Assignment

The objective of this exercise is to deploy a LAMP stack (Linux, Apache, MySQL, PHP) on an Apache2 server, and use PHP to test it operates normally.

You can use any linux machine with a GUI, the Desktop installations already have one provided. Root account will be used to deploy the system.

Suggested environment (all modules can be the latest version available):

- Linux: Ubuntu 22.04
- Web Server: Apache2
- Database: MySQL, installed to support PHP
- Script Engine: PHP
- The above combination is a LAMP stack, and has a complete package available.

There is a more in-depth coverage of installing a LAMP stack at https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-22-04, however we'll do the bare minimum to see it in operation.

It is recommended to update the environment to the latest patch state. On Ubuntu, you perform the update using Aptitude packet manager. The command is:

~$ apt update

(use sudo apt update if super user rights are required)

We'll begin by logging in as root:

~$ sudo -i

This makes us no longer require the sudo command, as we are above a super user level already.

You may want to temporarily turn off firewall so it won't interfere with testing:

~# ufw disable

We'll begin by installing Apache2:

~# apt install apache2

You can now test if Apache2 is installed correctly visiting http://localhost on your desktop VMs browser. You should get a default description page for the web server. The hostname "localhost" loops back to the originating computer due to the hostname being included in the OS hosts file.

Install MySQL:

~# apt install mysql-server

Like Apache2, MySQL service should start automatically. It can be administered in it's own prompt using SQL commands:

~# mysql

Start by checking if you are the root user:

mysql> SELECT USER();

If you aren't root, you may exit and force start MySQL as the root user:

~# mysql -u root

Create a database called "testing", then make it active:

mysql> CREATE DATABASE testing;

mysql> USE testing

!NOTE the database must always be selected with the query "USE <database>" before modifying or requesting information from it. Always make sure the database is selected before data creation, or the data will not go into the database.

Create a user called "user" in "localhost" so the machine itself, whose password is "student". Then give rights to the database "testing" to "user". In this case all the rights to the database are given, but in this exercise only access to read would suffice (ALL → READ):

mysql> CREATE USER ‘user’@’localhost’ IDENTIFIED BY ‘student’;

mysql> GRANT ALL ON testing.* TO 'user’@’localhost’;

Create a simple table, with the fields “ID”, “name”, and “info”. Press ENTER after each line:

mysql> create table personnel (

ID int primary key not null,

name varchar(30),

info varchar(50)

);

You can check the contents of the table:

mysql> DESCRIBE personnel;

Then start adding information to the fields:

mysql > INSERT INTO personnel VALUES

(1,’Mattila Matti’,’Esquire’),

(2,’Pekkala Pekka’,’Hunter’),

(3,’Jussila Jussi’,’Lord’);

Finally verify the information entered:

mysql> SELECT * FROM personnel;

As the database is set up, we’ll install PHP and the PHP binary designed to work with Apache2:

~# apt install php libapache2-mod-php

Installing PHP modules to connect with MySQL and perform graphical representation:

~# apt install php-mysql php-gd

Finally enable PHP Data Objects module to connect to MySQL, as this is the standard way to make database calls in PHP 8. Enabling modules requires restarting the web server:

~# phpenmod pdo_mysql

~# systemctl restart apache2

You can now test the behavior of PHP. Create a PHP file inside the default public Apache directory "/var/www/html/":

~# nano info.php

Then enter the next contents inside the file and save:

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

You can now load the PHP info page pointing the browser to http://localhost/info.php.

Create another PHP file to test the MySQL database you've created:

~# nano sqltest.php

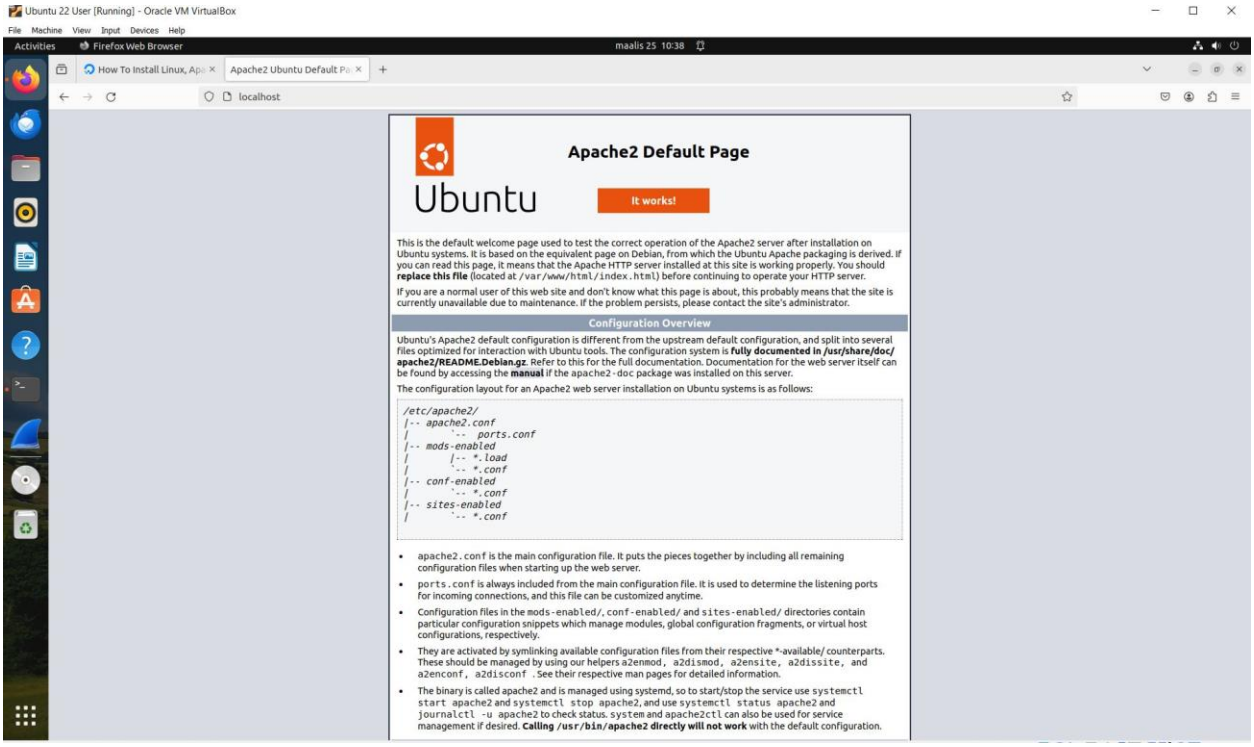
Enter the next contents inside the file and save. To prevent typos it might be a good idea to have VBox Guest Additions installed, and clipboard copy between host and guest systems enabled. Lines starting with "//" are comments that clarify what each part of the script is for, and are not required for the script to run. However commented lines are not read by the system, so they can be included in the PHP file. The actual script is ten (10) lines long:

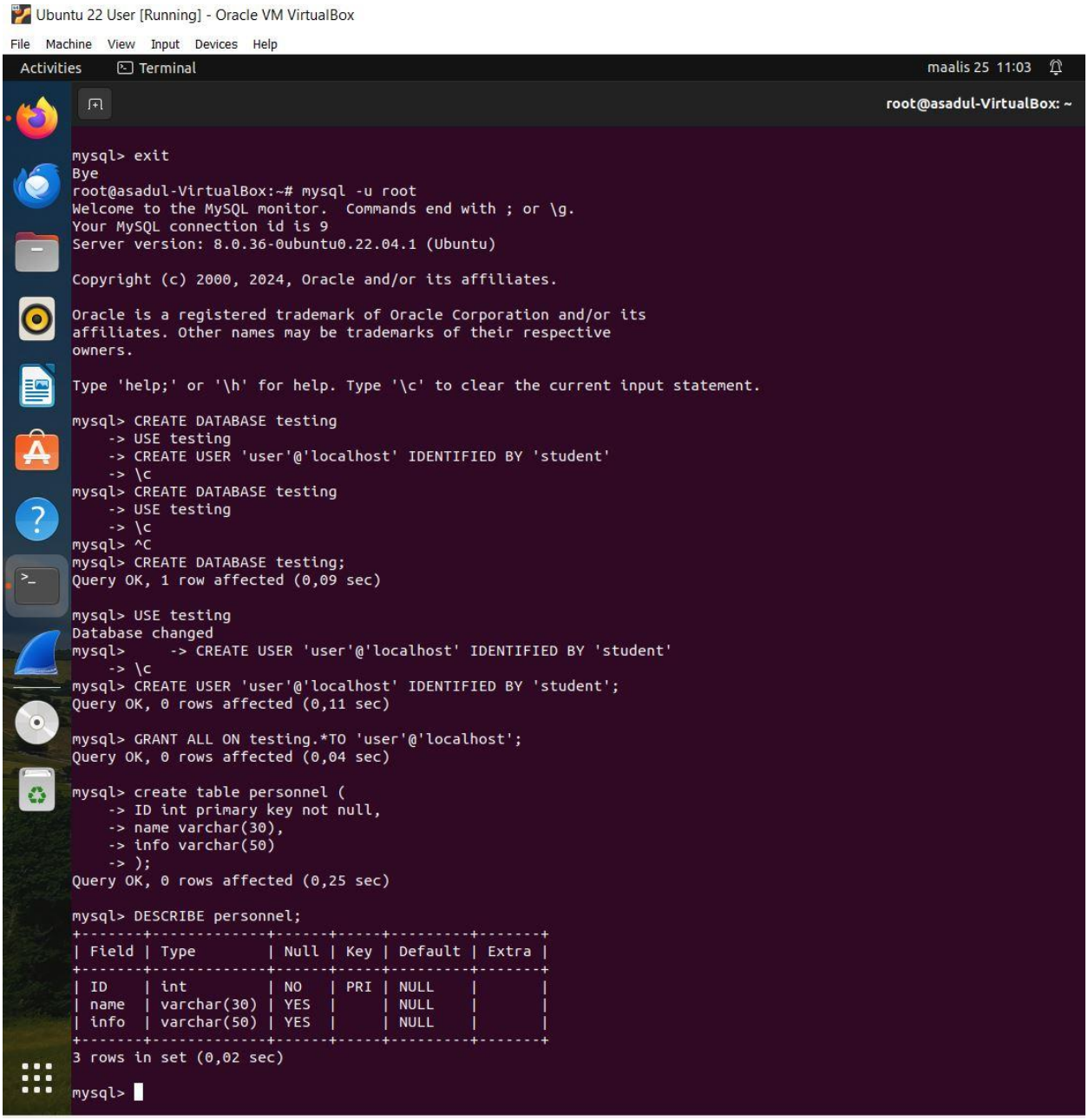
```
<?php
// make our configuration variables for the script
$user = "user";
$password = "student";
$database = "testing";
$table = "personnel";

// create a code block that can be "caught"
try {
```

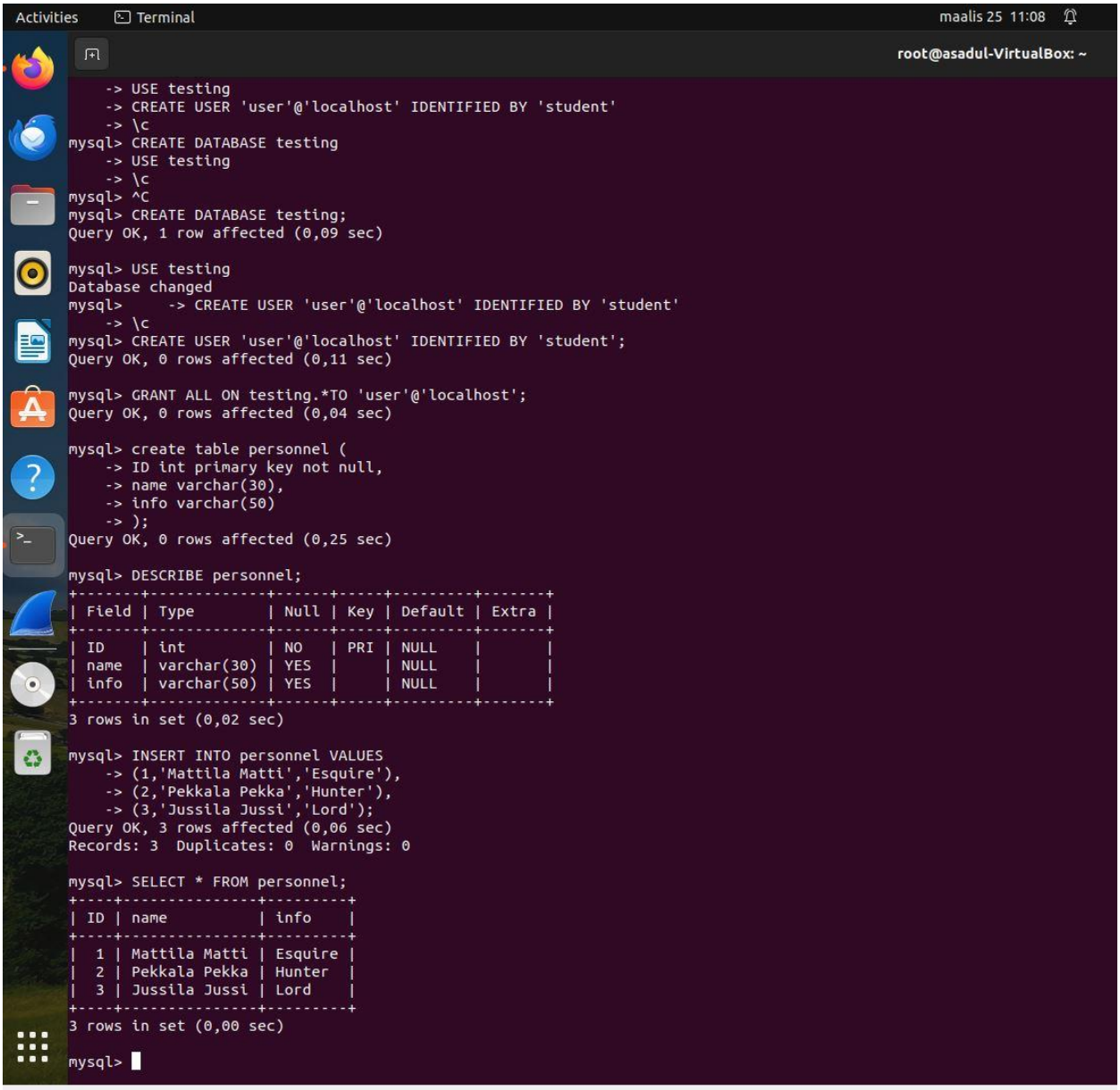
```
// make a PHP Data Objects connection to the database
 $db = new PDO("mysql:host=localhost;dbname=$database", $user, $password);
// print a pre-defined string with a line break
 echo "<h2>LASTNAME FIRSTNAME Personnel File</h2><br>";
// go through every row in the database table, then print results as list items
 foreach($db->query("SELECT ID,name,info FROM $table") as $row) {
  echo "<li>" . $row['ID'] . ' ' . $row['name'] . ' ' . $row['info'] . "</li>";
 }
// "catch" any exception the code block produced, IF it did
} catch (PDOException $e) {
  print "Error!: " . $e->getMessage() . "<br/>";
  die();
}
?>
```

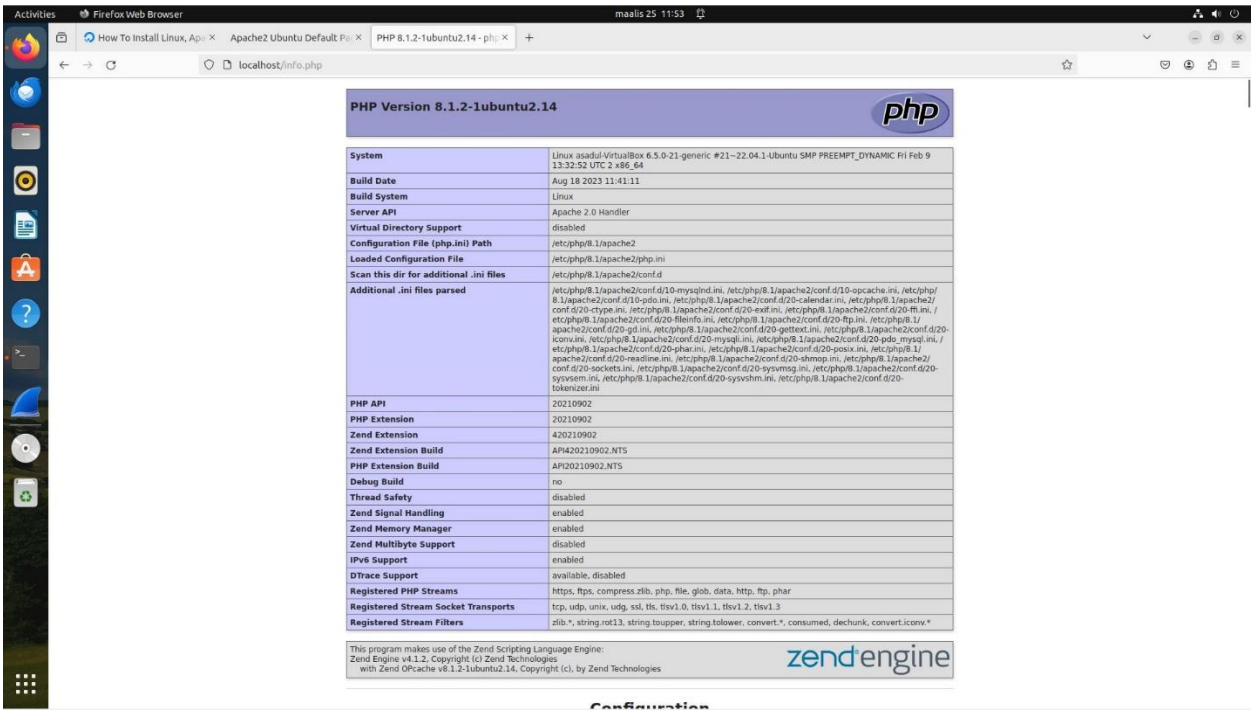
For exercise returns, in the above script REPLACE “lastname” and “firstname” with your own, EXECUTE modified sqltest.php USING YOUR VM BROWSER, and return A SCREENCAP of the resulting web page.

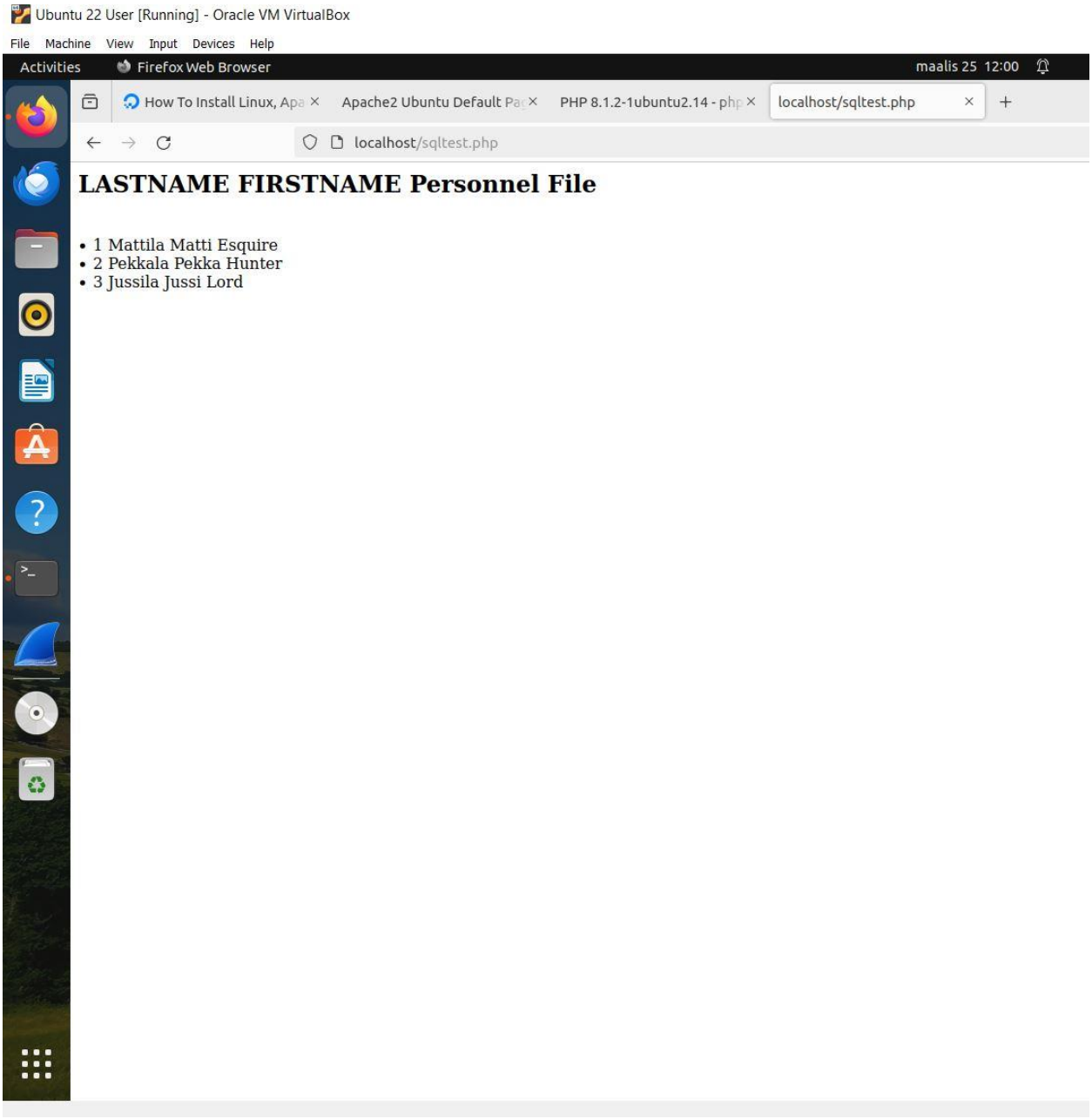
Solution

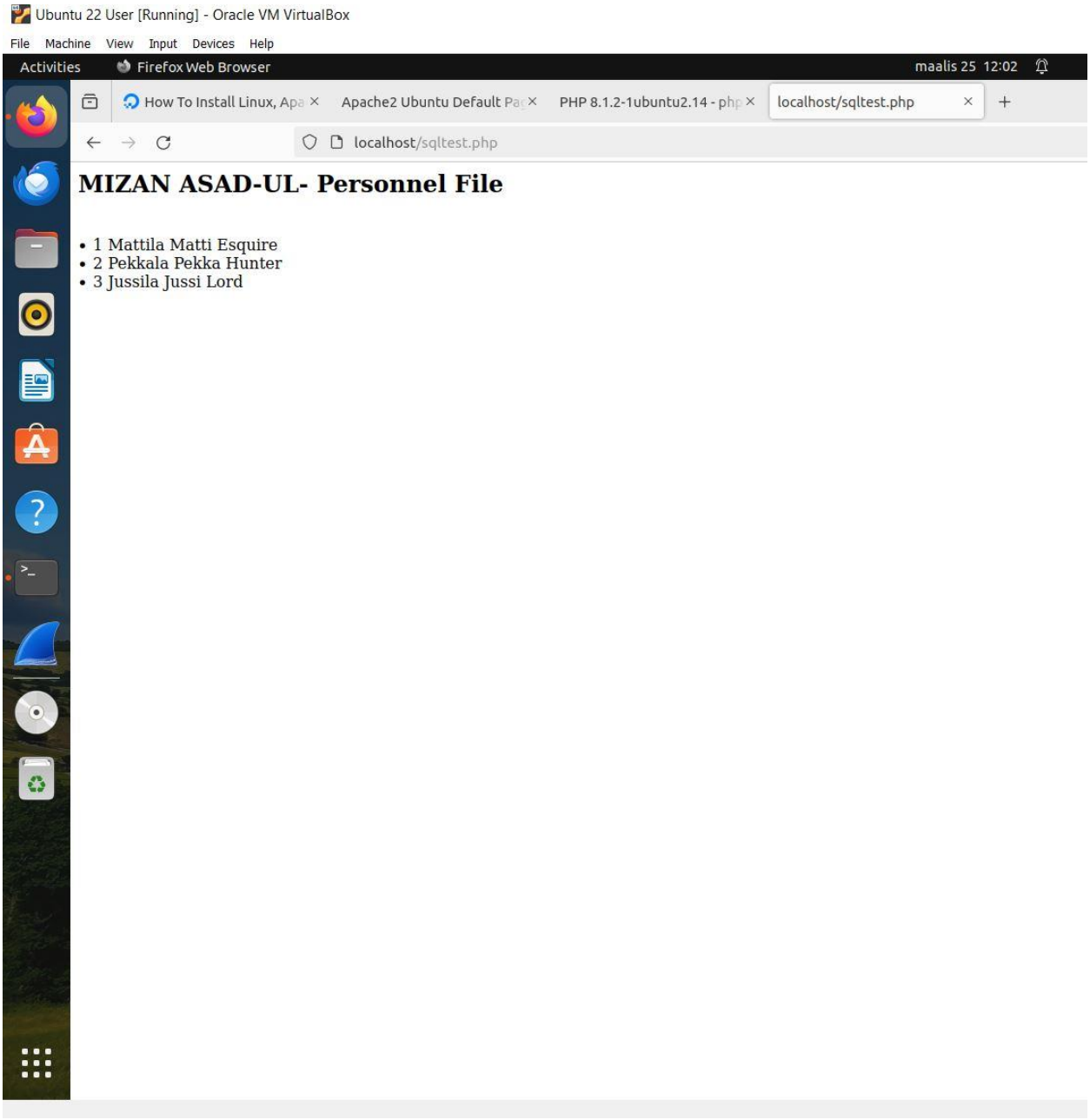

Ubuntu 22 User [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
maalis 25 10:38
How To Install Linux, Apa
Apache2 Ubuntu Default Pa
localhost
Apache2 Default Page
Ubuntu
It works!
This is the default welcome page used to test the correct operation of 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 /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the manual if the apache2-doc package was installed on this server.
The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:
/etc/apache2/
|-- apache2.conf
| `-- ports.conf
|-- mods-enabled
| |-- *.load
| `-- *.conf
|-- conf-enabled
| `-- *.conf
|-- sites-enabled
| `-- *.conf
apache2.conf is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
ports.conf is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
Configuration files in the mods-enabled/, conf-enabled/ and sites-enabled/ directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers a2enmod, a2dismod, a2ensite, a2dissite, and a2enconf, a2disconf . See their respective man pages for detailed information.
The binary is called apache2 and is managed using systemd, so to start/stop the service use systemctl start apache2 and systemctl stop apache2, and use systemctl status apache2 and journalctl -u apache2 to check status. system and apache2ctl can also be used for service management if desired. Calling /usr/bin/apache2 directly will not work with the default configuration.


Ubuntu 22 User [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
maalis 25 11:03
root@asadul-VirtualBox: ~
mysql> exit
Bye
root@asadul-VirtualBox:~# mysql -u root
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)
Copyright (c) 2000, 2024, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql> CREATE DATABASE testing
-> USE testing
-> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student'
-> \c
mysql> CREATE DATABASE testing
-> USE testing
-> \c
mysql> ^C
mysql> CREATE DATABASE testing;
Query OK, 1 row affected (0,09 sec)
mysql> USE testing
Database changed
mysql> -> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student'
-> \c
mysql> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student';
Query OK, 0 rows affected (0,11 sec)
mysql> GRANT ALL ON testing.*TO 'user'@'localhost';
Query OK, 0 rows affected (0,04 sec)
mysql> create table personnel (
-> ID int primary key not null,
-> name varchar(30),
-> info varchar(50)
->);
Query OK, 0 rows affected (0,25 sec)
mysql> DESCRIBE personnel;
+-------+-------------+------+-----+---------+-------+
| Field | Type | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
ID	int	NO	PRI	NULL	
name	varchar(30)	YES		NULL	
info	varchar(50)	YES		NULL	
+-------+-------------+------+-----+---------+-------+
3 rows in set (0,02 sec)
mysql>


Activities
Terminal
maalis 25 11:08
root@asadul-VirtualBox: ~
-> USE testing
-> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student'
-> \c
mysql> CREATE DATABASE testing
-> USE testing
-> \c
mysql> ^C
mysql> CREATE DATABASE testing;
Query OK, 1 row affected (0,09 sec)
mysql> USE testing
Database changed
mysql> -> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student'
-> \c
mysql> CREATE USER 'user'@'localhost' IDENTIFIED BY 'student';
Query OK, 0 rows affected (0,11 sec)
mysql> GRANT ALL ON testing.*TO 'user'@'localhost';
Query OK, 0 rows affected (0,04 sec)
mysql> create table personnel (
-> ID int primary key not null,
-> name varchar(30),
-> info varchar(50)
->);
Query OK, 0 rows affected (0,25 sec)
mysql> DESCRIBE personnel;
+-------+-------------+------+-----+---------+-------+
| Field | Type | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
ID	int	NO	PRI	NULL	
name	varchar(30)	YES		NULL	
info	varchar(50)	YES		NULL	
+-------+-------------+------+-----+---------+-------+
3 rows in set (0,02 sec)
mysql> INSERT INTO personnel VALUES
-> (1,'Mattila Matti','Esquire'),
-> (2,'Pekkala Pekka','Hunter'),
-> (3,'Jussila Jussi','Lord');
Query OK, 3 rows affected (0,06 sec)
Records: 3 Duplicates: 0 Warnings: 0
mysql> SELECT * FROM personnel;
+----+--------------+---------+
| ID | name | info |
+----+--------------+---------+
1	Mattila Matti	Esquire
2	Pekkala Pekka	Hunter
3	Jussila Jussi	Lord
+----+--------------+---------+
3 rows in set (0,00 sec)
mysql>


Activities
Firefox Web Browser
maalis 25 11:53
How To Install Linux, Apa
Apache2 Ubuntu Default Pa
PHP 8.1.2-1ubuntu2.14 - php
localhost/info.php
PHP Version 8.1.2-1ubuntu2.14
php
System
Linux asadul-VirtualBox 6.5.0-21-generic #21~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Fri Feb 9 13:32:52 UTC 2 x86_64
Build Date
Aug 18 2023 11:41:11
Build System
Linux
Server API
Apache 2.0 Handler
Virtual Directory Support
disabled
Configuration File (php.ini) Path
/etc/php/8.1/apache2
Loaded Configuration File
/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files
/etc/php/8.1/apache2/conf.d
Additional .ini files parsed
/etc/php/8.1/apache2/conf.d/10-mysqlnd.ini, /etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/conf.d/10-pdo.ini, /etc/php/8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20-ctype.ini, /etc/php/8.1/apache2/conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-ffi.ini, /etc/php/8.1/apache2/conf.d/20-fileinfo.ini, /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/20-gd.ini, /etc/php/8.1/apache2/conf.d/20-gettext.ini, /etc/php/8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-mysqli.ini, /etc/php/8.1/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.1/apache2/conf.d/20-phar.ini, /etc/php/8.1/apache2/conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /etc/php/8.1/apache2/conf.d/20-shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.1/apache2/conf.d/20-sysvsem.ini, /etc/php/8.1/apache2/conf.d/20-sysvshm.ini, /etc/php/8.1/apache2/conf.d/20-tokenizer.ini
PHP API
20210902
PHP Extension
20210902
Zend Extension
420210902
Zend Extension Build
API420210902,NTS
PHP Extension Build
API20210902,NTS
Debug Build
no
Thread Safety
disabled
Zend Signal Handling
enabled
Zend Memory Manager
enabled
Zend Multibyte Support
disabled
IPv6 Support
enabled
DTrace Support
available, disabled
Registered PHP Streams
https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports
tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters
zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*
This program makes use of the Zend Scripting Language Engine:
Zend Engine v4.1.2, Copyright (c) Zend Technologies
with Zend OPcache v8.1.2-1ubuntu2.14, Copyright (c), by Zend Technologies
zend engine
Configuration


Ubuntu 22 User [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
maalis 25 12:00
How To Install Linux, Apa
Apache2 Ubuntu Default Pa
PHP 8.1.2-1ubuntu2.14 - php
localhost/sqltest.php
localhost/sqltest.php
LASTNAME FIRSTNAME Personnel File
1 Mattila Matti Esquire
2 Pekkala Pekka Hunter
3 Jussila Jussi Lord


Ubuntu 22 User [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
maalis 25 12:02
How To Install Linux, Apa
Apache2 Ubuntu Default Pa
PHP 8.1.2-1ubuntu2.14 - php
localhost/sqltest.php
localhost/sqltest.php
MIZAN ASAD-UL- Personnel File
1 Mattila Matti Esquire
2 Pekkala Pekka Hunter
3 Jussila Jussi Lord