|  |  |  |
| --- | --- | --- |
| MySQL Datatypes: | | |
| Type | Size | Description |
| CHAR[Length] | Length bytes | A fixed-length field from 0 to 255 characters long. |
| VARCHAR(Length) | String length + 1 bytes | A variable-length field from 0 to 255 characters long. |
| TINYTEXT | String length + 1 bytes | A string with a maximum length of 255 characters. |
| TEXT | String length + 2 bytes | A string with a maximum length of 65,535 characters. |
| MEDIUMTEXT | String length + 3 bytes | A string with a maximum length of 16,777,215 characters. |
| LONGTEXT | String length + 4 bytes | A string with a maximum length of 4,294,967,295 characters. |
| TINYINT[Length] | 1 byte | Range of -128 to 127 or 0 to 255 unsigned. |
| SMALLINT[Length] | 2 bytes | Range of -32,768 to 32,767 or 0 to 65535 unsigned. |
| MEDIUMINT[Length] | 3 bytes | Range of -8,388,608 to 8,388,607 or 0 to 16,777,215 unsigned. |
| INT[Length] | 4 bytes | Range of -2,147,483,648 to 2,147,483,647 or 0 to 4,294,967,295 unsigned. |
| BIGINT[Length] | 8 bytes | Range of -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807 or 0 to 18,446,744,073,709,551,615 unsigned. |
| FLOAT | 4 bytes | A small number with a floating decimal point. |
| DOUBLE[Length, Decimals] | 8 bytes | A large number with a floating decimal point. |
| DECIMAL[Length, Decimals] | Length + 1 or Length + 2 bytes | A DOUBLE stored as a string, allowing for a fixed decimal point. |
| DATE | 3 bytes | In the format of YYYY-MM-DD. |
| DATETIME | 8 bytes | In the format of YYYY-MM-DD HH:MM:SS. |
| TIMESTAMP | 4 bytes | In the format of YYYYMMDDHHMMSS; acceptable range ends inthe year 2037. |
| TIME | 3 bytes | In the format of HH:MM:SS |
| ENUM | 1 or 2 bytes | Short for enumeration, which means that each column can haveone of several possible values. |
| SET | 1, 2, 3, 4, or 8 bytes | Like ENUM except that each column can have more than one ofseveral possible values. |

Variables in php

Boolean variables  
Boolean variables contain only TRUE or FALSE as values.  
$var = TRUE;  
This is boolean variable. TRUE or FALSE should not be in quotes. Then it will be considered as strings.

Arithmetic Operations  
It usually includes addition, subtraction,multiplication, division, modulus which gives the remainder of the division and exponentiation.

| Example | Name | Result |
| --- | --- | --- |
| $a + $b | Addition | Sum of $a and $b |
| $a - $b | Subtraction | Difference of $a and $b |
| $a \* $b | Multiplication | Product of $a and $b |
| $a / $b | Division | Quotient of $a and $b |
| $a % $b | Modulo | Remainder of $a divided by $b |
| $a \*\* $b | Exponentiation | Result of raising $a to the $b th power. Introduced in PHP 5.6 |

Logical Operations  
Logical operators, Relational operators are used for comparisons. These are the valid operators for PHP. These operations either evaluate to TRUE or a FALSE state.

| Example | Name | Result |
| --- | --- | --- |
| $a and $b | And | TRUE if both $a and $b are TRUE, FALSE otherwise |
| $a || $b | Or | TRUE if either $a or $b is TRUE. If $a is TRUE and $b is false then also this condition will output true as one of the two options or both the option should be true in OR case |
| $a xor $b | Xor | TRUE if either $a or $b is TRUE, but not both; otherwise FALSE |
| ! $a | Not | TRUE if $a is not TRUE, otherwise FALSE |
| $a % $b | Modulo | Remainder of $a divided by $b |
|  |  |  |

Strings

Difference between single quotes and double quotes:

| Single Quotes | Double Quotes |
| --- | --- |
| Double quotes inside single quotes will be considered as strings. | Single quotes inside double quotes will be considered as strings. |
| Variables inside single quotes will be considered as strings and will display the variable name only. | Variables inside double quotes will be considered as variables only and will display the value of the variable instead of name. |
| Input: $variable = 5; echo 'The output is $variable.';  Output: The output is $variable. | Input: $variable = 5; echo "The output is $variable.";  Output: The output is 5. |

Commonly used string functions are:

* strrev($string): Returns the reversed string.
* strtolower($string): Returns the string with all the characters in lowercase only.
* strtoupper($string): Returns the string with all the characters in uppercase only.
* strpos($string,$find\_substring,start): This fucntion finds the position of the first occurrence of a substring in a string.  
  e.g. Input: echo strpos("I am a web developer", "web", 1); Output: 7.