CACS 205: Script Language

PREPARED BY KRISHNA PD. ACHARYA

(MECHI MULTIPLE CAMPUS)

MySQL 7 Hrs.

- Introduction to MySQL,
- PHP MySQL Connect to a Database
- Closing a Connection.
    - MySql Data Types
    - MySql Insert
    - MySql Update
    - MySql Select
    - MySql Delete
    - MySql Where Clause
    - MySql Aggregate
    - Functions(sum,avg,count etc.)
    - MySql Order by Group by
    - MySql Subquries
    - MySql Joins

MySQL 7 Hrs.

Introduction to MySQL:

- MySQL is a database management system.
- MySQL databases are relational.
- MySQL software is Open Source.
- The MySQL Database Server is fast, reliable, scalable, and easy to use.
- MySQL Server works in client/server or embedded systems.
- Initial release : 23 May 1995
- Current stable release : 5.6.13 / 30 July 2013
- Written in : C, C++
- Operating system : Cross-platform
- Available in : English
- The license of MySQL is available under GNU General Public License.

MySQL 7 Hrs.

Introduction to MySQL

- MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses.
- MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company.
- MySQL is becoming so popular because of many good reasons −
- MySQL is released under an open-source license. So you have nothing to pay to use it.
- MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc.
- MySQL works very quickly and works well even with large data sets.
- MySQL is very friendly to PHP, the most appreciated language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).
- MySQL is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

MySQL 7 Hrs.

```
mysql -p -u root
CREATE USER 'user'@'localhost' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON database.table TO 'user'@'localhost‘;
GRANT ALL PRIVILEGES ON database.* TO 'user'@'localhost';
GRANT ALL PRIVILEGES ON *.* TO 'user'@'localhost';
GRANT SELECT, INSERT, DELETE ON database.* TO 'user'@'localhost';
REVOKE ALL PRIVILEGES, GRANT OPTION FROM user [, user]
REVOKE INSERT ON *.* FROM ‘krishna'@'localhost';
REVOKE SELECT, INSERT, DELETE, UPDATE ON Users TO 'Amit'@'localhost;
DROP USER 'user'@'localhost‘
show databases;
use DBNAME;
SHOW WARNINGS;
```

Database Connectivity

Syntax: CREATE DATABASE <dbname>;
CREATE DATABASE mmc;

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";/* Put your password here */
/* Create connection */
$conn = new mysqli($servername, $username, $password);
/* Create database */
$sql = "CREATE DATABASE MMC";
if ($conn->query($sql) === TRUE) {
    echo "Database admin created successfully";

}
else
{
    echo "Error creating database: " . $conn->error;
}
$conn->close();
?>
```

Database Connectivity

Student				
ID	FirstName	LastName	Rollno	City
1	Ram	Rai	101	KTM

```
CREATE TABLE table_name (
    column1_name data_type constraints,
    column2_name data_type constraints,
    ....
);
```

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";/* Put your password */
$dbname = "mmc";/* Put your database name */
/* Create connection */
$conn = new mysqli($servername, $username, $password, $dbname);
/* Check connection*/
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
/* sql to create table */
$sql = "CREATE TABLE Student
(
ID int NOT NULL AUTO_INCREMENT,
FirstName varchar(50),
LastName varchar(50),
RollNo varchar(50),
City varchar(50),
PRIMARY KEY (ID)
)";

if ($conn->query($sql) === TRUE) {
    echo "Table test created successfully";
}
 else {
    echo "Error creating table: " . $conn->error;
}
$conn->close();
?>
```

Database Connectivity

Student				
ID	FirstName	LastName	Rollno	City
1	Zimpa	Sherpa	1011	Ktm

INSERT INTO table_name (column1,column2,...)
VALUES (value1,value2,...);

```php
<?php
$mysqli = new mysqli("localhost", "root", "", "mmc");
if($mysqli === false){
    die("ERROR: Could not connect. " . $mysqli->connect_error);
}
$sql = "INSERT INTO student (FirstName, LastName, RollNo, City)
                    VALUES ('zimpa', 'Sherpa', '1011', 'Ktm')";

if($mysqli->query($sql) === true){
    echo "Records inserted successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . $mysqli->error;
}
$mysqli->close();
?>
```

```
UPDATE table_name SET column1=value,
column2=value2,... WHERE column_name=some_value
```

Student				
ID	FirstName	LastName	Rollno	City
1	Zimpa	Sherpa	1011	Jhapa

```php
<?php
$mysqli = new mysqli("localhost", "root", "", "mmc");
if($mysqli === false){
    die("ERROR: Could not connect. " . $mysqli->connect_error);
}
$sql = "UPDATE student SET City='Jhapa' WHERE id=9";
if($mysqli->query($sql) === true){
    echo "Records were updated successfully.";
} else{
    echo "ERROR: Could not able to execute $sql. " . $mysqli->error;
}
$mysqli->close();
?>
```

Database Connectivity

Student				
ID	FirstName	LastName	Rollno	City
1	Zimpa	Sherpa	1011	Jhapa

DELETE FROM table_name WHERE column_name=some_value

```php
<?php
$mysqli = new mysqli("localhost", "root", "", "mmc");
if($mysqli === false){
  die("ERROR: Could not connect. " . $mysqli->connect_error);
}
$sql = "DELETE FROM student WHERE id=10003";
if($mysqli->query($sql) === true){
    echo "Records were deleted successfully.".$mysqli->affected_rows;
} else{
    echo "ERROR: Could not able to execute $sql. " . $mysqli->error;
  }
$mysqli->close();
?>
```

Select Query

```php
<?php
$mysqli = new mysqli("localhost", "root", "", "mmc");
if($mysqli === false){
    die("ERROR: Could not connect. " . $mysqli->connect_error);
}
$sql = "SELECT * FROM student";
if($result = $mysqli->query($sql)){
    if($result->num_rows > 0){
        echo "<table border='1'>";
            echo "<tr>";
                echo "<th>ID</th>";
                echo "<th>First Name</th>";
                echo "<th>Last Name</th>";
                echo "<th>RollNo</th>";
                echo "<th>City</th>";
            echo "</tr>";
        while($row = $result->fetch_assoc()){
                echo "<tr>";
                echo "<td>" . $row['ID'] . "</td>";
                echo "<td>" . $row['FirstName'] . "</td>";
                echo "<td>" . $row['LastName'] . "</td>";
                echo "<td>" . $row['RollNo'] . "</td>";
                echo "<td>" . $row['City'] . "</td>";
            echo "</tr>";
        }
        echo "</table>";
        $result->free();
    } else{
        echo "No records matching your query were found.";
    }
} else{
    echo "ERROR: Could not able to execute $sql. " . $mysqli->error;
}
$mysqli->close();
?>
```

SELECT column1_name, column2_name, columnN_name FROM table_name;

SELECT column_name(s) FROM table_name WHERE column_name= operator value

SELECT column_name(s) FROM table_name LIMIT row_offset, row_count;

SELECT * FROM persons LIMIT 3;

SELECT * FROM persons LIMIT 1, 3;

SELECT column_name(s) FROM table_name ORDER BY column_name(s) ASC | DESC

SELECT * FROM persons ORDER BY first_name DESC

Data types in MySQL

Binary data type	Nonbinary data type	Maximum length
BINARY	CHAR	255
VARBINARY	VARCHAR	65,535
TINYBLOB	TINYTEXT	255
BLOB	TEXT	65,535
MEDIUMBLOB	MEDIUMTEXT	16,777,215
LONGBLOB	LONGTEXT	4,294,967,295

Data Type	Range (Signed)	Range (Unsigned)
TINYINT	-128 to 127	0 to 255
SMALLINT	-32768 to 32767	0 to 65535
MEDIUMINT	-8388608 to 8388607	0 to 16777215
INT	-2147483648 to 2147483647	0 to 4294967295
BIGINT	-9223372036854775808 to 9223372036854775807	0 to 18446744073709551615

Types	Description
FLOAT	A precision from 0 to 23 results in a four-byte single-precision FLOAT column
DOUBLE	A precision from 24 to 53 results in an eight-byte double-precision DOUBLE column.

MySQL allows a nonstandard syntax: FLOAT(M,D) or REAL(M,D) or DOUBLE PRECISION(M,D).

Type	Length in Bytes	Minimum Value (Signed)	Maximum Value (Signed)	Minimum Value (Unsigned)	Maximum Valu (Unsigned)
FLOAT	4	-3.402823466E+38	-1.175494351E-38	1.175494351E-38	3.402823466E+
DOUBLE	8	-1.7976931348623157E+ 308	-2.22507385850720 14E- 308	0, and 2.22507385850720 14E- 308	1.79769313486 7E+ 308

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