

Introduction to Databases

How Do RDBMS Work? Managing DBs Using IDEs



SoftUni Team
Technical Trainers



SoftUni



Software University

<https://softuni.bg>

Table of Contents

1. Data Management
2. Database Engines
3. Data Types in SQL Server
4. Database Modeling





Data Management

When Do We Need a Database?

Storage vs. Management (1)

- Conventional Data Storage
 - Notes
 - Receipts



Storage vs. Management (2)

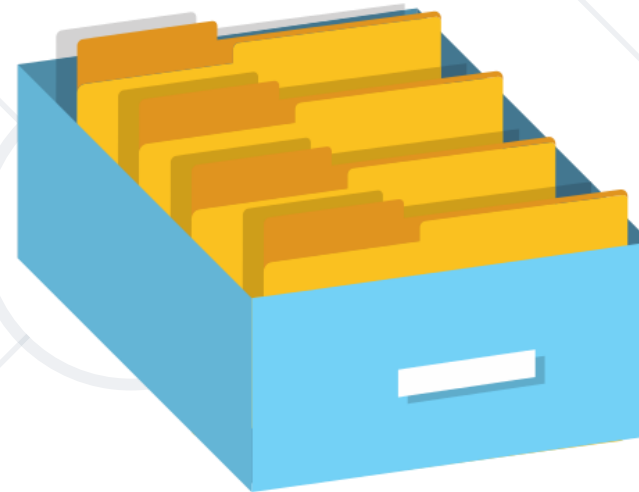
- We can group related pieces of data into separate columns



Order#	Date	Customer	Product	S/N	Unit Price	Qty	Total
315	07/16/2016	David Rivers	Oil Pump	OP147-0623	69.90	1	69.90

Storage vs. Management (3)

- Storing data is **not** the primary reason to use a Database
- Flat storage **eventually** runs into **issues** with
 - Size
 - Ease of updating
 - Searching
 - Concurrency
 - Security
 - Consistency



Databases and RDBMS

- A database is an **organized** collection of information
 - It imposes **rules** on the contained data
 - Relational storage first proposed by **Edgar Codd** in 1970
- A **R**elational **D**ata **B**ase **M**anagement **S**ystem provides tools to **manage** the database
 - It **parses requests** from the user and takes the **appropriate** action
 - The user doesn't have direct access to the stored data

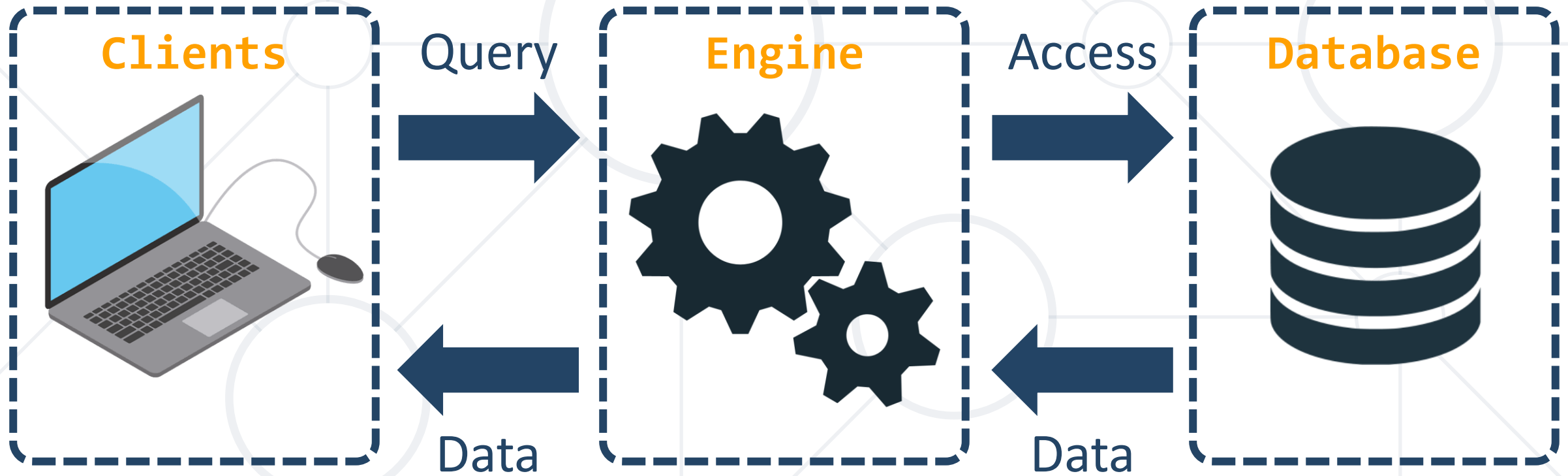




Database Engines

Database Engine Flow

- SQL Server uses the Client-Server Model



- Download **SQL Server Express** Edition from Microsoft

go.microsoft.com/fwlink/?linkid=866662

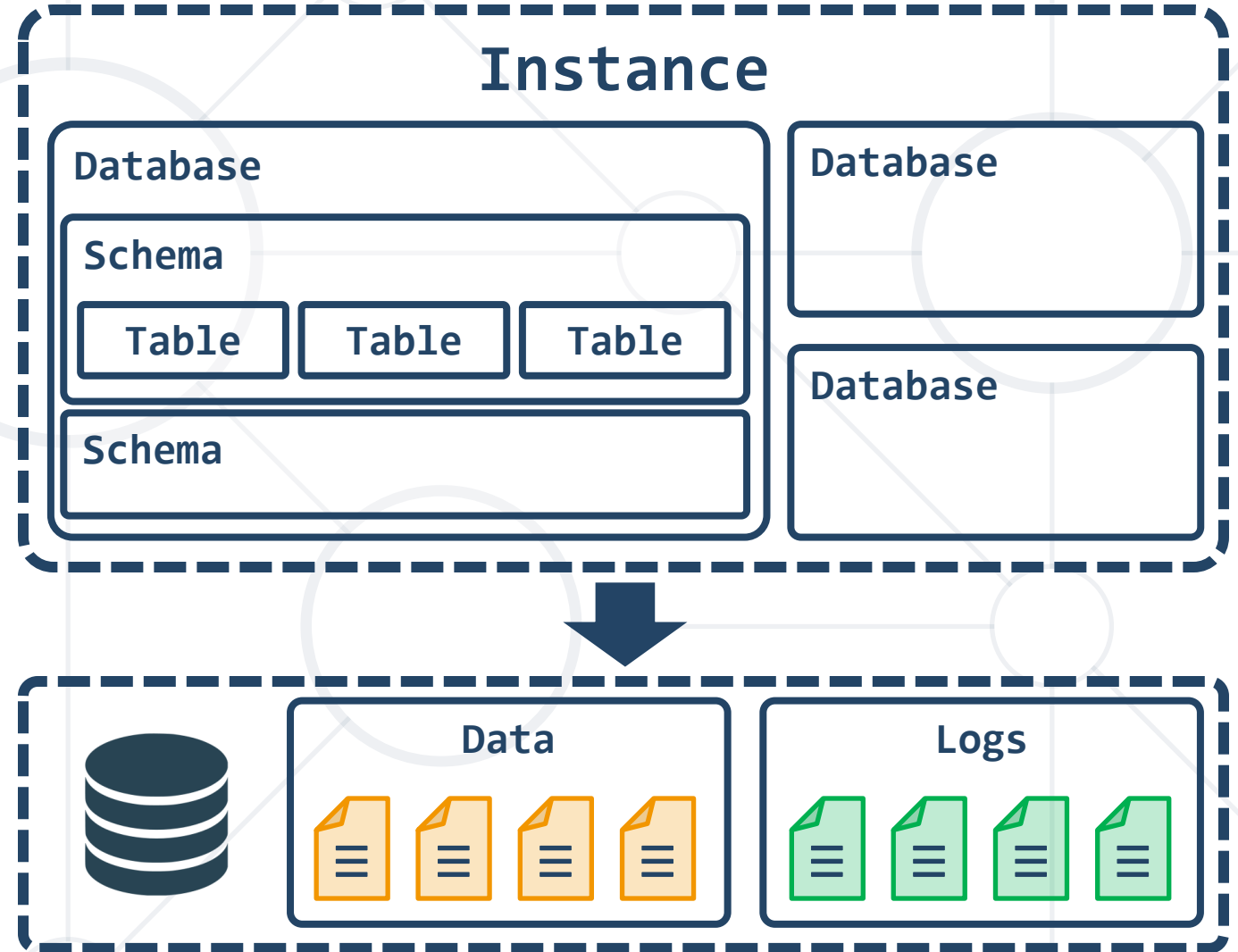
- Download SQL Server **Management Studio** separately

<https://aka.ms/ssmsfullsetup>



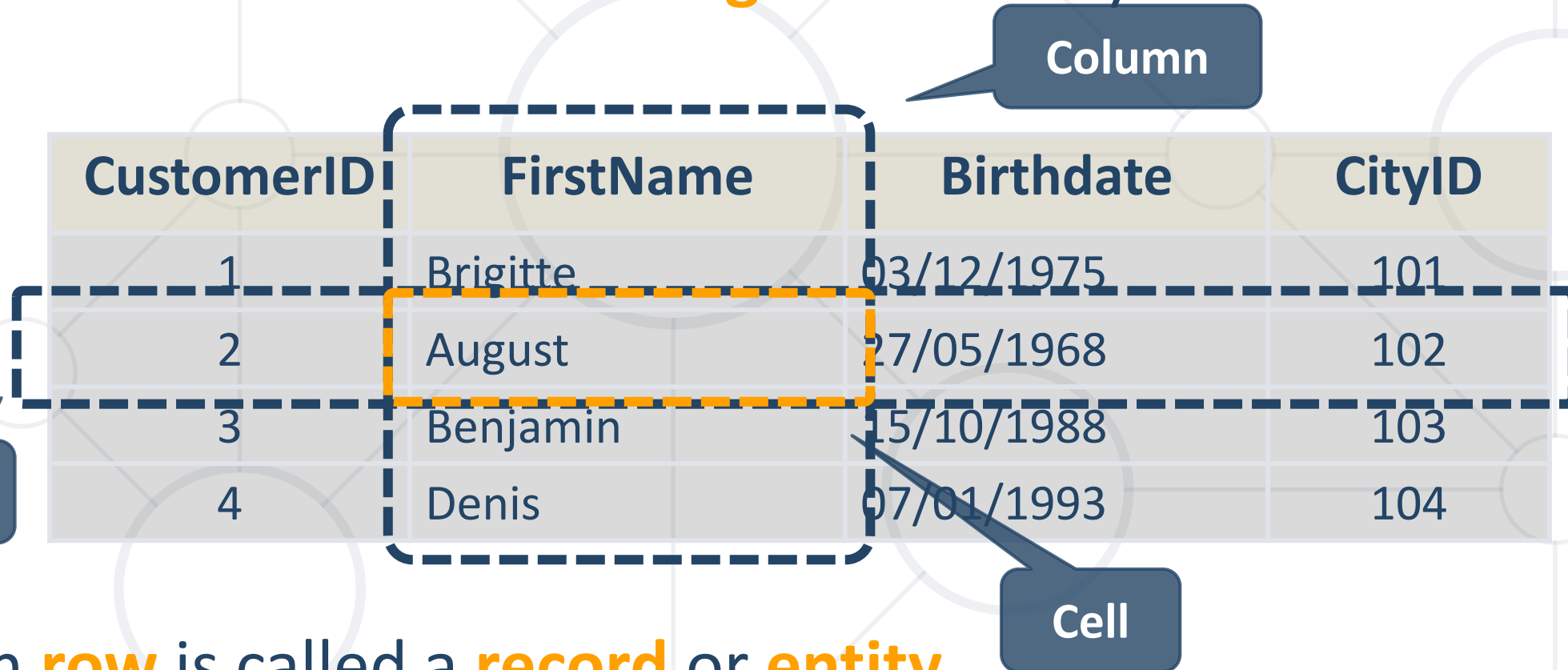
SQL Server Architecture

- Logical Storage
 - Instance
 - Database
 - Schema
 - Table
 - Table
 - Table
 - Schema
 - Database
 - Database
- Physical Storage
 - Data Files and Log files
 - Data Pages



Database Table Elements

- The table is the main **building block** of any database



CustomerID	FirstName	Birthdate	CityID
1	Brigitte	03/12/1975	101
2	August	27/05/1968	102
3	Benjamin	15/10/1988	103
4	Denis	07/01/1993	104

- Each **row** is called a **record** or **entity**
- Columns (**fields**) define the **type** of data they contain

- To communicate with the Engine we use **SQL**
 - **Declarative** language
- Logically divided in four sections
 - **Data Definition** – describe the structure of our data
 - **Data Manipulation** – store and retrieve data
 - **Data Control** – define who can access the data
 - **Transaction Control** – bundle operations and allow rollback



Data Types in SQL Server

- Numeric
 - **BIT** (1-bit), **TINYINT** (8-bit), **SMALLINT** (16-bit)
 - **INT** (32-bit), **BIGINT** (64-bit)
 - **FLOAT**, **REAL**, **DECIMAL**(**precision**, **scale**)
- Textual
 - **CHAR**(**size**) – fixed size string
 - **VARCHAR**(**size**) – variable size string
 - **NCHAR**(**size**) – Unicode fixed size string
 - **NVARCHAR**(**size**) – Unicode variable size string

Size of Textual Characters

```
DECLARE @VarcharVar VARCHAR(5) = 'Test';  
DECLARE @NVarcharVar NVARCHAR(5) = 'Test';  
DECLARE @CharVar CHAR(5) = 'Test';  
DECLARE @NCharVar NCHAR(5) = 'Test';  
  
SELECT DATALENGTH(@VarcharVar),  
       DATALENGTH(@NVarcharVar),  
       DATALENGTH(@CharVar),  
       DATALENGTH(@NCharVar)
```


Data Types in SQL Server (2)

- Binary data
 - **BINARY(size)** – fixed length sequence of bits
 - **VARBINARY(size)** – a sequence of bits, 1-8000 bytes or **MAX** (2GB)
- Date and time
 - **DATE** – date in range 0001-01-01 through 9999-12-31
 - **DATETIME** – date and time with precision of 1/300 sec
 - **DATETIME2** – type that has a larger date range
 - **SMALLDATETIME** – date and time (1 minute precision)
 - **TIME** – defines a time of a day (no time zone)
 - **DATETIMEOFFSET** – date and time that has time zone

Date and Time in SQL Server

DATA TYPE	① RANGE OF VALUES	② ACCURACY	③ STORAGE SPACE
<i>SMALLDATETIME</i>	01/01/1900 to 06/06/2079	<u>1 minute</u>	4 bytes
<i>DATETIME</i>	01/01/ <u>1753</u> to 12/31/9999	0.00333 seconds	8 bytes
<i>DATETIME2</i>	01/01/ <u>0001</u> to 12/31/9999	100 <u>nanoseconds</u>	<u>6 to 8 bytes</u>
<i>DATETIMEOFFSET</i>	01/01/0001 to 12/31/9999	100 nanoseconds	8 to 10 bytes
<i>DATE</i>	01/01/0001 to 12/31/9999	1 day	<u>3 bytes</u>
<i>TIME</i>	00:00:00.0000000 to 23:59:59.9999999	100 nanoseconds	3 to 5 bytes

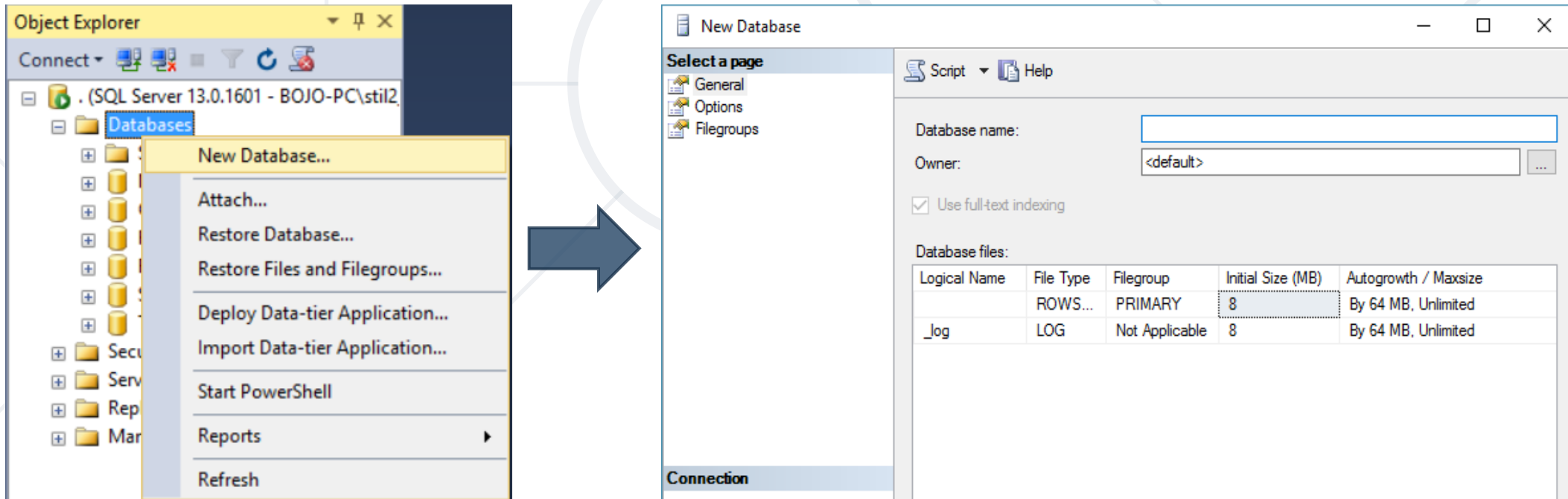


Database Modelling

Data Definition Using SSMS

Creating a New Database

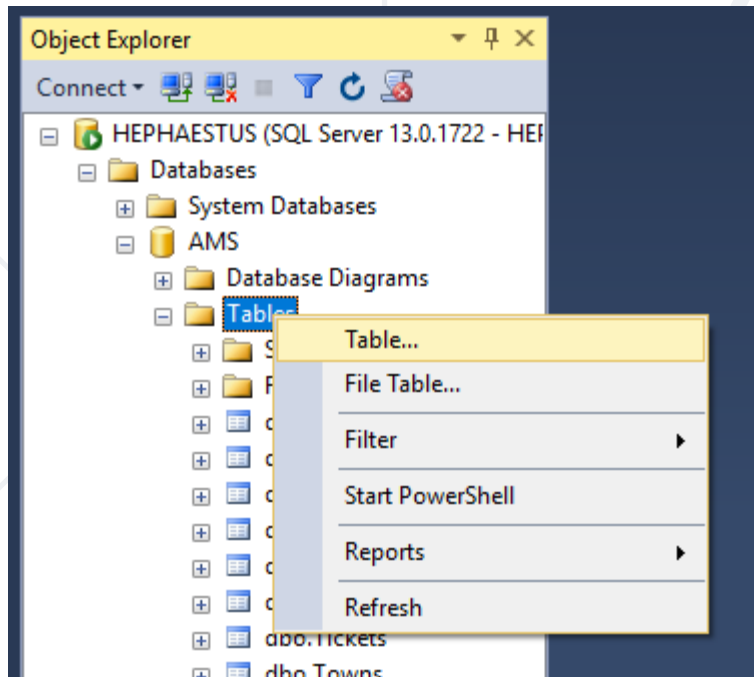
- Select **New Database** from the **context menu** under "Databases"




- You may need to **Refresh [F5]** to see the results

Creating Tables (1)

- From the **context menu** under "**Tables**" inside the desired database

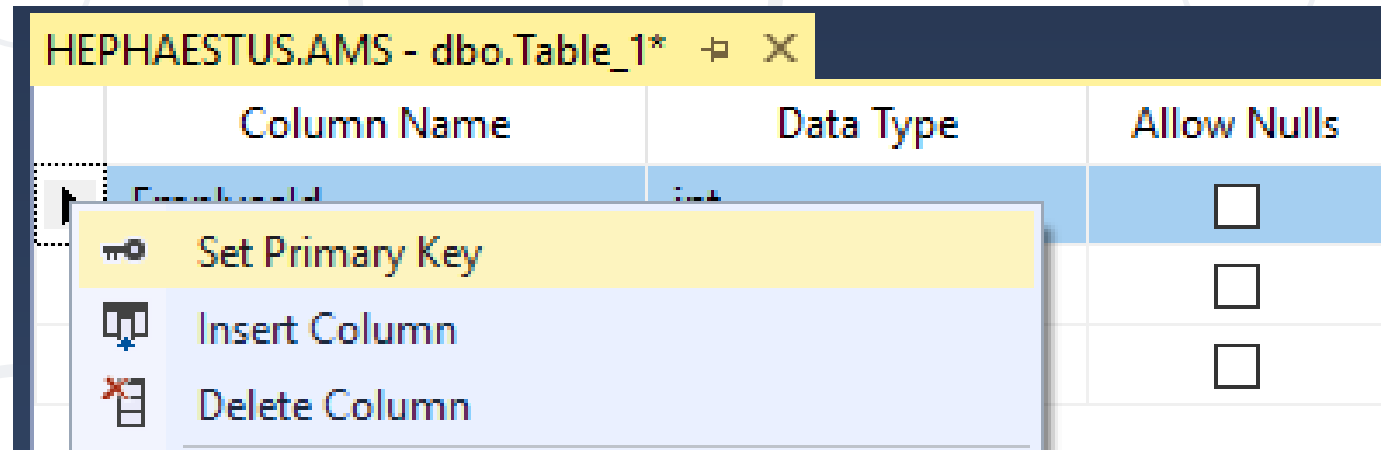


	Column Name	Data Type	Allow Nulls
	EmployeeID	int	<input type="checkbox"/>
	FirstName	nvarchar(50)	<input type="checkbox"/>
	LastName	nvarchar(50)	<input type="checkbox"/>
			<input type="checkbox"/>

- Table name can be set from its **Properties [F4]** or when it is **saved**

Creating Tables (2)

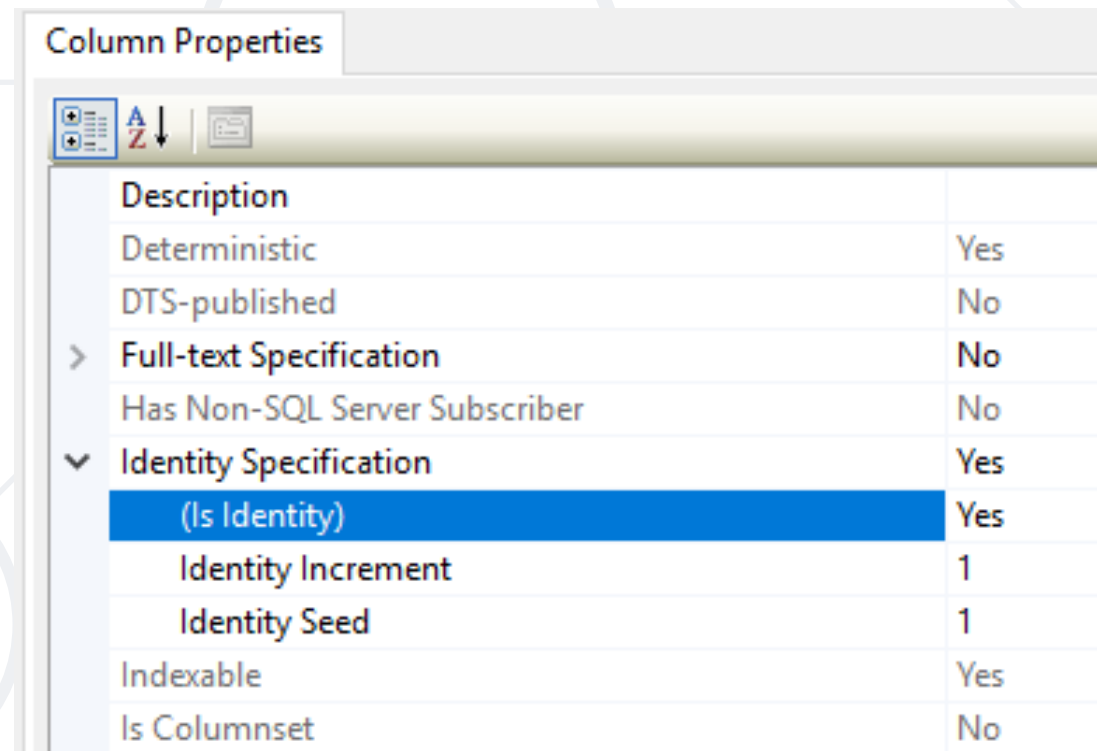
- A **Primary Key** is used to uniquely identify and index records
- Setting **primary key** on a column:



- **Identity** – The value in the column is automatically incremented when a new record is added
 - These values cannot be assigned manually
 - **Identity Seed** – the initial number (1 by default)
 - **Identity Increment** – how much each consecutive value is incremented

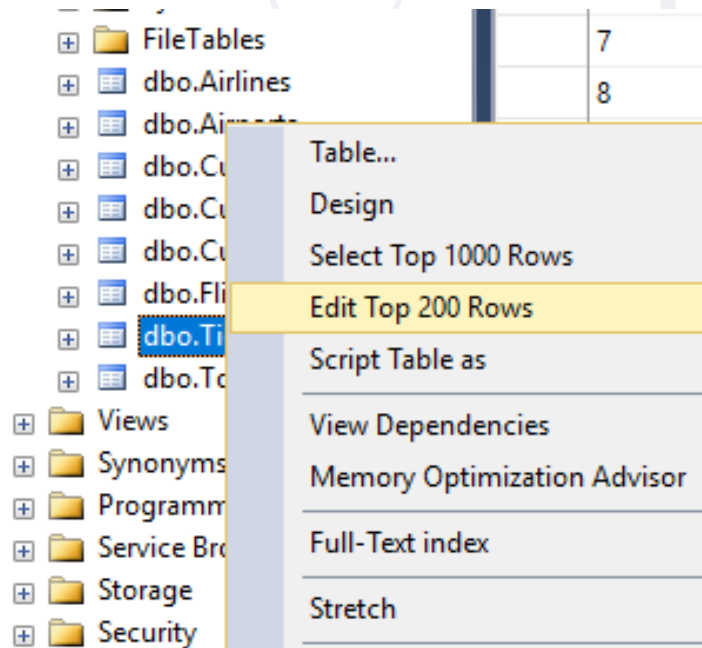
Creating Tables (4)

- Setting an identity through the "**Column Properties**" window:



Storing and Retrieving Data (1)

- We can **add**, **modify** and **read** records with Management Studio
- To insert or edit a record, click **Edit** from the context menu

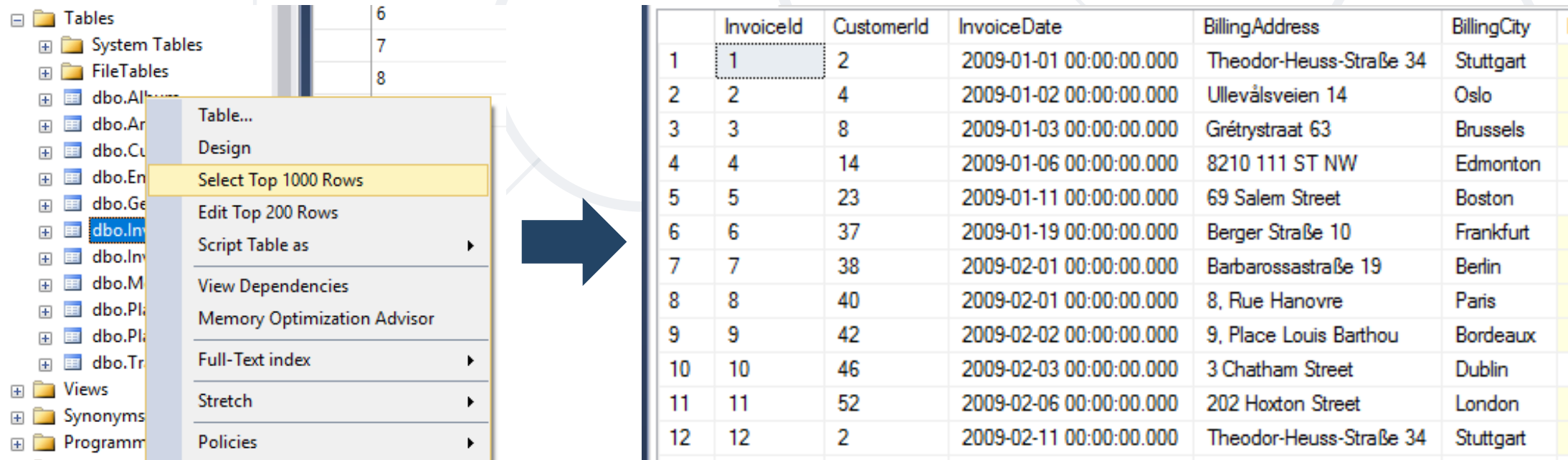


	TicketID	Price	Class	Seat	CustomerID
▶	1	4500.00	First	233-A	3
	2	2699.85	Second	123-D	1
	3	1800.75	Second	12-Z	2
	4	616.02	Third	45-Q	2
	5	840.00	Third	201-R	4
	6	3150.00	Second	13-T	1
	7	5500.00	First	98-O	2
	8	100.00	First	1	1
*	NULL	NULL	NULL	NULL	NULL

Enter data at the end to add a new row

Storing and Retrieving Data (2)

- To retrieve records, click **Select** from the context menu



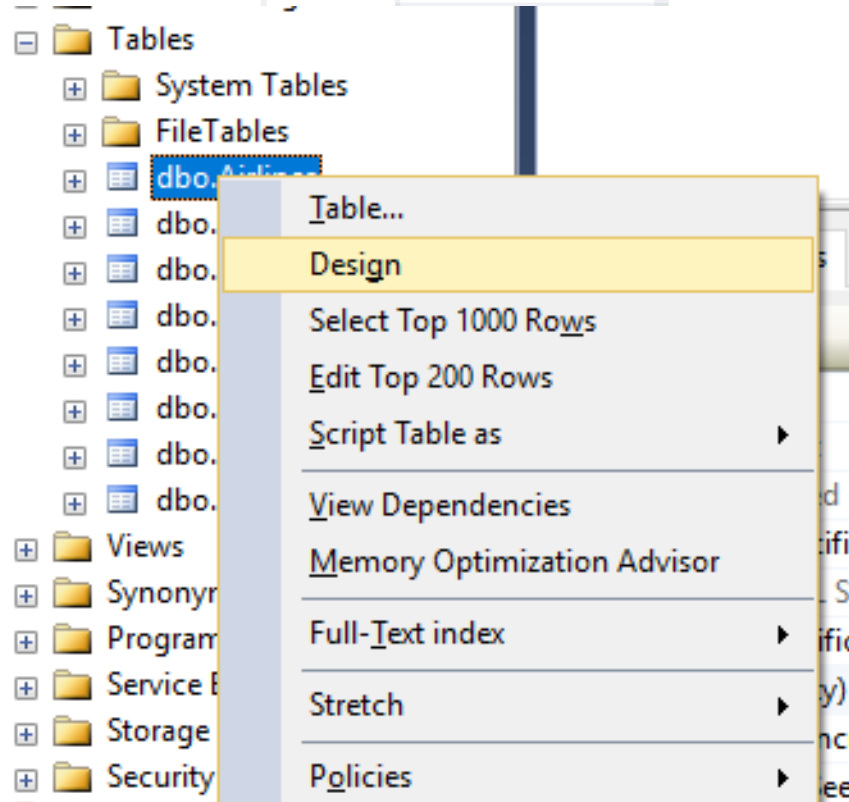
The screenshot shows the SQL Server Enterprise Manager interface. On the left, a tree view displays the database structure, including 'Tables', 'System Tables', 'FileTables', and various user tables like 'dbo.Albums', 'dbo.Artists', etc. The 'dbo.Invoices' table is selected. A right-click context menu is open over this table, with the option 'Select Top 1000 Rows' highlighted in yellow. A large blue arrow points from this menu option to a data table on the right. This table displays the first 12 rows of the 'Invoices' table, with columns: InvoiceId, CustomerId, InvoiceDate, BillingAddress, and BillingCity. The first row is highlighted with a dotted border.

	InvoiceId	CustomerId	InvoiceDate	BillingAddress	BillingCity
1	1	2	2009-01-01 00:00:00.000	Theodor-Heuss-Straße 34	Stuttgart
2	2	4	2009-01-02 00:00:00.000	Ullevålsveien 14	Oslo
3	3	8	2009-01-03 00:00:00.000	Grétrystraat 63	Brussels
4	4	14	2009-01-06 00:00:00.000	8210 111 ST NW	Edmonton
5	5	23	2009-01-11 00:00:00.000	69 Salem Street	Boston
6	6	37	2009-01-19 00:00:00.000	Berger Straße 10	Frankfurt
7	7	38	2009-02-01 00:00:00.000	Barbarossastraße 19	Berlin
8	8	40	2009-02-01 00:00:00.000	8, Rue Hanovre	Paris
9	9	42	2009-02-02 00:00:00.000	9, Place Louis Barthou	Bordeaux
10	10	46	2009-02-03 00:00:00.000	3 Chatham Street	Dublin
11	11	52	2009-02-06 00:00:00.000	202 Hoxton Street	London
12	12	2	2009-02-11 00:00:00.000	Theodor-Heuss-Straße 34	Stuttgart

- The received information can be customized with **SQL queries**

Altering Tables

- You can change the properties of a table after its creation
- Select **Design** from the table's context menu



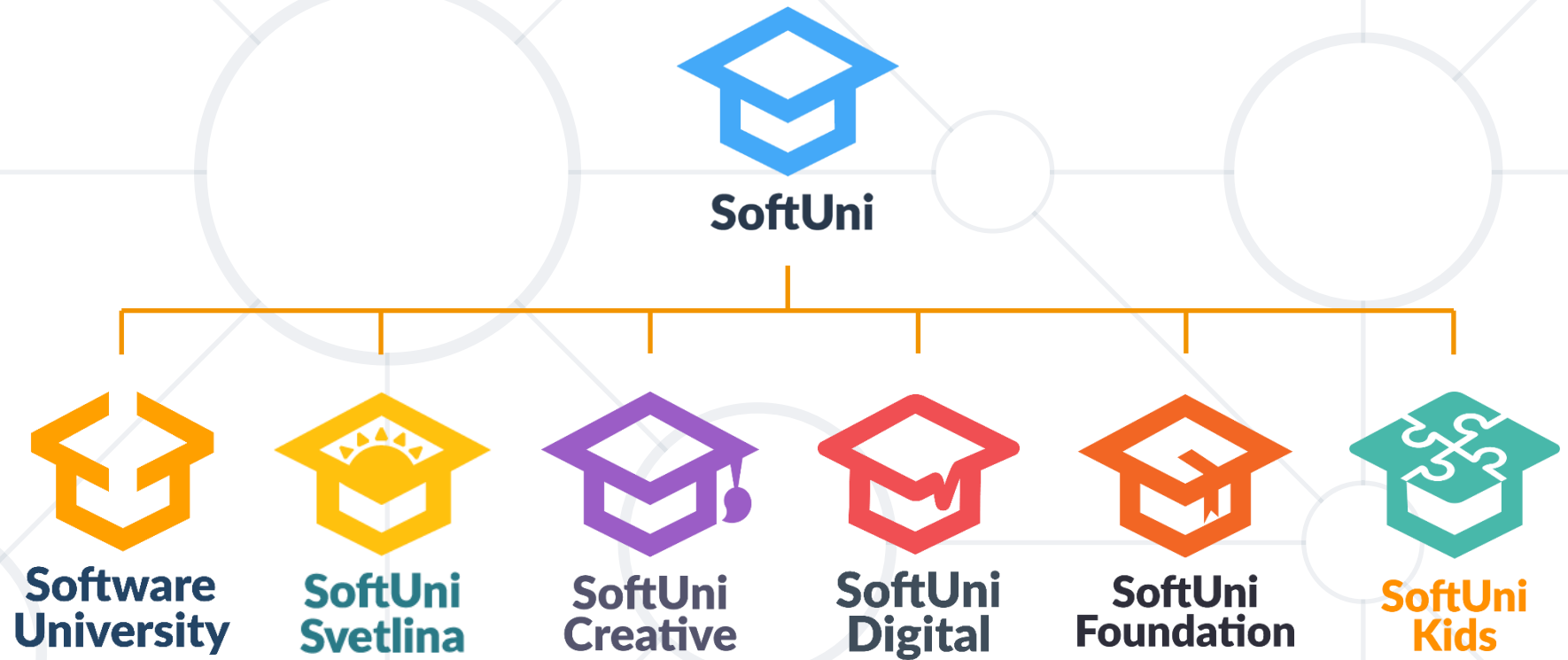
	Column Name	Data Type	Allow Nulls
PK	AirlineID	int	<input type="checkbox"/>
	AirlineName	varchar(30)	<input type="checkbox"/>
	Nationality	varchar(30)	<input type="checkbox"/>
	Rating	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Changes cannot conflict with existing rules!

- RDBMS store and manage data
- Table relations reduce repetition and complexity
- Table columns have **fixed types**
- We can use Management Studio to **create** and **customize** tables



Questions?



- Software University – High-Quality Education, Profession and Job for Software Developers

- softuni.bg, softuni.org

- Software University Foundation

- softuni.foundation

- Software University @ Facebook

- facebook.com/SoftwareUniversity

- Software University Forums

- forum.softuni.bg



- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – <https://softuni.org>
- © Software University – <https://softuni.bg>

