

# Database applications for the web with PHP and MySQL

## SQL Commands

**Select** (Retrieve rows from a table)

```
SELECT [ALL|DISTINCT [ON column] ]
       expression [ AS
       name ] [, ...]
       [ INTO [TEMP] [TABLE] new_table ]
       [ FROM table
       [alias] [, ...] ]
       [ WHERE condition ]
       [ GROUP BY column [, ...] ]
       [ HAVING condition [, ...] ]
       [ { UNION [ALL] | INTERSECT | EXCEPT } select ]
       [ ORDER BY column [ ASC | DESC ] [, ...] ]
       [ FOR UPDATE [OF class_name...]]
       [ LIMIT count [OFFSET|, count]]
```

SELECT will return rows from one or more tables. Candidates for selection are rows which satisfy the WHERE condition; if WHERE is omitted, all rows are candidates.

DISTINCT will eliminate all duplicate rows from the selection. DISTINCT ON *column* will eliminate all duplicates in the specified column; this is equivalent to using GROUP BY *column*. ALL will return all candidate rows, including duplicates.

The GROUP BY clause allows a user to divide a table conceptually into groups.

The HAVING clause specifies a grouped table derived by the elimination of groups from the result of the previously specified GROUP BY clause.

The ORDER BY clause allows a user to specify that he/she wishes the rows sorted according to the ASCending or DESCending mode operator.

The UNION clause allows the result to be the collection of rows that is created by doing a *set union* of the rows returned by this SELECT statement and another SELECT statement. Similarly, INTERSECT does a set intersection, and EXCEPT does a set minus.

The LIMIT...OFFSET clause allows control over which rows are returned by the query.

You must have SELECT privilege to a table to read its values.

```
SELECT * FROM tbl_movies WHERE title LIKE '%007%';
```

## **UPDATE** (Replaces values of columns in a table)

```
UPDATE table SET column = expression [, ...]  
  [ FROM fromlist ]  
  [ WHERE condition ]
```

UPDATE changes the values of the columns specified for all rows which satisfy condition. Only the columns to be modified need appear as column.

Array references use the same syntax found in SELECT. That is, either single array elements, a range of array elements or the entire array may be replaced with a single query.

You must have write access to the table in order to modify it, as well as read access to any table whose values are mentioned in the WHERE condition.

```
UPDATE tbl_movies SET genre = 'Dramatic' WHERE genre = 'Drama';
```

## **DELETE** (Deletes rows from a table)

```
DELETE FROM table [ WHERE condition ]
```

DELETE removes rows which satisfy the WHERE condition from the specified table.

If the condition is absent, the effect is to delete all rows in the table. The result is a valid, but empty table.

You must have write access to the table in order to modify it, as well as read access to any table whose values are read in the condition.

```
DELETE FROM tbl_movies WHERE genre = 'Western';
```

## **INSERT** (Inserts new rows into a table)

```
INSERT INTO table [ ( column [, ...] ) ]  
  { VALUES ( expression [, ...] ) | SELECT query }
```

INSERT allows one to insert new rows into a table. One can insert a single row at a time or several rows as a result of a query. The columns in the target list may be listed in any order. In every column *not* present in the target list, its *default value* will be inserted. If a column has no declared default value it will be assumed as NULL. If the *expression* for each column is not of the correct data type, automatic type coercion will be attempted.

```
INSERT INTO tbl_movies(title,year) VALUES('Alexander','2004')
```

## MySQL data types

MySQL supports a number of column types, which may be grouped into 3 categories:

- numeric types,
- date and time types, and
- string (character) types.

This section first gives an overview of the types available.

Type	Use for	Size
TINYINT	A very small integer	The signed range is -128 to 127. The unsigned range is 0 to 255.
SMALLINT	A small integer	The signed range is -32768 to 32767. The unsigned range is 0 to 65535
MEDIUMINT	A medium-size integer	The signed range is -8388608 to 8388607. The unsigned range is 0 to 16777215
INT or INTEGER	A normal-size integer	The signed range is -2147483648 to 2147483647. The unsigned range is 0 to 4294967295
BIGINT	A large integer	The signed range is -9223372036854775808 to 9223372036854775807. The unsigned range is 0 to 18446744073709551615
FLOAT	A small (single-precision) floating-point number. Cannot be unsigned	Ranges are -3.402823466E+38 to -1.175494351E-38, 0 and 1.175494351E-38 to 3.402823466E+38. If the number of Decimals is not set or <= 24 it is a single-precision floating point number
DOUBLE, DOUBLE PRECISION, REAL	A normal-size (double-precision) floating-point number. Cannot be unsigned	Ranges are -1.7976931348623157E+308 to -2.2250738585072014E-308, 0 and 2.2250738585072014E-308 to 1.7976931348623157E+308. If the number of Decimals is not set or 25 <= Decimals <= 53 stands for a double-precision floating point number
DECIMAL, NUMERIC	An unpacked floating-point number. Cannot be unsigned	Behaves like a CHAR column: "unpacked" means the number is stored as a string, using one character for each digit of the value. The decimal point, and, for negative numbers, the '-' sign is not counted in Length. If Decimals is 0, values will have no decimal point or fractional part. The maximum range of DECIMAL values is the same as for DOUBLE, but the actual range for a given DECIMAL column may be constrained by the choice of Length and Decimals. If Decimals is left out it's set to 0. If Length is left out it's set to 10. Note that in MySQL 3.22 the Length includes the sign and the decimal point
DATE	A date	The supported range is '1000-01-01' to '9999-12-31'. MySQL displays DATE values in 'YYYY-MM-DD' format
DATETIME	A date and time combination	The supported range is '1000-01-01 00:00:00' to '9999-12-31 23:59:59'. MySQL displays DATETIME values in 'YYYY-MM-DD HH:MM:SS' format
TIMESTAMP	A timestamp	The range is '1970-01-01 00:00:00' to sometime in the year 2037. MySQL displays TIMESTAMP values in YYYYMMDDHHMMSS, YMMDDHHMMSS, YYYYMMDD or YMMDD format, depending on whether M is 14 (or missing), 12, 8 or 6, but allows you to assign values to TIMESTAMP columns using either strings or numbers. A TIMESTAMP column is useful for recording the date and time of an INSERT or UPDATE operation because it is automatically set to the date and time of the most recent operation if you don't give it a value yourself
TIME	A time	The range is '-838:59:59' to '838:59:59'. MySQL displays TIME values in 'HH:MM:SS' format, but allows you to assign values to TIME columns using either strings or numbers
YEAR	A year in 2- or 4- digit formats (default is 4-digit)	The allowable values are 1901 to 2155, and 0000 in the 4 year format and 1970-2069 if you use the 2 digit format (70-69). MySQL displays YEAR values in YYYY format, but allows you to assign values to YEAR columns using either strings or numbers. (The YEAR type is new in MySQL 3.22.)
CHAR	A fixed-length string that is always right-padded with spaces to the specified length when stored	The range of Length is 1 to 255 characters. Trailing spaces are removed when the value is retrieved. CHAR values are sorted and compared in case-insensitive fashion according to the default character set unless the BINARY keyword is given
VARCHAR	A variable-length string. Note: Trailing spaces are removed when the value is stored (this differs from the ANSI SQL specification)	The range of Length is 1 to 255 characters. VARCHAR values are sorted and compared in case-insensitive fashion unless the BINARY keyword is given
TINYBLOB, TINYTEXT		A BLOB or TEXT column with a maximum length of 255 (2^8 - 1) characters
BLOB, TEXT		A BLOB or TEXT column with a maximum length of 65535 (2^16 - 1) characters
MEDIUMBLOB, MEDIUMTEXT		A BLOB or TEXT column with a maximum length of 16777215 (2^24 - 1) characters
LOB, LONGBLOB, LONGTEXT		A BLOB or TEXT column with a maximum length of 4294967295 (2^32 - 1) characters

ENUM	An enumeration	A string object that can have only one value, chosen from the list of values 'value1', 'value2', ..., or NULL. An ENUM can have a maximum of 65535 distinct values.
SET	A set	A string object that can have zero or more values, each of which must be chosen from the list of values 'value1', 'value2', ... A SET can have a maximum of 64 members

## Connecting MySQL with PHP

The following Steps are necessary to connect to a MySQL database with PHP:

### Step 1:

Connect to host

```
$db = mysql_connect('localhost', 'root', '');
```

### Step 2:

Select the default database

```
mysql_select_db('video', $db);
```

### Step 3:

Start SQL query

```
$sql = "SELECT titel, genre, jahr FROM tbl_movies WHERE titel LIKE '%007%'";
```

### Step 4:

Send query to host

```
$qry = mysql_query($sql, $db);
```

### Step 5:

Calculating the number of results

```
$result_size = mysql_num_rows($qry);
```

### Step 6:

Print virtual table into HTML

```
for ($i = 0; $i < $result_size; $i++){
    $title = mysql_result($qry, $i, "title");
    $genre = mysql_result($qry, $i, "genre");
    $year = mysql_result($qry, $i, "year");

    // HTML output
    echo "$title; $genre; $year <br>";
}
```

To avoid error messages, e.g. if a database connection failed, you can use the following command instead of the ones above:

```
$db = @mysql_connect('localhost', 'root', '') or die („Connection failed“);
mysql_select_db('video', $db) or die ('Database not found');
```

