Marwadi University

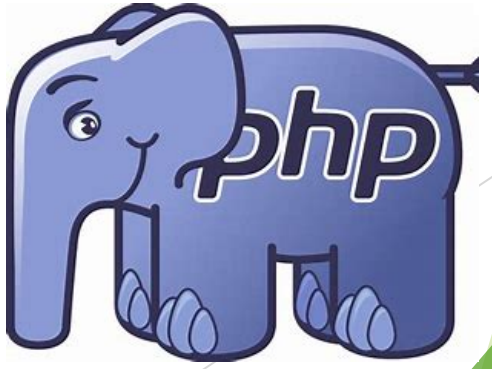
Web Application Development (PHP)

Unit – 4

Interacting with MySQL

Prof. Hardik Chavda

FOCA,MU



- Learning the MySQL Data Types,
- Frequently used String functions in MySQL,
- Using Date and Time functions in MySQL,
- Interacting with MySQL using PHP,
- MySQL Versus MySQLi Functions,
- Connecting to MySQL with PHP,
- Working with MySQL Data,
- PHPMyAdmin interface.

Introduction to MySQL

- MySQL is a relational database management system (RDBMS).
- RDBMS is software which is used to store and retrieve data, and manage permissions as to who or what services can access that data.
- Structured Query Language (SQL) is a powerful and widely used language to create and manage relational databases. MySQL uses SQL as its language to interact with the database.
- One of the key aspects of MySQL is its support for various data types that enable efficient storage and retrieval of different kinds of data.
- MySQL data types include numeric data types such as **integers** and **decimals**, **character** and **string** data types, **date** and **time** data types, **enum** and **set** data types, and **binary** data types.
- Understanding the different data types and how they are used is essential for you to have effective database design and query optimization.

Learning the MySQL Data Types

Data type	Description
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

Integer Datatypes

Data type	Description
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

Data type	Description
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.

MySQL String Functions

Function	Description
CHAR_LENGTH	Returns the length of a string (in characters)
CONCAT	Adds two or more expressions together
FORMAT	Formats a number to a format like "#,###,###.##", rounded to a specified number of decimal places
INSERT	Inserts a string within a string at the specified position and for a certain number of characters
LENGTH	Returns the length of a string (in bytes)
LOWER	Converts a string to lower-case
LTRIM	Removes leading spaces from a string
REPEAT	Repeats a string as many times as specified

REPLACE	Replaces all occurrences of a substring within a string, with a new substring
REVERSE	Reverses a string and returns the result
RTRIM	Removes trailing spaces from a string
SPACE	Returns a string of the specified number of space characters
STRCMP	Compares two strings
SUBSTR	Extracts a substring from a string (starting at any position)
SUBSTRING	Extracts a substring from a string (starting at any position)
TRIM	Removes leading and trailing spaces from a string
UPPER	Converts a string to upper-case

MySQL Date Functions

Function	Description
CURRENT_DATE	Returns the current date
CURRENT_TIME	Returns the current time
DATE	Extracts the date part from a datetime expression
DATEDIFF	Returns the number of days between two date values
DATE_FORMAT	Formats a date
DAY	Returns the day of the month for a given date
DAYNAME	Returns the weekday name for a given date
DAYOFMONTH	Returns the day of the month for a given date
DAYOFWEEK	Returns the weekday index for a given date
DAYOFYEAR	Returns the day of the year for a given date
HOUR	Returns the hour part for a given date

MICROSECOND	Returns the microsecond part of a time/datetime
MINUTE	Returns the minute part of a time/datetime
MONTH	Returns the month part for a given date
MONTHNAME	Returns the name of the month for a given date
NOW	Returns the current date and time
SECOND	Returns the seconds part of a time/datetime
SYSDATE	Returns the current date and time
TIME	Extracts the time part from a given time/datetime
TIMEDIFF	Returns the difference between two time/datetime expressions
TIMESTAMP	Returns a datetime value based on a date or datetime value
WEEK	Returns the week number for a given date
WEEKDAY	Returns the weekday number for a given date
WEEKOFYEAR	Returns the week number for a given date
YEAR	Returns the year part for a given date
YEARWEEK	Returns the year and week number for a given date

Interacting with MySQL using PHP & Connecting MySQL using PHP

There are three ways of working with MySQl and PHP

- MySQLi (object-oriented)
- MySQLi (procedural)
- PDO

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";

// Creating connection
$conn = new mysqli($servername, $username, $password);

// Checking connection
if ($conn->connect_error)  or   (!$conn)          {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

MySQL vs MySQLi

- MySQL and MySQLi are PHP database extensions implemented by using the PHP extension framework. PHP database extensions are used to write PHP code for accessing the database. They expose database API to provide interfaces to use database functions.
- MySQL extension is deprecated and will not be available in future PHP versions. It is recommended to use the MySQLi extension with PHP 5.5 and above.
- There are too many differences between these PHP database extensions. These differences are based on some factors like performance, library functions, features, benefits, and others.
- Note: Though MySQL extension is deprecated, for backward compatibility it will be available. But do not use if you are starting something new and recommend migrating the older from MySQL to MySQLi extension.

MySQL	MySQLi
MySQL extension added in PHP version 2.0. and deprecated as of PHP 5.5.0.	MySQLi extension added in PHP 5.5 and will work on MySQL 4.1.3 or above.
Does not support prepared statements.	MySQLi supports prepared statements.
MySQL provides the procedural interface.	MySQLi provides both procedural and object-oriented interfaces.
MySQL extension does not support stored procedures.	MySQLi supports store procedures.
MySQL extension lags in security and other special features, comparatively.	MySQLi extension with enhanced security and improved debugging.
Transactions are handled by SQL queries only.	MySQLi supports transactions through API.
Extension directory: ext/mysql.	Extension directory: ext/mysqli.

Phpmyadmin interface

- phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web.
- phpMyAdmin supports a wide range of operations on MySQL and MariaDB.
- Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

Features of PHPmyadmin

- Support for most MySQL features:
  - browse and drop databases, tables, views, fields and indexes
  - create, copy, drop, rename and alter databases, tables, fields and indexes
  - maintenance server, databases and tables, with proposals on server configuration
  - execute, edit and bookmark any SQL-statement, even batch-queries
  - manage MySQL user accounts and privileges
  - manage stored procedures and triggers
- Import data from CSV and SQL
- Export data to various formats: CSV, SQL, XML, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet, Word, LATEX and others
- Administering multiple servers
- Creating graphics of your database layout in various formats
- Creating complex queries using **Query-by-example** (QBE)
- Searching globally in a database or a subset of it
- Transforming stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link