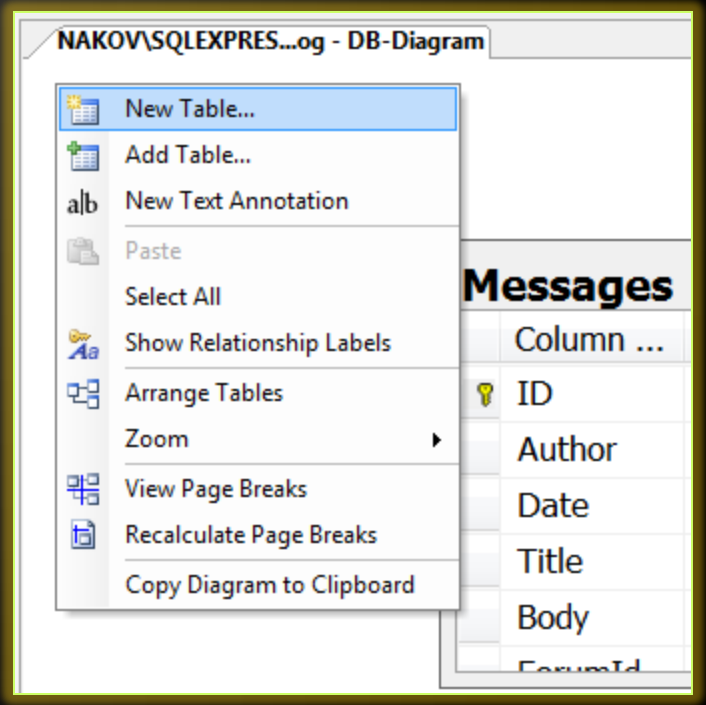
Creating Tables



Creating Tables

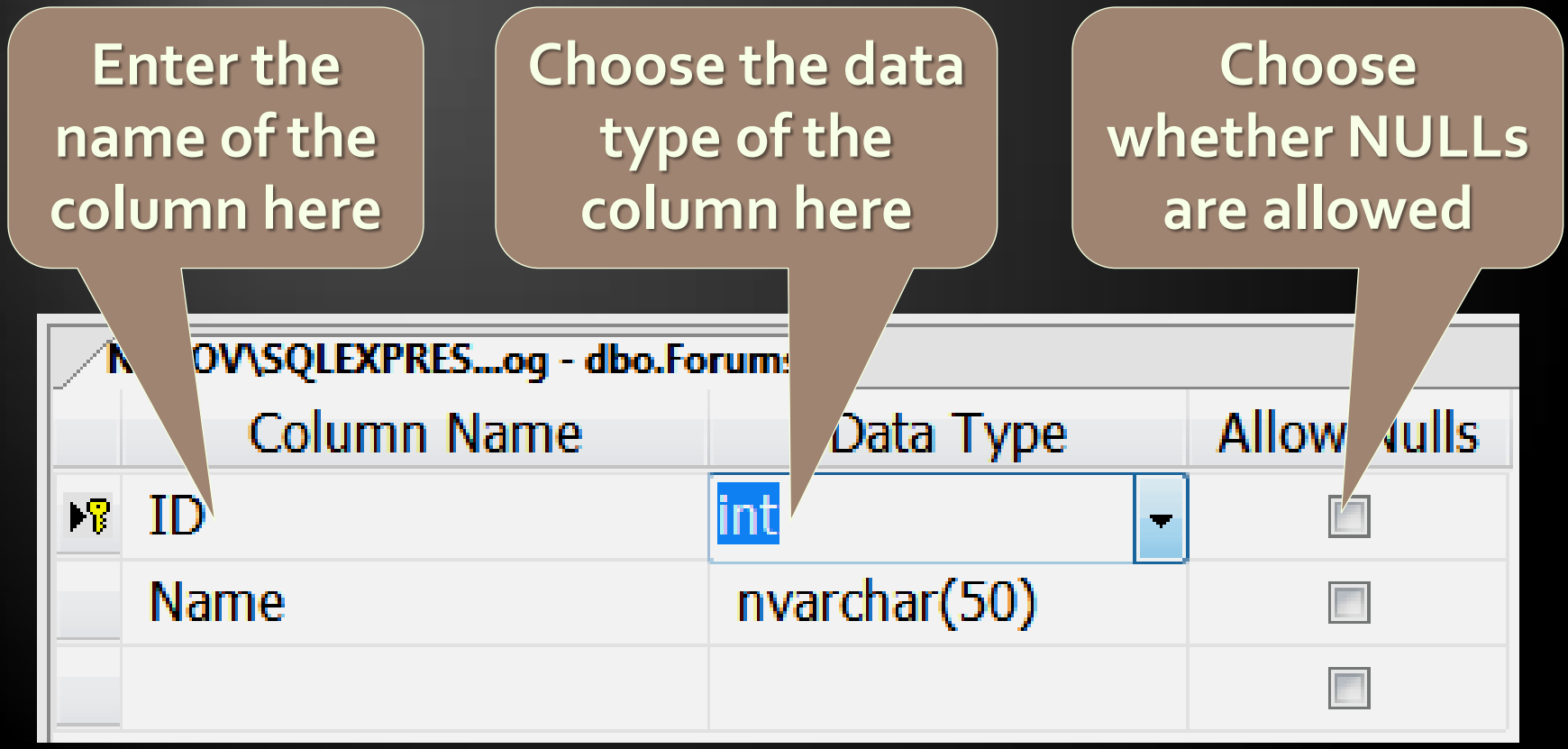
- If the database doesn't show immediately in Object Explorer perform "Refresh" [F5]
- Creating new table:





Creating Tables (2)

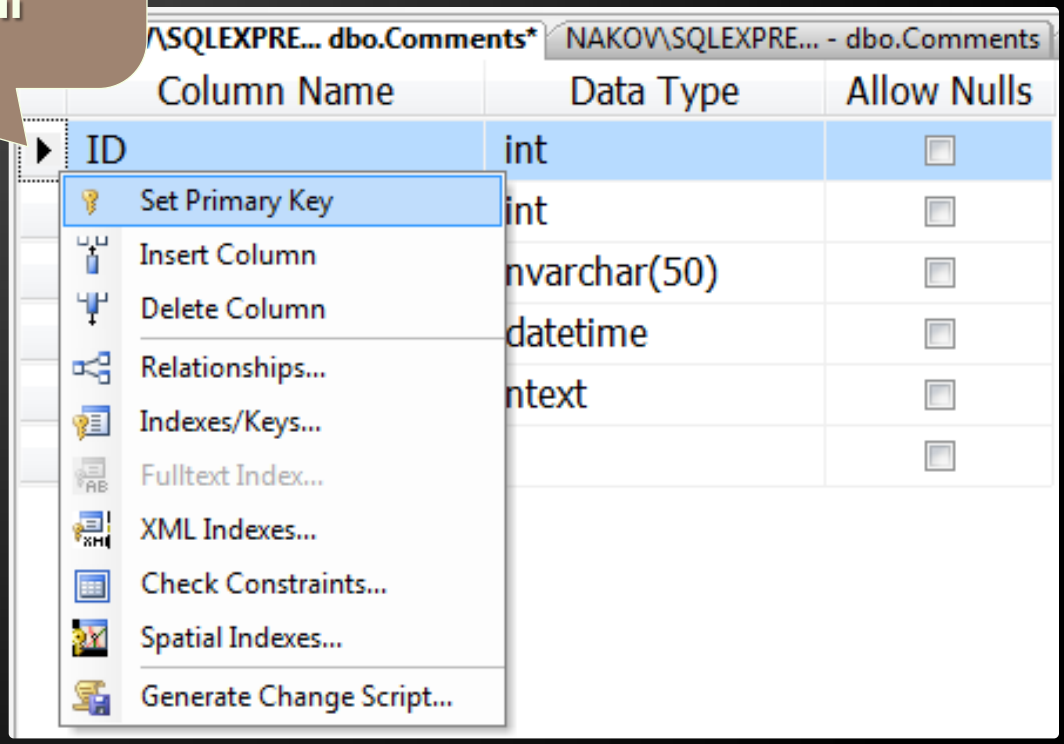
- Enter table name and define the table columns (name and type):



Creating Tables (3)

- Defining a primary key

Right click on the column start and select "Set Primary Key"

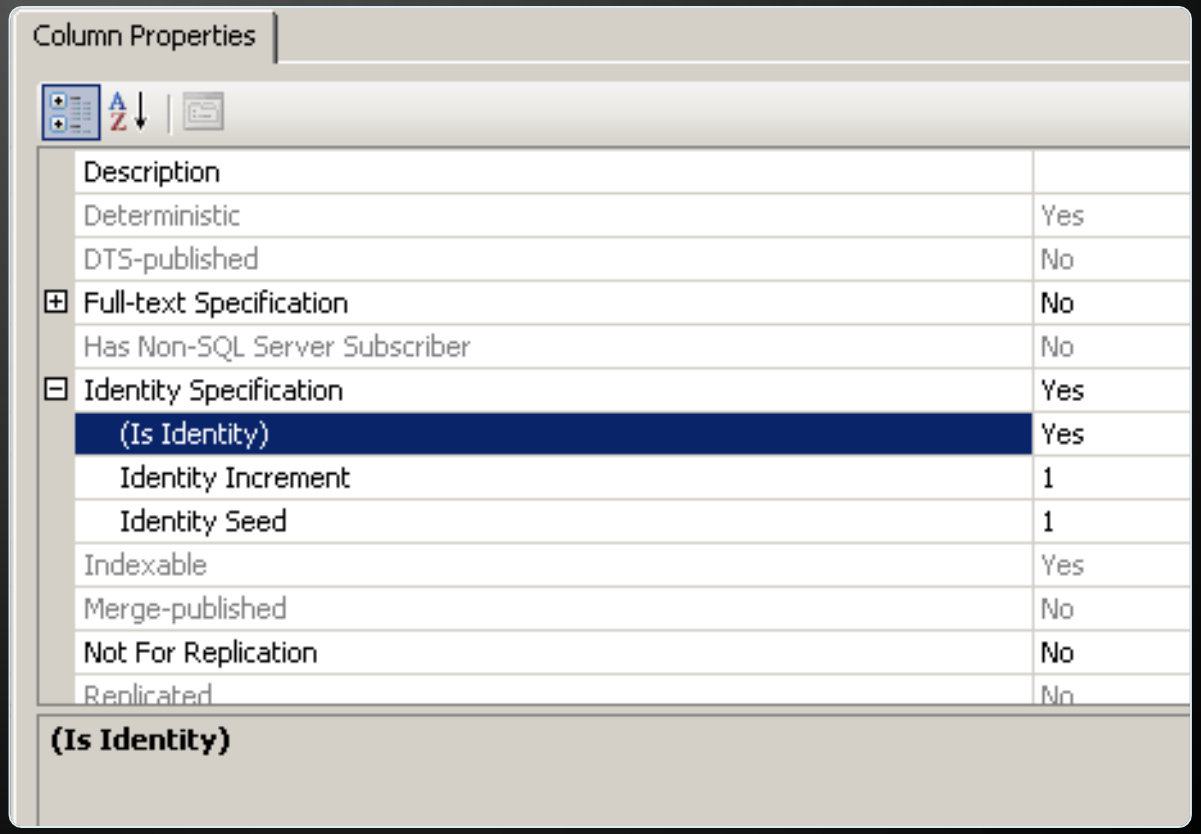


Creating Tables (4)

- Defining an identity columns
  - Identity means that the values in a certain column are auto generated (for `int` columns)
  - These values cannot be assigned manually
  - Identity Seed – the starting number from which the values in the column begin to increase.
  - Identity Increment – by how much each consecutive value is increased

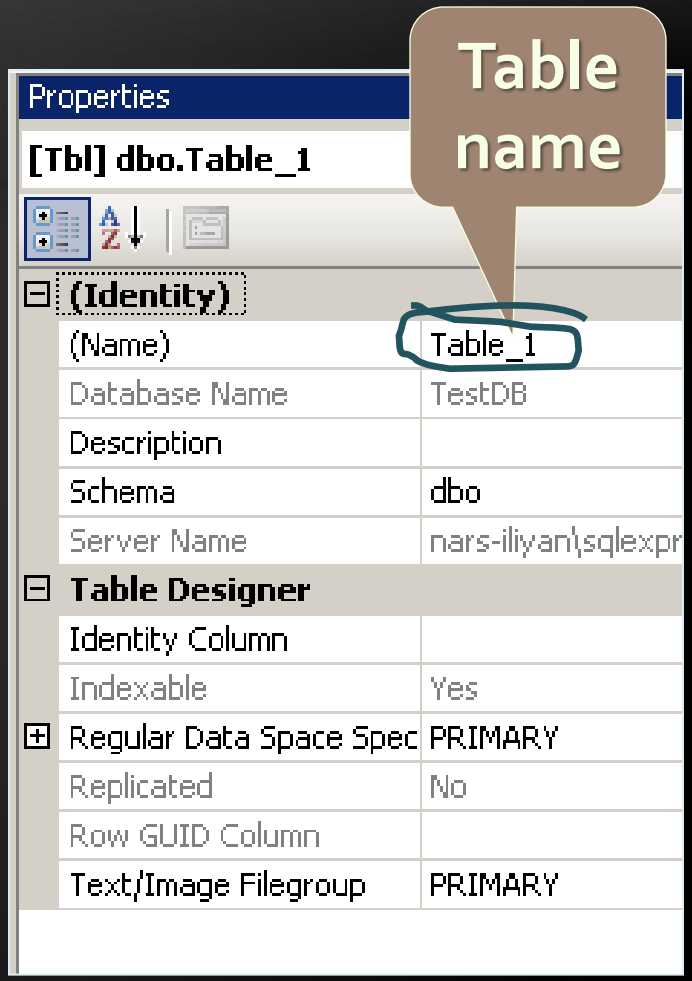
Creating Tables (5)

- Setting an identity through the "Column Properties" window



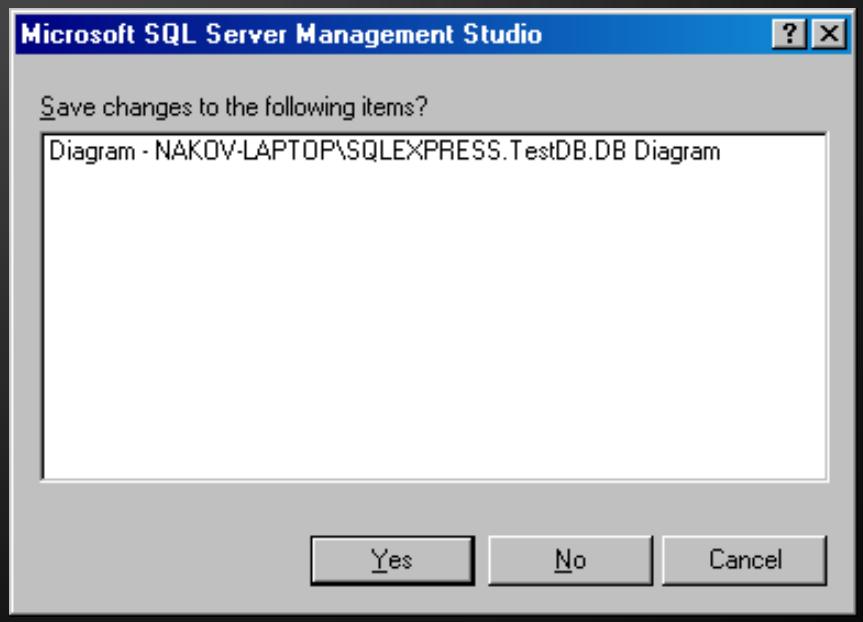
Creating Tables (6)

- It is a good practice to set the name of the table at the time it is created
  - Use the "Properties" window
  - If it's not visible use "View" → "Properties Window" or press [F4]



Creating Tables (7)

- When closing the window for the table, SSMS asks whether to save the table
  - You can do it manually by choosing "Save Table" from the "File" menu or by pressing Ctrl + S



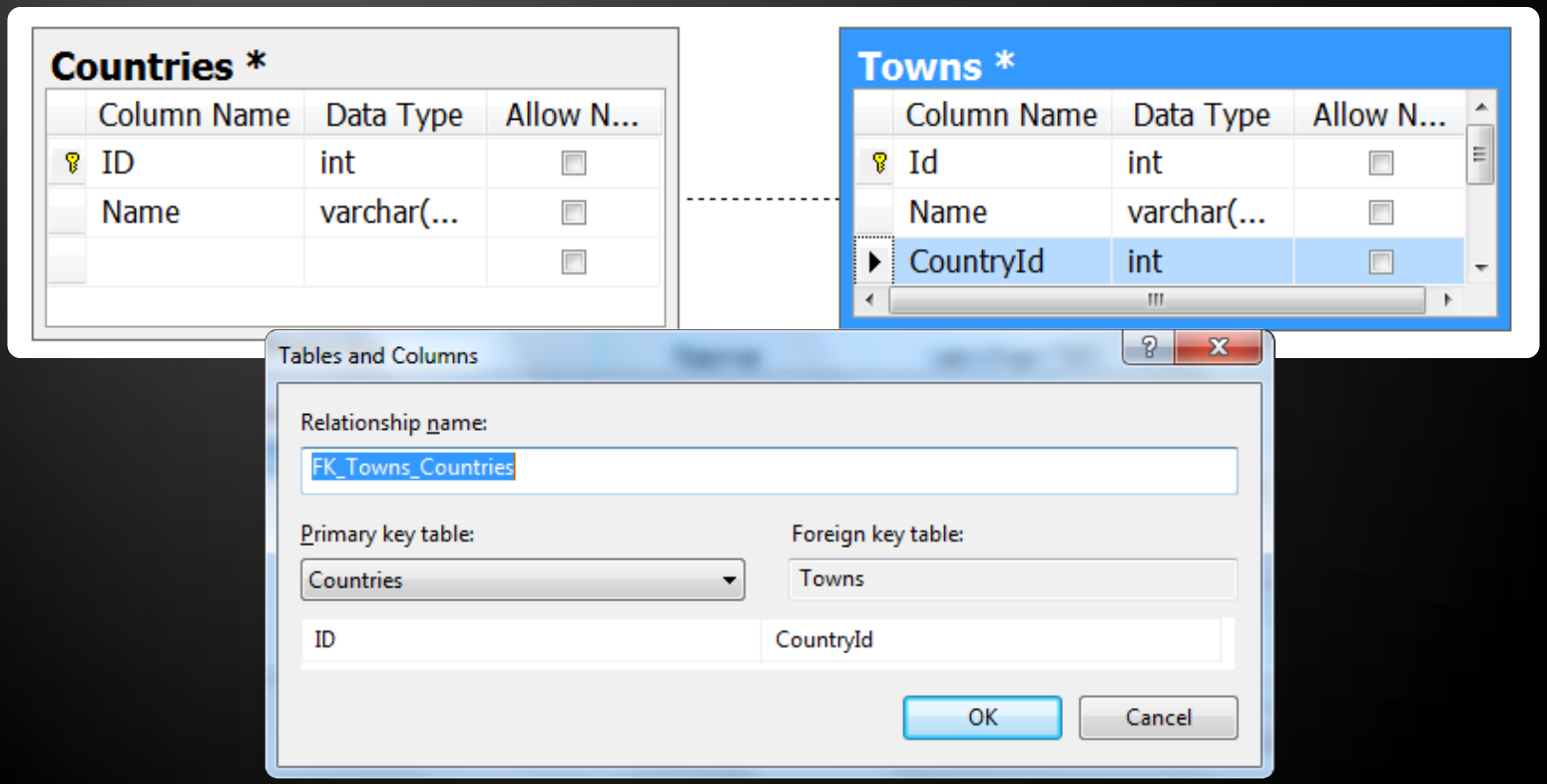
Telerik Academy

Database Modeling with SQL Server Management Studio

Creating Relationships between Tables

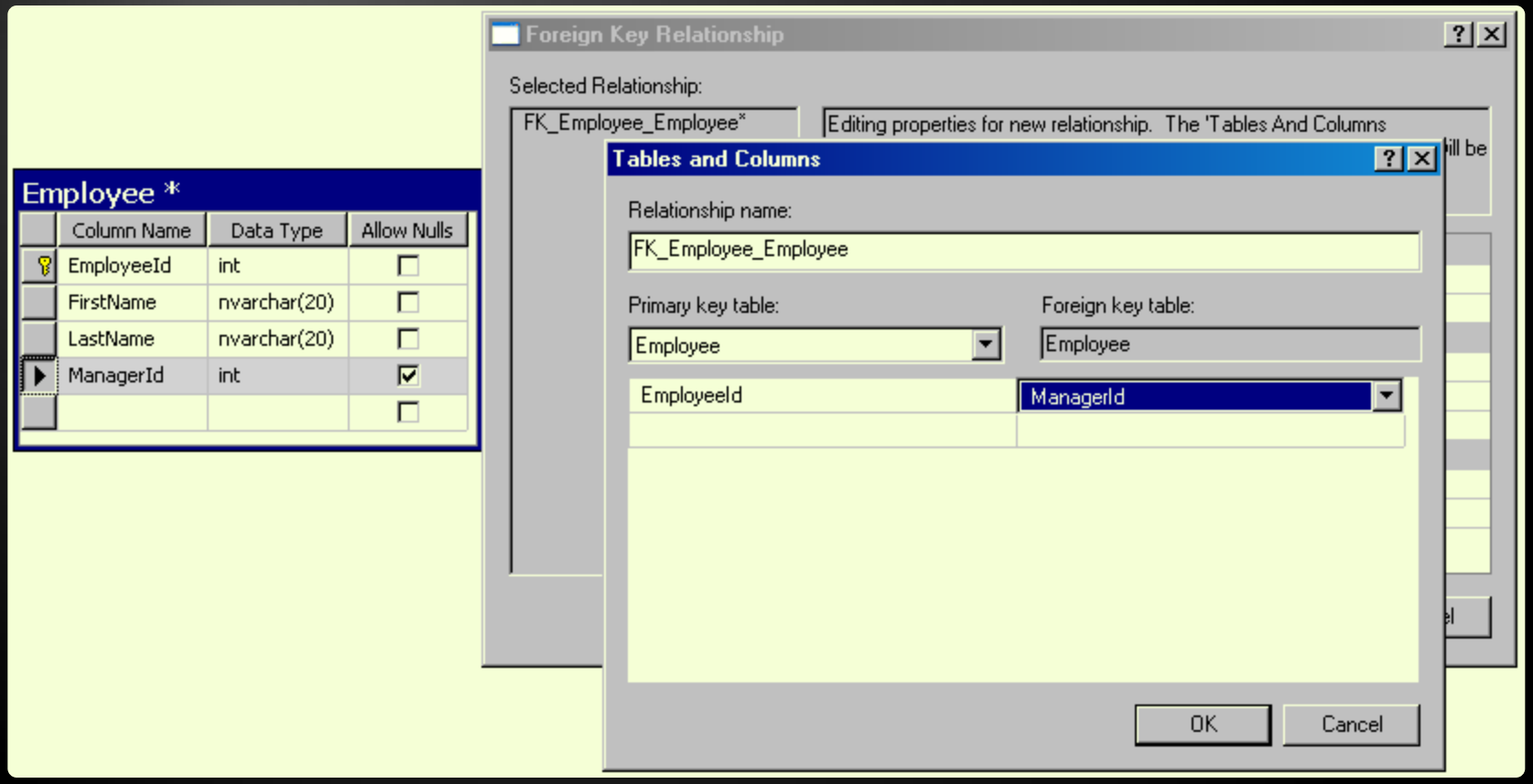
Creating Relationships

- To create one-to-many relationship drag the foreign key column onto the other table
  - Drag from the child table to the parent table



Self-Relationships

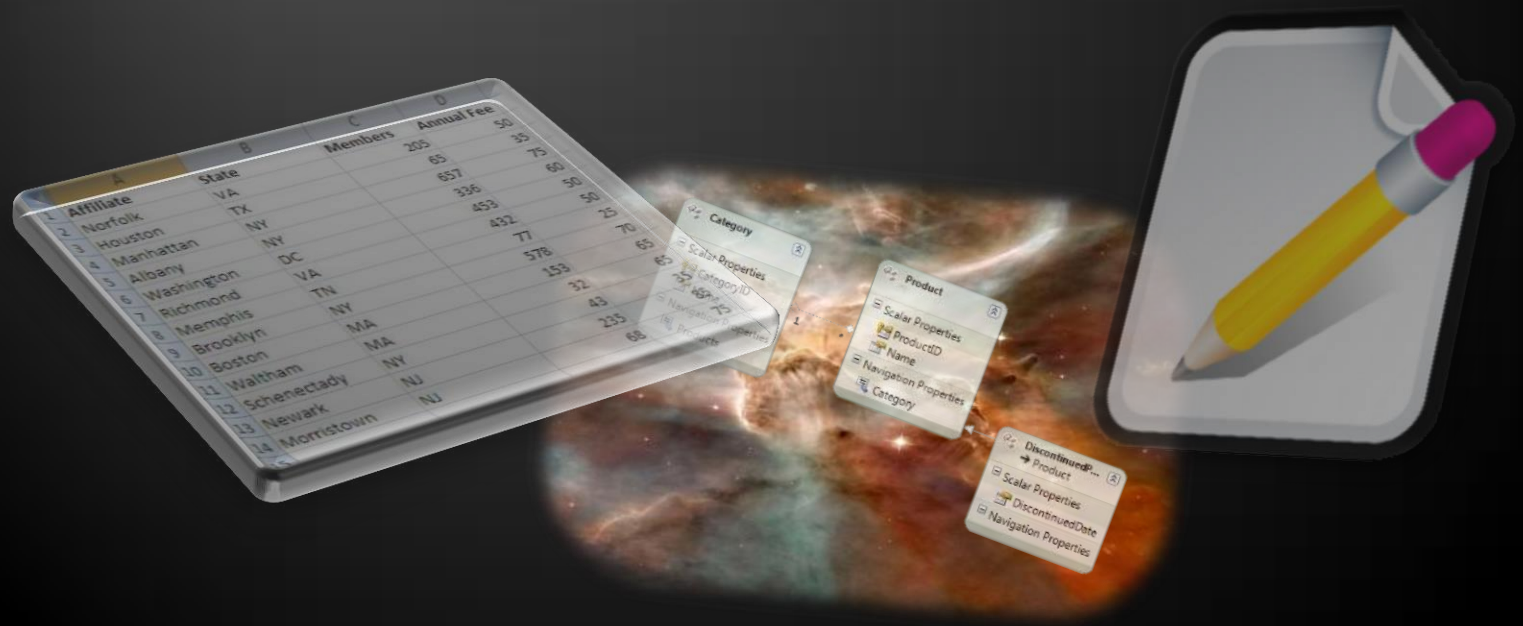
- Self-relationship can be created by dragging a foreign key onto the same table



Telerik Academy

Database Modeling with SQL Server Management Studio

Naming Conventions



Naming Conventions

- Tables
  - Each word is capitalized (Pascal Case)
  - In English, plural
  - Examples: `Users`, `PhotoAlbums`, `Countries`
- Columns
  - In English, singular
  - Each word is capitalized (Pascal Case)
  - Avoid reserved words (e.g. `key`, `int`, `date`)
  - Examples: `FirstName`, `OrderDate`, `Price`

Naming Conventions (2)

- Primary key
  - Use "Id" or name_of_the_table + "Id"
  - Example: in the Users table the PK column should be be called Id or UserId
- Foreign key
  - Use the name of the referenced table + "Id"
  - Example: in the Users table the foreign key column that references the Groups table should be named GroupId

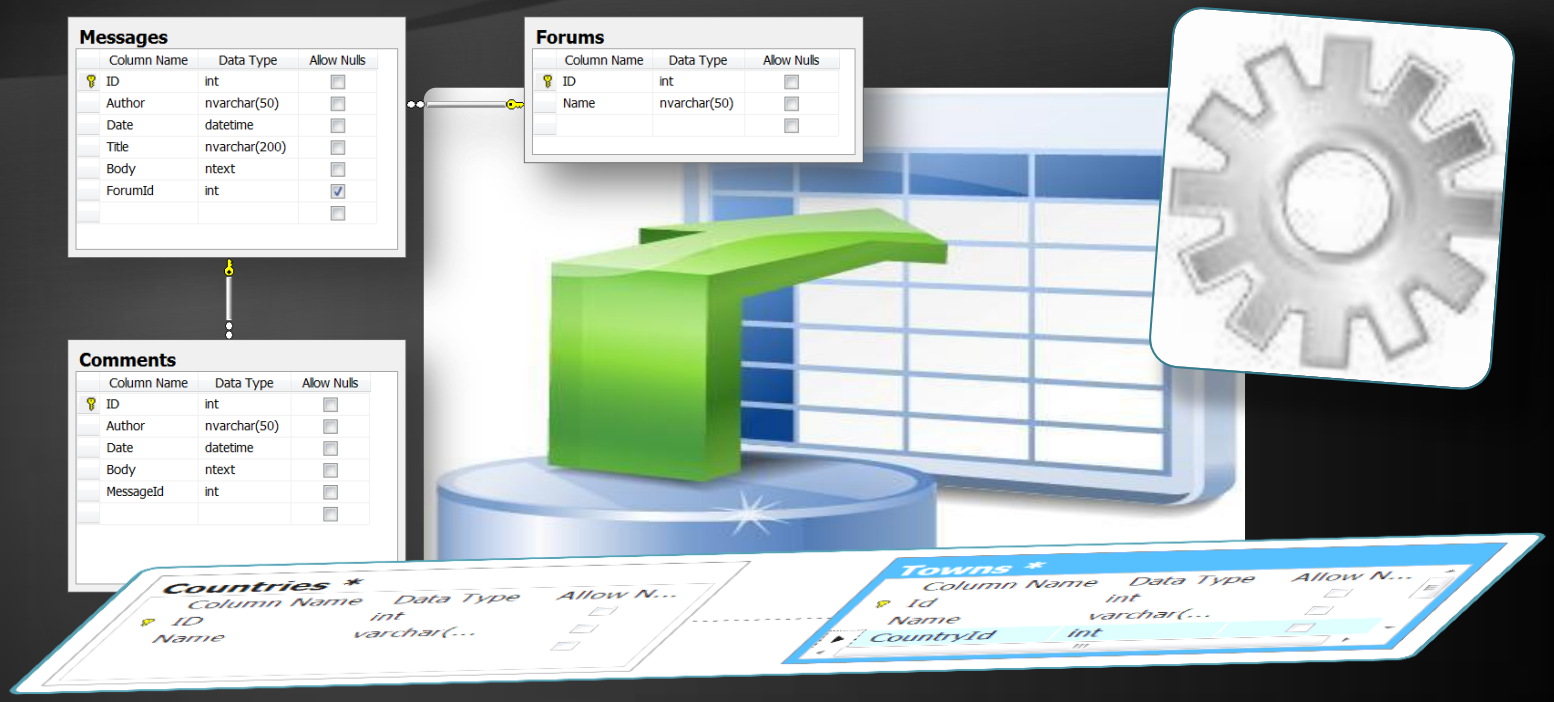
Naming Conventions (3)

- Relationship names (constraints)
  - In English, Pascal Case
  - `"FK_" + table1 + "_" + table2`
  - For example: `FK_Users_Groups`
- Index names
  - `"IX_" + table + column`
  - For example: `IX_Users_UserName`

Naming Conventions (4)

- Unique key constraints names
  - "UK_" + table + column
  - For instance: UK_Users_UserName
- Views names
  - V_ + name
  - Example: V_BGCompanies
- Stored procedures names
  - usp_ + name
  - Example: usp_InsertCustomer(@name)

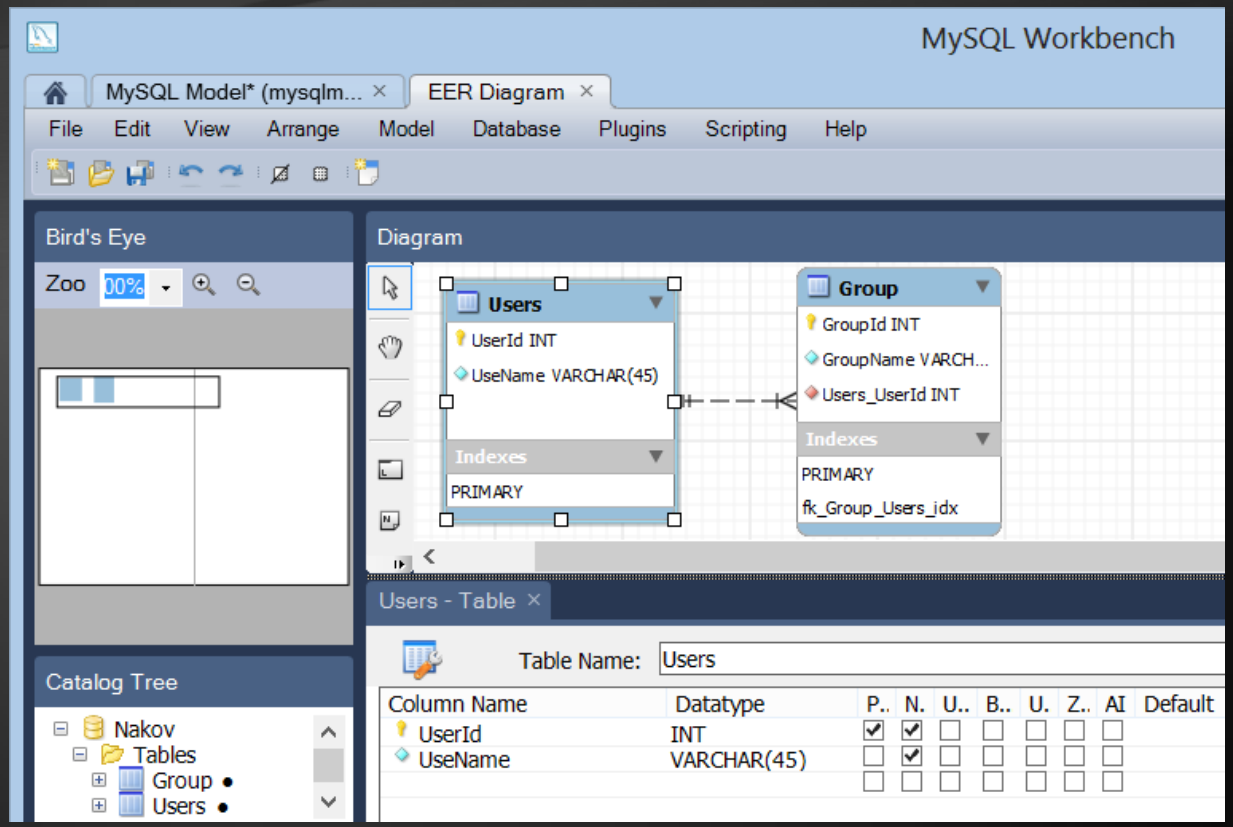
Telerik Academy



Database Modeling with SQL Server Management Studio

Live Demo

Telerik Academy



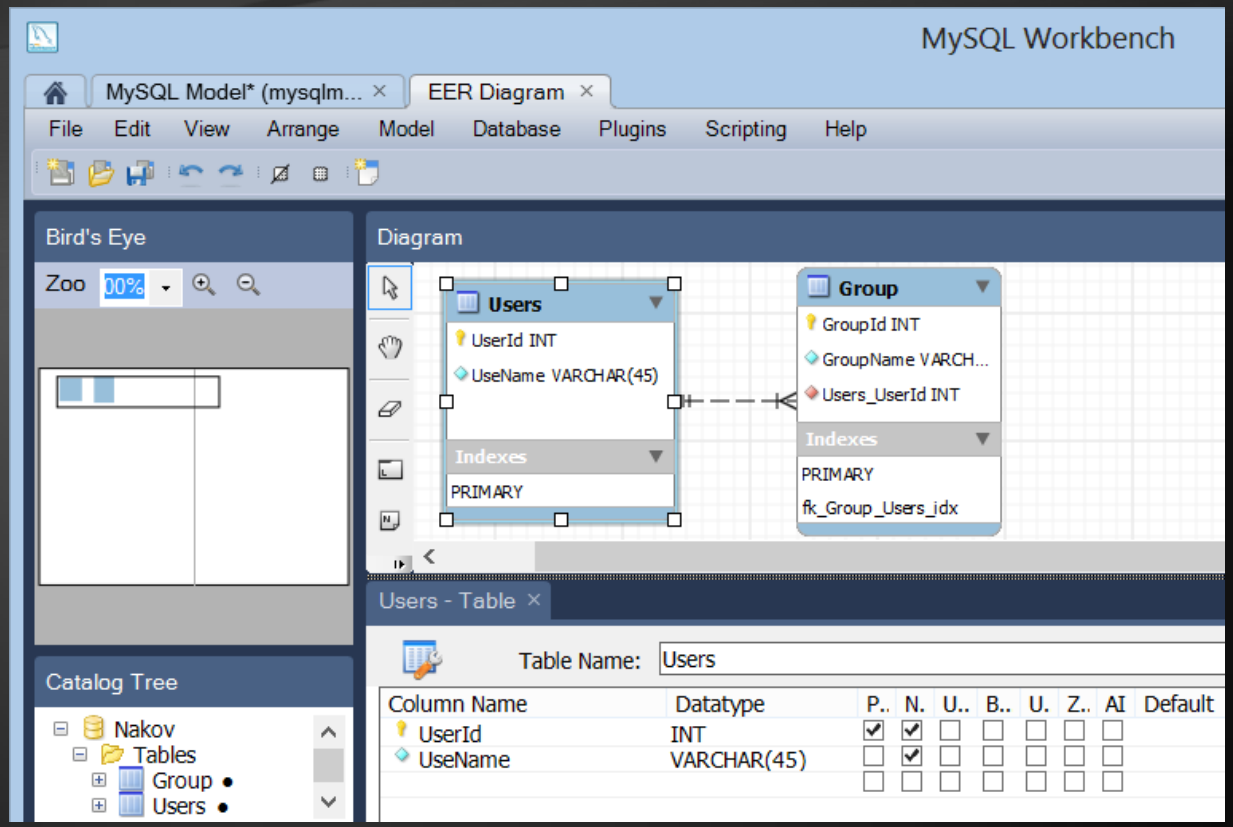
Data Modeling in MySQL

Creating E/R Diagrams with MySQL Workbench

E/R Diagrams in MySQL Workbench

- MySQL Workbench supports database schema design (E/R diagrams)
  - Can reverse engineer an existing database
  - Can forward engineer the diagram into SQL script / existing / new database
  - Can synchronize schema changes with existing database
  - User-unfriendly UI but better than nothing
    - Edit tables, relationships, indices, triggers, …

Telerik Academy



Data Modeling in MySQL

Live Demo

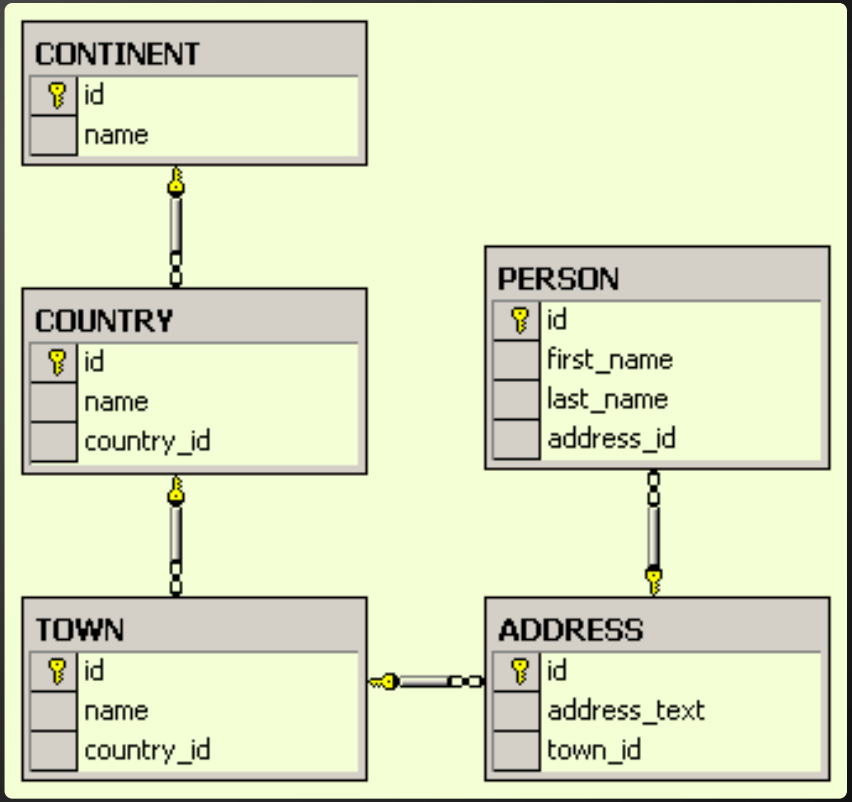
Telerik Academy

Database Modeling

http://academy.telerik.com

Exercises

1. Create the following database diagram in SQL Server:



2. Fill some sample data in the tables with SQL Server Management Studio.

Exercises (2)

3. Typical universities have: faculties, departments, professors, students, courses, etc. Faculties have name and could have several departments. Each department has name, professors and courses. Each professor has name, a set of titles (Ph. D, academician, senior assistant, etc.) and a set of courses. Each course consists of several students. Each student belongs to some faculty and to several of the courses. Your task is to create a data model (E/R diagram) for the typical university in SQL Server using SQL Server Management Studio (SSMS).
4. Create the same data model in MySQL.

Exercises (3)

5. We should design a multilingual dictionary. We have a set of words in the dictionary.

   Each word can be in some language and can have synonyms and explanations in the same language and translation words and explanations in several other languages.

   The synonyms and translation words are sets of words from the dictionary. The explanations are textual descriptions.

   Design a database schema (a set of tables and relationships) to store the dictionary.

Exercises (4)

6. Add support in the previous database for storing antonym pairs.

   Add support for storing part-of-speech information (e.g. verb, noun, adjective, …).

   Add support for storing hypernym / hyponym chains (e.g. tree → oak, pine, walnut-tree, …).

Telerik Academy

Free Trainings @ Telerik Academy

- C# Programming @ Telerik Academy
  - csharpfundamentals.telerik.com

- Telerik Software Academy
  - academy.telerik.com

- Telerik Academy @ Facebook
  - facebook.com/TelerikAcademy

- Telerik Software Academy Forums
  - forums.academy.telerik.com