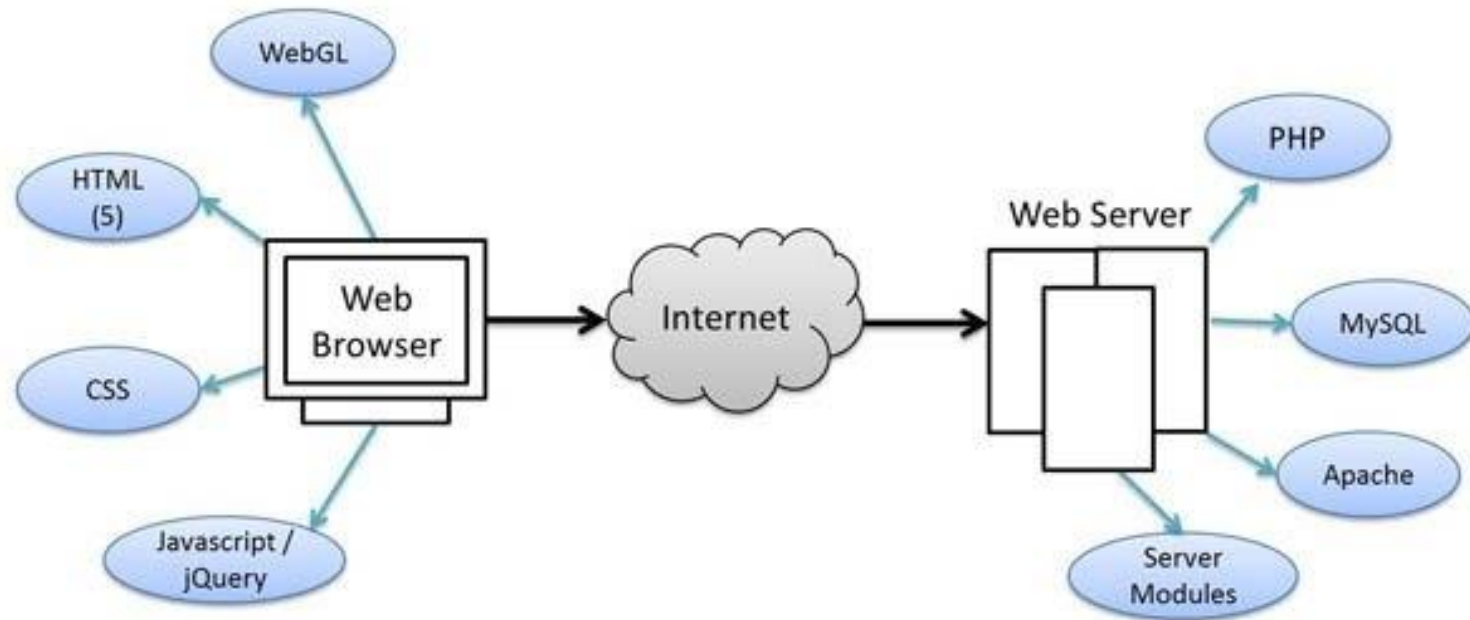
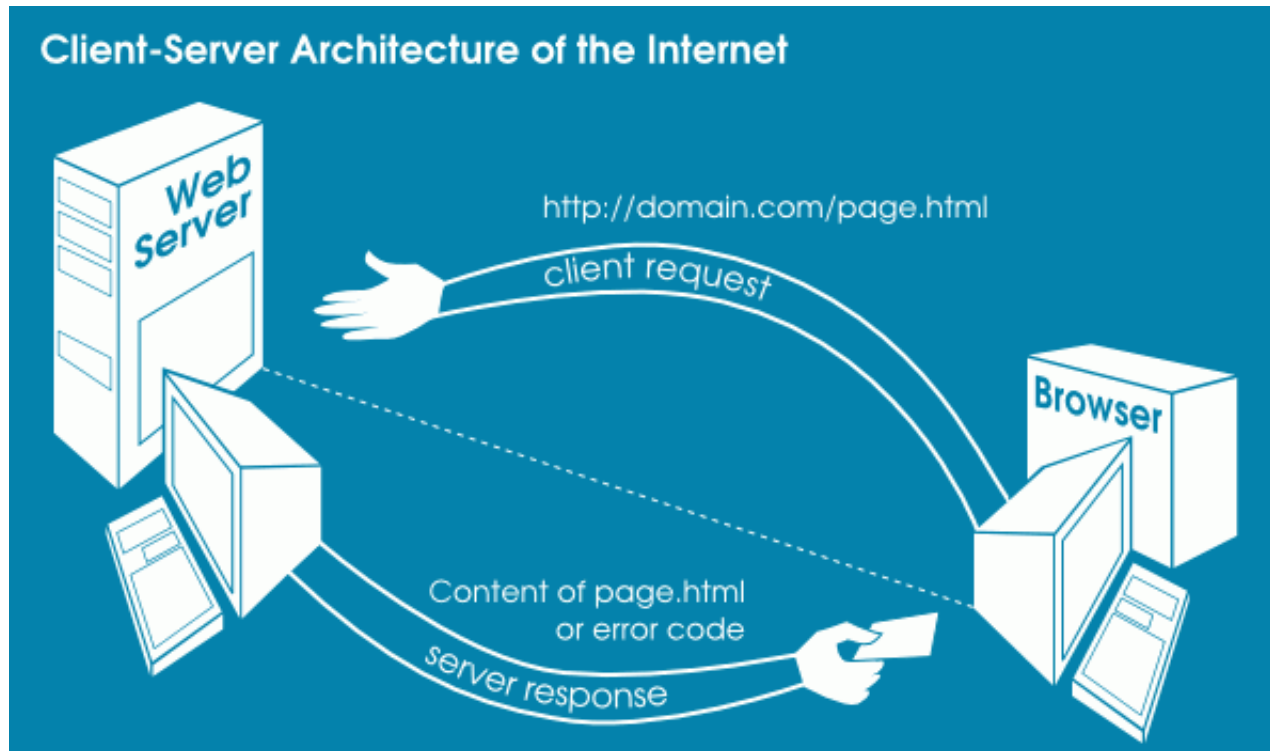


Apache HTTP and PHP

Web Server



How it works



Client-Server model and HTTP

- A request is generated by a client (by browser software)
 - Most common requests are “Get” and “Post”
- Request reaches the appropriate web-server
- Request is processed by the web-server
- A response is formulated by the web server and sent back to the client (e.g. web page contents)
- HTTP is the de facto standard for transferring World Wide Web documents
- Usually to port 80

HTTP: Requests from client: HTML Examples

<body>

<form method= “**post**” action =“from-process.jsp”>

Word to look up: <Input type = text Name = “word”>

<input types = “submit”>

</form>

</body>

HTTP Requests from client: HTML Examples

<body>

<form method= “**get**” action =“from-process.jsp”>

Word to look up: <Input type = text Name = “word”>

<input types = “submit”>

</form>

</body>

- **http://hostname?weight=200&height=60**

HTTP: Response from web server

```
root@tecadmin:~# wget --server-response --spider http://example.com/index.php
Spider mode enabled. Check if remote file exists.
--2018-01-12 16:36:59-- http://example.com/index.php
Resolving example.com (example.com)... 192.168.1.237
Connecting to example.com (example.com)[192.168.1.237]:80... connected.
HTTP request sent, awaiting response...
  HTTP/1.1 200 OK
  Date: Fri, 12 Jan 2018 11:06:59 GMT
  Server: Apache/2.4.18 (Ubuntu)
  X-Powered-By: PHP/7.2.0-2+ubuntu16.04.1+deb.sury.org+2
  Keep-Alive: timeout=5, max=100
  Connection: Keep-Alive
  Content-Type: text/html; charset=UTF-8
Length: unspecified [text/html]
Remote file exists and could contain further links,
but recursion is disabled -- not retrieving.
```

Accessing web servers

Must know host name on which web server resides

- **Remote web servers accessed using**
 - URL: `http://www.usask.ca/default.asp`
 - IP address: `http://207.60.134.230`
- **Local web servers (on same machine) accessed using *machine name* or *localhost***

Apache

HTTP SERVER

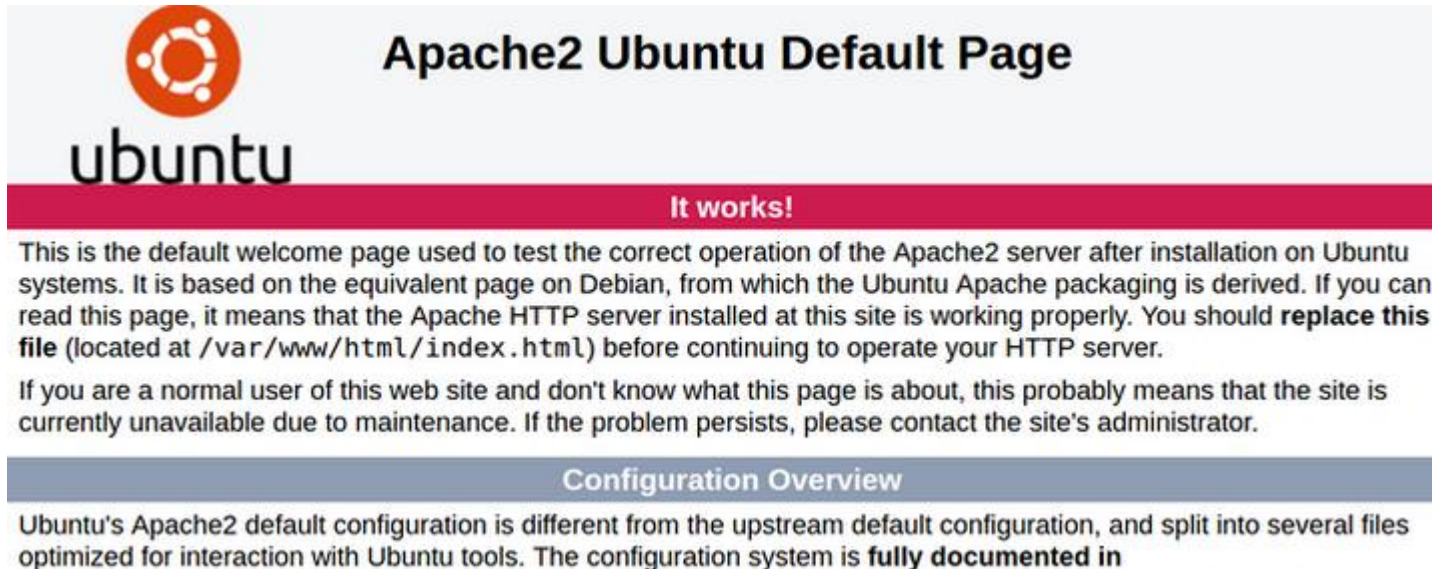


Installation and Setup - Apache, MySQL, and PHP on Linux

`sudo apt-get update`

`sudo apt-get install apache2`

<http://localhost/>



The screenshot shows the default Apache2 welcome page on Ubuntu. At the top left is the Ubuntu logo (a red circle with white dots) and the word "ubuntu" in a black sans-serif font. To the right of the logo, the title "Apache2 Ubuntu Default Page" is displayed in a bold black font. Below the title is a red horizontal bar with the text "It works!" in white. Underneath the bar, a paragraph explains that this is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It states that if the page can be read, the Apache HTTP server is working properly, and advises replacing the file located at `/var/www/html/index.html` before continuing to operate the HTTP server. Another paragraph follows, stating that if a normal user of the web site doesn't know what the page is about, it probably means the site is currently unavailable due to maintenance, and suggests contacting the site's administrator. At the bottom, there is a blue horizontal bar with the text "Configuration Overview" in white. Below this bar, a paragraph explains that Ubuntu's Apache2 default configuration is different from the upstream default configuration, is split into several files optimized for interaction with Ubuntu tools, and that the configuration system is fully documented in the `README` file.

Apache2 Ubuntu Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in** the `README` file.

Installation and Setup - Apache, MySQL, and PHP on Linux

```
sudo apt-get install mysql-server
```

```
sudo apt-get install python-software-properties
```

```
sudo add-apt-repository ppa:ondrej/php-7.0
```

```
sudo apt-get install php7.0-cli php7.0-common libapache2-mod-php7.0 php7.0 php7.0-  
mysql php7.0-fpm php7.0-curl php7.0-gd php7.0-bz2
```

Working Directory

```
var/html/www
```

In case of permission problem:

```
sudo chown -R username /var/www/html/
```

Installation and Setup - Apache, MySQL, and PHP on Linux

Working Directory

var/html/www

Create a new file : phpinfo.php

```
<?php  
phpinfo();  
?>
```



System	Linux ubuntu 4.4.0-101-generic #124-Ubuntu SMP Fri Nov 10 18:29:59 UTC 2017 x86_64
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2

Installation and Setup - Apache, MySQL, and PHP on Windows

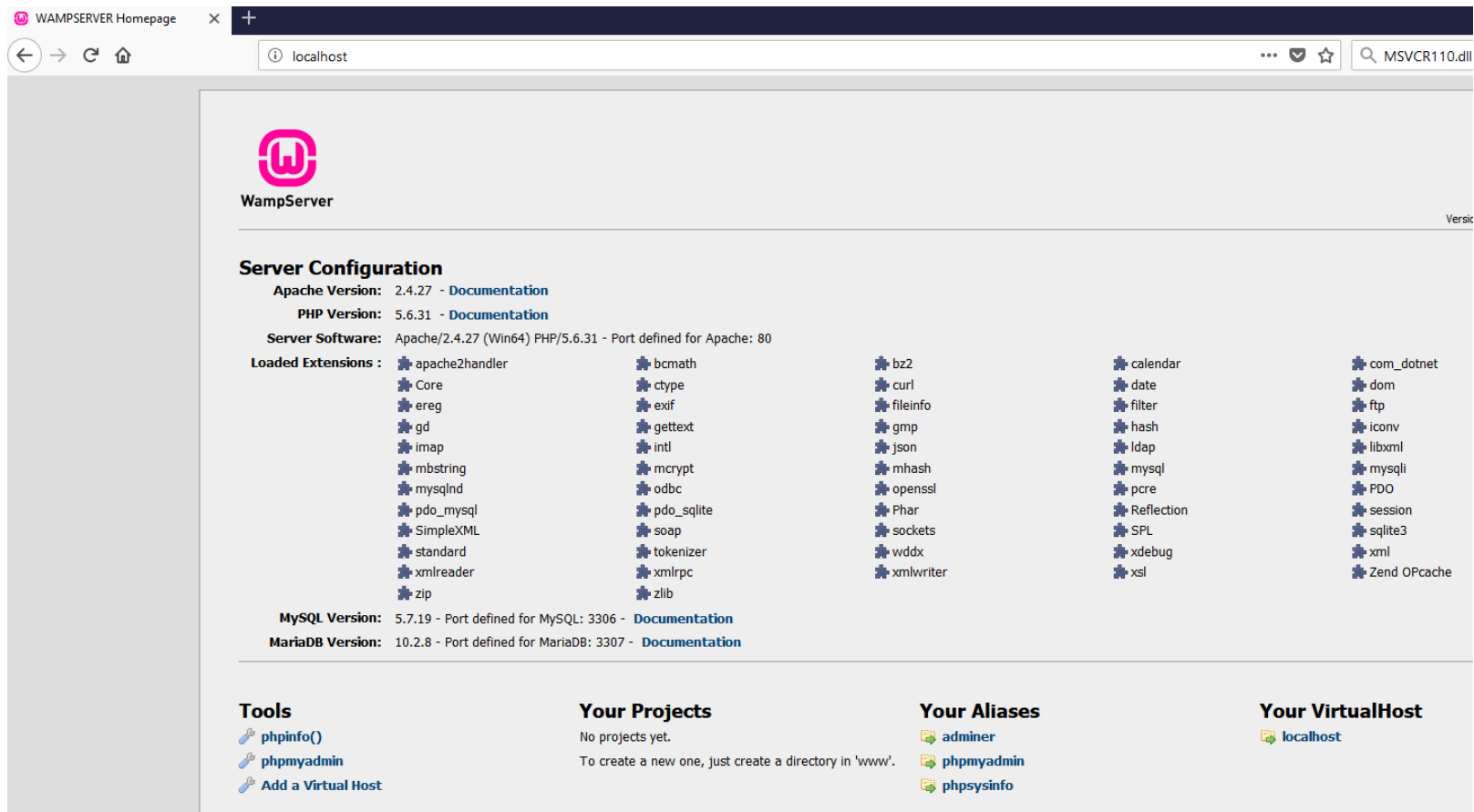


WampServer

Apache, PHP, MySQL sous Windows



Installation and Setup - Apache, MySQL, and PHP on Windows



The screenshot shows the WampServer homepage in a web browser. The browser's address bar shows 'localhost'. The page features the WampServer logo and a 'Server Configuration' section. This section lists the installed versions of Apache (2.4.27), PHP (5.6.31), MySQL (5.7.19), and MariaDB (10.2.8), along with links to their respective documentation. A 'Loaded Extensions' section displays a grid of 40 PHP extensions, including apache2handler, Core, ereg, gd, imap, mbstring, mysqlnd, pdo_mysql, SimpleXML, standard, xmlreader, zip, bcmath, ctype, exif, gettext, intl, mcrypt, odbc, pdo_sqlite, soap, tokenizer, xmlrpc, zlib, bz2, curl, fileinfo, gmp, json, mhash, openssl, Phar, sockets, vld, xmlwriter, calendar, date, filter, hash, ldap, mysql, pcres, Reflection, SPL, xdebug, xsl, com_dotnet, dom, ftp, iconv, libxml, mysqli, PDO, session, sqlite3, xml, and Zend OPcache. At the bottom, there are four sections: 'Tools' (with links to phpinfo(), phpmyadmin, and 'Add a Virtual Host'), 'Your Projects' (showing 'No projects yet.' and instructions to create a new project), 'Your Aliases' (listing adminer, phpmyadmin, and phpsysinfo), and 'Your VirtualHost' (showing localhost).

WampServer

Server Configuration

Apache Version: 2.4.27 - [Documentation](#)

PHP Version: 5.6.31 - [Documentation](#)

Server Software: Apache/2.4.27 (Win64) PHP/5.6.31 - Port defined for Apache: 80

Loaded Extensions :

- apache2handler
- Core
- ereg
- gd
- imap
- mbstring
- mysqlnd
- pdo_mysql
- SimpleXML
- standard
- xmlreader
- zip
- bcmath
- ctype
- exif
- gettext
- intl
- mcrypt
- odbc
- pdo_sqlite
- soap
- tokenizer
- xmlrpc
- zlib
- bz2
- curl
- fileinfo
- gmp
- json
- mhash
- openssl
- Phar
- sockets
- vld
- xmlwriter
- calendar
- date
- filter
- hash
- ldap
- mysql
- pcres
- Reflection
- SPL
- xdebug
- xsl
- com_dotnet
- dom
- ftp
- iconv
- libxml
- mysqli
- PDO
- session
- sqlite3
- xml
- Zend OPcache

MySQL Version: 5.7.19 - Port defined for MySQL: 3306 - [Documentation](#)

MariaDB Version: 10.2.8 - Port defined for MariaDB: 3307 - [Documentation](#)

Tools

- [phpinfo\(\)](#)
- [phpmyadmin](#)
- [Add a Virtual Host](#)

Your Projects

No projects yet.

To create a new one, just create a directory in 'www'.

Your Aliases

- [adminer](#)
- [phpmyadmin](#)
- [phpsysinfo](#)

Your VirtualHost

- [localhost](#)

An introduction to PHP web programming



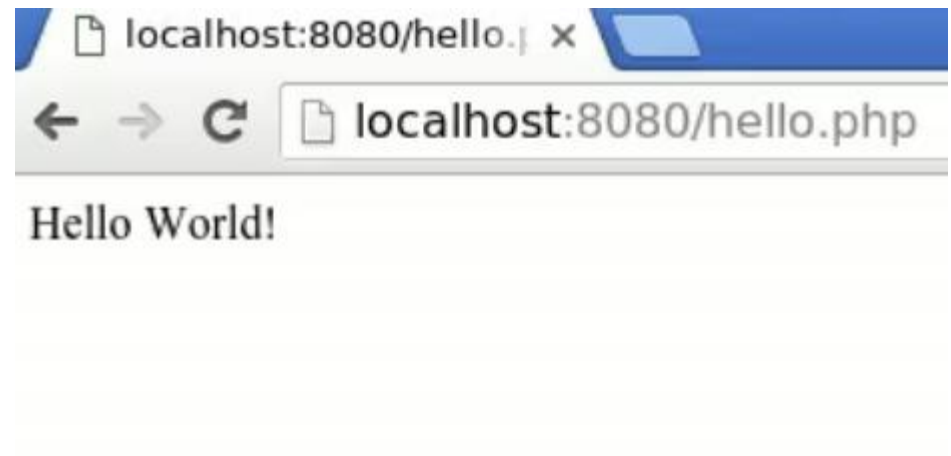
```
<? php ?>
```

An introduction to PHP web programming

- **PHP is a general-purpose server-side scripting language**
- **PHP can interact with MySQL databases**
- **All PHP statements end with a semi-colon**
- **PHP == 'Hypertext Preprocessor'**
- **Open-source and free**
- **Used to generate dynamic web-pages**
- **Easy to learn**

Basic PHP syntax

```
hello.php x
1 <?php
2 //one line comment
3
4 /*
5 Hi
6 I am
7 multi-line
8 comment
9 */
10
11 /**
12  * I am a doc block
13  * comment
14  */
15
16 # I am a shell style comment
17
18 echo 'Hello World!';
19 ?>
```



Variables in PHP

- PHP variables must begin with a "\$" sign
- Case-sensitive (\$Foo != \$foo != \$fOo)
- Global and locally-scoped variables
 - Global variables can be used anywhere
 - Local variables restricted to a function or class
- Certain variable names reserved by PHP
 - Form variables (\$_POST, \$_GET)
 - Server variables (\$_SERVER)

Echo

- The PHP command 'echo' is used to output the parameters passed to it
- The typical usage for this is to send data to the client's web-browser

```
<?php
$foo = 25;           // Numerical variable
$bar = "Hello";      // String variable

echo $bar;           // Outputs Hello
echo $foo, $bar;      // Outputs 25Hello
echo "5x5=", $foo;    // Outputs 5x5=25
echo "5x5=$foo";      // Outputs 5x5=25
echo '5x5=$foo';      // Outputs 5x5=$foo
?>
```

Concatenation

- Use a period to join strings into one.

```
<?php
$string1="Hello";
$string2="PHP";
$string3=$string1 . " " .
$string2;
Print $string3;
?>
```

Hello PHP

Functions

- Functions **MUST** be defined before then can be called
- Unlike variables, function names are not case sensitive (foo(...) == Foo(...) == FoO(...))

Syntax

```
function functionName() {  
    code to be executed;  
}
```

```
<?php  
function writeMsg() {  
    echo "Hello world!";  
}  
  
writeMsg(); // call the function  
?>
```

Include Files

- Include “database-config.php”;

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

// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```

PHP References

- <http://www.php.net> <-- php home page
- <http://www.phpbuilder.com/>
- <http://www.devshed.com/>
- <http://www.hotscripts.com/PHP/>
- <http://geocities.com/stuprojects/ChatroomDescription.htm>
- <http://www.academic.marist.edu/~kbhkj/chatroom/chatroom.htm>
- <http://www.aus-ettrade.com/Scripts/php.php>
- <http://www.codeproject.com/asp/CDIChatSubmit.asp>
- <http://php.resourceindex.com/> <-- PHP resources like sample programs, text book references, etc.

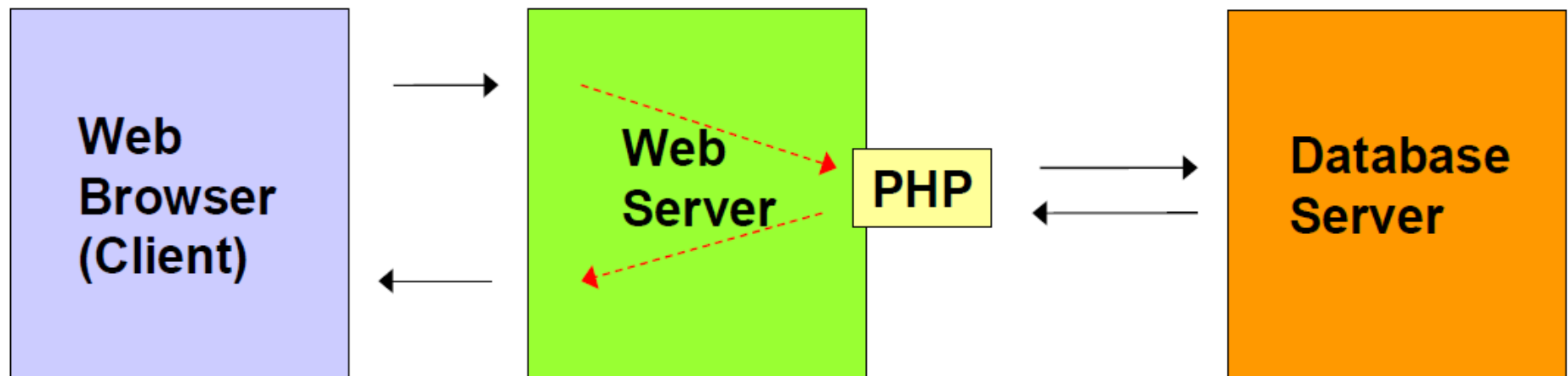


PHP & MySQL

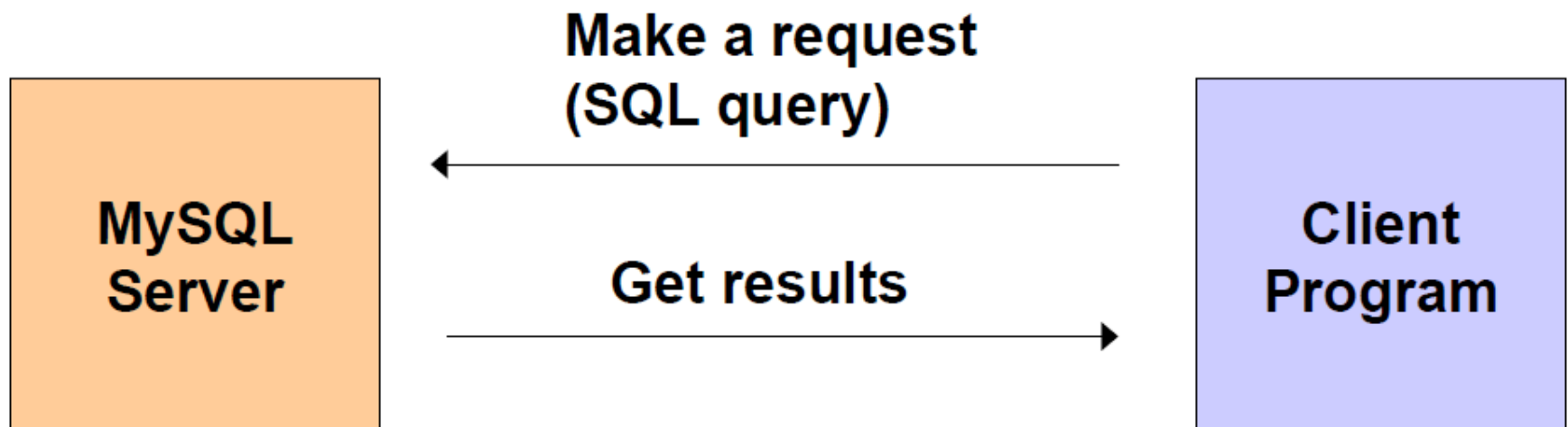
What is MySQL?

- **A relational database management system**
- **The MySQL Database Server is very fast, reliable, and easy to use**
- **SQL: Structured Query Language**
- **A server providing multi-user access to a number of databases**
- **MySQL software is Open Source**

Architecture



Client-Server Interaction



Database concepts

- A relational database management system consists of a number of databases.
- Each database consists of a number of tables.

books table

isbn	title	author	pub	year	price

column headings

rows (records)

WARNING

- **Always assume that everything is case sensitive, especially table names and column names.**
- **This is not the case in Windows but it is the case in Linux**

An introduction to MySQL

- Show all the databases

```
mysql> SHOW DATABASES;  
+-----+  
| Database |  
+-----+  
| bookstore |  
| employee_db |  
| mysql |  
| student_db |  
| test |  
| web_db |  
+-----+
```

An introduction to MySQL

- Choosing a database and showing its tables

```
USE test;  
SHOW tables;
```

```
mysql> USE test;  
Database changed  
mysql> SHOW tables;  
+-----+  
| Tables_in_test |  
+-----+  
| books          |  
| name2          |  
| names          |  
| test           |  
+-----+  
4 rows in set (0.00 sec)  
mysql>
```

An introduction to MySQL

- Show the structure of a table

■ **DESCRIBE names;**

```
mysql> DESCRIBE names;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)       |      | PRI | NULL    | auto_increment |
| firstName  | varchar(20)   |      |     |         |                |
| lastName   | varchar(20)   |      |     |         |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```


An introduction to MySQL

- Show the rows of a table

■ **SELECT * FROM names;**

```
mysql> SELECT * FROM names;
+----+-----+-----+
| id | firstName | lastName |
+----+-----+-----+
|  1 | Fred      | Flintstone |
|  2 | Barney    | Rubble     |
+----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

An introduction to MySQL

- Inserting a new record

```
INSERT INTO names (firstName, lastName) VALUES ('John', 'Smith');
```

```
SELECT * FROM names;
```

```
mysql> INSERT INTO names (firstName, lastName) VALUES ('John', 'Smith');
```

```
Query OK, 1 row affected (0.02 sec)
```

```
mysql> SELECT * FROM names;
```

```
+----+-----+-----+
| id | firstName | lastName |
+----+-----+-----+
| 1 | Fred | Flintstone |
| 2 | Barney | Rubble |
| 3 | John | Smith |
+----+-----+-----+
```

```
3 rows in set (0.00 sec)
```

```
mysql>
```

An introduction to MySQL

- Updating a record

```
UPDATE names SET firstName = 'Jane' WHERE id=3;  
SELECT * FROM names;
```

```
mysql> UPDATE names SET firstName = 'Jane' WHERE id=3;  
Query OK, 1 row affected (0.28 sec)  
Rows matched: 1 Changed: 1 Warnings: 0  
mysql> SELECT * FROM names;  
+----+-----+-----+  
| id | firstName | lastName |  
+----+-----+-----+  
| 1 | Fred | Flintstone |  
| 2 | Barney | Rubble |  
| 3 | Jane | Smith |  
+----+-----+-----+  
3 rows in set (0.00 sec)  
mysql>
```

SQL data types

- Each entry in a row has a type specified by the column.
- Numeric data types
- TINYINT, SMALLINT, MEDIUMINT,
- INT, BIGINT
- FLOAT(display_length, decimals)
- DOUBLE(display_length, decimals)
- DECIMAL(display_length, decimals)
- NUMERIC is the same as DECIMAL

SQL data types

String types

CHAR: fixed length string, e.g., CHAR(20)

VARCHAR: variable length string, e.g., VARCHAR(20)

BLOB, TINYBLOB, MEDIUMBLOB, LONGBLOB: same as TEXT, TINYTEXT ...

The CREATE Command

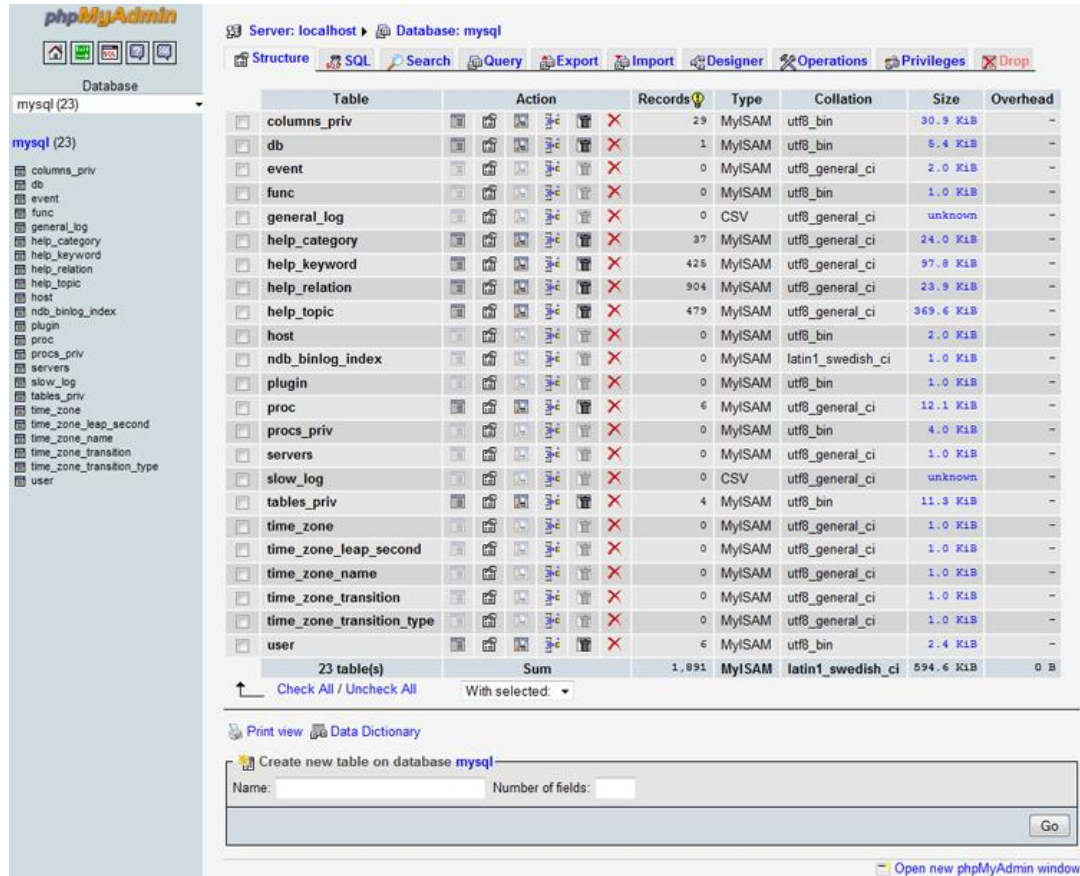
CREATE creates a database table

```
CREATE TABLE table_name
(
    column_name1 column_type1,
    column_name2 column_type2,
    ...
    column_nameN column_typeN
);
```

Note: To create a database use the statement

```
CREATE db_name;
```

phpMyAdmin



Server: localhost Database: mysql

Structure SQL Search Query Export Import Designer Operations Privileges Drop

Table	Action	Records	Type	Collation	Size	Overhead
<input type="checkbox"/> columns_priv		29	MyISAM	utf8_bin	30.9 KiB	-
<input type="checkbox"/> db		1	MyISAM	utf8_bin	6.4 KiB	-
<input type="checkbox"/> event		0	MyISAM	utf8_general_ci	2.0 KiB	-
<input type="checkbox"/> func		0	MyISAM	utf8_bin	1.0 KiB	-
<input type="checkbox"/> general_log		0	CSV	utf8_general_ci	unknown	-
<input type="checkbox"/> help_category		37	MyISAM	utf8_general_ci	24.0 KiB	-
<input type="checkbox"/> help_keyword		425	MyISAM	utf8_general_ci	97.8 KiB	-
<input type="checkbox"/> help_relation		904	MyISAM	utf8_general_ci	23.9 KiB	-
<input type="checkbox"/> help_topic		479	MyISAM	utf8_general_ci	369.6 KiB	-
<input type="checkbox"/> host		0	MyISAM	utf8_bin	2.0 KiB	-
<input type="checkbox"/> ndb_binlog_index		0	MyISAM	latin1_swedish_ci	1.0 KiB	-
<input type="checkbox"/> plugin		0	MyISAM	utf8_bin	1.0 KiB	-
<input type="checkbox"/> proc		6	MyISAM	utf8_general_ci	12.1 KiB	-
<input type="checkbox"/> procs_priv		0	MyISAM	utf8_bin	4.0 KiB	-
<input type="checkbox"/> servers		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> slow_log		0	CSV	utf8_general_ci	unknown	-
<input type="checkbox"/> tables_priv		4	MyISAM	utf8_bin	11.3 KiB	-
<input type="checkbox"/> time_zone		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> time_zone_leap_second		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> time_zone_name		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> time_zone_transition		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> time_zone_transition_type		0	MyISAM	utf8_general_ci	1.0 KiB	-
<input type="checkbox"/> user		6	MyISAM	utf8_bin	2.4 KiB	-
23 table(s)	Sum	1,891	MyISAM	latin1_swedish_ci	594.6 KiB	0 B

Check All / Uncheck All With selected: ▾

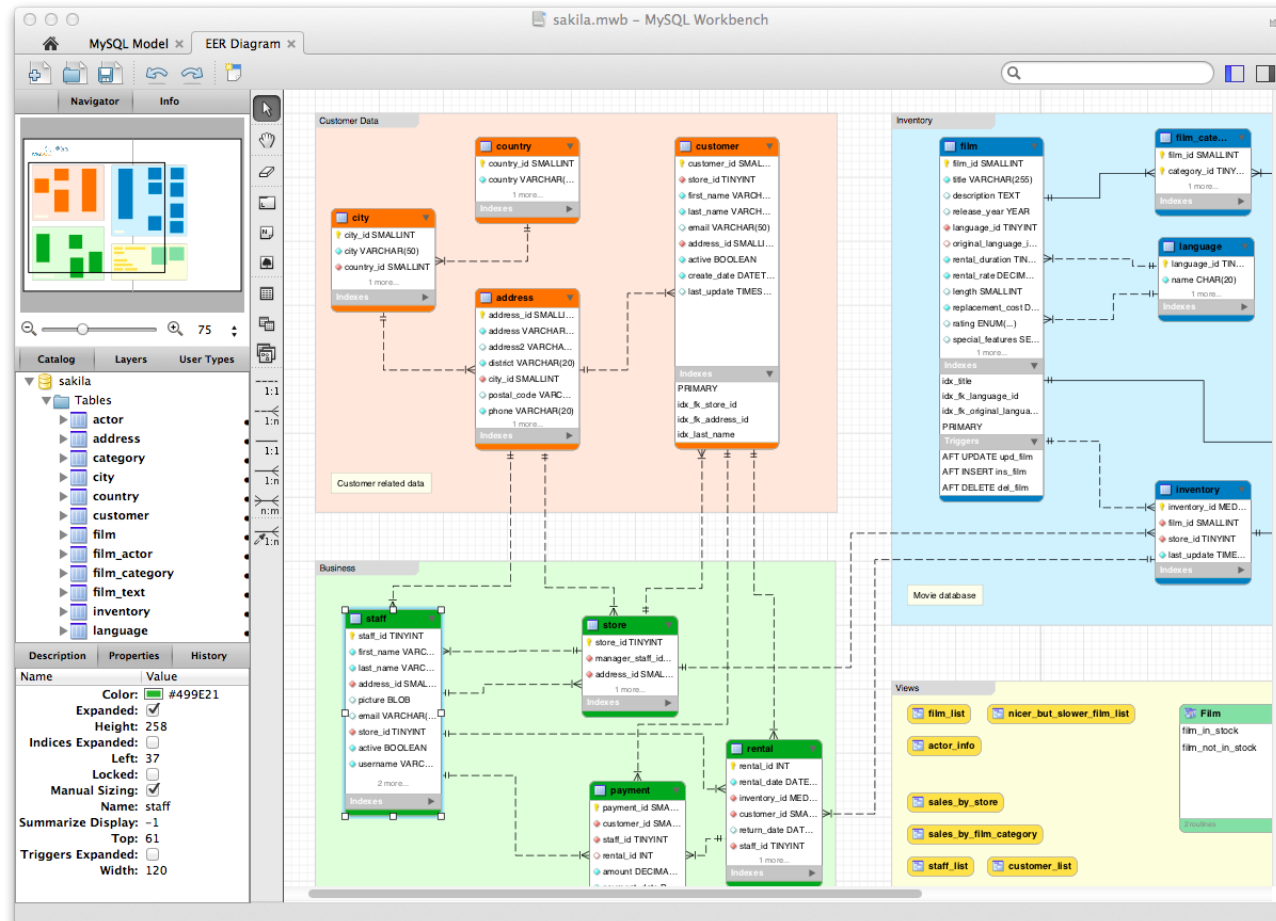
Print view Data Dictionary

Create new table on database **mysql**

Name: Number of fields:

[Open new phpMyAdmin window](#)

MySQL Workbench



Creating a Simple PHP and MySQL-Based Login System

Step 1: Creating the Database Table

```
1. CREATE TABLE users (  
2.     id INT NOT NULL PRIMARY KEY AUTO_INCREMENT,  
3.     username VARCHAR(50) NOT NULL UNIQUE,  
4.     password VARCHAR(255) NOT NULL,  
5.     created_at DATETIME DEFAULT CURRENT_TIMESTAMP  
6. );
```

Creating a Simple PHP and MySQL-Based Login System

Step 2: Creating the Config File

```
01. <?php
02. /* Database credentials. Assuming you are running MySQL
03. server with default setting (user 'root' with no password) */
04. define('DB_SERVER', 'localhost');
05. define('DB_USERNAME', 'root');
06. define('DB_PASSWORD', '');
07. define('DB_NAME', 'demo');
08.
09. /* Attempt to connect to MySQL database */
10. $link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);
11. "
```

Creating a Simple PHP and MySQL-Based Login System

Step 3: Creating the Registration Form

```
01. <?php
02. // Include config file
03. require_once 'config.php';
04.
05. // Define variables and initialize with empty values
06. $username = $password = $confirm_password = "";
07. $username_err = $password_err = $confirm_password_err = "";
08.
09. // Processing form data when form is submitted
10. if($_SERVER["REQUEST_METHOD"] == "POST"){
11.
12.     // Validate username
13.     if(empty(trim($_POST["username"]))){
14.         $username_err = "Please enter a username.";
15.     } else{
16.         // Prepare a select statement
17.         $sql = "SELECT id FROM users WHERE username = ?";
18.     }
```

Sign Up

Please fill this form to create an account.

Username

Password

Confirm Password

Submit

Reset

Already have an account? [Login here.](#)

Creating a Simple PHP and MySQL-Based Login System

Step 4: Creating the Login Form

```
01. <?php
02. // Include config file
03. require_once 'config.php';
04.
05. // Define variables and initialize with empty values
06. $username = $password = "";
07. $username_err = $password_err = "";
08.
09. // Processing form data when form is submitted
10. if($_SERVER["REQUEST_METHOD"] == "POST"){
11.
12.     // Check if username is empty
13.     if(empty(trim($_POST["username"]))){
14.         $username_err = 'Please enter username.';
15.     } else{
```

Login

Please fill in your credentials to login.

Username

Password

Login

Don't have an account? [Sign up now.](#)

Creating a Simple PHP and MySQL-Based Login System

Step 5: Creating the Welcome Page

```
01. <?php
02. // Initialize the session
03. session_start();
04.
05. // If session variable is not set it will redirect to login page
06. if(!isset($_SESSION['username']) || empty($_SESSION['username'])){
07.     header("location: login.php");
08.     exit;
09. }
10. ?>
```