CS 547

Week 3 Day 1

PHP & MySQL

Agenda

- Web Forms
- MySQL

Announcements

Review

- From previous lecture
- Web forms

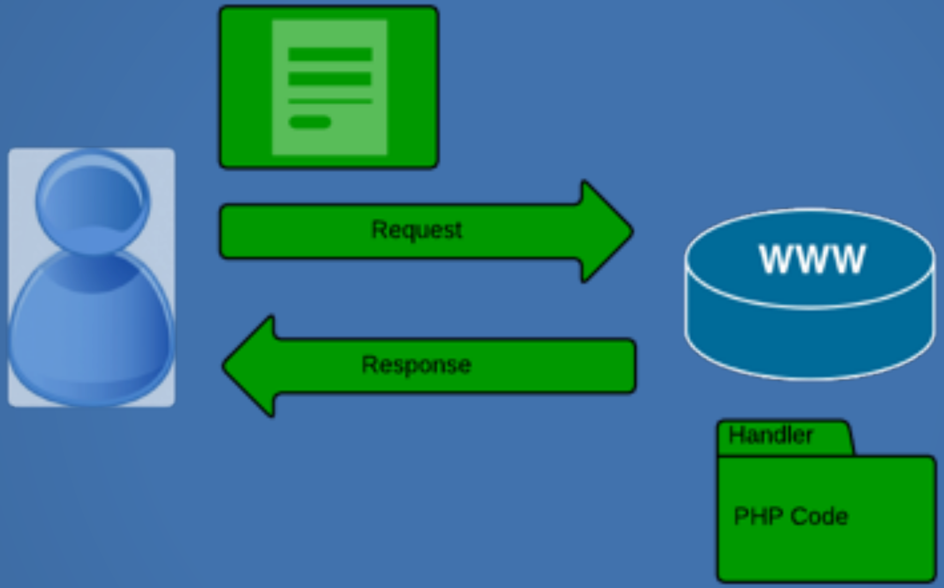
PHP Language

-

Today:

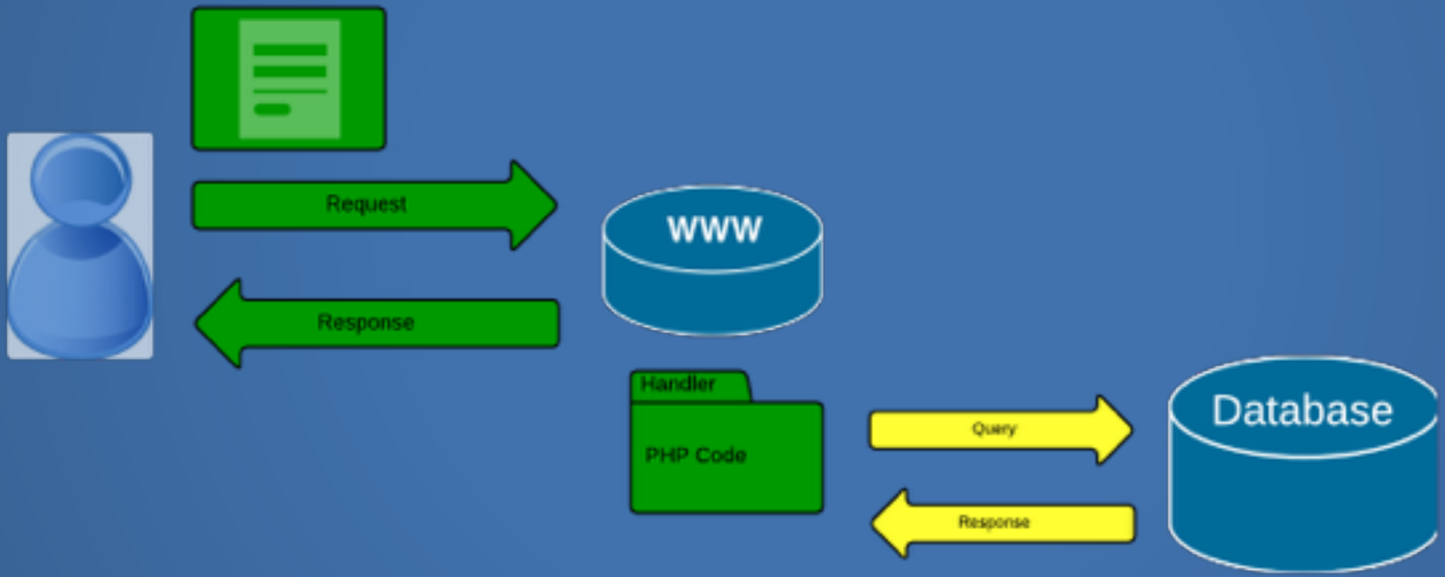
MySQL

Recall

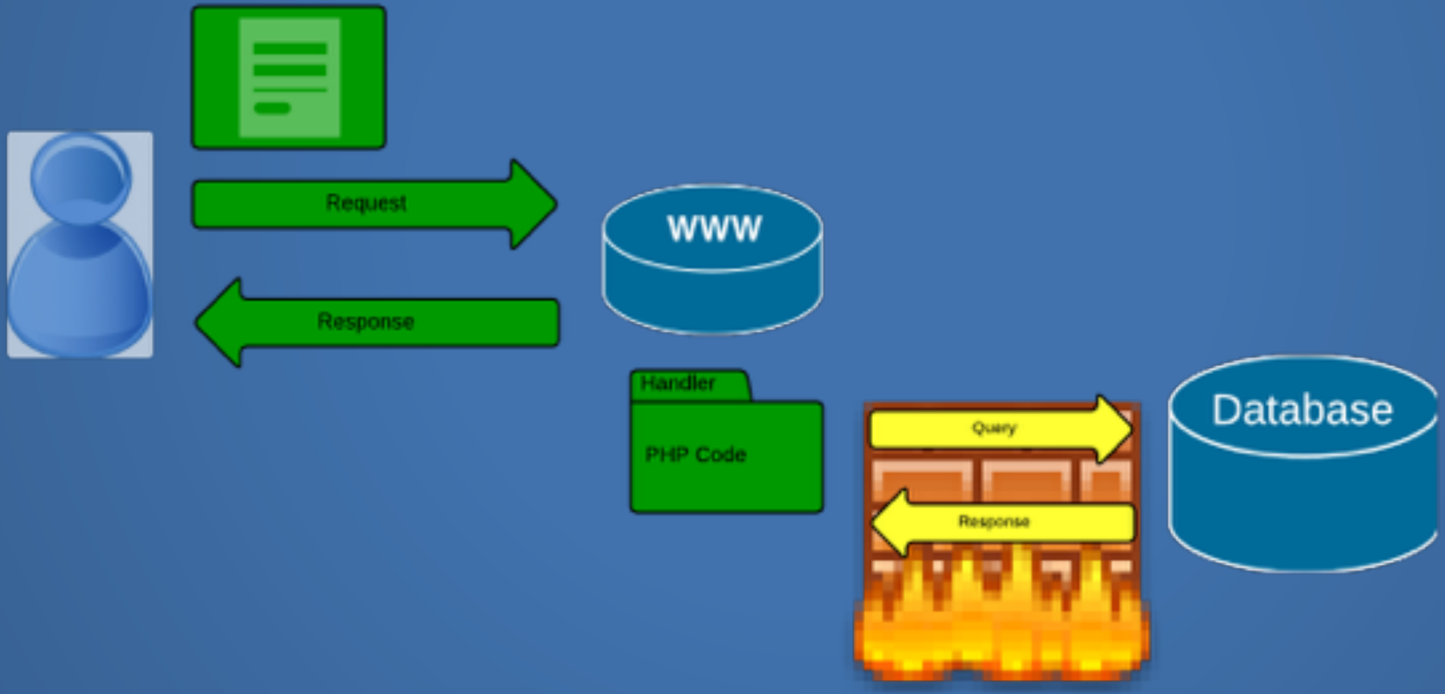


Dynamic Web Sites

- Introduce a new ‘layer’ to the picture

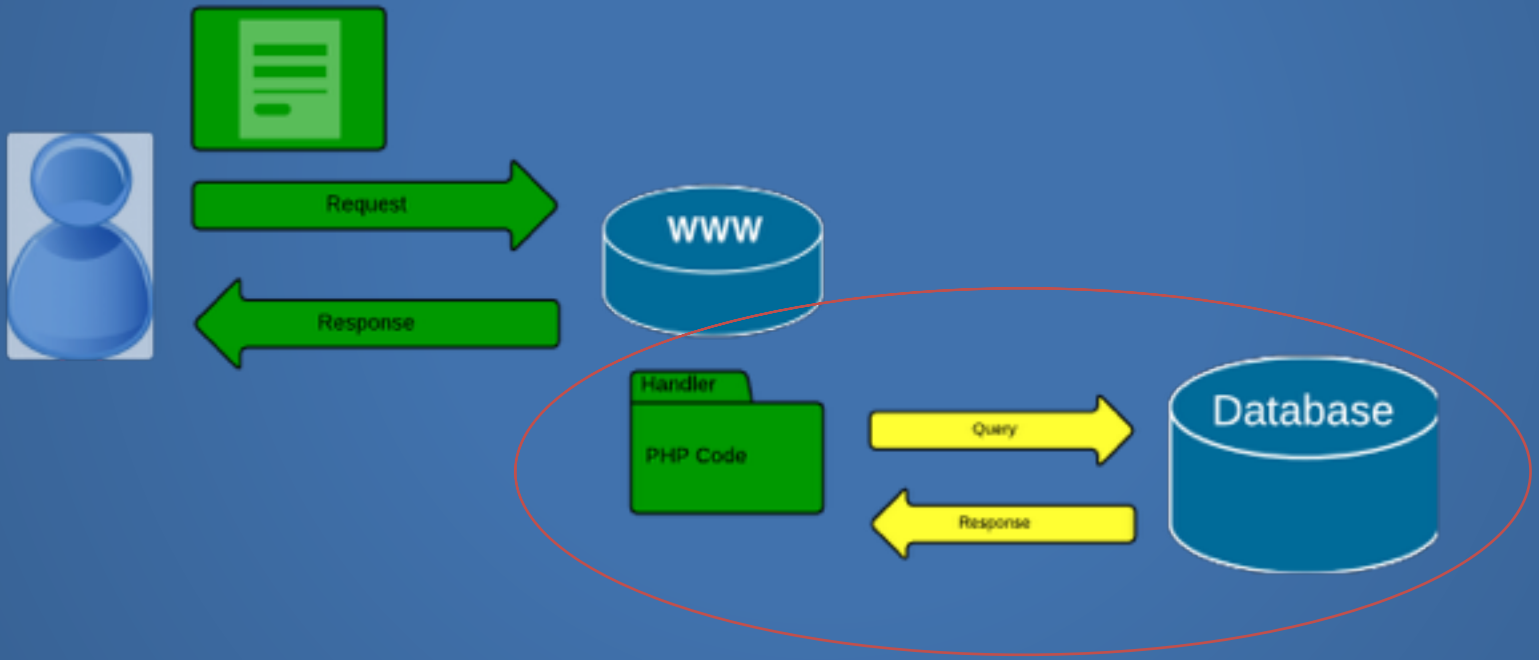


Dynamic Web

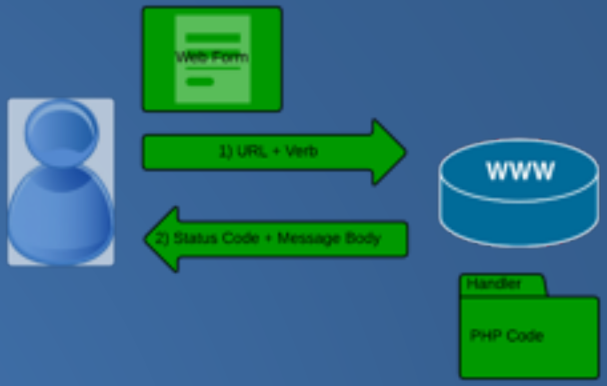

Request
Response
WWW
Handler
PHP Code
Query
Response
Database

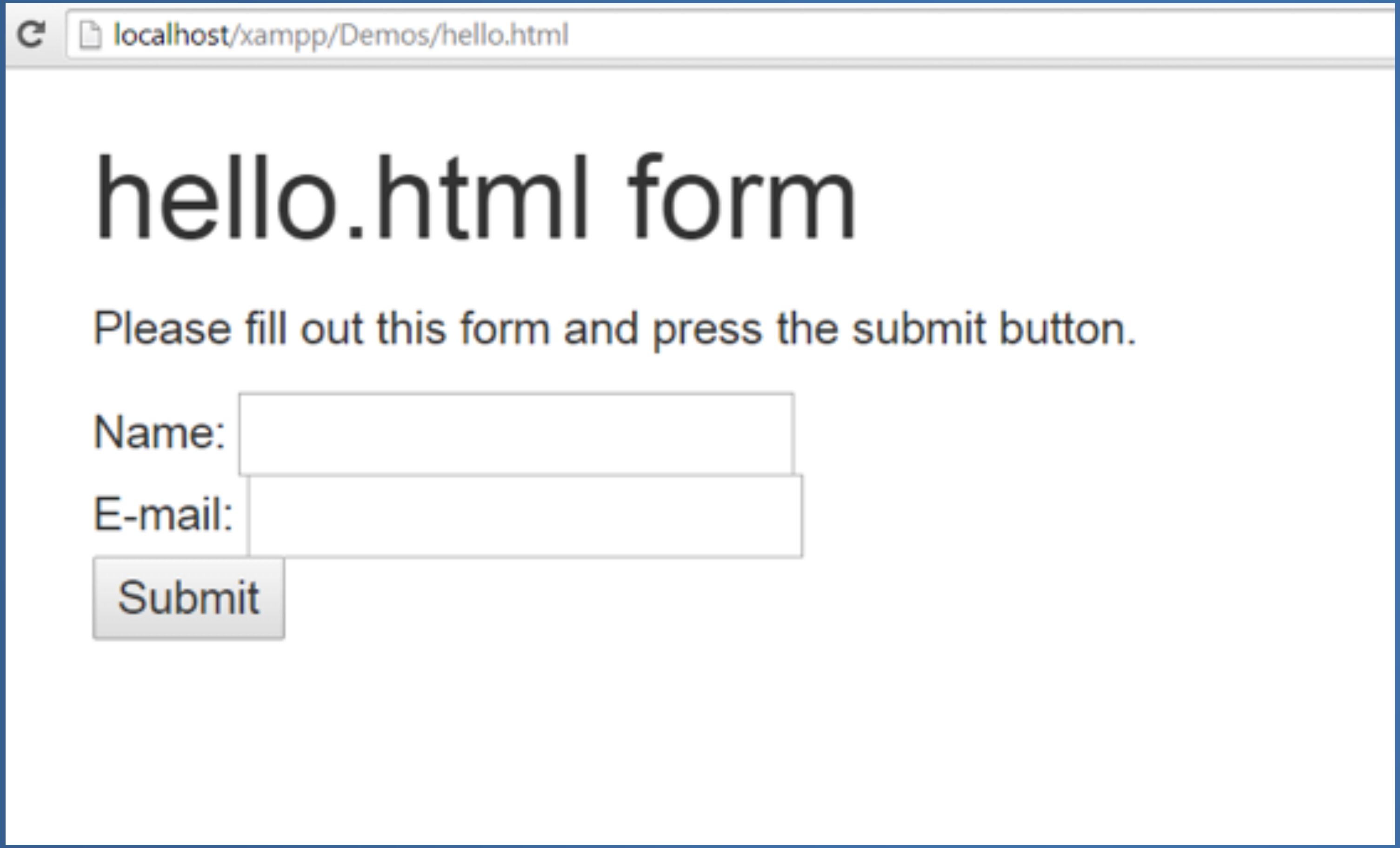
Dynamic Web Sites

- We will focus on the php – database part



HTML Forms




localhost/xampp/Demos/hello.html

hello.html form

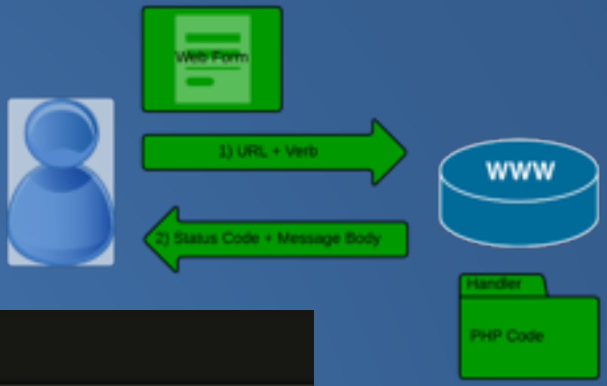
Please fill out this form and press the submit button.

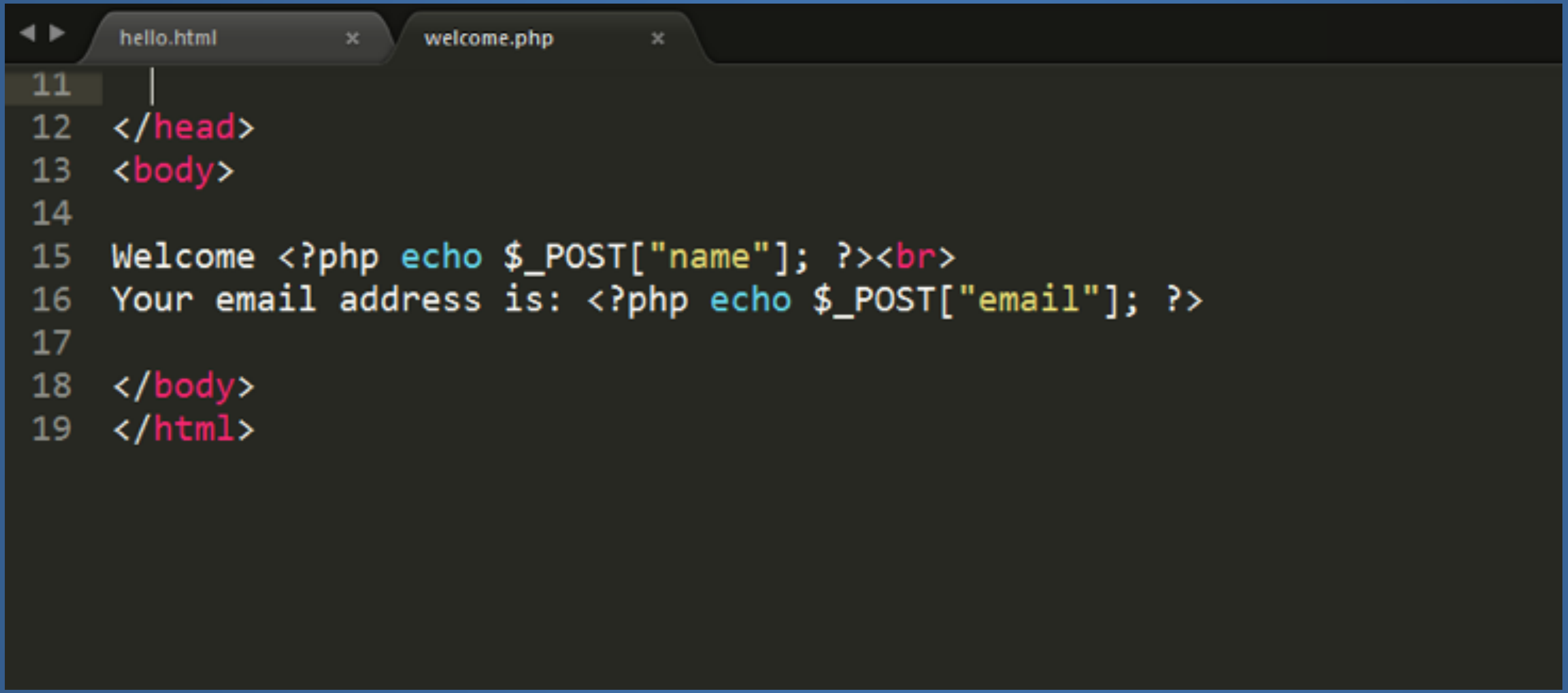
Name:

E-mail:

Submit

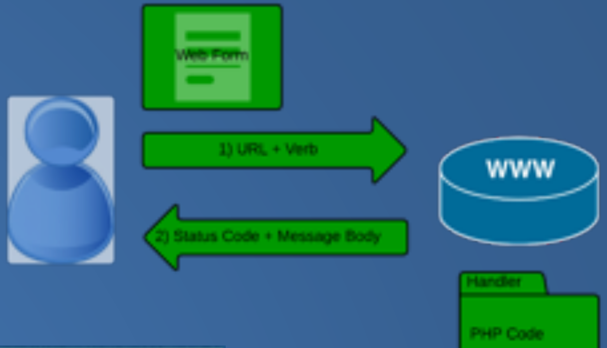
Form Handler

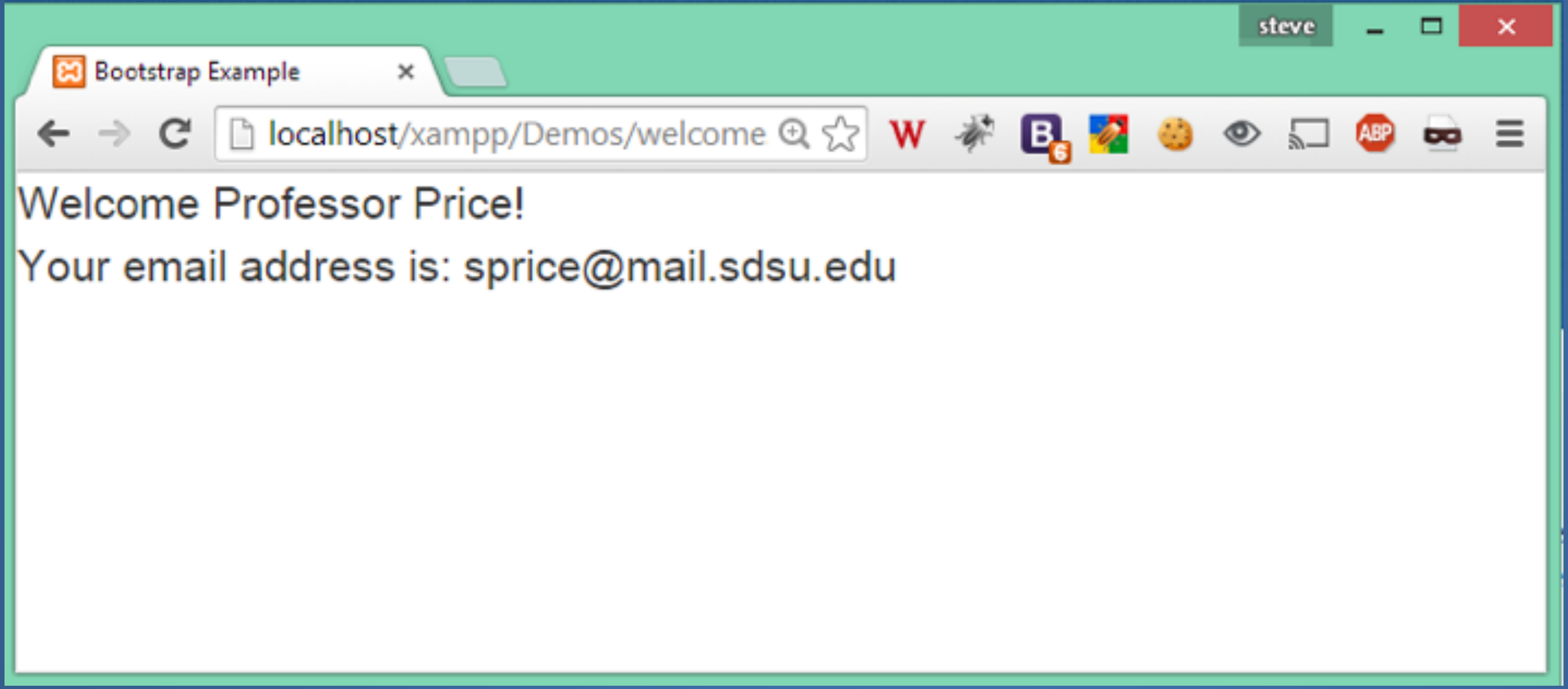




```

</head>
<body>

Welcome <?php echo $_POST["name"]; ?><br>
Your email address is: <?php echo $_POST["email"]; ?>

</body>
</html>
```

Form Handler output



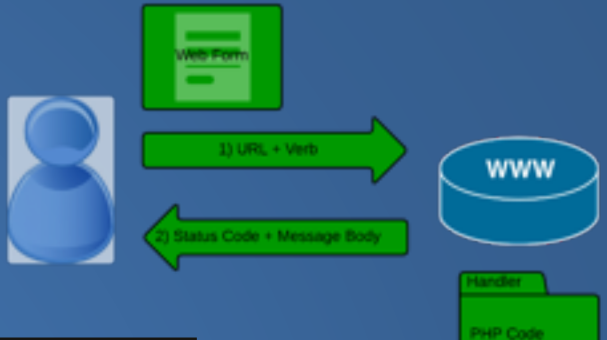


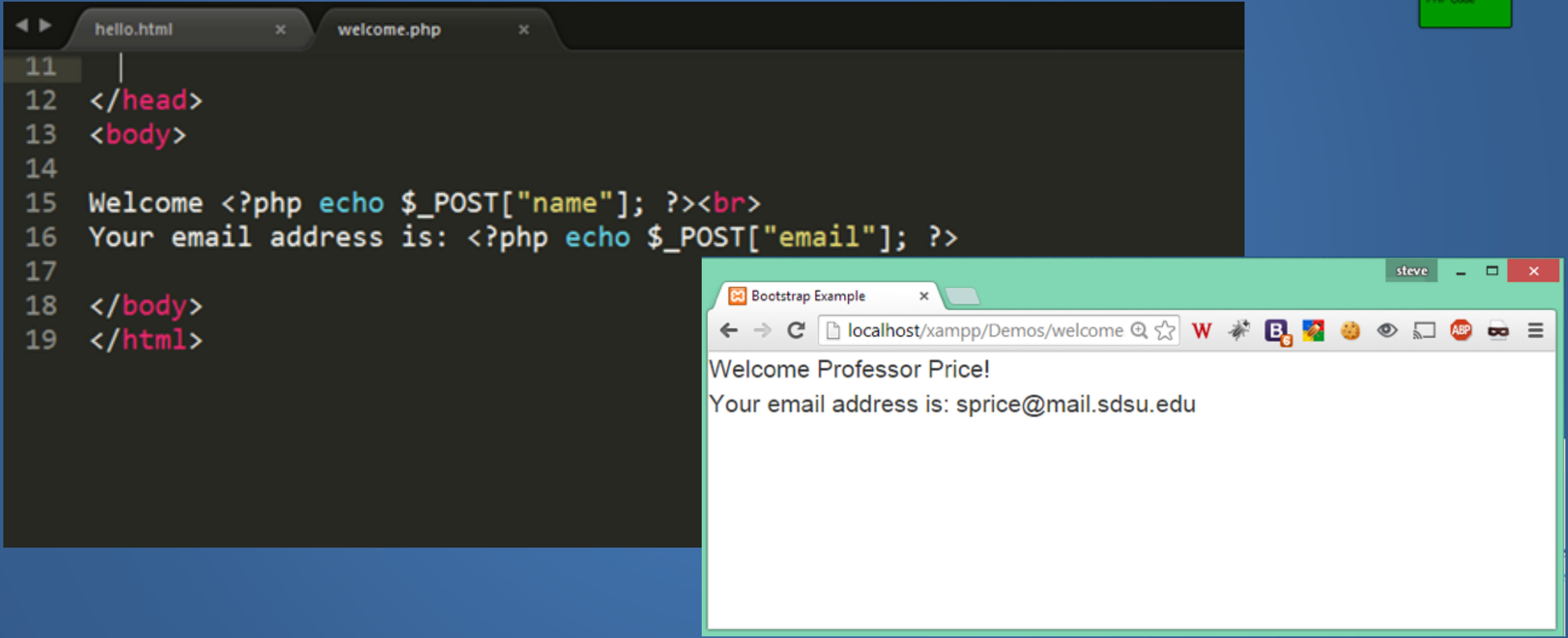
Welcome Professor Price!

Your email address is: sprice@mail.sdsu.edu

Form Handler output

Great! But how does it work?



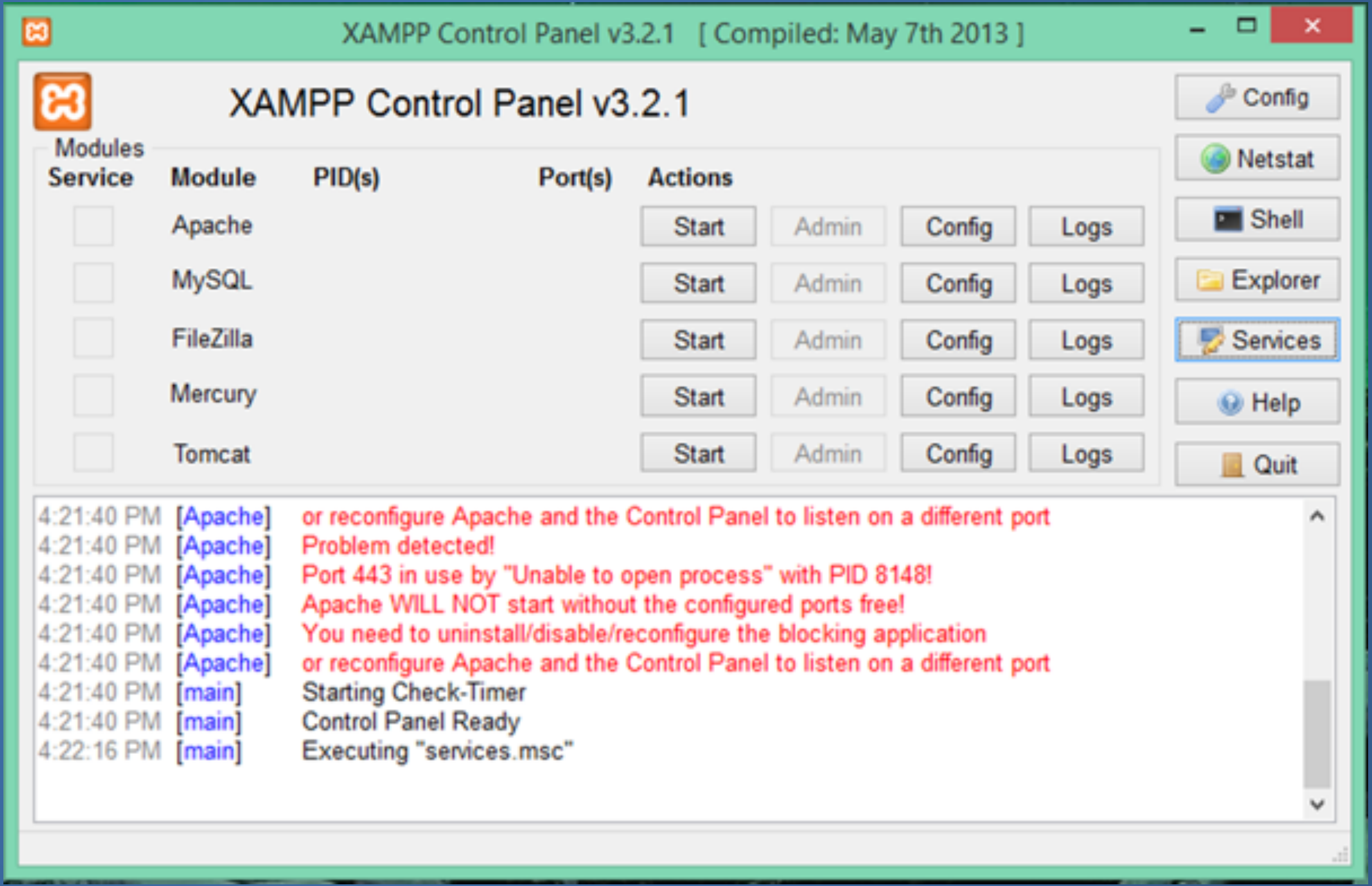


```

</head>
<body>

Welcome <?php echo $_POST["name"]; ?><br>
Your email address is: <?php echo $_POST["email"]; ?>

</body>
</html>
```

```
Welcome Professor Price!
Your email address is: sprice@mail.sdsu.edu
```

XAMPP

- Open the XAMPP Control Panel



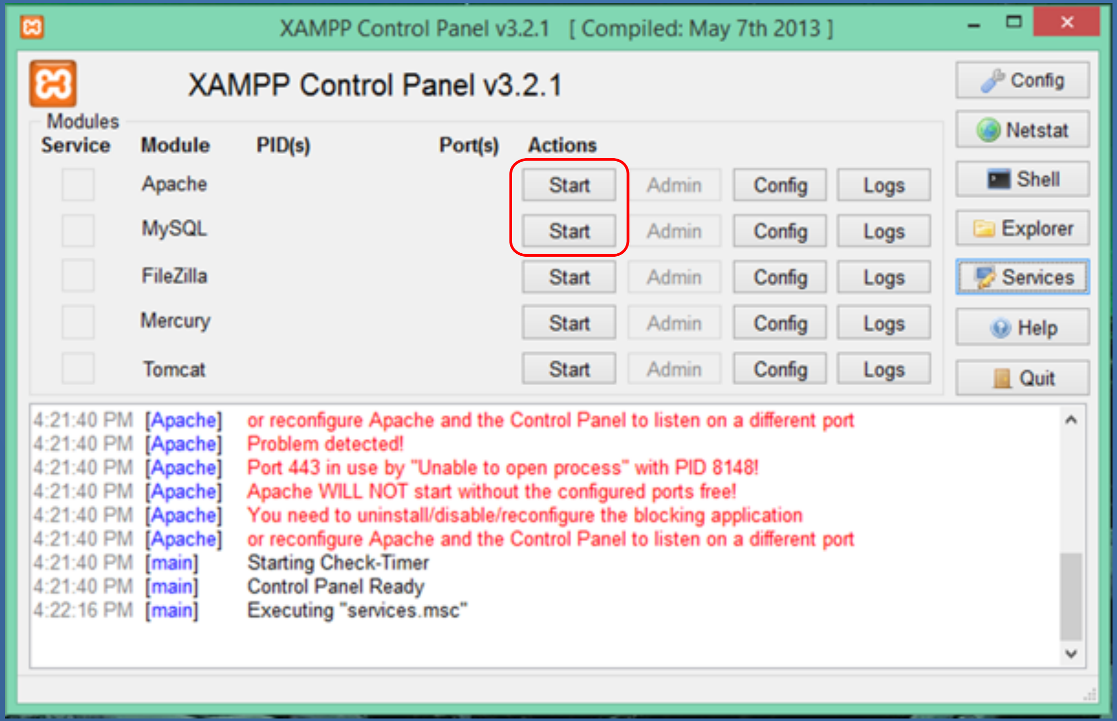
XAMPP Control Panel v3.2.1 [Compiled: May 7th 2013]

XAMPP Control Panel v3.2.1

```
4:21:40 PM  [Apache]   or reconfigure Apache and the Control Panel to listen on a different port
4:21:40 PM  [Apache]   Problem detected!
4:21:40 PM  [Apache]   Port 443 in use by "Unable to open process" with PID 8148!
4:21:40 PM  [Apache]   Apache WILL NOT start without the configured ports free!
4:21:40 PM  [Apache]   You need to uninstall/disable/reconfigure the blocking application
4:21:40 PM  [Apache]   or reconfigure Apache and the Control Panel to listen on a different port
4:21:40 PM  [main]     Starting Check-Timer
4:21:40 PM  [main]     Control Panel Ready
4:22:16 PM  [main]     Executing "services.msc"
```

Start using the tools

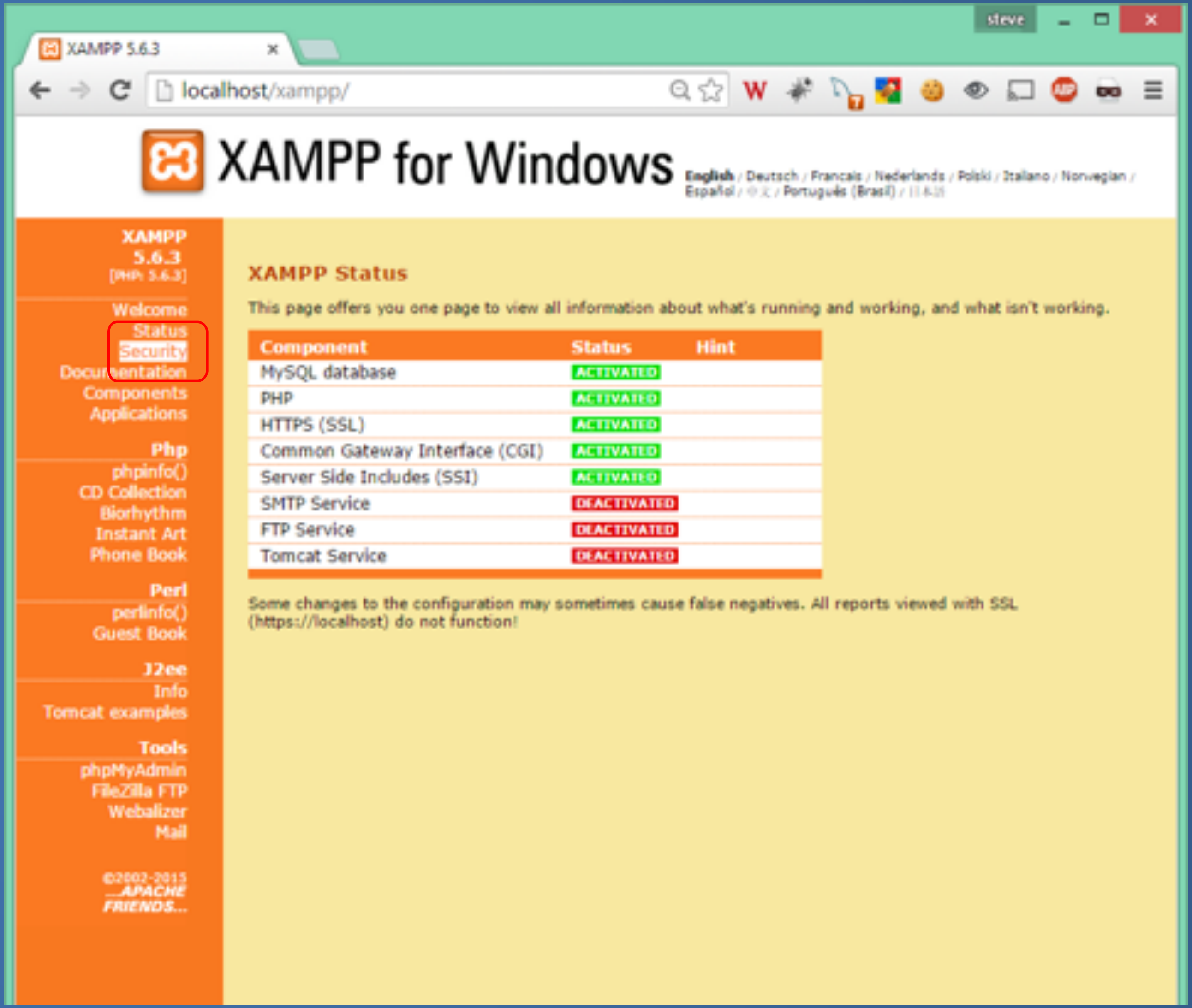
1. Start the xamp control panel
2. Start the apache web service



```
4:21:40 PM  [Apache]   or reconfigure Apache and the Control Panel to listen on a different port
4:21:40 PM  [Apache]   Problem detected!
4:21:40 PM  [Apache]   Port 443 in use by "Unable to open process" with PID 8148!
4:21:40 PM  [Apache]   Apache WILL NOT start without the configured ports free!
4:21:40 PM  [Apache]   You need to uninstall/disable/reconfigure the blocking application
4:21:40 PM  [Apache]   or reconfigure Apache and the Control Panel to listen on a different port
4:21:40 PM  [main]     Starting Check-Timer
4:21:40 PM  [main]     Control Panel Ready
4:22:16 PM  [main]     Executing "services.msc"
```

XAMPP

1. Navigate to the main page
2. Click the Security link



XAMPP for Windows

English / Deutsch / Francais / Nederlands / Polski / Italiano / Norwegian / Español / 中文 / Português (Brasil) / 日本語

XAMPP Status

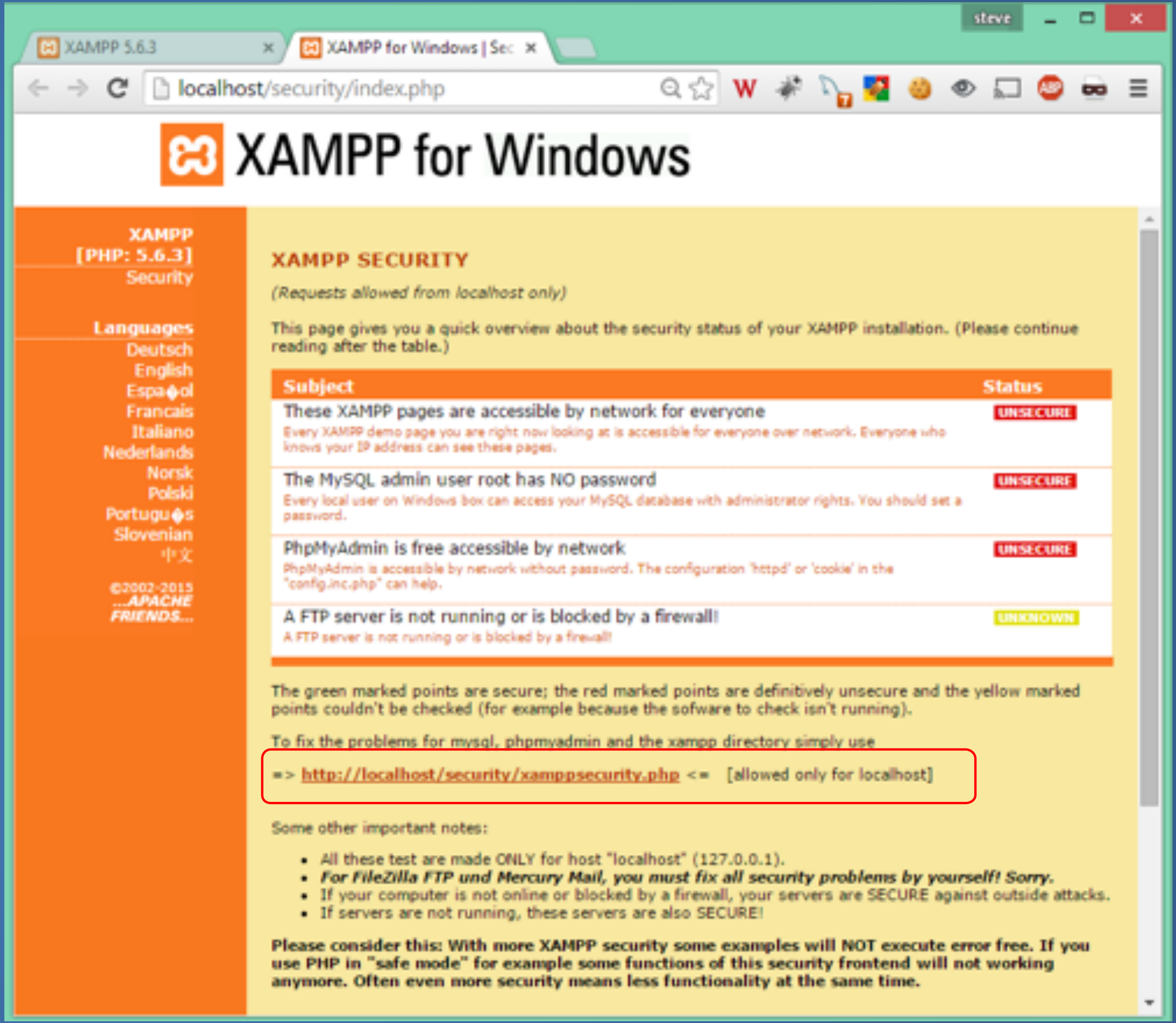
This page offers you one page to view all information about what's running and working, and what isn't working.

Component	Status	Hint
MySQL database	ACTIVATED	
PHP	ACTIVATED	
HTTPS (SSL)	ACTIVATED	
Common Gateway Interface (CGI)	ACTIVATED	
Server Side Includes (SSI)	ACTIVATED	
SMTP Service	DEACTIVATED	
FTP Service	DEACTIVATED	
Tomcat Service	DEACTIVATED	

Some changes to the configuration may sometimes cause false negatives. All reports viewed with SSL (https://localhost) do not function!

XAMPP

1. This will open a new tab
2. Click the xamppsecurity.php link to secure your system



XAMPP for Windows

XAMPP SECURITY

(Requests allowed from localhost only)

This page gives you a quick overview about the security status of your XAMPP installation. (Please continue reading after the table.)

Subject	Status
These XAMPP pages are accessible by network for everyone Every XAMPP demo page you are right now looking at is accessible for everyone over network. Everyone who knows your IP address can see these pages.	UNSECURE
The MySQL admin user root has NO password Every local user on Windows box can access your MySQL database with administrator rights. You should set a password.	UNSECURE
PhpMyAdmin is free accessible by network PhpMyAdmin is accessible by network without password. The configuration 'httpd' or 'cookie' in the "config.inc.php" can help.	UNSECURE
A FTP server is not running or is blocked by a firewall! A FTP server is not running or is blocked by a firewall!	UNKNOWN

The green marked points are secure; the red marked points are definitively unsecure and the yellow marked points couldn't be checked (for example because the sofware to check isn't running).

To fix the problems for mysql, phpmyadmin and the xampp directory simply use

=> http://localhost/security/xamppsecurity.php <= [allowed only for localhost]

Some other important notes:

- All these test are made ONLY for host "localhost" (127.0.0.1).
- ***For FileZilla FTP und Mercury Mail, you must fix all security problems by yourself! Sorry.***
- If your computer is not online or blocked by a firewall, your servers are SECURE against outside attacks.
- If servers are not running, these servers are also SECURE!

Please consider this: With more XAMPP security some examples will NOT execute error free. If you use PHP in "safe mode" for example some functions of this security frontend will not working anymore. Often even more security means less functionality at the same time.

XAMPP demo

PHP MySQL

Two correct ways for connecting to database

MySQLi – mysq

PDO – php Data Objects

PHP MySQL

One old – Deprecated way

mysql_* api

Which api to use?

MySQLi

Simpler syntax

Richer API

PDO

Faster

Supports other Databases

Object Oriented

MySQLi example

```
<?php

$DBServer = 'localhost'; // e.g 'localhost' or '192.168.1.100'
$DBUser   = 'root';
$DBPass   = '';
$DBName   = 'test547';


$
```

MySQLi example

```
$conn = new mysqli($DBServer, $DBUser, $DBPass, $DBName);

// check connection
if ($conn->connect_error) {
trigger_error('Database connection failed: '  . $conn->connect_error, E_USER_ERROR);
}
else {
echo ("Connection Success.");
```

MySQLi example

```
                 echo ("Connection Success.");

$sql='SELECT * FROM mytable ';  // $sql='SELECT * FROM mytable WHERE condition';

                                 $rs=$conn->query($sql);
               /*                    if($rs === false) {
    trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
                                          } else {
                                      $rows_returned = $rs->num_rows;
                                 echo ("Found " + $rows_returned + " rows.");
                                              $rs->data_seek(0);
```

MySQLi example

```
                                                            <?php

                     $DBServer = 'localhost'; // e.g 'localhost' or '192.168.1.100'
                                                  $DBUser   = 'root';
                                                     $DBPass   = '';
                                               $DBName   = 'test547';


                $conn = new mysqli($DBServer, $DBUser, $DBPass, $DBName);

                                                  // check connection
                                            if ($conn->connect_error) {
trigger_error('Database connection failed: '  . $conn->connect_error, E_USER_ERROR);
                                                               }
                                                            else {
                                                                               echo ("Connection Success.");

                                    $sql='SELECT * FROM mytable ';  // $sql='SELECT * FROM mytable WHERE condition';

                                                                                   $rs=$conn->query($sql);
              /*                                                                         if($rs === false) {
                                             trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
                                                                                                 } else {
                                                                                                                  $rows_returned = $rs->num_rows;
```

MySQLi example

```
<?php

$DBServer = 'localhost'; // e.g 'localhost' or '192.168.1.100'
$DBUser   = 'root';
$DBPass   = '';
$DBName   = 'test547';


$conn = new mysqli($DBServer, $DBUser, $DBPass, $DBName);

// check connection
if ($conn->connect_error) {
trigger_error('Database connection failed: '  . $conn->connect_error, E_USER_ERROR);
}
else {
        echo ("Connection Success.");

$sql='SELECT * FROM mytable ';  // $sql='SELECT * FROM mytable WHERE condition';

        $rs=$conn->query($sql);
/*      if($rs === false) {
trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
        } else {
                $rows_returned = $rs->num_rows;
```

MySQLi examples

- **Iterate over recordset**
- **Using MYSQLI_ASSOC an associated array is returned, MYSQLI_NUM an enumerated one and MYSQLI_BOTH both of them.**
- **WARNING: fetch_all is available only with MySQL Native Driver.**

More sql commands

Record count

```
$rows_returned = $rs->num_rows;
```

Move inside recordset

```
$rs->data_seek(10);
```

Free memory

Optional:

```
$rs->free();
```

SQL Insert Command

```
$v1="'" . $conn->real_escape_string('col1_value') . "'";

$sql="INSERT INTO mytable (name_varchar, age_number) VALUES ($v1,10)";

if($conn->query($sql) === false) {
trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
} else {
$last_inserted_id = $conn->insert_id;
$affected_rows = $conn->affected_rows;
}
```

SQL UpDate

```
$v1="'" . $conn->real_escape_string('col1_value') . "'";

$sql="UPDATE mytable SET name_varchar=$v1, age_number=21 WHERE id>10";

if($conn->query($sql) === false) {
trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
} else {
$affected_rows = $conn->affected_rows;
}
```

SQL Delete

```
$sql="DELETE FROM mytable WHERE id>10";

if($conn->query($sql) === false) {
trigger_error('Wrong SQL: ' . $sql . ' Error: ' . $conn->error, E_USER_ERROR);
} else {
$affected_rows = $conn->affected_rows;
}
```

SQL Pattern

The commands we just covered follow a common pattern: CRUD

C == Create a record

R == Read one or more records

U == UpDate one or more records

D == Delete a records

This CRUD pattern is at the heart of modern dynamic websites.