Bases de données

Prof. Giovanna Di Marzo Serugendo
Assistant: Mohammad Parhizkar
mohammad.parhizkar@unige.ch

Semestre: printemps 2019

session 5- 25/03/2019

Agenda

25 / 03 / 2019

- PL/SQL Tables
- PL/SQL Exception handling (Error Management)

- Introduction to MySQL
- Introduction to PHP
- XAMPP, MAMP
- TP5

PL/SQL Tables

- PL/SQL tables help you **move bulk data**. So, PL/SQL tables make it easy to move collections of data into and out of database tables or between client-side applications and stored subprograms.

Syntax:

```
TYPE table_type_name IS TABLE OF datatype [NOT NULL]
   INDEX BY BINARY_INTEGER;
```

The INDEX BY clause must specify datatype BINARY_INTEGER, which has a magnitude range of -2147483647 .. 2147483647.

Example:

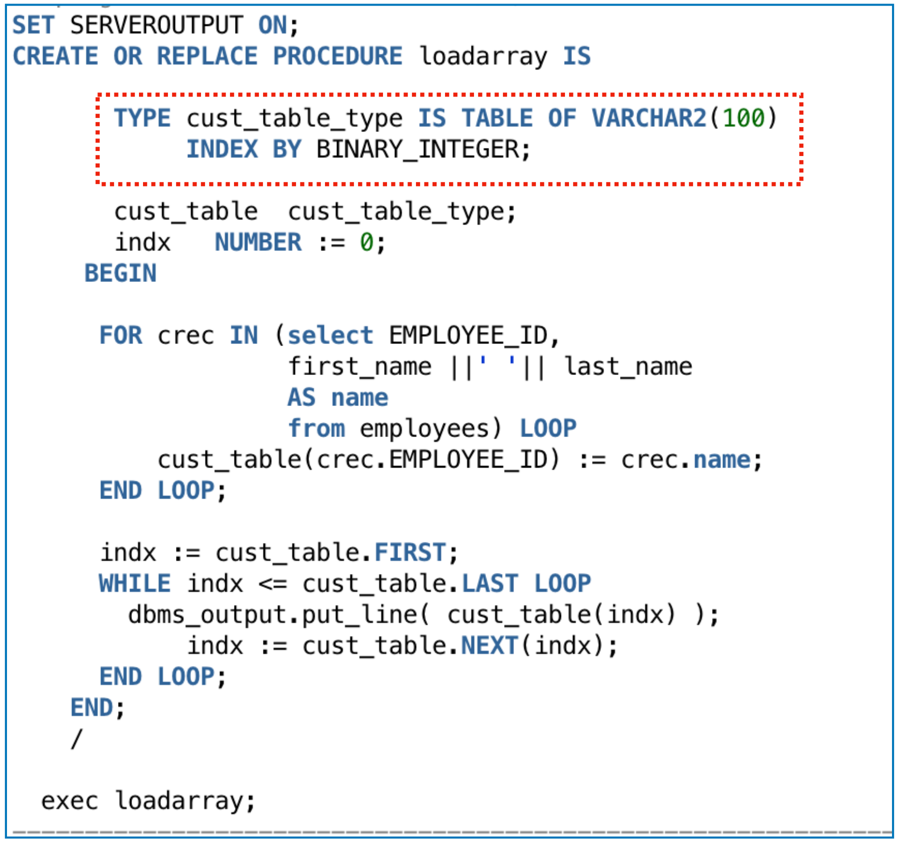
```
DECLARE
 TYPE EnameTabTyp IS TABLE OF employees.first_name%TYPE NOT NULL
    INDEX BY BINARY_INTEGER;
```

To specify the element type, you can use %TYPE to provide the datatype of a variable or database column.

You can add the NOT NULL constraint to a TABLE type definition and so prevent the storing of nulls in PL/SQL tables of that type.

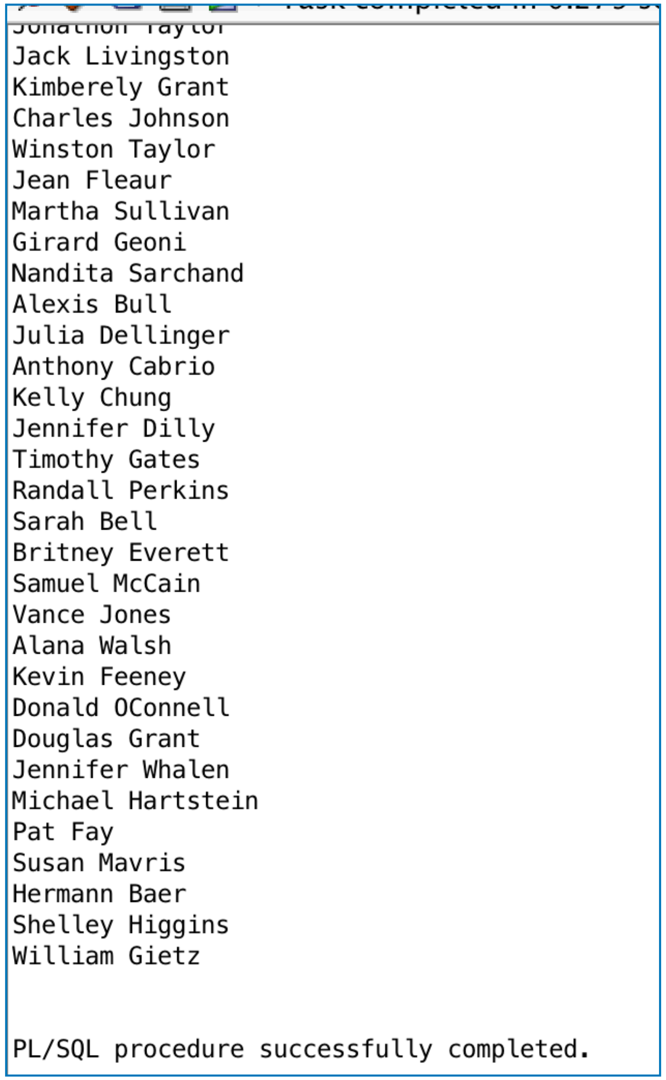
PL/SQL Tables

Example:



```
SET SERVEROUTPUT ON;
CREATE OR REPLACE PROCEDURE loadarray IS

      TYPE cust_table_type IS TABLE OF VARCHAR2(100)
           INDEX BY BINARY_INTEGER;

      cust_table  cust_table_type;
      indx    NUMBER := 0;
     BEGIN

      FOR crec IN (select EMPLOYEE_ID,
                   first_name ||' '|| last_name
                   AS name
                   from employees) LOOP
         cust_table(crec.EMPLOYEE_ID) := crec.name;
      END LOOP;

      indx := cust_table.FIRST;
      WHILE indx <= cust_table.LAST LOOP
        dbms_output.put_line( cust_table(indx) );
           indx := cust_table.NEXT(indx);
      END LOOP;
    END;
    /

  exec loadarray;
```

Script output:



```
Jack Livingston
Kimberely Grant
Charles Johnson
Winston Taylor
Jean Fleaur
Martha Sullivan
Girard Geoni
Nandita Sarchand
Alexis Bull
Julia Dellinger
Anthony Cabrio
Kelly Chung
Jennifer Dilly
Timothy Gates
Randall Perkins
Sarah Bell
Britney Everett
Samuel McCain
Vance Jones
Alana Walsh
Kevin Feeney
Donald OConnell
Douglas Grant
Jennifer Whalen
Michael Hartstein
Pat Fay
Susan Mavris
Hermann Baer
Shelley Higgins
William Gietz


PL/SQL procedure successfully completed.
```

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PL / SQL Exception Handling

- An exception is an error condition during a program execution.
- Oracle PL/SQL supports users to define different conditions and catch the errors by using **EXCEPTION section** in their program.
- A proper action in the circumstances is taken against the error condition.

PL / SQL Exceptions

a simple example to illustrate the concept

Example:

```
SET SERVEROUTPUT ON;

DECLARE
    emp_id employees.employee_id%TYPE := 4;
    emp_name employees.last_name%TYPE;

 BEGIN
   SELECT  last_name INTO  emp_name
   FROM employees
   WHERE employee_id = emp_id;
   DBMS_OUTPUT.PUT_LINE ('Name: '||  emp_name);

EXCEPTION
   WHEN no_data_found THEN
      dbms_output.put_line('No such employee!');
   WHEN others THEN
      dbms_output.put_line('Just an Error!');
END;
```

```
SET SERVEROUTPUT ON;

DECLARE
    emp_id employees.employee_id%TYPE := 45;
    emp_name employees.last_name%TYPE;

 BEGIN
   SELECT  last_name INTO  emp_name
   FROM employees
   WHERE employee_id = emp_id;
   DBMS_OUTPUT.PUT_LINE ('Name: '||  emp_name);

EXCEPTION
   WHEN no_data_found THEN
      dbms_output.put_line('No such employee!');
   WHEN others THEN
      dbms_output.put_line('Just an Error!');
END;
/
```

EXCEPTION block, which has 2 exceptions to handle

Output:

```
Name: Hanson


PL/SQL procedure successfully completed.
```

Output:

```
No such employee!


PL/SQL procedure successfully completed.
```

Since there is no employee with ID value 45 in our HR database, the program raises the run-time exception **NO_DATA_FOUND**.

The **WHEN OTHERS** clause is used to trap all remaining exceptions that have not been handled.

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PL / SQL Exceptions

example 2 to illustrate the concept

without EXCEPTION section

Example:

```
SET SERVEROUTPUT ON;

DECLARE
    emp_id employees.employee_id%TYPE := -2;
    emp_name employees.last_name%TYPE;

 BEGIN
       SELECT  last_name INTO  emp_name
       FROM employees
       WHERE employee_id = emp_id;
       DBMS_OUTPUT.PUT_LINE ('Name: '||  emp_name);
END;
/
```

Output:

```
Error starting at line : 3 in command -
DECLARE
    emp_id employees.employee_id%TYPE := -2;
    emp_name employees.last_name%TYPE;

 BEGIN
       SELECT  last_name INTO  emp_name
       FROM employees
       WHERE employee_id = emp_id;
       DBMS_OUTPUT.PUT_LINE ('Name: '||  emp_name);
END;
Error report -
ORA-01403: no data found
ORA-06512: at line 6
01403. 00000 -  "no data found"
*Cause:    No data was found from the objects.
*Action:   There was no data from the objects which may be due to end of fetch.
```

Example:

```
SET SERVEROUTPUT ON;

DECLARE
    emp_id employees.employee_id%TYPE := -2;
    emp_name employees.last_name%TYPE;

 -- user-defined exception
    error_id  EXCEPTION;

 BEGIN
   IF emp_id < 1 THEN
       RAISE error_id;
   ELSE
      SELECT  last_name INTO  emp_name
      FROM employees
      WHERE employee_id = emp_id;
      DBMS_OUTPUT.PUT_LINE ('Name: '||  emp_name);
  END IF;

   EXCEPTION
      WHEN error_id THEN
        dbms_output.put_line('ID must be greater (or equal) than 1!');

      WHEN no_data_found THEN
        dbms_output.put_line('No such employee!');
      WHEN others THEN
        dbms_output.put_line('Just an Error!');
END;
/
```

Step 1 ← `error_id  EXCEPTION;`

Step 2 ← `IF emp_id < 1 THEN RAISE error_id;`

Step 3 ← `WHEN error_id THEN dbms_output.put_line('ID must be greater (or equal) than 1!');`

Output:

```
ID must be greater (or equal) than 1!


PL/SQL procedure successfully completed.
```

UNIVERSITÉ DE GENÈVE

CENTRE UNIVERSITAIRE D'INFORMATIQUE

PL / SQL Exception Types

1. System-defined Exceptions

They are named in the STANDARD package in PL/SQL and do not need to be defined by the programmer.

2. User-defined Exceptions

Sometimes, it is necessary for programmers to name and trap their own exceptions ones that are not defined already by PL/SQL.

In this section, we will also see ….

- `SQLCODE Function`
- `SQLERRM Function`

PL / SQL Exception Handling

System-defined Exceptions

The syntax for the Named System Exception in a procedure:

```
CREATE [OR REPLACE] PROCEDURE procedure_name
   [ (parameter [,parameter]) ]
IS
   [declaration_section]

BEGIN
   executable_section

EXCEPTION
   WHEN exception_name1 THEN
      [statements]

   WHEN exception_name2 THEN
      [statements]

   WHEN exception_name_n THEN
      [statements]

   WHEN OTHERS THEN
      [statements]

END [procedure_name];
```

EXCEPTION block

Here you list down the exceptions you want to handle.

The syntax for the Named System Exception in a function:

```
CREATE [OR REPLACE] FUNCTION function_name
   [ (parameter [,parameter]) ]
   RETURN return_datatype
IS | AS
   [declaration_section]

BEGIN
   executable_section

EXCEPTION
   WHEN exception_name1 THEN
      [statements]

   WHEN exception_name2 THEN
      [statements]

   WHEN exception_name_n THEN
      [statements]

   WHEN OTHERS THEN
      [statements]

END [function_name];
```

EXCEPTION block

https://docs.oracle.com/cd/E11882_01/timesten.112/e21639/exceptions.htm#TTPLS194

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PL / SQL Exception Handling

Named System Exceptions

Example:

```
--Here is an example (from TP1) of a procedure that uses a Named System Exception:

CREATE OR REPLACE PROCEDURE add_new_director
   (director_id_in IN NUMBER, director_first_name_in IN VARCHAR2, director_last_name_in IN VARCHAR2,
   director_BD_in IN DATE, country_in IN VARCHAR2 )
IS

BEGIN
   INSERT INTO directors (director_id, director_first_name, director_last_name, director_BD, country)
   VALUES ( supplier_id_in, director_first_name_in,director_last_name_in , director_BD_in,country_in );

EXCEPTION
   WHEN DUP_VAL_ON_INDEX THEN
      raise_application_error (-20001,'You have tried to insert a duplicate director_id.');

   WHEN OTHERS THEN
      raise_application_error (-20002,'An error has occurred inserting a director.');

END;
```

The *raise_application_error* is a system-defined procedure. It allows us to raise an exception and associate an error **number** and **message** with the procedure.

Error numbers are defined between -20,000 and -20,999

DUP_VAL_ON_INDEX: You tried to execute an INSERT or UPDATE statement that has created a duplicate value in a field restricted by a unique index.

https://docs.oracle.com/cd/E11882_01/timesten.112/e21639/exceptions.htm#TTPLS194

PL / SQL Exception Handling

Named programer-defined Exceptions

The syntax for the Named Programmer-Defined Exception in a procedure:

```
CREATE [OR REPLACE] PROCEDURE procedure_name
    [ (parameter [,parameter]) ]
IS
    [declaration_section]

    exception_name EXCEPTION;        Step 1

BEGIN
    executable_section
    RAISE exception_name;            Step 2

EXCEPTION                            Step 3
    WHEN exception_name THEN
        [statements]

    WHEN OTHERS THEN
        [statements]

END [procedure_name];
```

EXCEPTION block

The syntax for the Named Programmer-Defined Exception in a function:

```
CREATE [OR REPLACE] FUNCTION function_name
    [ (parameter [,parameter]) ]
    RETURN return_datatype

IS | AS
    [declaration_section]

    exception_name EXCEPTION;

BEGIN
    executable_section

    RAISE exception_name;

EXCEPTION
    WHEN exception_name THEN
        [statements]

    WHEN OTHERS THEN
        [statements]

END [function_name];
```

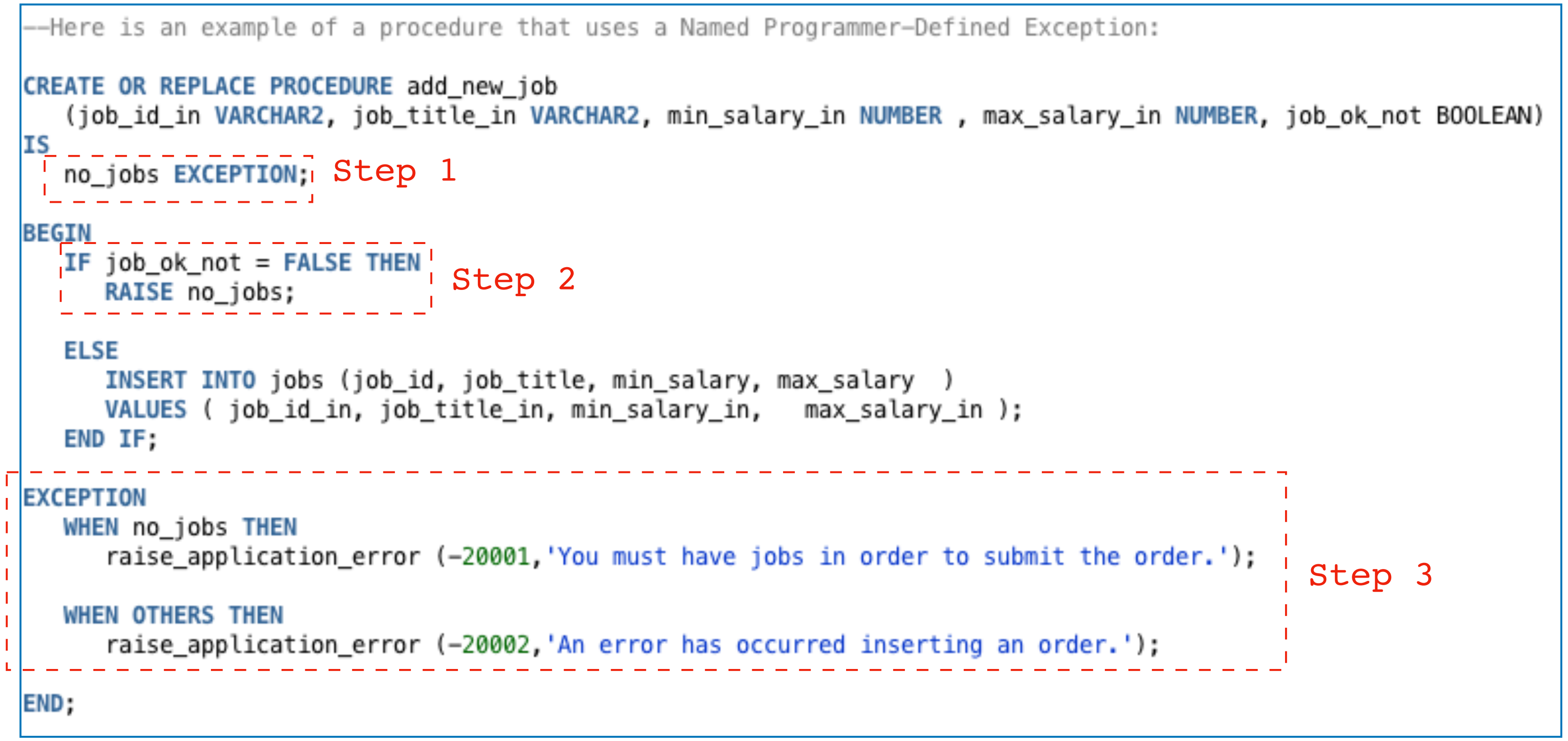
EXCEPTION block

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PL / SQL Exception Handling

Named programer-defined Exceptions

Example:



```
--Here is an example of a procedure that uses a Named Programmer-Defined Exception:

CREATE OR REPLACE PROCEDURE add_new_job
   (job_id_in VARCHAR2, job_title_in VARCHAR2, min_salary_in NUMBER , max_salary_in NUMBER, job_ok_not BOOLEAN)
IS
   no_jobs EXCEPTION;

BEGIN
   IF job_ok_not = FALSE THEN
      RAISE no_jobs;

   ELSE
      INSERT INTO jobs (job_id, job_title, min_salary, max_salary  )
      VALUES ( job_id_in, job_title_in, min_salary_in,   max_salary_in );
   END IF;

EXCEPTION
   WHEN no_jobs THEN
      raise_application_error (-20001,'You must have jobs in order to submit the order.');

   WHEN OTHERS THEN
      raise_application_error (-20002,'An error has occurred inserting an order.');

END;
```

- In this example, we have declared a Named Programmer-Defined Exception called ***no_jobs*** in the declaration statement.
- Now if the ***job_ok_not*** variable contains false, the code will jump directly to the Named Programmer-Defined Exception called ***no_jobs***.
- Finally, we tell our procedure what to do when the ***no_jobs*** exception is encountered by including code in the WHEN clause.

we are also using the **WHEN OTHERS** clause to trap all remaining exceptions…

PL / SQL Exception Handling

WHEN OTHERS Clause Exceptions

The syntax for WHEN OTHERS clause Exception in a procedure:

```
CREATE [OR REPLACE] PROCEDURE procedure_name
   [ (parameter [,parameter]) ]
IS
   [declaration_section]

BEGIN
   executable_section

EXCEPTION
   WHEN exception_name1 THEN
      [statements]

   WHEN exception_name2 THEN
      [statements]

   WHEN exception_name_n THEN
      [statements]

   WHEN OTHERS THEN
      [statements]

END [procedure_name];
```

The syntax for WHEN OTHERS clause Exception in a function:

```
CREATE [OR REPLACE] FUNCTION function_name
   [ (parameter [,parameter]) ]
   RETURN return_datatype
IS | AS
   [declaration_section]

BEGIN
   executable_section

EXCEPTION
   WHEN exception_name1 THEN
      [statements]

   WHEN exception_name2 THEN
      [statements]

   WHEN exception_name_n THEN
      [statements]

   WHEN OTHERS THEN
      [statements]

END [function_name];
```

to trap all remaining exceptions…

SQLCODE function & SQLERRM function

are Oracle's built-in error reporting functions in PL/SQL.

- The **SQLCODE** function returns the **error number** associated with the most recently raised error exception.
- This function should only be used within the Exception Handling section of your code.

Syntax:

```
EXCEPTION
   WHEN exception_name1 THEN
      [statements]

   WHEN exception_name2 THEN
      [statements]

   WHEN exception_name_n THEN
      [statements]

   WHEN OTHERS THEN
      [statements]

END [procedure_name];
```

```
EXCEPTION
   WHEN OTHERS THEN
      raise_application_error(-20001,'An error was encountered - '||SQLCODE||' -ERROR-
'||SQLERRM);
END;
```

The **SQLERRM** function returns the **error message** associated with the most recently raised error exception.

You can put it in your code to see the difference:

```
exception
   when others then
     dbms_output.put_line('SQLCODE: '|| SQLCODE);
     dbms_output.put_line('SQLERRM: '|| SQLERRM);
```

More examples:
https://www.youtube.com/watch?v=01BKhYiZ_sE

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

MySQL

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

for Mac

Uninstall MySQL

You are unable to install a new version of MySQL even you believe you have removed everything about previous versions :(

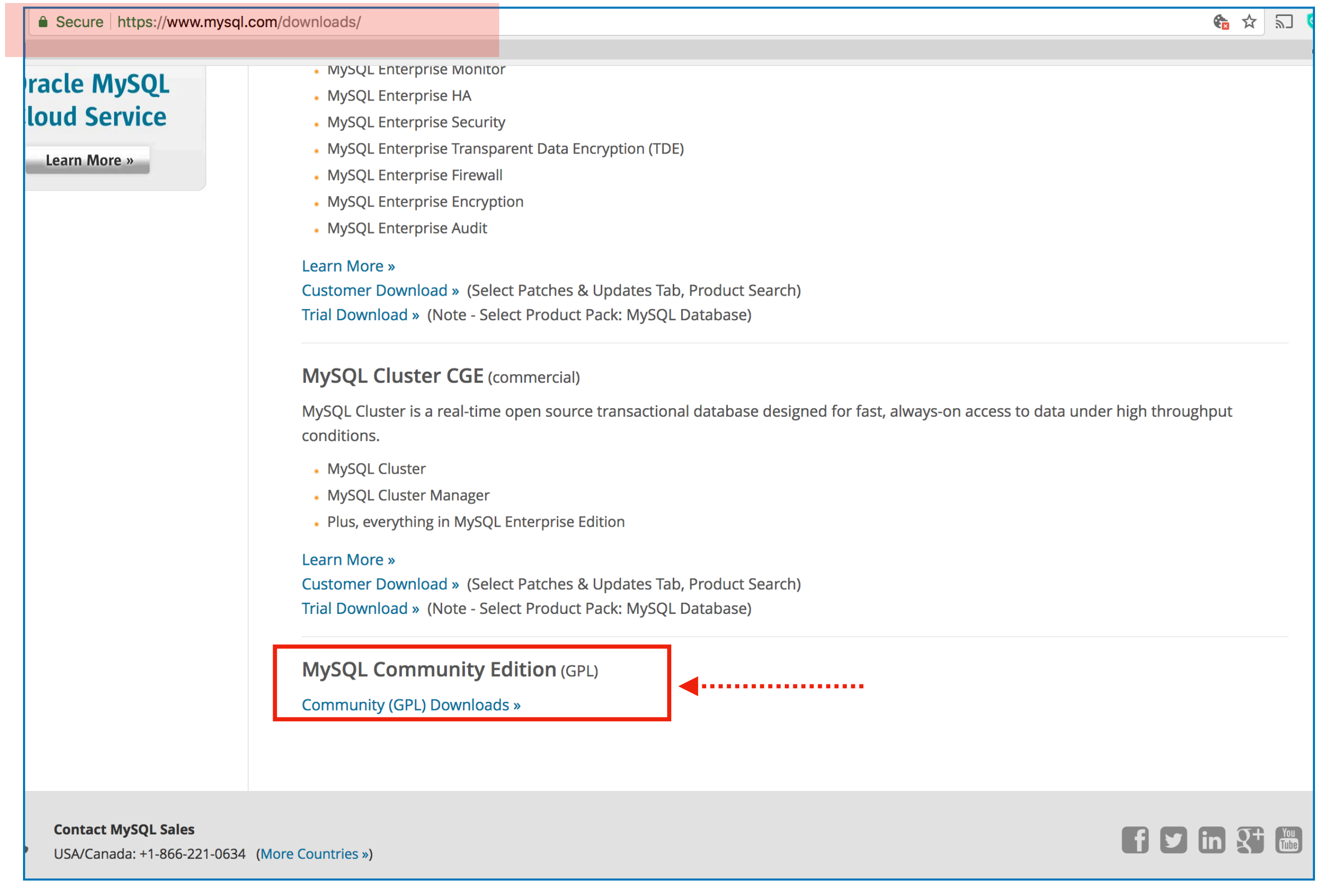
To uninstall MySQL and completely remove it (including all databases) from your Mac, do the following:

Open a terminal window:

- sudo rm /usr/local/mysql
- sudo rm -rf /usr/local/mysql*
- sudo rm -rf /Library/StartupItems/MySQLCOM
- sudo rm -rf /Library/PreferencePanes/My*
- rm -rf ~/Library/PreferencePanes/My*
- sudo rm -rf /Library/Receipts/mysql*
- sudo rm -rf /Library/Receipts/MySQL*
- sudo rm -rf /private/var/db/receipts/*mysql*

Installation

All downloads for MySQL are located at *MySQL Downloads*...


Secure | https://www.mysql.com/downloads/

racle MySQL
loud Service
Learn More »

- MySQL Enterprise Monitor
- MySQL Enterprise HA
- MySQL Enterprise Security
- MySQL Enterprise Transparent Data Encryption (TDE)
- MySQL Enterprise Firewall
- MySQL Enterprise Encryption
- MySQL Enterprise Audit

Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download » (Note - Select Product Pack: MySQL Database)

MySQL Cluster CGE (commercial)

MySQL Cluster is a real-time open source transactional database designed for fast, always-on access to data under high throughput conditions.

- MySQL Cluster
- MySQL Cluster Manager
- Plus, everything in MySQL Enterprise Edition

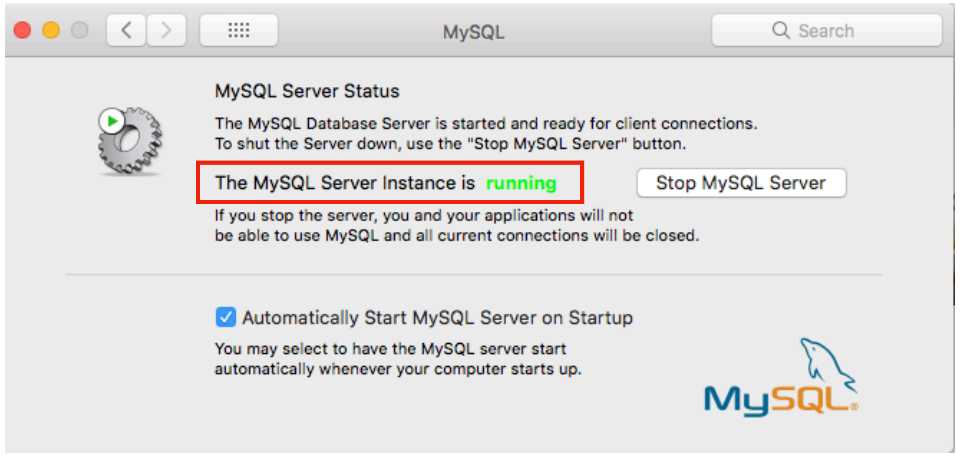
Learn More »
Customer Download » (Select Patches & Updates Tab, Product Search)
Trial Download » (Note - Select Product Pack: MySQL Database)

MySQL Community Edition (GPL)
Community (GPL) Downloads »

Contact MySQL Sales
USA/Canada: +1-866-221-0634 (More Countries »)

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

Installation



MySQL Server Status

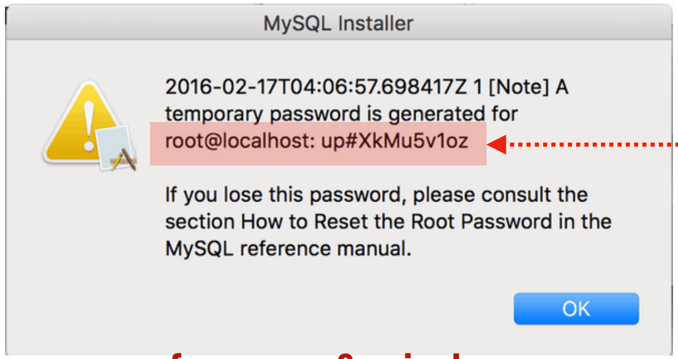
The MySQL Database Server is started and ready for client connections.
To shut the Server down, use the "Stop MySQL Server" button.

The MySQL Server Instance is running

If you stop the server, you and your applications will not
be able to use MySQL and all current connections will be closed.

☑ Automatically Start MySQL Server on Startup

You may select to have the MySQL server start
automatically whenever your computer starts up.



2016-02-17T04:06:57.698417Z 1 [Note] A
temporary password is generated for
root@localhost: up#XkMu5v1oz

If you lose this password, please consult the
section How to Reset the Root Password in the
MySQL reference manual.

you will need this password

for mac & windows

For windows 10: `https://www.youtube.com/watch?v=fwQyZz6cNGU`

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

Connecting to MySQL using terminal

for mac

```
1. Open Terminal …
2. /usr/local/mysql/bin/mysql -uroot -p
3. Enter password:
4. show databases;
5. set password=password ('12345');
6. show databases;
7. exit
8. /usr/local/mysql/bin/mysql -uroot -p12345
9. exit
```

You will be asked here to change your password

again

then, you can use new password

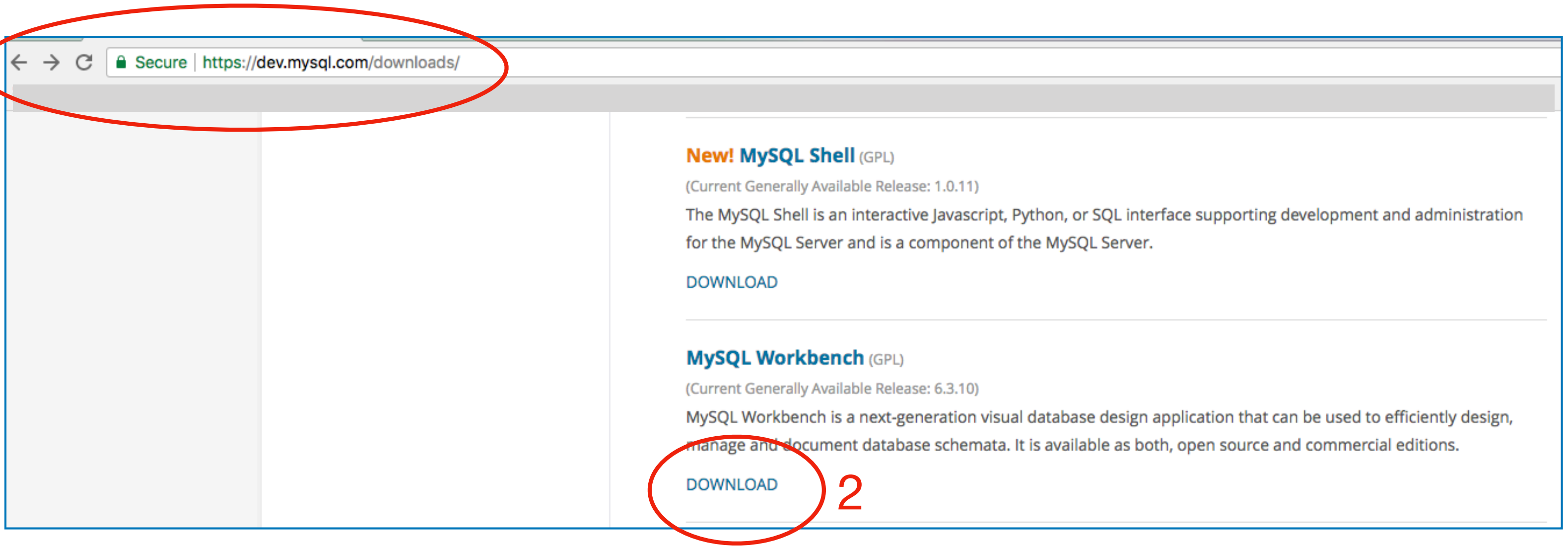
```
1.sudo vim /etc/paths.d/mysql
2.past « /usr/local/mysql/bin »
3.mysql -uroot -p12345
4.exit
```

create new file name mysql

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

Installing MySQL workbench

1



Secure | https://dev.mysql.com/downloads/

New! MySQL Shell (GPL)

(Current Generally Available Release: 1.0.11)

The MySQL Shell is an interactive Javascript, Python, or SQL interface supporting development and administration for the MySQL Server and is a component of the MySQL Server.

DOWNLOAD

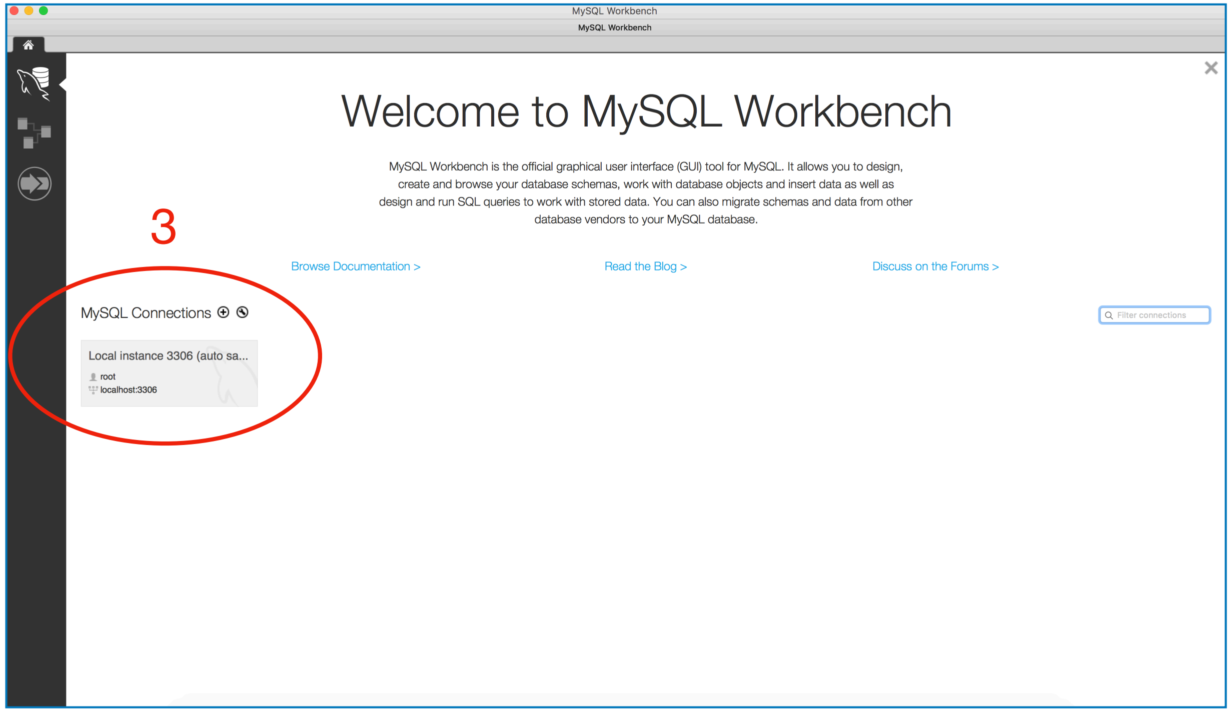
MySQL Workbench (GPL)

(Current Generally Available Release: 6.3.10)

MySQL Workbench is a next-generation visual database design application that can be used to efficiently design, manage and document database schemata. It is available as both, open source and commercial editions.

DOWNLOAD

2



MySQL Workbench

Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Browse Documentation >

Read the Blog >

Discuss on the Forums >

3

MySQL Connections

Local instance 3306 (auto sa...

root

localhost:3306

MySQL

- MySQL is a **database** system used on the web
- MySQL is a database system that runs on a **server**
- MySQL is ideal for both **small** and **large** applications
- MySQL is very fast, reliable, and easy to use
- MySQL uses standard SQL
- MySQL compiles on a number of platforms
- MySQL is free to download and use
- MySQL is developed, distributed, and supported by Oracle Corporation

Look at http://www.mysql.com/customers/ for an overview of companies using MySQL.

Create a new database with MySQL Workbench

"Schema" means "Database" in MySQL

2 possible ways:

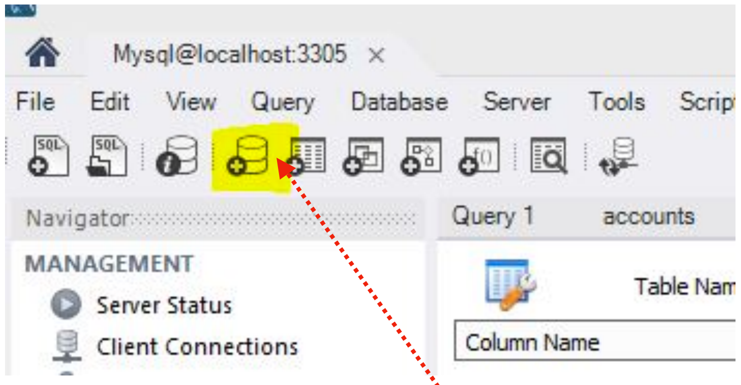
1

If you'd prefer to do it in SQL, enter this query into the query window:

```
CREATE SCHEMA Test
```

Press CTRL + Enter to submit it

2



create a new schema in the connected server

TP5

Building a Website with PHP

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

What do you need?

To start using PHP, you can:

- Find a web host with PHP and MySQL support
- Install a **web server** on your own PC, and then install PHP and MySQL

your localhost

- install a web server -usually apache
- install PHP
- install a database, such as MySQL

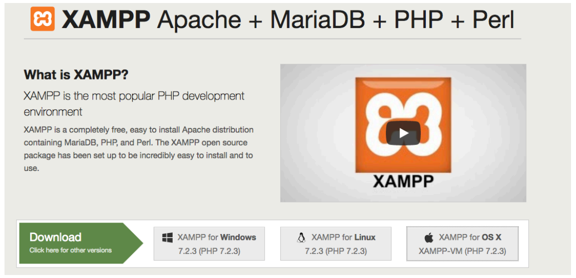
There are tools which install once simply and easily these three tools all together: **XAMPP**(It’s a package…)

MacOS

MAMP

run dynamic web sites on Mac OS computers
M-Mac OS
A-Apache
M-MYSQL
P-PHP or Perl

XAMPP **or** **WAMP**
X-cross platform: windows, mac , linux
A-Apache
M-MYSQL or MARIADB
P-PHP
P-Perl



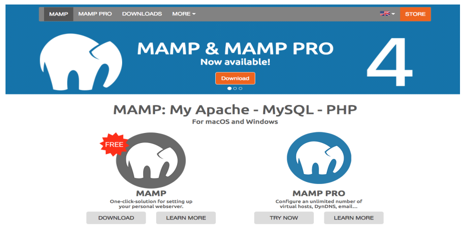
XAMPP Apache + MariaDB + PHP + Perl

What is XAMPP?

XAMPP is the most popular PHP development environment

XAMPP is a completely free, easy to install Apache distribution containing MariaDB, PHP, and Perl. The XAMPP open source package has been set up to be incredibly easy to install and to use.

XAMPP
Download
Click here for other versions
XAMPP for Windows 7.2.3 (PHP 7.2.3)
XAMPP for Linux 7.2.3 (PHP 7.2.3)
XAMPP for OS X XAMPP-VM (PHP 7.2.3)



MAMP
MAMP PRO
DOWNLOADS
MORE
STORE
MAMP & MAMP PRO
Now available!
Download
4

MAMP: My Apache - MySQL - PHP
For macOS and Windows

FREE
MAMP
One-click-solution for setting up your personal webserver.
DOWNLOAD
LEARN MORE
MAMP PRO
Configure an unlimited number of virtual hosts, DynDNS, email...
TRY NOW
LEARN MORE

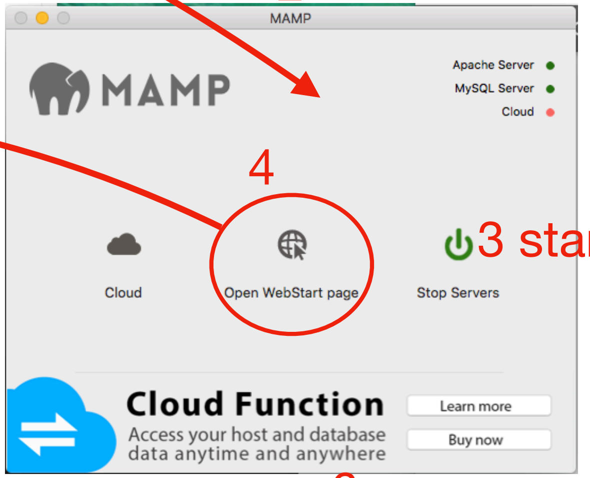
UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D’INFORMATIQUE

for Mac OS

How to start using phpMyAdmin …

1. Install MAMP (mac)

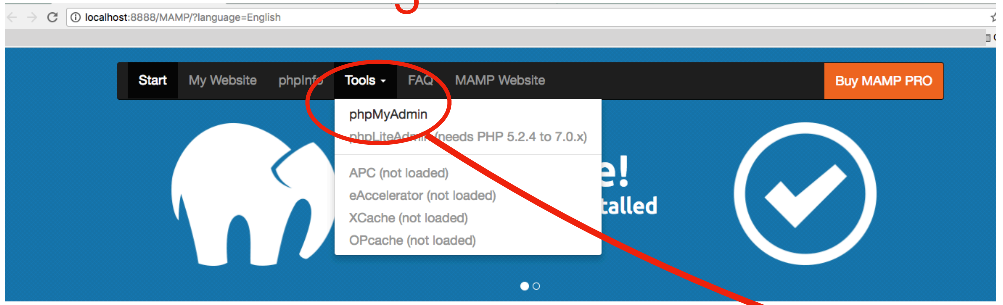
2



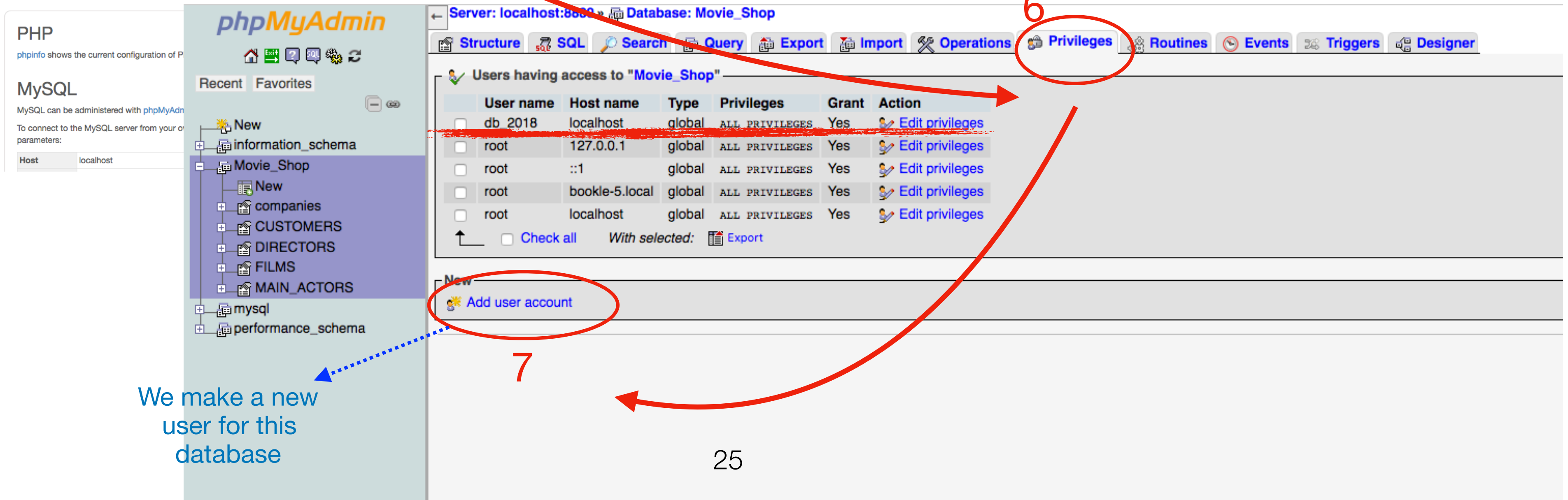
3 start server

4

5



6



Users having access to "Movie_Shop"

User name	Host name	Type	Privileges	Grant	Action
~~db_2018~~	~~localhost~~	~~global~~	~~ALL PRIVILEGES~~	~~Yes~~	Edit privileges
root	127.0.0.1	global	ALL PRIVILEGES	Yes	Edit privileges
root	::1	global	ALL PRIVILEGES	Yes	Edit privileges
root	bookle-5.local	global	ALL PRIVILEGES	Yes	Edit privileges
root	localhost	global	ALL PRIVILEGES	Yes	Edit privileges

Check all *With selected:* Export

New

Add user account

7

We make a new user for this database

for windows

How to Install **XAMPP** and start Using phpMyAdmin …

https://www.youtube.com/watch?v=dV3JjLhi4Jk

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PHP: Basic

Basic PHP Syntax

A PHP script starts with **<?php** and ends with **?>**:

```
<?php
// PHP code goes here
?>
```

between **tags**

- The default file extension for PHP files is ".php".
- A PHP file normally contains HTML tags, and some PHP scripting code

```
<!DOCTYPE html>
<html>
<body>

<h1>My first PHP page</h1>

<?php
echo "Hello World!";
?>

</body>
</html>
```

PHP: Basic

- You embed PHP code between **tags**
- A **semicolon** has to finish every php statement
- echo statement puts what ever is between quotes in the browser
- **Single quotes**: Print what is between them and ignore escape sequences except for \' and \\
- **Double quotes**: Print many escape sequences, the values for variables, and more

```
echo "<p>Data Processed at </p>";
```

```
/* Multiline
   comment */

// Single line comment

# Another single line comment
```

In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive.

However; all variable names are case-sensitive.

Example:

```
<html>
<body>

<?php
$color = "red";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
echo "My boat is " . $coLOR . "<br>";
?>

</body>
</html>
```

only the first statement will display the value of the $color variable (this is because $color, $COLOR, and $coLOR are treated as three different variables)

PHP: Basic

Creating (Declaring) PHP Variables

Example:

```php
<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;
?>
```

in text value: quotes around the value

- A variable starts with the **$ sign**
- A variable name must **start with a letter** or the underscore character (_)
- A variable name **cannot** start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive ($age and $AGE are two different variables)

Example:

```php
<?php
$x = 5;
$y = 4;
echo $x + $y;
?>
```

output: the sum of two variables

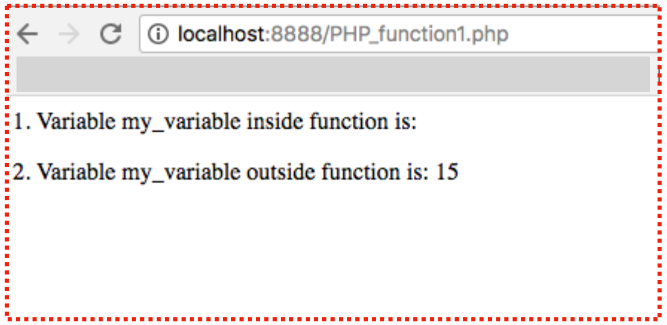
UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PHP: Variables + functions

Example 1:

```
<!--  PHP_function1.php -->
<html>
<body>

<?php
$my_variable = 15; // global scope

function func_1() {

    echo "<p>1. Variable my_variable inside function is: $my_variable</p>";
}
func_1();          // call the function

echo "<p>2. Variable my_variable outside function is: $my_variable</p>";
?>

</body>
</html>
```

webpage:



```
localhost:8888/PHP_function1.php

1. Variable my_variable inside function is:

2. Variable my_variable outside function is: 15
```

Example 2:

```
<!-- global variable -- >
<!--  PHP_function2.php -->
<html>
<body>

<?php
$my_variable2 = 15; // global scope

function func_2() {
    global $my_variable2;

    echo "<p>1. Variable my_variable2 inside function is: $my_variable2</p>";
}
func_2();

echo "<p>2. Variable my_variable2 outside function is: $my_variable2</p>";
?>

</body>
</html>
```

webpage:

```
1. Variable my_variable2 inside function is: 15

2. Variable my_variable2 outside function is: 15
```

The `global` keyword is used to access a global variable from within a function:

```
function func_1() {
      global $my_variable;
}
func_1();
```

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PHP: Basic

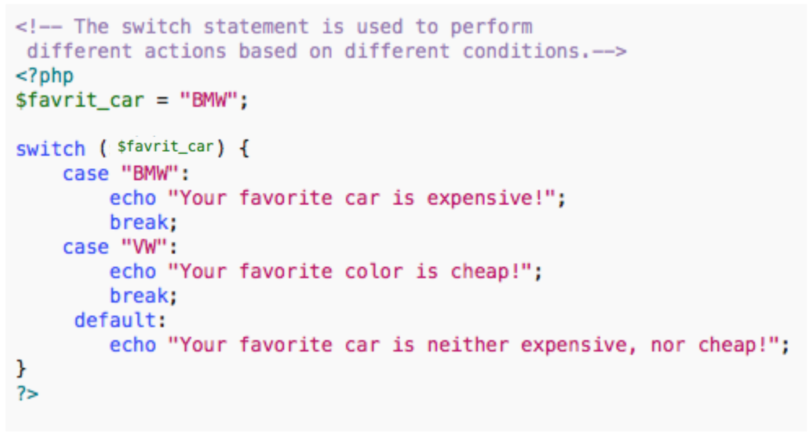
Create a PHP Constant:
To create a constant, use the `define()` function.

```
define(name, value, case-insensitive)
```

PHP - The if...else Statement:

```
if (condition) {
    code to be executed if condition is true;
} else {
    code to be executed if condition is false;
}
```

PHP - switch Statement:



```
<!-- The switch statement is used to perform
 different actions based on different conditions.-->
<?php
$favrit_car = "BMW";

switch ( $favrit_car) {
    case "BMW":
        echo "Your favorite car is expensive!";
        break;
    case "VW":
        echo "Your favorite color is cheap!";
        break;
     default:
        echo "Your favorite car is neither expensive, nor cheap!";
}
?>
```

PHP: Basic

PHP- While loop:

```
<!-- PHP While loop -->
<?php
$my_number = 1;

while($my_number <= 10) {
  echo ”MY number: $my_number <br> ”;
    $my_number++;
}
?>
```

Example 2:

```
<!-- PHP While loop 2 -->
<?php
$my_number = 6;

do {
    echo "The number is: $my_number <br>";
    $my_number++;
} while ($my_number<= 5);
?>
```

The example sets the $my_number variable to 6, then it runs the loop, **and then the condition is checked**

PHP- For loop:

```
<!-- PHP for loop-->
<?php
for ($my_v = 0; $my_v <= 8; $my_v++) {
    echo "The number is: $my_v <br>";
}
?>
```

PHP- arrays:

```
<!-- PHP Arrays-->
<?php
$colors = array("red", "blue", "green");
echo "I like " . $colors[0] . ", " . $colors[1] . " and " . $colors[2] . ".";
?>
```

The `count()` function is used to return the length (the number of elements) of an array

```
echo count($colors);
```

```
for($i = 0; $i < $arrlength; $i++) {
    echo $colors[$i];
    echo "<br>";
}
```

PHP + HTML: easy example

Form Handling

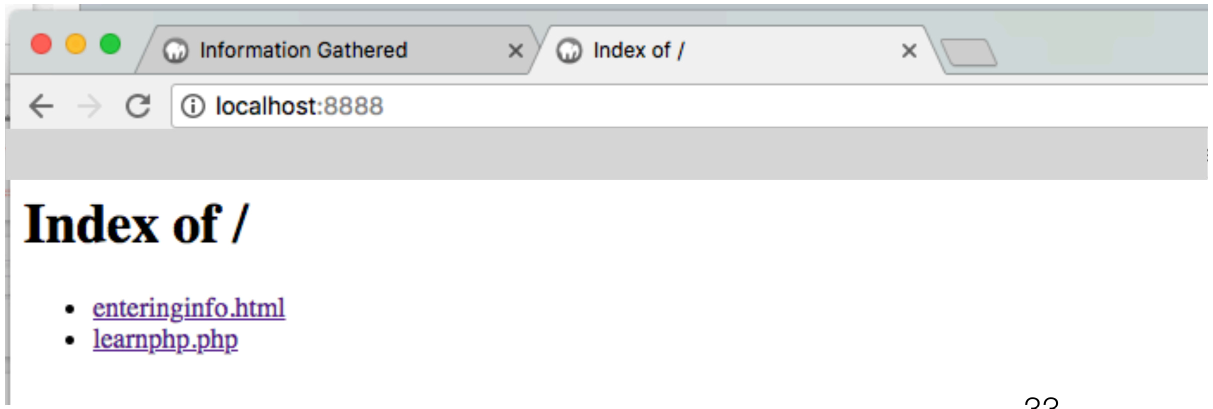
enteringinfo.html

```
<html>
<body>
<form action="learnphp.php" method="post">

<table border="0">

<tr>
    <td>Name</td>
    <td align="center"><input type="text" name="name" size="30" /></td>
</tr>

<tr>
    <td>Last name</td>
    <td align="center"><input type="text" name="lastname" size="30" /></td>
</tr>

<tr>
    <td>Email</td>
    <td align="center"><input type="text" name="emailaddress" size="30"  /></td>
</tr>

<tr>
<td colspan="2" align="center"><input type="submit" value="Submit"/></td>
</tr>

</table>
</form>
</body>
</html>
```

submit button,

learnphp.php

```
<html>

    <head>
        <title>Information Gathered</title>
    </head>

    <body>

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

    ?>


    </body>

</html>
```

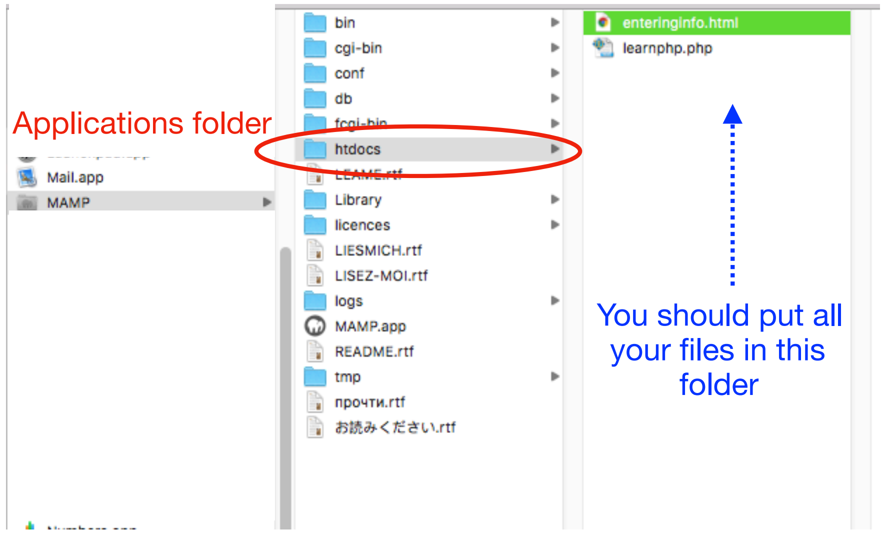


Information Gathered ×   Index of / ×

localhost:8888

Index of /

- enteringinfo.html
- learnphp.php

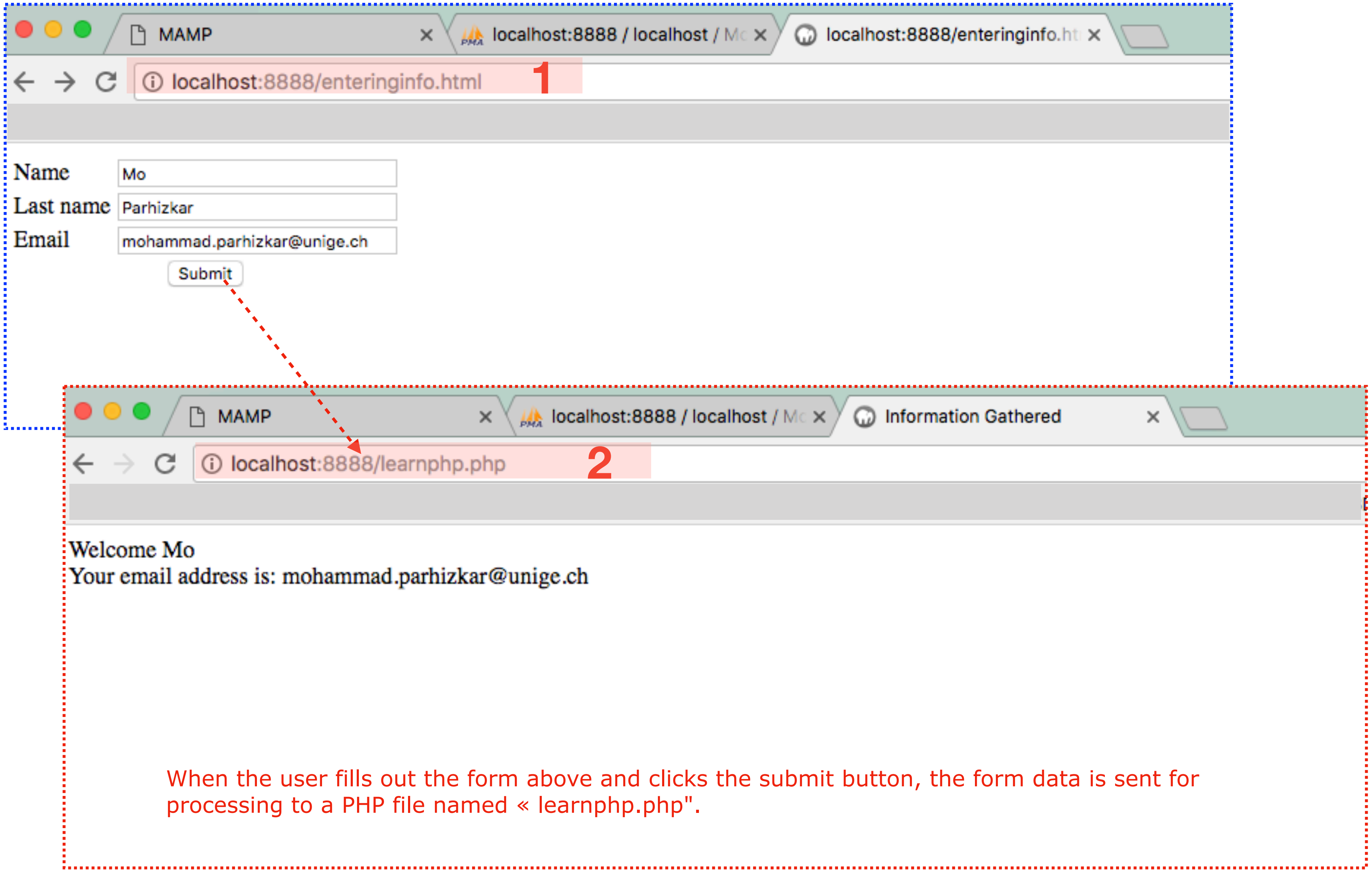


You should put all your files in this folder

PHP + HTML: easy example

Form Handling

first webpage:

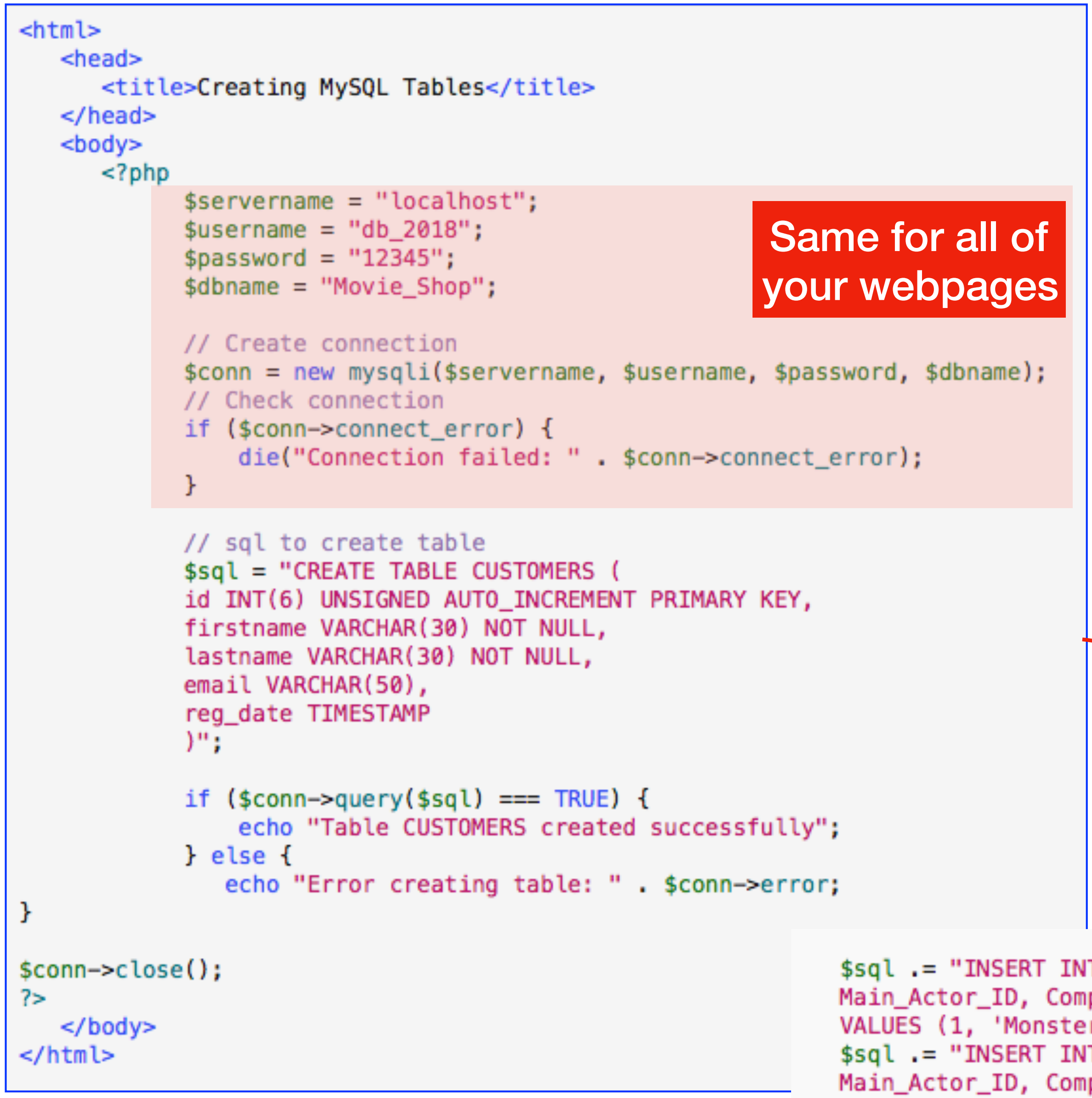

MAMP
localhost:8888 / localhost / Mc
localhost:8888/enteringinfo.ht
localhost:8888/enteringinfo.html
1
Submit
Information Gathered
localhost:8888/learnphp.php
2

Name	Mo
Last name	Parhizkar
Email	mohammad.parhizkar@unige.ch

Welcome Mo
Your email address is: mohammad.parhizkar@unige.ch

When the user fills out the form above and clicks the submit button, the form data is sent for processing to a PHP file named « learnphp.php".

HTML: PHP + phpMyAdmin: Create Tables + Insert

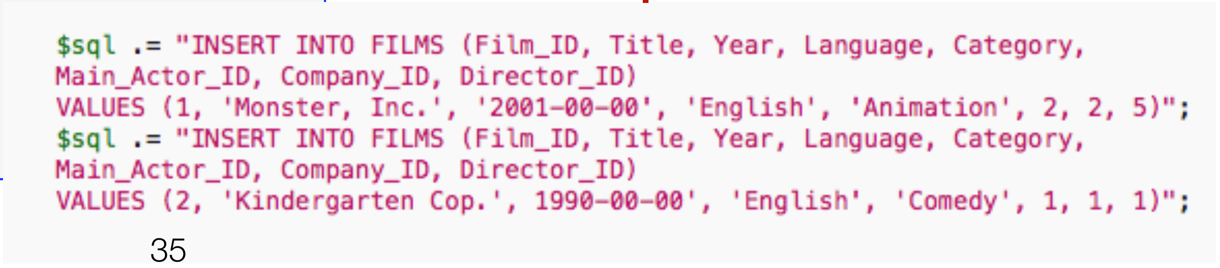


```
<html>
   <head>
      <title>Creating MySQL Tables</title>
   </head>
   <body>
      <?php
         $servername = "localhost";
         $username = "db_2018";
         $password = "12345";
         $dbname = "Movie_Shop";

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

         // sql to create table
         $sql = "CREATE TABLE CUSTOMERS (
         id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
         firstname VARCHAR(30) NOT NULL,
         lastname VARCHAR(30) NOT NULL,
         email VARCHAR(50),
         reg_date TIMESTAMP
         )";

         if ($conn->query($sql) === TRUE) {
             echo "Table CUSTOMERS created successfully";
         } else {
            echo "Error creating table: " . $conn->error;
}

$conn->close();
?>
   </body>
</html>
```

Same for all of your webpages

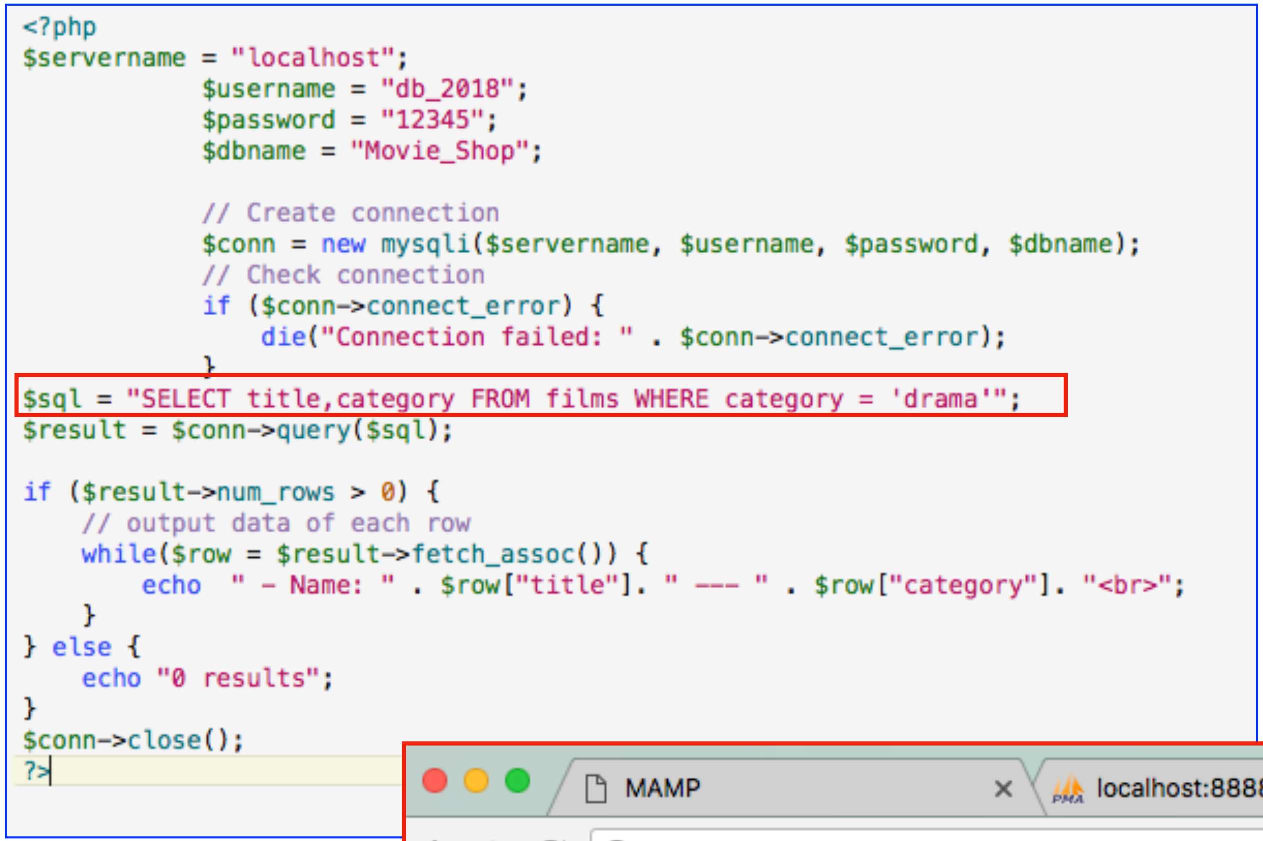
next step…

Insert Multiple Records

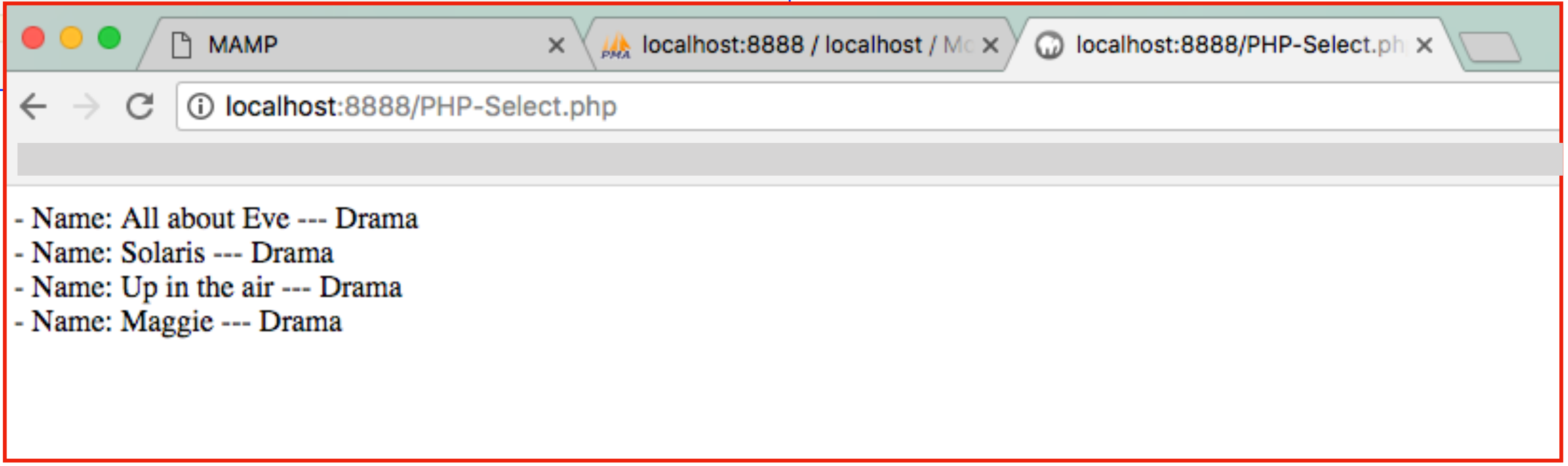


```
$sql .= "INSERT INTO FILMS (Film_ID, Title, Year, Language, Category,
Main_Actor_ID, Company_ID, Director_ID)
VALUES (1, 'Monster, Inc.', '2001-00-00', 'English', 'Animation', 2, 2, 5)";
$sql .= "INSERT INTO FILMS (Film_ID, Title, Year, Language, Category,
Main_Actor_ID, Company_ID, Director_ID)
VALUES (2, 'Kindergarten Cop.', 1990-00-00', 'English', 'Comedy', 1, 1, 1)";
```

HTML: PHP + phpMyAdmin: Select Data From a MySQL Database



```php
<?php
$servername = "localhost";
            $username = "db_2018";
            $password = "12345";
            $dbname = "Movie_Shop";

            // Create connection
            $conn = new mysqli($servername, $username, $password, $dbname);
            // Check connection
            if ($conn->connect_error) {
                die("Connection failed: " . $conn->connect_error);
            }
$sql = "SELECT title,category FROM films WHERE category = 'drama'";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo  " - Name: " . $row["title"]. " --- " . $row["category"]. "<br>";
    }
} else {
    echo "0 results";
}
$conn->close();
?>
```



MAMP
localhost:8888 / localhost / Mc
localhost:8888/PHP-Select.ph
localhost:8888/PHP-Select.php

```
- Name: All about Eve --- Drama
- Name: Solaris --- Drama
- Name: Up in the air --- Drama
- Name: Maggie --- Drama
```

phpMyAdmin: foreign key

```
ALTER TABLE films
    ADD CONSTRAINT fk_foreign_films_3
    FOREIGN KEY (Company_ID)
    REFERENCES companies(Company_ID);
```

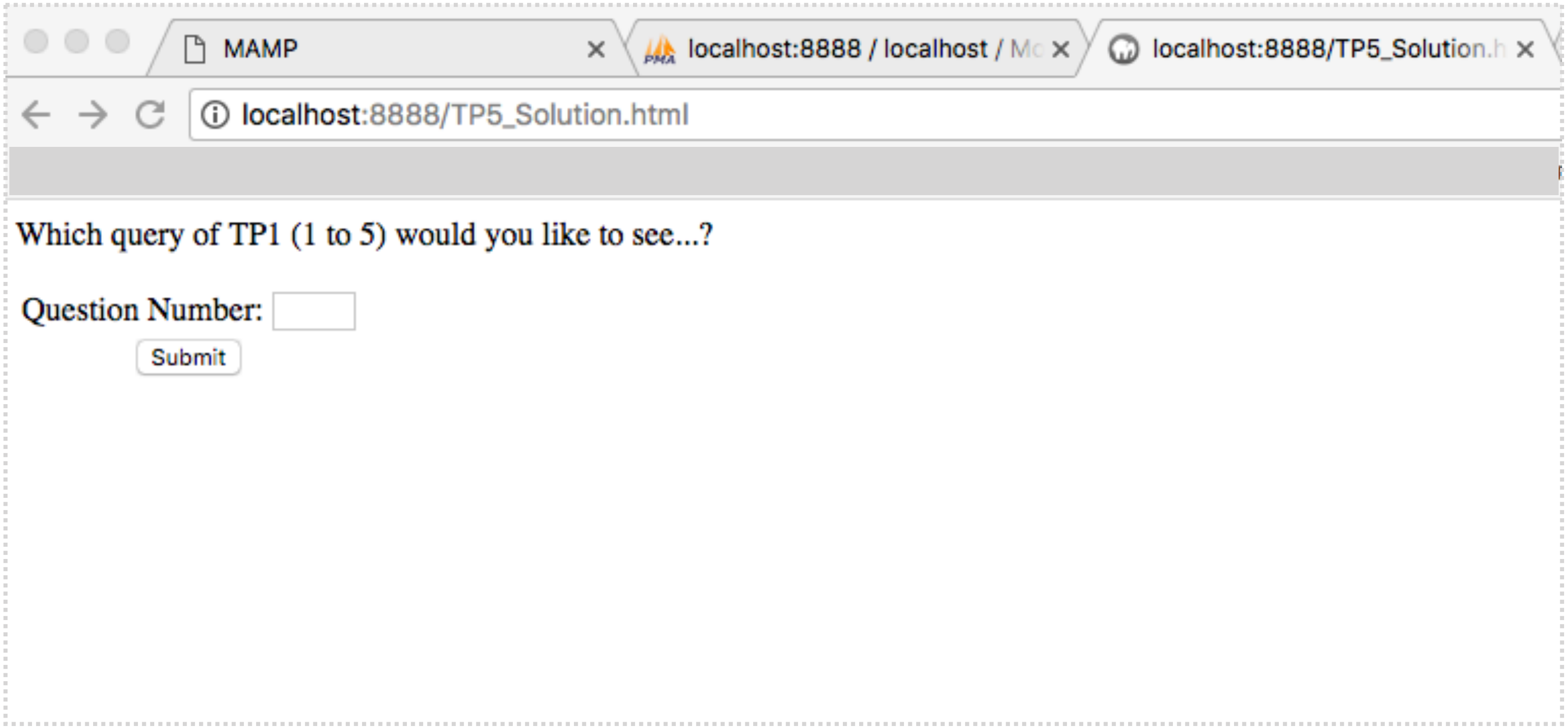
✓

TP5

phpMyAdmin: Year function

```
$sql = "SELECT language FROM films
     WHERE Year(Year)>’1980'";
```

✓


MAMP
localhost:8888 / localhost / Mc
localhost:8888/TP5_Solution.h
localhost:8888/TP5_Solution.html

Which query of TP1 (1 to 5) would you like to see...?

Question Number:

UNIVERSITÉ DE GENÈVE
CENTRE UNIVERSITAIRE D'INFORMATIQUE

PHP Basics

👍 PHP Forms

PHP Form Handling

Next Sessions:

- PHP Form Validation
- PHP Form Required
- PHP Form URL/E-mail
- PHP Form Complete

Thank you!