SHANIMZ ACADEMY

SQL - Syllabus

Sql introduction,Sql Datatypes Sql basics-DDL,DML,DCL,Comparison Operators,Constraints,Aggregate Functions,Practice questions,Joins,Stored Procedures,Practice Question,Views,Group By,Having ,Order By Clauses, Practice Question.

Day 1 -SQL

LIVE CLASS : https://youtu.be/_nvngE0Mb8g

Database is an organised collection of data,stored & accessed electronically.

Server is a Collection of N number of databases.

What is SQL?

SQL stands for Structured Query Language. It is a language used to manage data in a relational database. It allows you to query, manipulate, and define the data in a relational database. It has become a standard language, which means that the language the database uses will be based on SQL despite which relational database you use. This makes it a very versatile language and an important skill to have if you work with databases or data. The language closely resembles English, making it easy for English-speakers to learn and read at a glance, even for a beginner. Statements consist of clauses, identifiers, operators, and constants.

Why SQL?

- The basic use of SQL for data professionals and SQL users is to insert, update, and delete the data from the relational database.
- SQL allows the data professionals and users to retrieve the data from the relational database management systems.
- It also helps them to describe the structured data.
- It allows SQL users to create, drop, and manipulate the database and its tables.
- It also helps in creating the view, stored procedure, and functions in the relational

database.

- It allows you to define the data and modify that stored data in the relational database.

SQL COMMANDS

- CREATE command
- UPDATE command
- DELETE command
- SELECT command
- DROP command
- INSERT command

SQL DATATYPES

Each column in a database table is required to have a name and a data type. An SQL developer must decide what type of data that will be stored inside each column when creating a table. The data type is a guideline for SQL to understand what type of data is expected inside of each column, and it also identifies how SQL will interact with the stored data

MYSQL DATATYPES

STRING DATA TYPES

- CHAR(size): A FIXED length string (can contain letters, numbers, and special characters). The size parameter specifies the column length in characters - can be from 0 to 255. Default is 1
- VARCHAR(size): A VARIABLE length string (can contain letters, numbers, and special characters). The size parameter specifies the maximum column length in characters - can be from 0 to 65535
- BINARY(size): Equal to CHAR(), but stores binary byte strings. The size parameter specifies the column length in bytes. Default is 1

- VARBINARY(size): Equal to VARCHAR(), but stores binary byte strings. The size parameter specifies the maximum column length in bytes.
- TINYBLOB: For BLOBs (Binary Large Objects). Max length: 255 bytes
- TINYTEXT: Holds a string with a maximum length of 255 characters
- TEXT(size): Holds a string with a maximum length of 65,535 bytes
- BLOB(size): For BLOBs (Binary Large Objects). Holds up to 65,535 bytes of data
- MEDIUMTEXT: Holds a string with a maximum length of 16,777,215 characters
- MEDIUMBLOB: For BLOBs (Binary Large Objects). Holds up to 16,777,215 bytes of data
- LONGTEXT: Holds a string with a maximum length of 4,294,967,295 characters
- LONGBLOB: For BLOBs (Binary Large Objects). Holds up to 4,294,967,295 bytes of data
- ENUM(val1, val2, val3, ...): A string object that can have only one value, chosen from a list of possible values. You can list up to 65535 values in an ENUM list. If a value is inserted that is not in the list, a blank value will be inserted. The values are sorted in the order you enter them
- SET(val1, val2, val3, ...): A string object that can have 0 or more values, chosen from a list of possible values. You can list up to 64 values in a SET list

NUMERIC DATA TYPE

- BIT(size) A bit-value type. The number of bits per value is specified in size. The size parameter can hold a value from 1 to 64. The default value for size is 1.
- TINYINT(size) A very small integer. Signed range is from -128 to 127. Unsigned range is from 0 to 255. The size parameter specifies the maximum display width (which is 255)
- BOOL Zero is considered as false, nonzero values are considered as true.
- BOOLEAN Equal to BOOL
- SMALLINT(size) A small integer. Signed range is from -32768 to 32767. Unsigned range is from 0 to 65535. The size parameter specifies the maximum display width (which is 255)

- MEDIUMINT(size) A medium integer. Signed range is from -8388608 to 8388607. Unsigned range is from 0 to 16777215. The size parameter specifies the maximum display width (which is 255)

- INT(size) A medium integer. Signed range is from -2147483648 to 2147483647. Unsigned range is from 0 to 4294967295. The size parameter specifies the maximum display width (which is 255)

- INTEGER(size) Equal to INT(size)

- BIGINT(size) A large integer. Signed range is from -9223372036854775808 to 9223372036854775807. Unsigned range is from 0 to 18446744073709551615. The size parameter specifies the maximum display width (which is 255)

- FLOAT(size, d) A floating point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter. This syntax is deprecated in MySQL 8.0.17, and it will be removed in future MySQL versions

- FLOAT(p) A floating point number. MySQL uses the p value to determine whether to use FLOAT or DOUBLE for the resulting data type. If p is from 0 to 24, the data type becomes FLOAT(). If p is from 25 to 53, the data type becomes DOUBLE()

- DOUBLE(size, d)A normal-size floating point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter

- DOUBLE PRECISION(size, d)

- DECIMAL(size, d) An exact fixed-point number. The total number of digits is specified in size. The number of digits after the decimal point is specified in the d parameter. The maximum number for size is 65. The maximum number for d is 30. The default value for size is 10. The default value for d is 0.

- DEC(size, d) Equal to DECIMAL(size,d)

Date and Time Data Types

- DATE A date. Format: YYYY-MM-DD. The supported range is from '1000-01-01' to '9999-12-31'

- DATETIME(fsp) A date and time combination. Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. Adding DEFAULT and ON UPDATE in the column definition to get automatic initialization and updating to the current date and time

- TIMESTAMP(fsp) A timestamp. TIMESTAMP values are stored as the number of seconds since the Unix epoch ('1970-01-01 00:00:00' UTC). Format: YYYY-MM-DD hh:mm:ss. The supported range is from '1970-01-01 00:00:01' UTC to '2038-01-09 03:14:07' UTC. Automatic initialization and updating to the current date and time can be specified using DEFAULT CURRENT_TIMESTAMP and ON UPDATE CURRENT_TIMESTAMP in the column definition

- TIME(fsp) A time. Format: hh:mm:ss. The supported range is from '-838:59:59' to '838:59:59'

- YEAR A year in four-digit format. Values allowed in four-digit format: 1901 to 2155, and 0000.

- MySQL 8.0 does not support year in two-digit format.

SQL Server Data Types

String Data Types

- char(n) Fixed width character string 8,000 characters Defined width

- varchar(n) Variable width character string 8,000 characters 2 bytes + number of chars

- varchar(max) Variable width character string 1,073,741,824 characters 2 bytes + number of chars

- text Variable width character string 2GB of text data 4 bytes + number of chars

- nchar Fixed width Unicode string 4,000 characters Defined width x 2

- nvarchar Variable width Unicode string 4,000 characters

- nvarchar(max) Variable width Unicode string 536,870,912 characters

- ntext Variable width Unicode string 2GB of text data

- binary(n) Fixed width binary string 8,000 bytes

- varbinary Variable width binary string 8,000 bytes

- varbinary(max) Variable width binary string 2GB
- image Variable width binary string 2GB

Numeric Data Types

- bit Integer that can be 0, 1, or NULL
- tinyint Allows whole numbers from 0 to 255 1 byte
- smallintAllows whole numbers between -32,768 and 32,767 2 bytes
- int Allows whole numbers between -2,147,483,648 and 2,147,483,647 4 bytes
- bigint Allows whole numbers between -9,223,372,036,854,775,808 and 9,223,372,036,854,775,807 8 bytes

Date and Time Data Types

- datetime From January 1, 1753 to December 31, 9999 with an accuracy of 3.33 milliseconds 8 bytes
- datetime2 From January 1, 0001 to December 31, 9999 with an accuracy of 100 nanoseconds 6-8 bytes
- smalldatetime From January 1, 1900 to June 6, 2079 with an accuracy of 1 minute 4 bytes
- date Store a date only. From January 1, 0001 to December 31, 9999 3 bytes
- time Store a time only to an accuracy of 100 nanoseconds 3-5 bytes
- datetimeoffset The same as datetime2 with the addition of a time zone offset 8-10 bytes
- timestamp Stores a unique number that gets updated every time a row gets created or modified. The timestamp value is based upon an internal clock and does not correspond to real time. Each table may have only one timestamp variable

LAB SETUP

DOWNLOAD Microsoft SQL SERVER Management Studio (SSMS) & Install It

https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16

DOWNLOAD Microsoft SQL Server Express 2019

https://www.microsoft.com/en-in/download/details.aspx?id=101064