

SQL Server 2008 Tutorial: Database Creation and Modification

IT 4153 Advanced Database

J.G. Zheng
Spring 2012



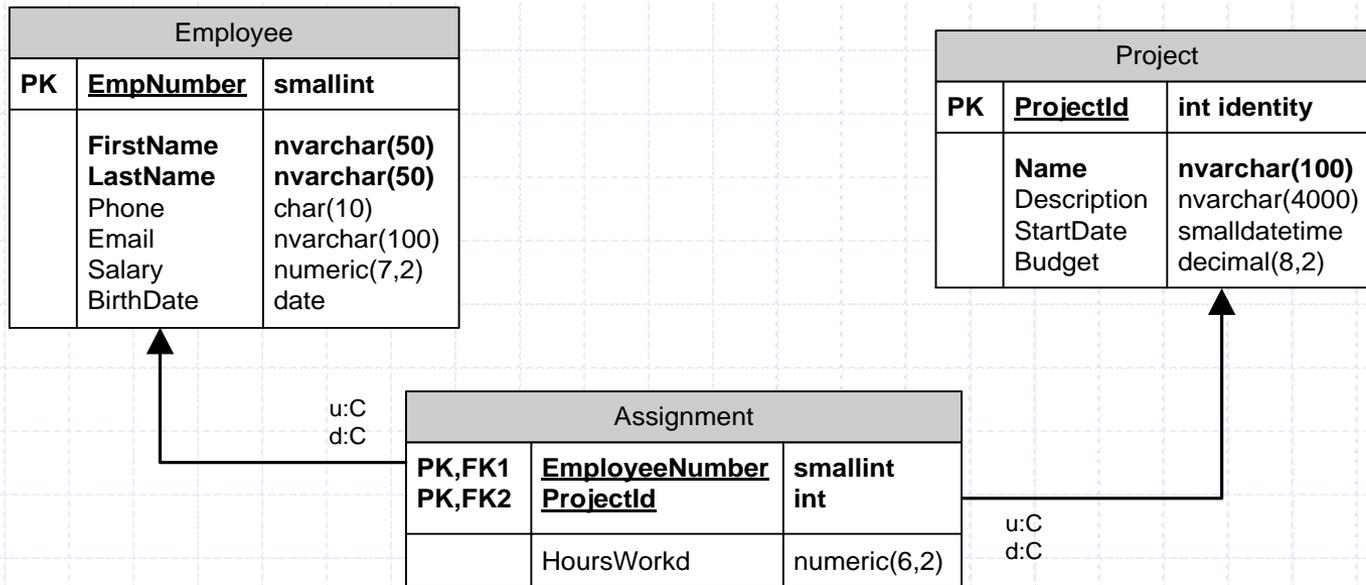
Overview

- ◆ Creating and modifying structures using the Management Studio GUI
 - Table designer
 - Relationship diagram designer
 - View designer

- ◆ Executing SQL scripts

Example

◆ Logical model in Visio



◆ Other requirements

- Email is unique
- Budget < 500000

Create a Database

Right-click and select "New Database"

Microsoft SQL Server Management Studio

File Edit View Project Debug Tools

New Query

Object Explorer

Connect

home-work-jack (SQL Server 10.0.5512.10)

Databases

System Databases

Database Snapshots

Northwind2003-mini

ReportServer

ReportServerTempDB

Project

Security

Server Objects

Replication

Management

SQL Server Agent

New Database

Select a page

General

Options

Filegroups

Script Help

Database name: Project

Owner: <default>

Use full-text indexing

Database files:

Logical Name	File ...	Fileg...	Initi...	Autogrowth	Path
Project	Ro...	PRI...	2	By 1 ...	D:\Documents\SQL Server 2008\Databases
Project_log	Log	Not ...	1	By 1...	D:\Documents\SQL Server 2008\Data...

Connection

Server: home-work-jack

Connection: Home-Work-Jack\Jack

[View connection properties](#)

Progress

Ready

Add Remove

OK Cancel

Name the new database

Customize and remember the file location.

Create Tables

1. Right-click and select "New Table"

The screenshot shows the Microsoft SQL Server Management Studio interface. The 'Object Explorer' on the left shows the 'Tables' folder selected under a database. The main window is the 'Table Designer' for a new table named 'Table_1*'. It contains a table with the following columns:

Column Name	Data Type	Allow Nulls
EmpNumber	smallint	<input type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nchar(10)	<input type="checkbox"/>

Below the column definitions is the 'Column Properties' pane, which is expanded to show the 'Table Designer' tab for the 'EmpNumber' column. The properties are as follows:

Property	Value
(Name)	EmpNumber
Allow Nulls	No
Data Type	smallint
Default Value or Binding	
Collation	<database default>
Computed Column Specification	
Condensed Data Type	smallint
Description	
Deterministic	Yes
DTS-published	No
Full-text Specification	No

Column definitions.

More Column definitions.

Default value.

Primary Key

Choose a column and set as PK.

Save table as "Employee"

Enter more columns

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure. The main window displays the Table Designer for a table named 'Employee' in the 'HOME-WORK-JACK...t - dbo.Employee' database. The table has three columns: 'EmpNumber' (smallint, not nullable), 'Email' (nvarchar(50), nullable), and 'FirstName' (nchar(10), not nullable). The 'EmpNumber' column is highlighted, and a callout box points to it with the text 'Choose a column and set as PK.'. Another callout box points to the 'Save' button in the top toolbar with the text 'Save table as "Employee"'. A third callout box points to the 'Add Column' button in the bottom toolbar with the text 'Enter more columns'. The Column Properties window at the bottom shows the properties for the selected 'EmpNumber' column, including its name, data type, and whether it allows nulls.

Column Name	Data Type	Allow Nulls
EmpNumber	smallint	<input type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nchar(10)	<input type="checkbox"/>

Column Properties

(General)

(Name)	EmpNumber
Allow Nulls	No
Data Type	smallint
Default Value or Binding	

Table Designer

Collation	<database default>
Computed Column Specification	

(General)

Surrogate Key (Identity Column)

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the 'Project' table under the 'dbo' schema. The Table Designer window is open, showing the table structure with columns 'ProjectId' (int) and 'Description' (nvarchar(10)). The 'ProjectId' column is highlighted in the Column Properties pane, and its 'Is Identity' property is set to 'Yes'.

Column Name	Data Type	Allow Nulls
ProjectId	int	<input type="checkbox"/>
Description	nchar(10)	<input checked="" type="checkbox"/>

Property	Value
(Name)	ProjectId
Allow Nulls	No
Data Type	int
Default Value or Binding	
Collation	<database default>
Computed Column Specification	
Condensed Data Type	int
Description	
Deterministic	Yes
DTS-published	No
Full-text Specification	No
Has Non-SQL Server Subscriber	No
Identity Specification	Yes
(Is Identity)	Yes
Identity Increment	1
Identity Seed	1

Create the second "Project" table.

ProjectId is the PK. To make it a surrogate key (auto number), first change the data type to int.

Then set the (Is Identity) property to Yes with increment and seed (starting value)

Composite Key

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'home-work-jack (SQL Server)'. The main window displays the 'HOME-WORK-JACK... dbo.Assignment' table design. The table has three columns: 'ProjectId' (int), 'EmployeeNumber' (nchar(10)), and 'Hours' (int). The 'ProjectId' and 'EmployeeNumber' columns are highlighted with a blue selection bar, and a yellow callout box points to them with the text 'Highlight two columns (hold Ctrl key) and set PK.'. The 'Hours' column has the 'Allow Nulls' checkbox checked. A second yellow callout box points to the table title with the text 'Create the intersection table.'. The Column Properties window at the bottom shows the 'Table Designer' tab, which is currently empty.

Column Name	Data Type	Allow Nulls
ProjectId	int	<input type="checkbox"/>
EmployeeNumber	nchar(10)	<input type="checkbox"/>
Hours	int	<input checked="" type="checkbox"/>

Column Properties

(General)

Allow Nulls: No

Data Type: <data type>

Default Value or Binding: <default value>

Table Designer

Collation: <database default>

Computed Column Specification

Condensed Data Type: <data type>

(General)

Create the intersection table.

Highlight two columns (hold Ctrl key) and set PK.

Relationship Diagram

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the 'Project' database under 'home-work-jack (SQL Serve)'. A context menu is open over the 'Database Diagrams' folder, with 'New Database Diagram' selected. The main work area shows a diagram titled 'HOME-WORK-JACK...ect - Diagram_0*' containing three tables: 'Employee' (with primary key 'EmpNumber' and fields 'Email', 'FirstName'), 'Project' (with primary key 'ProjectId' and field 'Description'), and 'Assignment' (with primary keys 'ProjectId' and 'EmployeeNumber', and field 'Hours').

Right click here to create a new relationship diagram

New Database Diagram
Filter
Reports
Refresh

First, add all tables and arrange them nicely in the work area.

Item(s) Saved

View More Metadata

Microsoft SQL Server Management Studio

File Edit View Project Debug Table Designer Database Diagram Tools Window Community Help

Table View 100%

Object Explorer

home-work-jack (SQL Serve)

Tables

- System Tables
- dbo.Employee
- dbo.Project
- dbo.Assignment
- Views
- Synonyms
- Security
- Server Objects
- Replication

Employee

Column Name	Data Type	Allow Nulls
EmpNumber	smallint	<input type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nchar(10)	<input type="checkbox"/>

Project

ProjectId
Description

Assignment

ProjectId
EmployeeNumber
Hours

Table View

- Standard
- Column Names
- Keys
- Name Only
- Custom
- Modify Custom ...

Set Primary Key

Insert Column

Delete Column

Delete Tables from Database

Remove from Diagram

Add Related Tables

Autosize Selected Tables

Arrange Selection

Relationships...

Indexes/Keys...

Fulltext Index...

Item(s) Saved

Use this button to add tables

Use these buttons to arrange tables, such as auto-sizing a table. You can also move table around and zoom the view manually.

Right click a table to change its table view. Use "standard" to display data type and null constraint.

Modify Columns in Relationship Diagram

Microsoft SQL Server Management Studio

File Edit View Project Debug Table Designer Database Diagram Tools Window Community Help

Object Explorer

home-work-jack (SQL Server)

- Databases
 - System Databases
 - Database Snapshots
 - Northwind2003-mini
 - ReportServer
 - ReportServerTempDB
 - Project
 - Database Diagrams
 - Tables
 - System Tables
 - dbo.Employee
 - dbo.Project
 - dbo.Assignment
 - Views
 - Synonyms
 - Programmability
 - Service Broker
 - Storage
 - Security
 - Security
 - Server Objects
 - Replication

HOME-WORK-JACK....ect - Diagram_0*

Employee

Column Name	Data Type	Allow Nulls
EmpNumber	smallint	<input type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nchar(10)	<input type="checkbox"/>

Project

Column Name	Data Type	Allow Nulls
ProjectId		<input type="checkbox"/>
Description		<input type="checkbox"/>

Assignment

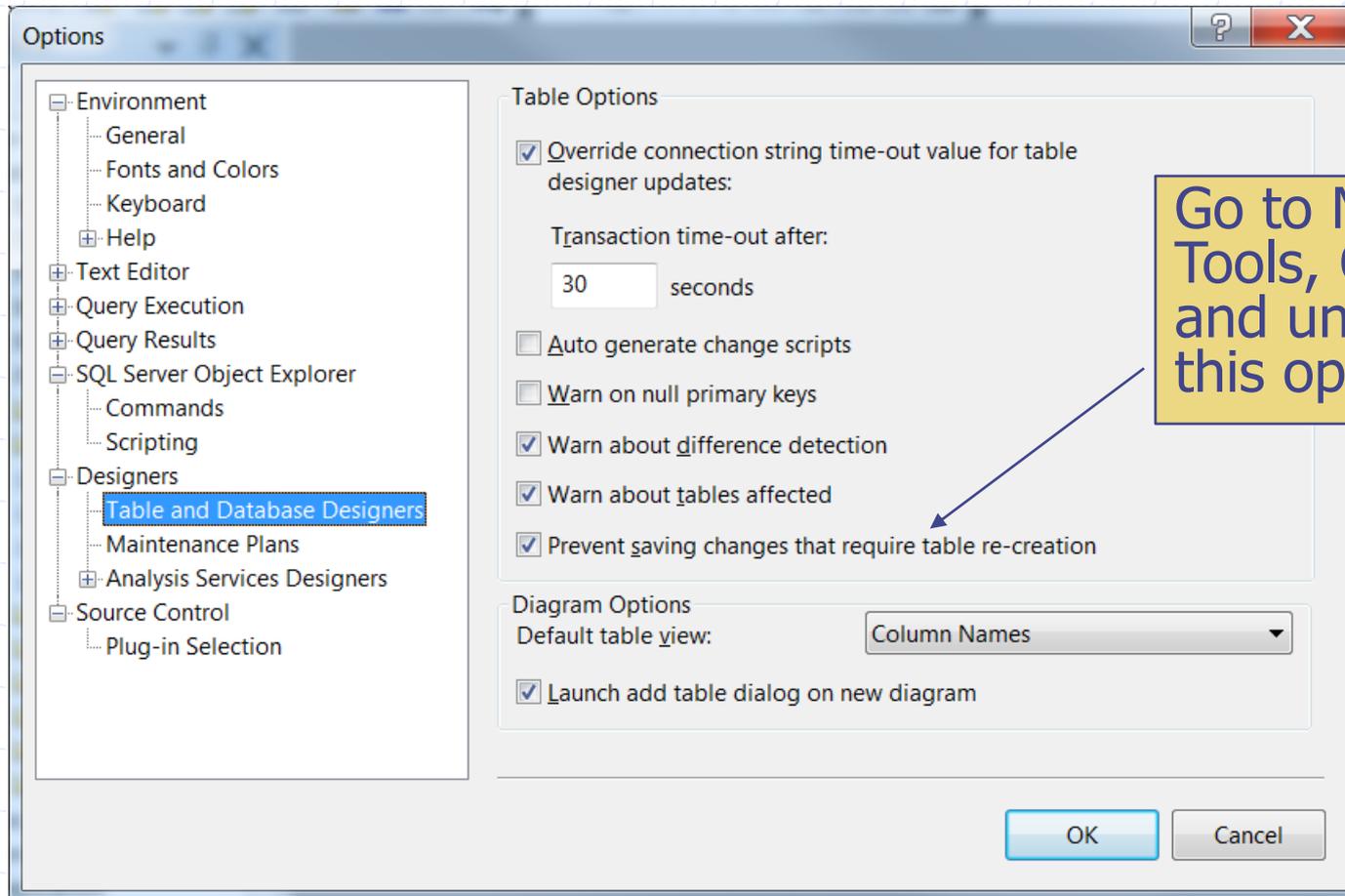
Column Name	Data Type	Allow Nulls
ProjectId	int	<input type="checkbox"/>
EmployeeNumber	nchar(10)	<input type="checkbox"/>
Hours	nchar(10)	<input checked="" type="checkbox"/>

Item(s) Saved

The FK and PK pair have to have the same data type. Change it to "smallint" and save the diagram.

Note: you may encounter a error message to prevent you to do it. If so, see next slide.

!!Set Option!!



Create Relationship

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the 'home-work-jack (SQL Server)' instance with a 'Project' database. The main window displays a diagram titled 'HOME-WORK-JACK...ect - Diagram_0*' containing three tables: 'Employee *', 'Project', and 'Assignment *'. The 'Employee *' table has columns: EmpNumber (smallint, primary key), Email (nvarchar(50)), and FirstName (nchar(10)). The 'Project' table has columns: ProjectId (primary key) and Description. The 'Assignment *' table has columns: ProjectId (foreign key to Project), EmployeeNumber (foreign key to Employee), and Hours (int). A relationship line connects the 'ProjectId' column in 'Assignment *' to the 'ProjectId' column in 'Project'. A context menu is open over this relationship line, showing the option 'Delete Relationships from Database'. Three callout boxes provide instructions: one points to the asterisk on the Employee table name, another points to the relationship line, and a third points to the 'Delete Relationships from Database' option.

Table	Column Name	Data Type	Allow Nulls
Employee *	EmpNumber	smallint	<input type="checkbox"/>
	Email	nvarchar(50)	<input checked="" type="checkbox"/>
	FirstName	nchar(10)	<input type="checkbox"/>
Project	ProjectId		<input type="checkbox"/>
	Description		
Assignment *	ProjectId	int	<input type="checkbox"/>
	EmployeeNumber	smallint	<input type="checkbox"/>
	Hours	int	<input checked="" type="checkbox"/>

* means not saved.

Drag the FK column to the primary key column. When releasing the mouse, a dialog will pop up to finish the rest steps. Once the relationship is created, you will see a symbol like this.

Right click on the relationship line and you can delete it.

More Commands

Manage keys and indexes

Highlight a table and you can see these command buttons enabled.

Remove a table from diagram

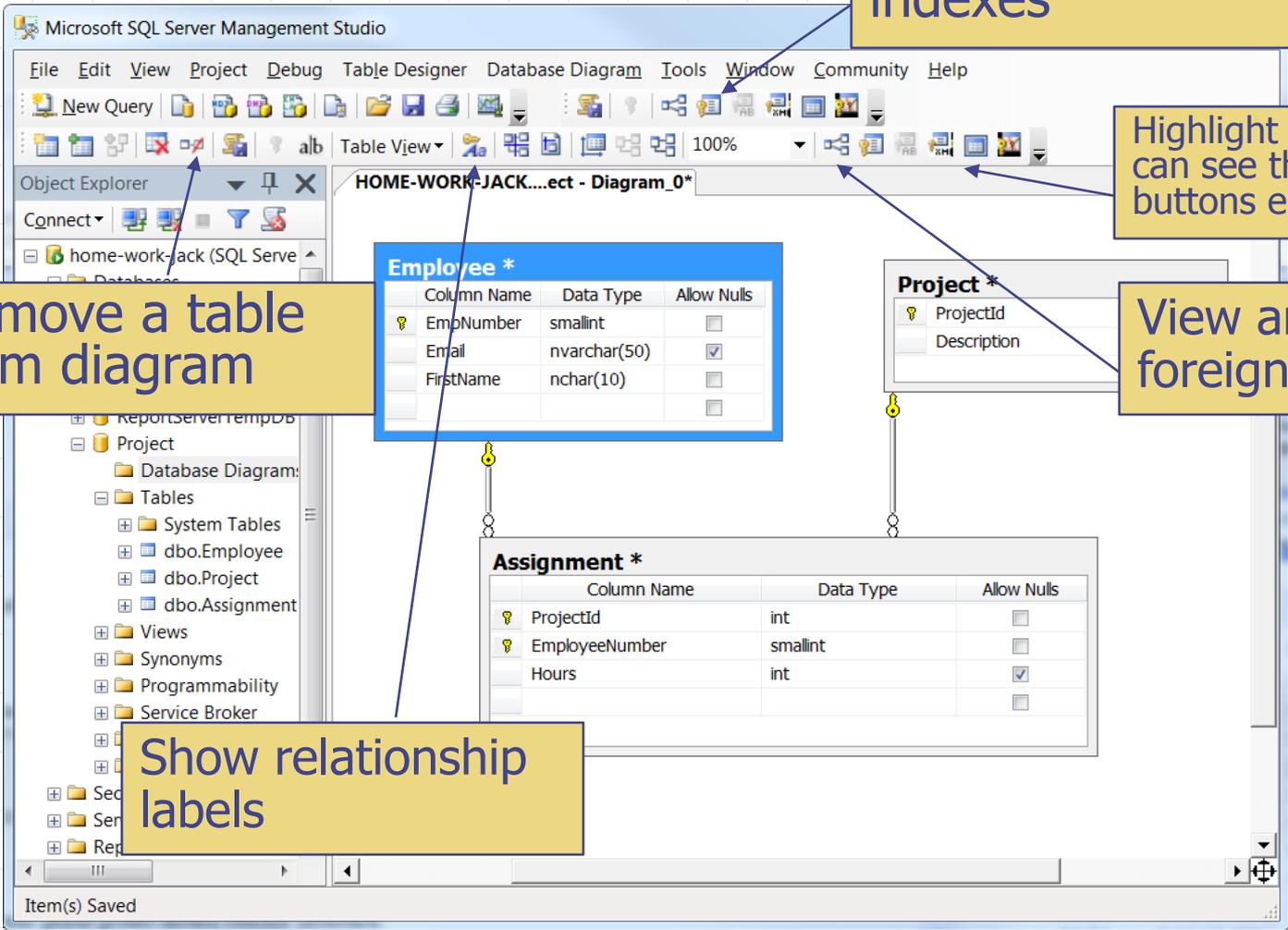
Column Name	Data Type	Allow Nulls
EmpNumber	smallint	<input type="checkbox"/>
Email	nvarchar(50)	<input checked="" type="checkbox"/>
FirstName	nchar(10)	<input type="checkbox"/>

View and configure foreign keys

Column Name	Data Type	Allow Nulls
ProjectId	int	<input type="checkbox"/>
Description	nvarchar(255)	<input checked="" type="checkbox"/>

Show relationship labels

Column Name	Data Type	Allow Nulls
ProjectId	int	<input type="checkbox"/>
EmployeeNumber	smallint	<input type="checkbox"/>
Hours	int	<input checked="" type="checkbox"/>



Referential Actions

Microsoft SQL Server Management Studio

File Edit View Project Debug Table Designer Database Diagram Tools Window Community Help

Object Explorer

Connect

home-work-jack

Databases

System Databases

Database

Northwind

ReportServer

ReportServer

Project

Database

dbo

Tables

Views

Synonyms

Programs

Services

Storage

Security

Security

Server Objects

Replication

Management

SQL Server Agent

Foreign Key Relationships

Selected Relationship:
FK_Assignment_Employee

Editing properties for existing relationship.

(General)

Check Existing Data On Cre Yes

Tables And Columns Specif

Database Designer

Enforce For Replication Yes

Enforce Foreign Key Constr: Yes

INSERT And UPDATE Speci

Delete Rule No Action

Update Rule No Action

Identity

(Name) FK_Assignment_Employee

Description

Add Delete Close

Ready

First, select a table and view its foreign keys

Change referential actions

Click close to finish.

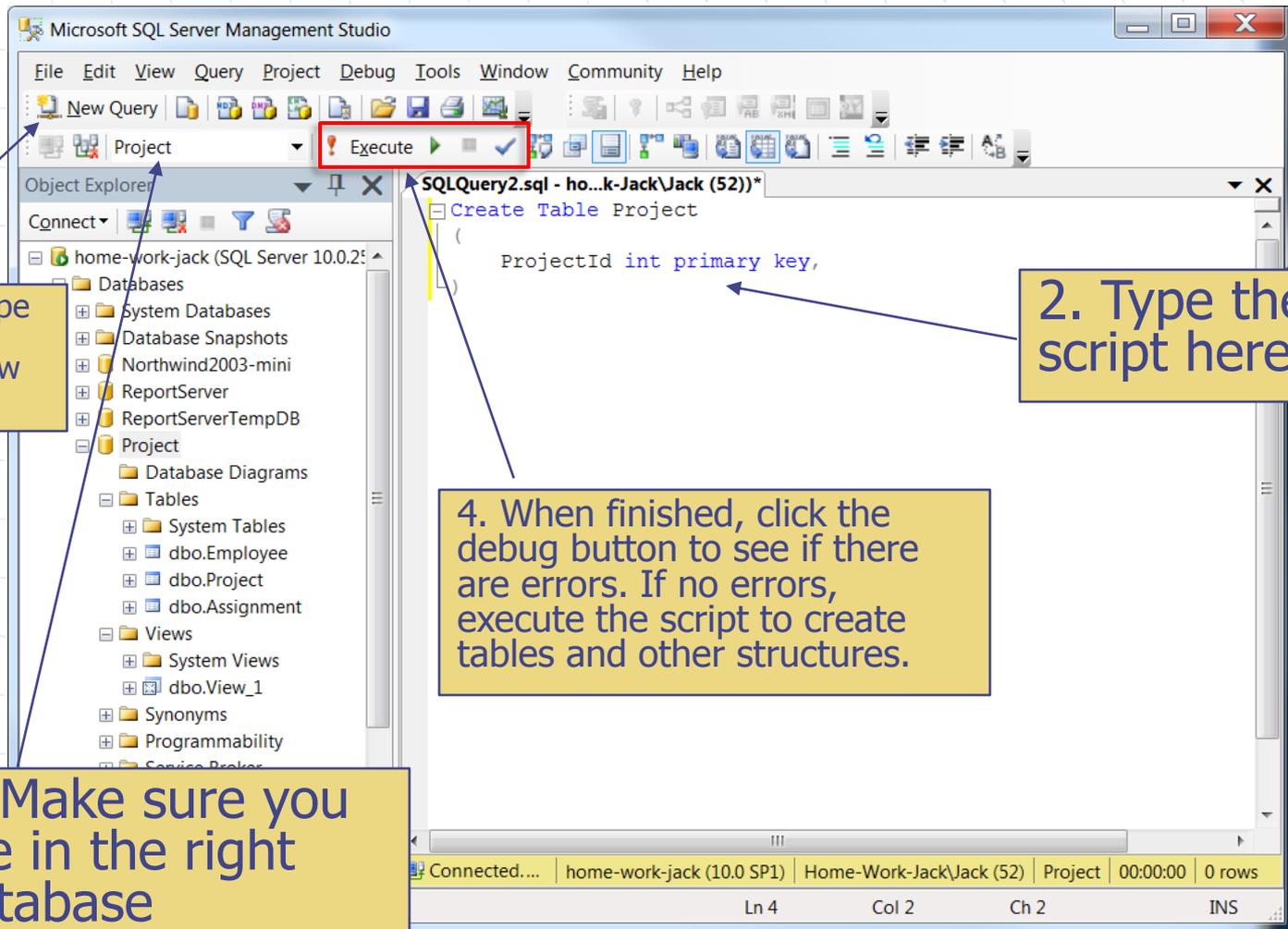
Type SQL in the Query Window

1. You can also type a new script by clicking on the New Query button.

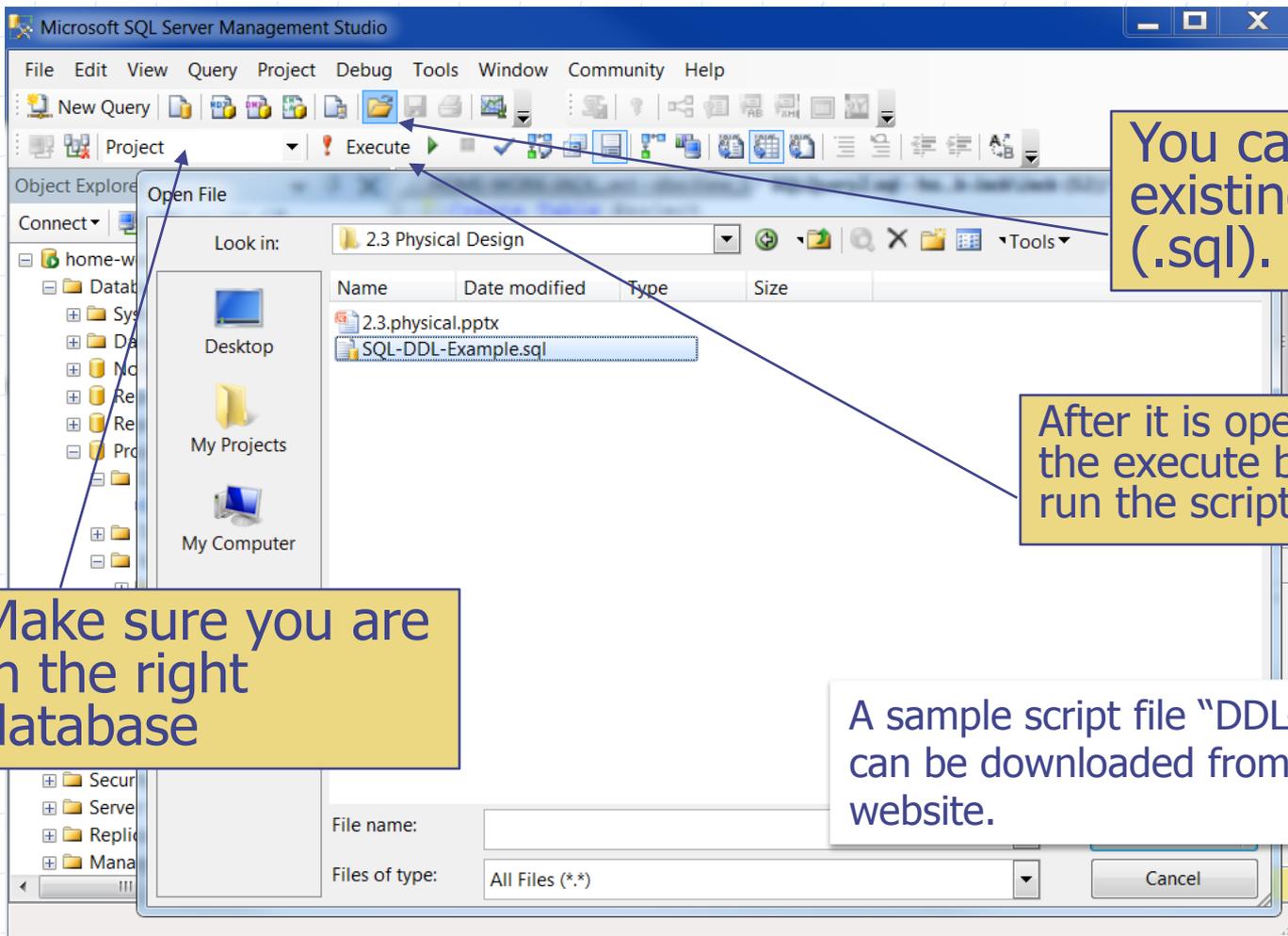
3. Make sure you are in the right database

2. Type the script here.

4. When finished, click the debug button to see if there are errors. If no errors, execute the script to create tables and other structures.



Use Script Files



You can open an existing script file (.sql).

After it is opened, click the execute button to run the script.

Make sure you are in the right database

A sample script file "DDL-Project.sql" can be downloaded from the course website.

Enter and Update Data

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the 'home-work-jack (SQL Server)' instance with the 'dbo.Employee' table selected. The main window displays the table data:

EmpNumber	Email	FirstName
101	2@j.com	jack
102	444@d.com	NULL
*	NULL	NULL

The table has a context menu open over it. The menu items are:

- New Table...
- Design
- Select Top 1000 Rows
- Edit Top 200 Rows
- Script Table as
- View Dependencies
- Full-Text index
- Storage
- Policies
- Facets
- Start PowerShell
- Reports

Callouts provide instructions:

- The red mark indicates the data is not saved to table yet. Select another row to save the data.
- New record
- Right-click the table and select "Edit Top 200 Rows" to enter and modify data.
- Right-click the table and select "Design" to modify the metadata.

Error Message

Microsoft SQL Server Management Studio

File Edit View Project Debug Query Designer Tools Window Community Help

Object Explorer

home-work-jack (SQL Serve)

- Databases
 - System Databases
 - Database Snapshots
 - Northwind2003-mini
 - ReportServer
 - ReportServerTempDB
 - Project
 - Database Diagrams
 - Tables
 - System Tables
 - dbo.Employee
 - Views
 - Synonyms
 - Programmability
 - Service Broker
 - Storage
 - Security
 - Security
 - Server Objects
 - Replication
 - Management
 - SQL Server Agent

EmpNumber	Email	FirstName
101	2@j.com	jack
102	444@d.com	peter
103	NULL	NULL
* NULL	NULL	NULL

Microsoft SQL Server Management Studio

No row was updated.

The data in row 3 was not committed.
Error Source: .Net SqlClient Data Provider.
Error Message: Cannot insert the value NULL into column 'FirstName', table 'Project.dbo.Employee'; column does not allow nulls. INSERT fails.
The statement has been terminated.

Correct the errors and retry or press ESC to cancel the change(s).

OK

Ready

If you move away from the current record without entering a value for the "required" column, you will get this error message.

Create Views

Microsoft SQL Server Management Studio

File Edit View Project Debug Query Designer Tools Window Community Help

Object Explorer

Connect

Databases

- System Databases
- Database Snapshots
- Northwind2003-mini
- ReportServer
- ReportServerTempDB
- Project
 - Database Diagrams
 - Tables
 - System Tables
 - dbo.Employee
 - dbo.Project
 - dbo.Assignment
 - Views
 - New View...
 - Filter
 - Start PowerShell
 - Reports
 - Refresh
- System
- Synon
- Progr
- Servic
- Stora
- Secur
- Security
- Server Objects
- Replication

HOME-WORK-JACK.P...ct - dbo.View_1*

Employee

- * (All Columns)
- EmpNumber
- Email
- FirstName

Column	Alias	Table	Output	Sort Type	Sort Order	Filter
*		Employee	<input checked="" type="checkbox"/>			

```
SELECT dbo.Employee.*  
FROM dbo.Employee
```

EmpNumber	Email	FirstName
101	2@j.com	jack
102	444@d.com	peter
*	NULL	NULL

Item(s) Saved

A view is like a virtual table. You can "SELECT FROM" a view just like a table.