

SQL DATA TYPES & **OPERATIONS**



By the end of this lecture students should be able to:

- ✓ Understand about the different types of data we can collect
- ✓ Use these data types while creating your tables
- ✓ Choose a appropriate data type for a table column based on your requirement
- ✓ Use operators to specify conditions in an SQL statement



!= or <>

Table of contents

- ◇ **Ms SQL Server Data Types**
- ◇ **SQL Operators**

Section 1

MS SQL SERVER DATA TYPES

Student:

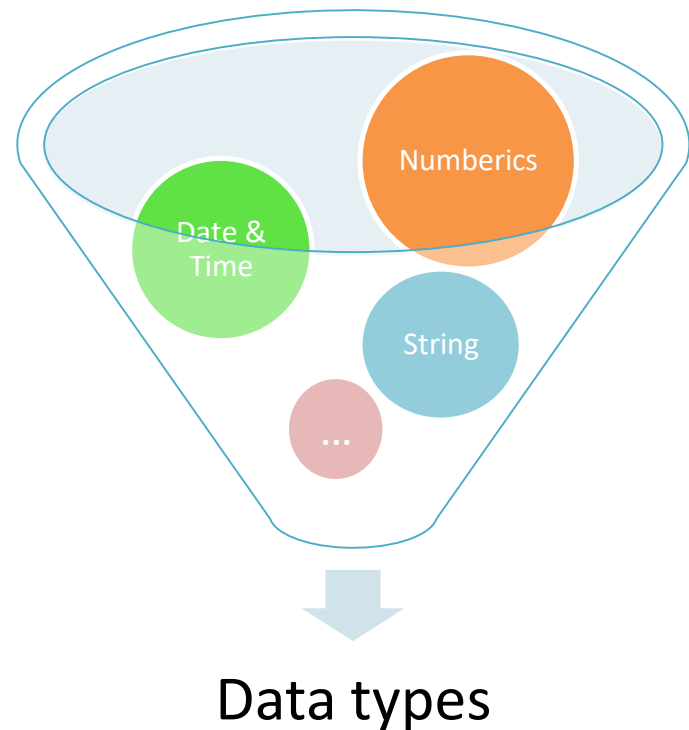
- ✓ Name
- ✓ Birthday
- ✓ Gender
- ✓ Address
- ✓ Marks...



What type of data each of field ???.....

➤ **SQL Server supports below data types. NULL is default value for most data type:**

- ✓ Exact Numerics
- ✓ Approximate Numerics
- ✓ Date and Time
- ✓ Character Strings
- ✓ Unicode Character Strings
- ✓ Binary Strings
- ✓ Other Data Types



➤ *Integer-based data type*

Data type	Size	Range of values
Bigint	8 Bytes	-2^{63} to $2^{63}-1$
Int	4 Bytes	-2^{31} to $2^{31}-1$
Smallint	2 Bytes	-2^{15} to $2^{15} - 1$
Tinyint	1 Byte	0 to 255
Bit	1 Bit	0 to 1



Exact decimal-based data type

Data type	Size	Range of values
Decimal(p,s)	5 - 17 Bytes (depending on precision)	- Varies based on precision setting. - Maximum values are $-10^{38} + 1$ through $10^{38} - 1$
<i>(p is the maximum number of all digits (both sides of the decimal point), s is the maximum number of digits after the decimal point)</i>		
Numeric(p,s)	...	Identical to Decimal type
Smallmoney	4 Bytes	- 214,748.3648 to 214,748.3647
Money	8 Bytes	- 922,337,203,685,477.5808 To 922,337,203,685,477.5807

Data type	Size	Range of values
Float	8 Bytes	- 1.79E+308 to 1.79E+308
<i>Depends on the value of n</i>		
Float(n)	If $1 \leq n \leq 24$: 4 Bytes (Precision: 7 digits)	4 Bytes: - 3.40E + 38 to 3.40E + 38
	If $25 \leq n \leq 53$: 8 Bytes (Precision: 15 digits)	8 Bytes: - 1.79E+308 to 1.79E+308
Real	...	- 3.40E + 38 to 3.40E + 38

Note: SQL Server treats n as one of two possible values. If $1 \leq n \leq 24$, n is treated as **24**. If $25 \leq n \leq 53$, n is treated as **53**.

Data Type	Description	Example
Date	Stores dates between January 1, 0001, and December 31, 9999	2008-01-15
Datetime	Stores dates and times between January 1, 1753, and December 31, 9999, with an accuracy of 3.33 milliseconds	2008-01-15 09:42:16.142
Datetime2	Stores date and times between January 1, 0001, and December 31, 9999, with an accuracy of 100 nanoseconds	2008-01-15 09:42:16.1420221
Datetimeoffset	Similar to the datetime2 data type, but also expects an offset designation of -14:00 to +14:00	2008-01-15 09:42:16.1420221 +05:00
Smalldatetime	Stores dates and times between January 1, 1900, and June 6, 2079, with an accuracy of 1 minute	2008-01-15 09:42:00
Time	Stores times with an accuracy of 100 nanoseconds	09:42:16.1420221

Non-Unicode string data types:

Data type	Description
Char(n)	<ul style="list-style-type: none">- Fixed-length- Maximum length of 8,000 characters ($1 \leq n \leq 8000$)
Varchar(n)	<ul style="list-style-type: none">- Variable-length- Maximum of 8,000 characters ($1 \leq n \leq 8000$)
Varchar(max)	<ul style="list-style-type: none">- Variable-length- Maximum length of 2,147,483,647 characters
Text	<ul style="list-style-type: none">- Variable-length- Maximum length of 2,147,483,647 characters- Use varchar(max) instead

Unicode string data types are “double width”:

Data type	Description
Nchar(n)	<ul style="list-style-type: none">- Fixed-length- Maximum specified length is 4,000 characters ($1 \leq n \leq 4000$)
Nvarchar(n)	<ul style="list-style-type: none">- Variable-length- Maximum specified length is 4,000 characters ($1 \leq n \leq 4000$)
Nvarchar(max)	<ul style="list-style-type: none">- Variable-length- Maximum length of 1,073,741,823 characters
Ntext	<ul style="list-style-type: none">- Variable-length- Maximum length of 1,073,741,823 characters

Data type	Description
Binary	<ul style="list-style-type: none">- Fixed-length binary data- Maximum length of 8,000 bytes
Varbinary	<ul style="list-style-type: none">- Variable length binary data- Maximum length of 8,000 bytes.
Image	<ul style="list-style-type: none">- Variable length binary data- Maximum length of 2,147,483,647 bytes.

Data Type	Description
Timestamp	Stores a database-wide unique number that gets updated every time a row gets updated
Hierarchyid	Special data type that maintains hierarchy positioning information
Uniqueidentifier	Stores a database-wide unique number that gets updated every time a row gets updated
Sql_variant	Stores values of various SQL Server-supported data types, except text, ntext, and timestamp
Xml	Stores XML data. You can store xml instances in a column or a variable (SQL Server 2005 only).
Table	Stores a result set for later processing

Section 2

SQL OPERATORS

What is an Operator in SQL?

- An **operator** is a reserved word or a character used primarily in an SQL statement's WHERE clause to perform operation(s), such as comparisons and arithmetic operations.
- Operators are used to specify conditions in an SQL statement and to serve as conjunctions for multiple conditions in a statement. Some types of most operators:
 - 1 Arithmetic operators
 - 2 Comparison operators
 - 3 Logical operators.

- Here is a list of the Arithmetic operators available in SQL

Operator	Description	Example
+	Addition	$a + b \rightarrow 30$
-	Subtraction	$a - b \rightarrow -10$
*	Multiplication	$a * b \rightarrow 200$
/	Division	$b / a \rightarrow 2$
%	Modulus	$b \% a \rightarrow 0$

(Assume variable **a** holds **10** and variable **b** holds **20**)

➤ Here is a list of all the Comparison operators available in SQL

Operator	Description	Operator	Description
=	equal to	>=	greater than or equal to
!=, <>	not equal to	<=	less than or equal to
<	less than	!<	not less than
>	greater than	!>	not greater than

❑ Example

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	MP	4500.00
7	Muffy	24	Indore	10000.00

CUSTOMERS TABLE

*SQL: SELECT * FROM CUSTOMERS WHERE SALARY > 5000;*



ID	NAME	AGE	ADDRESS	SALARY
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
7	Muffy	24	Indore	10000.00

Operator	Description
ALL	• Used to compare a value to all values in another value set.
AND	• Used when both conditions are included
ANY	• Used to compare a value to any applicable value in the list according to the condition
BETWEEN	• Used to limit the values in a range e.g.
EXISTS	• Used to search for the presence of a row in a specified table that meets certain criteria
IN	• Included in the list e.g.
LIKE	• Equal to some character (use quotes)
NOT	• Opposite of the logical value
OR	• Used when either of the condition is true
IS NULL	• This checks if the field has a null
UNIQUE	• Searches every row of a specified table for uniqueness

✓ Ms SQL Server Data Types

- What is Ms SQL Server Data Type?
- Some Ms SQL Server Data Types

✓ SQL Operators

- What is an Operator in SQL?
- Some category of Operators

✓ Demo

- Ms SQL Server Data Types
- Operators in SQL



Thank you

