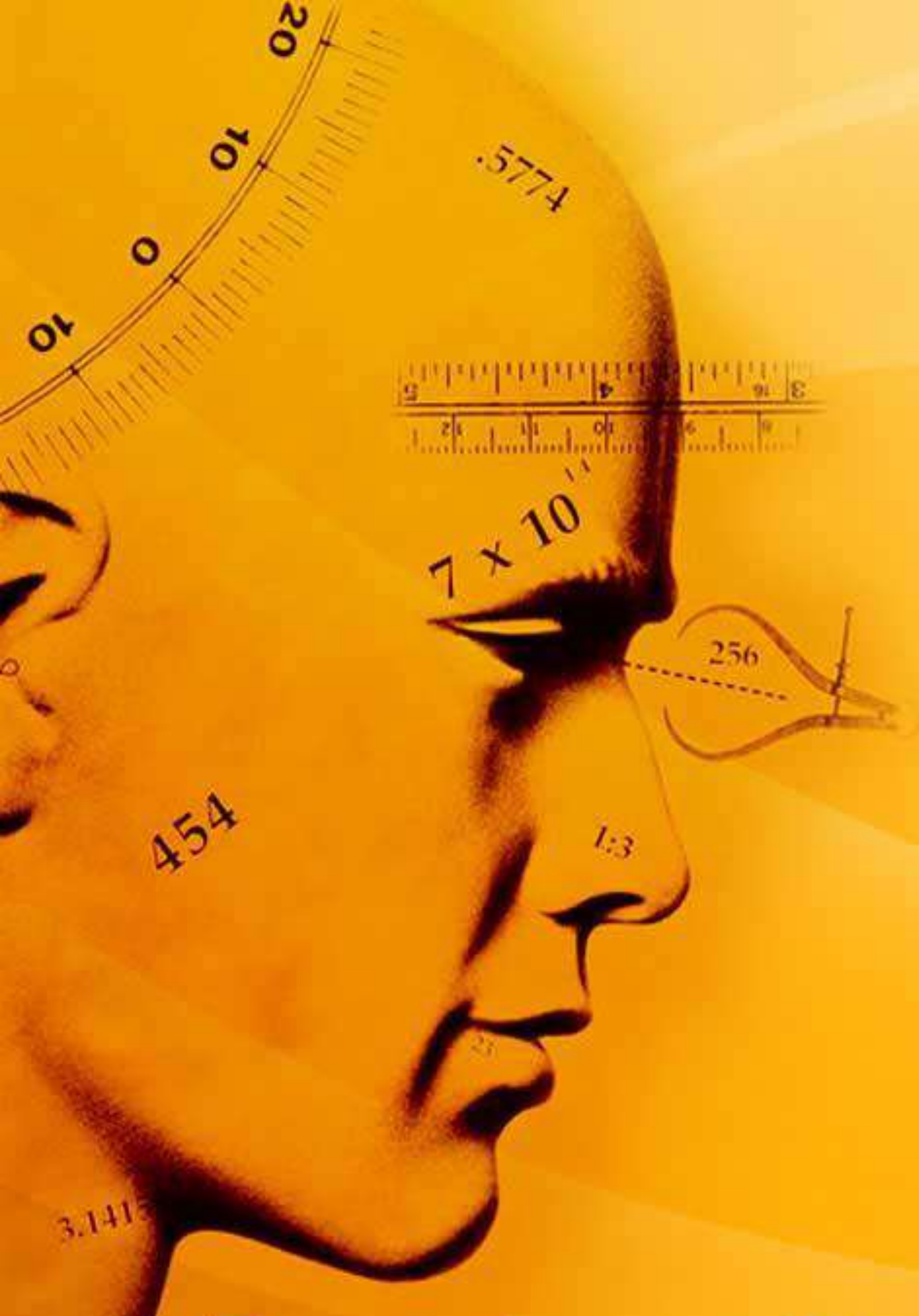

changepond

MY SQL- DATA TYPES

Experience Trust & Lasting Solutions

OBJECTIVES

After completing this session you will be able to

- Describe System data types
- Explain user defined data types

Retrieving Data

- The database user might need to retrieve complete or selected data from a table.
- Database can contain different type of data. Hence , before querying the data it is important to identify the various data types.

WHAT ISADATA TYPE?

- A data type specifies a particular type of data, such as integer, floating-point, Boolean etc…….

- A data type also specifies the possible values for that type, the operations that can be performed on that type and the way the values of that type are stored

MySQL DATA TYPES

- Supports a number of SQL standard data types in various categories.

- Has three main categories
  - Numeric types
  - DATE and TIME type
  - String

MYSQL NUMERIC TYPES

- MySQL supports all standard SQL numeric data types
- Include
  - INTEGER
  - SMALLINT
  - DECIMAL
  - NUMERIC

MYSQL NUMERIC TYPESCONT. ....

- Also supports approximate numeric data types
  - FLOAT
  - REAL
  - DOUBLE PRECISION

INTEGER TYPES

- SQL standard integer types INTEGER(or INT) and SMALLINT are supported by MySQL.

- Also supports the integer types;

- TINYINT
- MEDIUMINT
- BIGINT

REQUIRED STORAGE ANDRANGE

Type	Length in bytes	Minimum Value (Signed)	Max Val (Signed)	Min Val (Unsigned)	Max Val (Unsigned)
TINYINT	1	-128	127	0	255
SMALLINT	2	-32768	32767	0	65535
MEDIUMINT	3	-8388608	8388607	0	16777215
INT	4	-2147483648	2147483648	0	429497295
BIGINT	8	-9223372036854775808	9223372036854775808	0	18446744073709551615

FLOATING-POINT TYPES

- The FLOAT and DOUBLE types represent approximate numeric data values.

- MySQL allows non standard syntax
- FLOAT(M,D)
- REAL(M,D)
- Here values can be stored up to M digits in total where D represent decimal point

REQUIRED STORAGE ANDRANGE

Type	**Length in byte**	**Minimum value (Signed)**	**Maximum value(Singed)**	Minimum Value (Unsigned)	Maximum Value (Unsigned)
FLOAT	4	-3.402823466E+38	-1.175494351E-38	1.175494351E-38	3.402823466E+3
DOUBLE	8	-1.7976931348623157E+ 308	-2.2250738585072014E- 308	0, and 2.2250738585072014E- 308	1.7976931348623157E+ 308

FIXED-POINT TYPES

- Used to preserve exact precision
- For example currency data.
- DECIMAL and NUMERIC type store exact numeric data values

MORE IN FIXED POINTTYPE

- DECIMAL(5,2) be able to store any value with five digits and 2 decimals
- Value range will be -999.99 to 999.99
- DECIMAL(M) is equivalent to DECIMAL(M,0)
- DECIMAL is equivalent to DECIMAL(M,0)
- MySQL supports both of these forms

MYSQL DATE AND TIME TYPES

- Represent DATE, TIME, DATETIME, TIMESTAMP and YEAR
- Each type has range of valid values as well as a zero values.

DATETIME, DATE, AND TIMESTAMP TYPES

Types	Description	Display Format	Range
DATETIME	Use when you need values containing both date and time information	YYYY-MM-DD HH:MM:SS	'1000-01-01 00:00:00' to '9999-12-31 23:59:59'.
DATE	Use when you need only date information.	YYYY-MM-DD	'1000-01-01' to '9999-12-31'.
TIMESTAMP	Values are converted from the current time zone to UTC while storing, and converted back from UTC to the current time zone when retrieved	YYYY-MM-DD HH:MM:SS	'1970-01-01 00:00:01' UTC to '2038-01-19 03:14:07' UTC

TIME TYPE

- Fetch and display time value in ‘HH:MM:SS’ or ‘HHH:MM:SS’ format
- Range : from ‘-838:59:59’ to ‘839:59:59’.
- MySQL explain abbreviated TIME values with colons as the time of the day.
- Suppose ’09:10’ means ’09:10:00’ not ’00:09:10’
- Two right most digits represent seconds.

Time Type cont.

- Fro example we think ‘0910’ and 0910 as meaning 09:10:00, i.e. 10 minutes after 9 o’clock but reality is MySQL understand them as 00:09:10, i.e. 9 minutes and 10 second.!

YEAR TYPE

- Year type is 1-byte type represent year values.
- Can be declared as YEAR(2) or YEAR(4) to specify a display width 2 or 4 characters
- Default width is four characters.
- YEAR(4) and YEAR(2) have different display format, but have the same range of values.
- For 2-digt format, MySQL displays only the last two digits; For example, 70 (1970) or 2070

YEAR TYPECONT.

- We also can specify YEAR values in a variant formats;

String length	Range
4-digit string	‘1901’ to ‘2155’
4-digit number	1901 to 2155
1- or 2-digit string	‘0’ to ‘99'. Values in the ranges '0' to '69' and '70' to '99' are converted to YEAR values in the ranges 2000 to 2069 and 1970 to 1999
1- or 2-digit number	1 to 99. Values in the ranges 1 to 69 and 70 to 99 are converted to YEAR values in the ranges 2001 to 2069 and 1970 to 1999.

STRING TYPES

- The string types:
  - CHAR
  - VARCHAR
  - BINARY
  - VARBINARY
  - BLOB
  - TEXT

CHAR AND VARCHAR TYPES

- CHAR and VARCHAR types are similar, but differ in the way they are stored and retrieved.

- They also differ in maximum length.

CHAR ANDVARCHAR TYPES CONT.

Type	Description	Display format	Range in characters
CHAR	Contains non-binary strings. Length is fixed as you declare while creating a table. When stored, they are right-padded with spaces to the specified length	Trailing spaces are removed.	The length can be any value from 0 to 255.
VARCHAR	Contains non-binary strings. Columns are variable-length strings	As stored.	A value from 0 to 255 before MySQL 5.0.3, and 0 to 65,535 in 5.0.3 and later versions

BINARY AND VARBINARYTYPES

- Similar to CHAR and VARCHAR
- But they contain binary strings rather than non binary strings.

Type	Description	Range in bytes
BINARY	Contains binary strings	0 to 255
VARBINARY	Contain binary strings.	A value from 0 to 255 before MySQL 5.0.3, and 0 to 65,535 in 5.0.3 and later versions

BLOB AND TEXTTYPES

- A BLOB is a binary large object that can hold a variable amount of data.

- Four types of BLOBS (differ only the maximum length)
  - TINYBLOB
  - MEDIUMBLOB
  - LONGBLOB

BLOB AND TEXT TYPES CONT.

- Four TEXT types

- TINYTEXT
- TEXT
- MEDIUMTEXT
- LONGTEXT

➢Corresponding blob types and text type have the same maximum length

BLOB AND TEXT TYPES CONT.

Type	Description	Category	Range
BLOB	Large binary object that containing a variable amount of data. Values are treated as binary strings. You don't need to specify length while creating a column	TINYBLOB	Maximum length of 255 characters.
		MEDIUMBLOB	Maximum length of 16777215 characters.
		LONGBLOB	Maximum length of 4294967295 characters
TEXT	Values are treated as character strings having a character set	TINYBLOB	Maximum length of 255 characters.
		MEDIUMBLOB	Maximum length of 16777215 characters.

Q & A

- Allow time for questions from participants

Try it Out

Problem Statement: Create a table by name **PurchaseOrderDetail** to store purchase order details of **Adventure Works** and also choose the appropriate system data type for the column

Try it Out (Contd.)

Code:

```
Use AdventureWorks
CREATE TABLE [dbo].[PurchaseOrderDetail]
(
[PurchaseOrderID] [int] NOT NULL,
[LineNumber] [smallint] NOT NULL,
[ProductID] [int] NULL,
[UnitPrice] [money] NULL,
[OrderQty] [smallint] NULL,
[ReceivedQty] [float] NULL,
[RejectedQty] [float] NULL,
[DueDate] [datetime] NULL,
[rowguid] [uniqueidentifier],
[ModifiedDate] [datetime] NOT NULL,
[LineTotal] [money] NULL,
[StockedQty] [money] NULL,
[Remarks] varchar(max),
)
Select * from [dbo].[PurchaseOrderDetail]
```

Try it Out (Contd.)

- **How it works:** When the create table script is executed, it creates a table by name **PurchaseOrderDetail** using system defined data types and varchar(max)

Try it Out (Contd.)

Code:

```
USE AdventureWorks;

CREATE SCHEMA CustomerTraining

CREATE TABLE [CustomerTraining].[Customer](
   [CustomerID] [int] NOT NULL,
   [CustomerName] [nvarchar](25) NOT NULL,
   [TerritoryID] [int] NULL,
   [AccountNumber] [nvarchar](20),
   [CustomerType] [nvarchar](10) NOT NULL,
   [CustomerAddress][nvarchar](30) NOT NULL,
   [CustomerPhone][nvarchar](20) NULL,
   [CustomerStateCode][nchar](2) NULL,
)
GO
```

Try it Out (Contd.)

```
Insert into CustomerTraining.Customer
values(101,'Raj Kumar',201,'CAN00001','Internal','R
   Nagar Chennai','91-81-2322322','TN')

Insert into CustomerTraining.Customer
values(102,'Narayan',202,'CAN00002','External','J Nagar
   Chennai','91-81-2342346','TN')

Insert into CustomerTraining.Customer
values(103,'Paul Anthony',203,'CAN00034','Internal','B
   Nagar Bangalore','91-81-2322533','KA')

Insert into CustomerTraining.Customer
values(101,'Murthy',241,'CAN00301','Internal','N Nagar
   Chennai','91-81-232342','KA')
```

Try it Out (Contd.)

- How it Works:
- When the SQL scripts in AdventureWorks database is executed:
  - » It will create a `CustomerTraining` schema
  - » It will create a `Customer` table under schema and insert four customers information modifies two customer `CustomerStateCode` to `‘TN’`
  - » Finally, it will drop `Customer` table and `CustomerTraining` Schema from database

Thank You