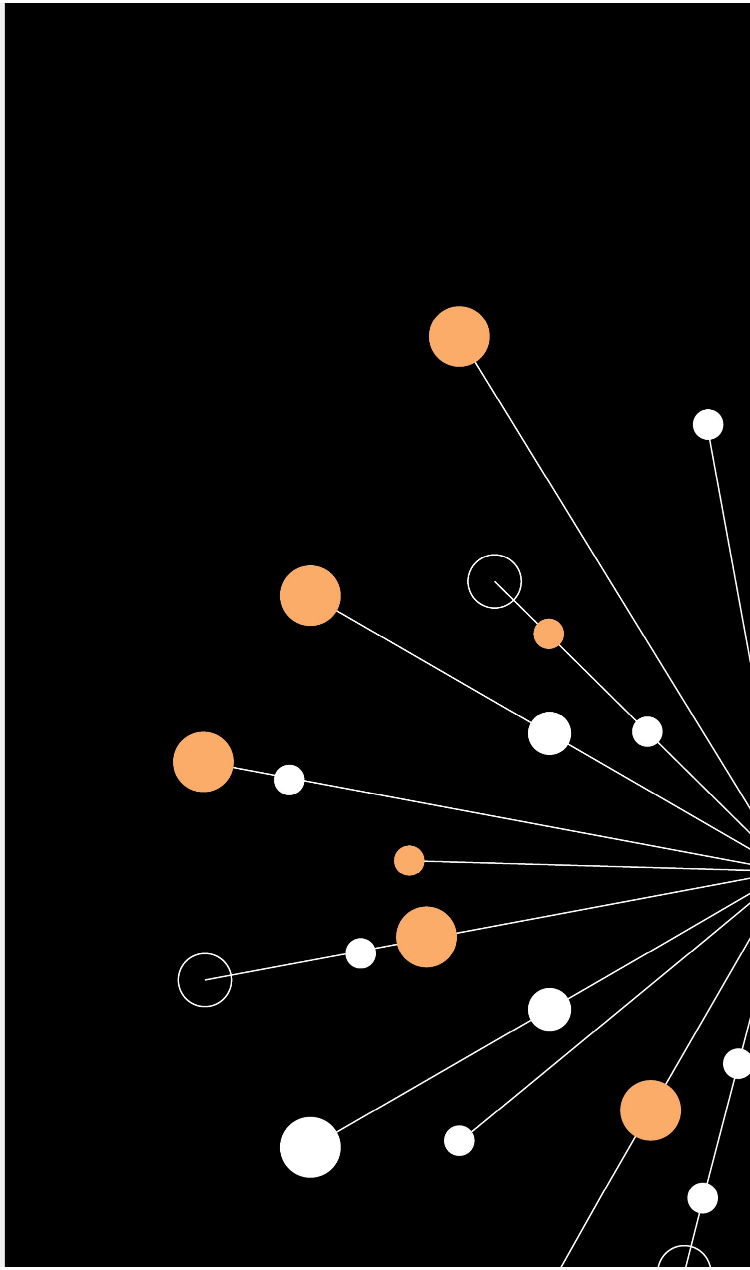
Physical Data Models

Data Modelling

Contents

- Physical Data Models
  - Uses table names
  - Column names
  - Column data types
  - Column constraints



Physical data model characteristics

Physical data models represent how the model will be built **in the database**.

A physical data model shows all table structures, including column name, column data type, column constraints, primary key, foreign key, and relationships between tables.

Features of a physical data model include:

- specification of all tables and columns.
- foreign keys that are used to identify relationships between tables.
- physical considerations that may cause the physical data model to be different from the logical data model.

Physical data model characteristics

The conceptual and logical data models are database agnostic – databases are not considered during their development.

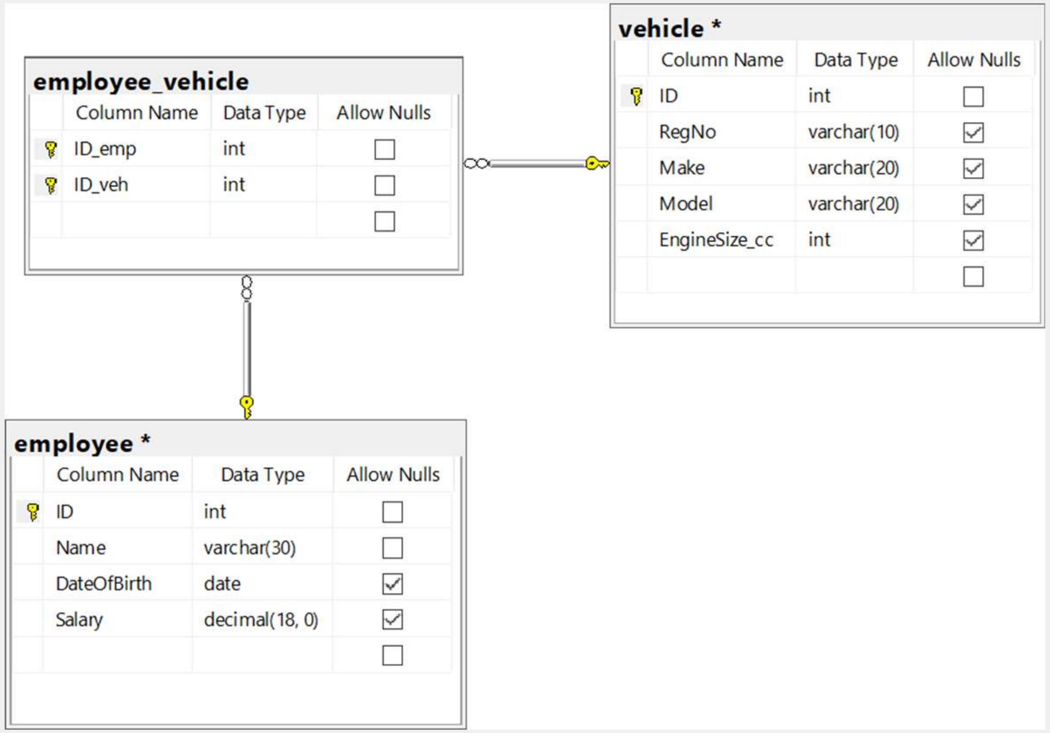
The physical data model will be different for different relational databases. For example, data type for a column may be different between MySQL and SQL Server. It’s important to choose a database that supports data types relevant to the data for the project.

Physical data model characteristics

Physical data models reflect the terminology and functionality of the database they are representing, rather than theoretical concepts, which are used in conceptual and logical data models. So, the vocabulary used in physical data models is different:

- Entities become tables.
- Attributes become columns.
- Instances of an entity become rows.

Physical data model: example



employee_vehicle

	Column Name	Data Type	Allow Nulls
🔑	ID_emp	int	□
🔑	ID_veh	int	□
			□

vehicle *

	Column Name	Data Type	Allow Nulls
🔑	ID	int	□
	RegNo	varchar(10)	☑
	Make	varchar(20)	☑
	Model	varchar(20)	☑
	EngineSize_cc	int	☑
			□

employee *

	Column Name	Data Type	Allow Nulls
🔑	ID	int	□
	Name	varchar(30)	□
	DateOfBirth	date	☑
	Salary	decimal(18, 0)	☑
			□

An example of a physical data model using the tool in SQL Server

SQL Server String Data Types

Data type	Description
char(n)	Fixed width character string
varchar(n)	Variable width character string
varchar(max)	Variable width character string
text	Variable width character string
nchar	Fixed width Unicode string
nvarchar	Variable width Unicode string

SQL Server Numeric Data Types

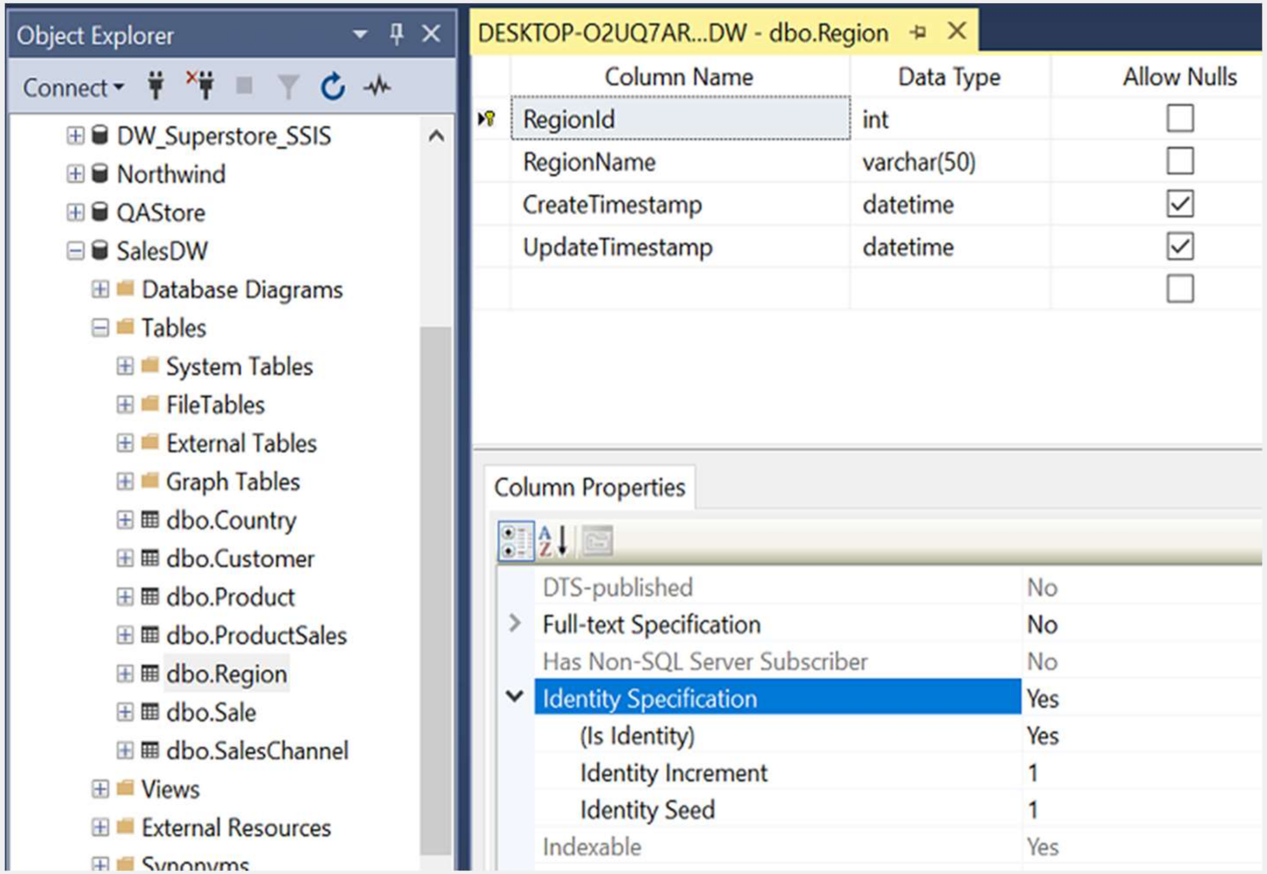
Data type	Description
bit	Integer that can be 0, 1, or NULL
tinyint	Allows whole numbers from 0 to 255
smallint	Allows whole numbers between -32,768 and 32,767
int	Allows whole numbers between -2,147,483,648 and 2,147,483,647
bigint	Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807
numeric(p,s)	Fixed precision and scale numbers. Allows numbers from -10^38 +1 to 10^38 –1. The p parameter indicates the maximum total number of digits that can be stored (both to the left and to the right of the decimal point). p must be a value from 1 to 38. Default is 18. The s parameter indicates the maximum number of digits stored to the right of the decimal point. s must be a value from 0 to p. Default value is 0

SQL Server Date Data Types

Data type	Description
datetime	From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds
datetime2	From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds
date	Store a date only. From January 1, 0001 to December 31, 9999
time	Store a time only to an accuracy of 100 nanoseconds

See more data types on
https://www.w3schools.com/sql/sql_datatypes.asp

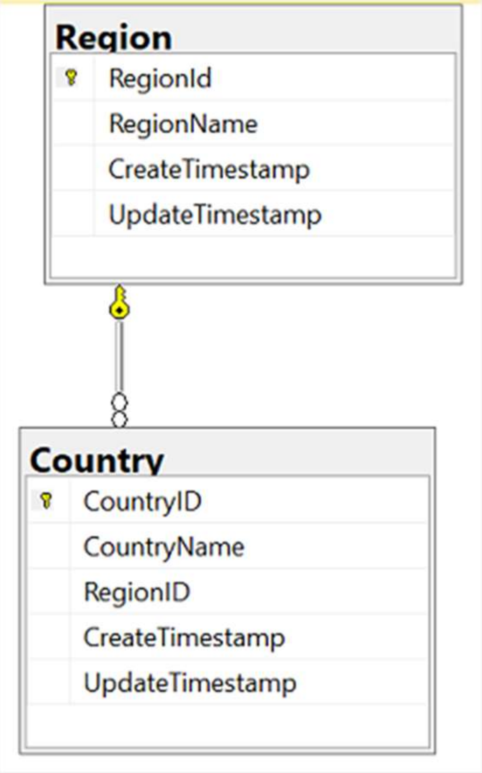
Primary Key Specification (SQL Server, SSMS)



Column Name	Data Type	Allow Nulls
RegionId	int	□
RegionName	varchar(50)	□
CreateTimestamp	datetime	☑
UpdateTimestamp	datetime	☑
		□

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
Indexable	Yes

foreign Key Specification (SQL Server, SSMS)

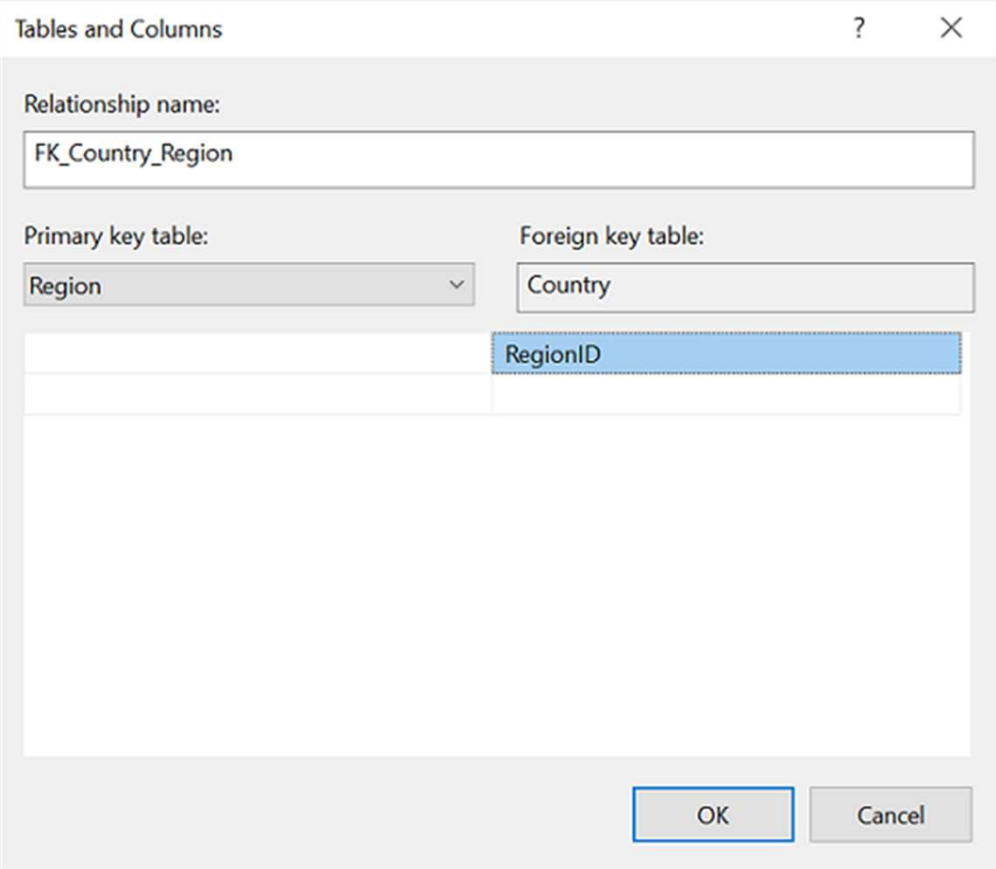


Region

- RegionId
- RegionName
- CreateTimestamp
- UpdateTimestamp

Country

- CountryID
- CountryName
- RegionID
- CreateTimestamp
- UpdateTimestamp



Tables and Columns

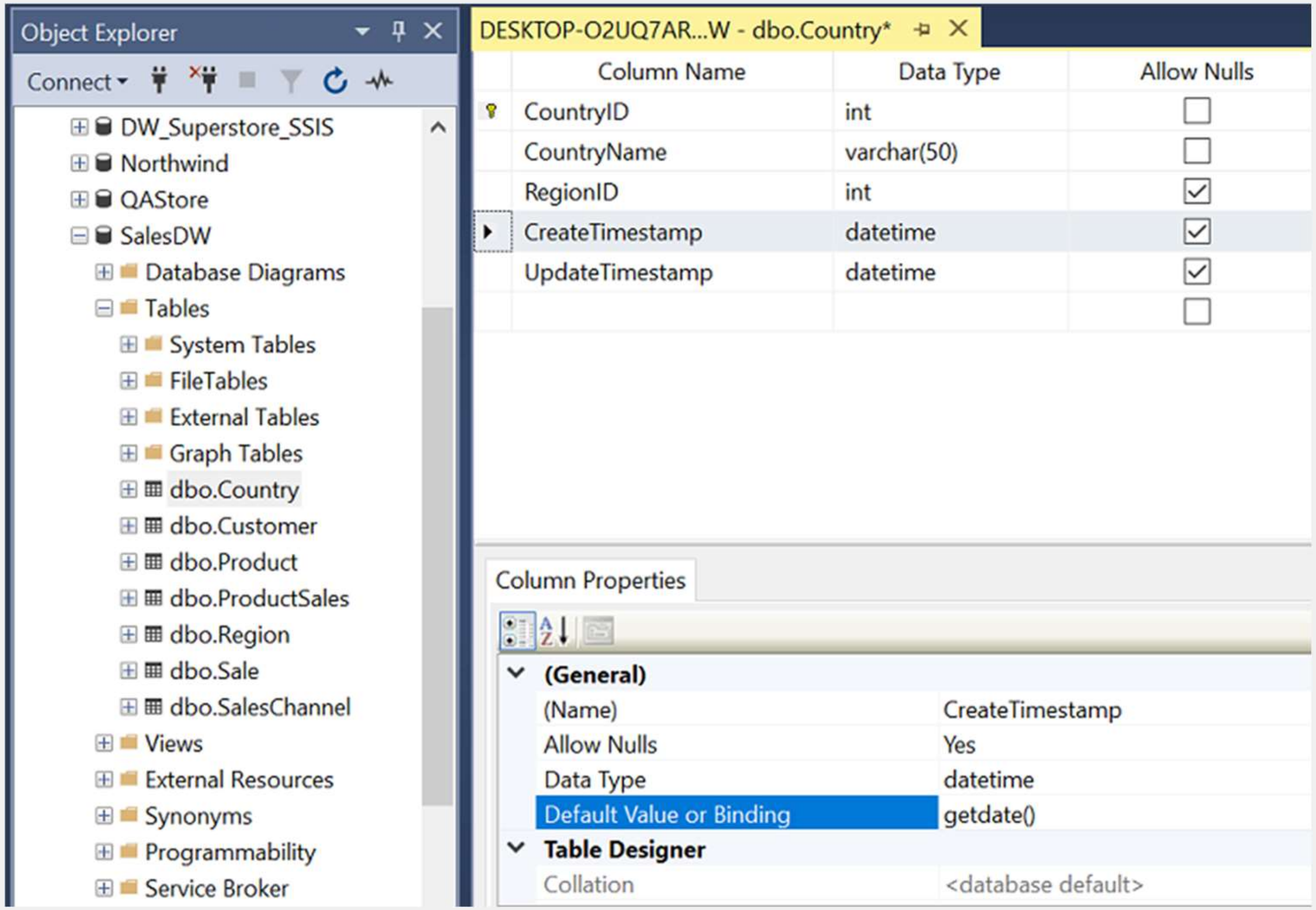
Relationship name: FK_Country_Region

Primary key table: Region

Foreign key table: Country

RegionID

Default Values specification (SQL Server, ssms)



Column Name	Data Type	Allow Nulls
CountryID	int	□
CountryName	varchar(50)	□
RegionID	int	☑
CreateTimestamp	datetime	☑
UpdateTimestamp	datetime	☑
		□

(General)	
(Name)	CreateTimestamp
Allow Nulls	Yes
Data Type	datetime
Default Value or Binding	getdate()
Table Designer	
Collation	<database default>