

# Databases

## Web applications in PHP to query a database

### Practice n. 4

The purpose of this practice is to build a simple web application in PHP to query a database.

### Preliminary steps

This practice is based on Apache Web server and MySQL database, in particular the versions available in XAMPP. It is necessary to boot both services in order to execute the exercises.

#### Boot MySQL server on localhost and start Apache

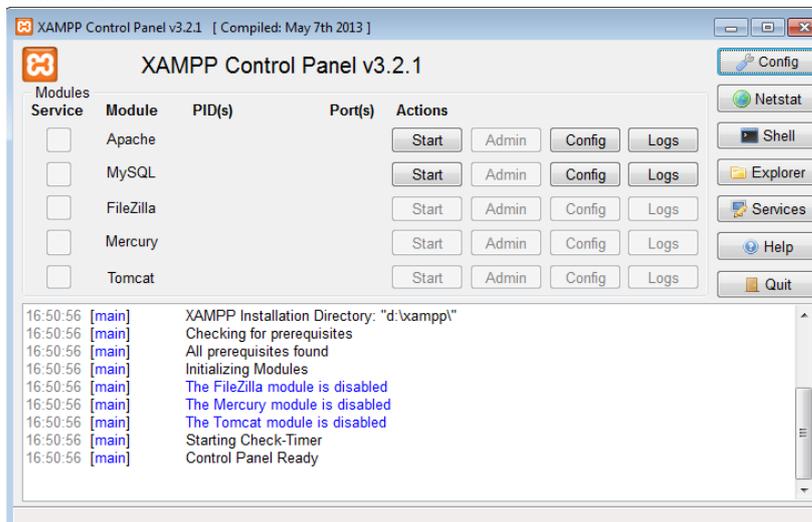
The execution of scripts with SQL commands for the creation and population of the database will be performed through the Web interface of MySQL.

Before opening the Web interface of MySQL it is necessary to:

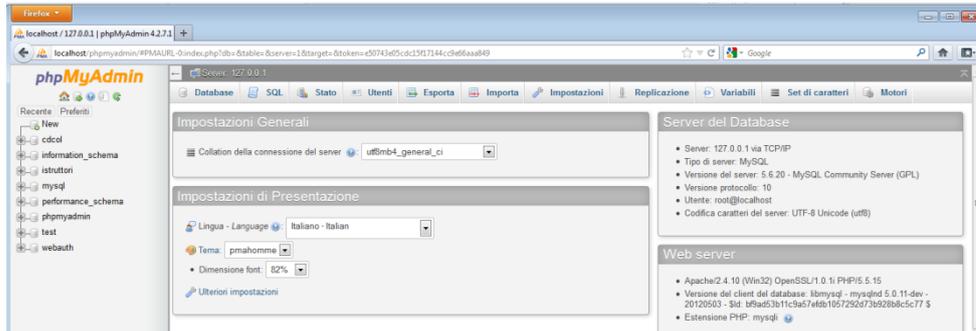
- Start the local Apache server
- Start the local MySQL server

Specifically, execute the following steps:

- 1) Start "XAMPP Control Panel"



- 2) Start Apache clicking the Start button in the row of "Apache" module.
- 3) Start MySQL clicking the Start button in the row of "MySQL" module.
- 4) Open the MySQL Web interface clicking the Admin button in the row of "MySQL" module (the browser will automatically open the URL associated to the page of administration and SQL querying, i.e., *phpMyAdmin*).



- 5) To execute a SQL script from the Web interface of MySQL:
  - a. Select the “Import” panel
  - b. Select the file with the script you want to execute and click on “Go” button.
- 6) To execute the creation/population script more than once, you need to cancel any existing instance of the database, either directly from the “Database” panel or by including at the beginning of the script the commands for deleting the existing tables.

## Creation and population of the database used for the practice

The database used for this practice is the same of LAB3. It is called *Gym* and is used to record the activities of a gym. It’s characterized by the following logical schema (primary keys are underlined, optional attributes are marked with asterisk):

TRAINER (SSN, Name, Surname, DateOfBirth, Email, PhoneNo\*)

COURSE (CId, Name, CType, CLevel)

SCHEDULE (SSN, WeekDay, StartTime, Duration, CId, GymRoom)

1. Create the Gym database and populate it using the createdB.sql and populatedB.sql scripts available on the webpage of the course.

Afterwards, the tables will contain the following data:

*TRAINER table*

<u>SSN</u>	Name	Surname	DateOfBirth	Email	PhoneNo
<b>SMTPLA80N31B791Z</b>	Paul	Smith	31/12/1980	p.smith@gym.it	NULL
<b>KHNJHN81E30C455Y</b>	John	Johnson	30/5/1981	j.johnson@gym.it	+2300110303444
<b>AAAGGG83E30C445A</b>	Peter	Johnson	30/5/1981	p.johnson@gym.it	+2300110303444

*COURSE table*

<u>CId</u>	Name	CType	CLevel
<b>CT100</b>	Spinning for beginners	Spinning	1
<b>CT101</b>	Fitdancing	Music	2

		activity	
CT104	Advanced spinning	Spinning	4

SCHEDULE table

SSN	WeekDay	StartTime	Duration	CId	GymRoom
SMTPLA80N31B791Z	Monday	10:00	45	CT100	R1
SMTPLA80N31B791Z	Tuesday	11:00	45	CT100	R1
SMTPLA80N31B791Z	Tuesday	15:00	45	CT100	R2
KHNJHN81E30C455Y	Monday	10:00	30	CT101	R2
KHNJHN81E30C455Y	Monday	11:30	30	CT104	R2
KHNJHN81E30C455Y	Wednesday	9:00	60	CT104	R1

## Publish/upload a PHP page in XAMPP

To publish a PHP page with XAMPP you need to copy the PHP file in the *htdocs* folder of XAMPP (C:\XAMPP\htdocs).

Once the file (for instance filename.php) has been copied, it will be accessible through the browser at the local address *http://localhost/filename.php*.

## Exercise

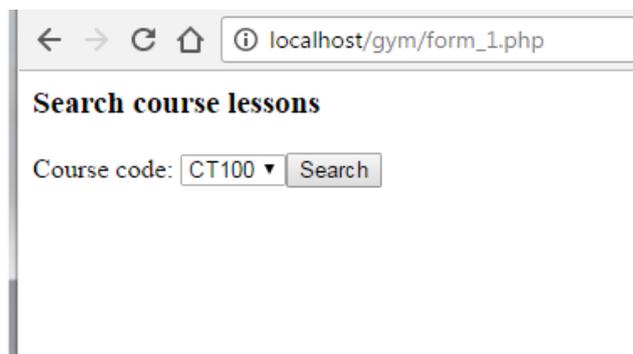
Design a web application in PHP for the online consultation of the Gym database. The application has to provide the following functionalities:

1. By selecting the code of a course (from a select box filled with data obtained by querying the content of the table COURSE), show all the scheduled weekly lessons for that course. For each lesson, show the day of the week, the start time, the duration, the room and the name and surname of the lesson trainer. Fig.1 and Fig. 2 show, respectively, the page used to perform the request and the page with the result.
2. By selecting the surname of a trainer (from a select box filled with data obtained by querying the content of the table TRAINER) and a day of the week (from another select box containing standard values), show all the lessons held by trainers with the given surname and scheduled for the given week day. For each lesson show the day of the week, the start time, the duration, the room, the name, type and level of the course, and the SSN, name and surname of the trainer. Lessons should be listed ordered by trainer SSN and by course name. If the selected trainer has no lessons scheduled for the selected day, show the message “No lesson scheduled for the trainer <surname> on <week day>”. Fig. 3 and Fig. 4 show, respectively, the page used to perform the request and the page containing the result.

3. (OPTIONAL) Write a different version of the pages at point 1, where the select box includes also the name of the course together with the code. Once the course is selected, show the same data at point 1. Fig. 5 and Fig. 6 show, respectively, the page used to perform the request and the page containing the result.
4. *Insert a new course.* Create a web page with a form that requests all the data needed to insert a new COURSE in the database (CId, Name, CType, CLevel). The application should check that all the fields are filled and that the value for CLevel is an integer between 1 and 4. In case of missing data, duplicated key (CId), values outside the allowed range for CLevel or incompatibilities with the schedule (instructor or room already busy with another course on that time and day), the application should generate an error message. Otherwise, if all values are correct and the insert operation is successful, the application should show a confirmation message. Fig.7 and Fig. 8 show two example pages used to perform the request and to show the result.
5. *Insert a new weekly lesson in the schedule.* Create a web page with a form to insert a new weekly lesson in the SCHEDULE table. The form should allow to insert all the needed fields (SSN, WeekDay, StartTime, Duration, CId, GymRoom) related to the schedule of a new lesson. The selection of the trainer should be performed through a select box containing the surname, name and SSN of the possible trainers, generated from the content of the database. Similarly, also the selection of the course should be performed through a select box populated with data from the database. The other fields are textual boxes instead, filled manually by the user. The application should check that the user does not insert lessons lasting more than 60 minutes and that the day is between Monday and Friday. The insertion of a new lesson should be allowed and executed only if no other lesson is scheduled for the same course in the same day of week. If the request fulfils such constraints and the insert operation is successful the application should show a confirmation message, otherwise it should show an error message with the specification of the problem.

**N.B.:** the SQL query, used to check whether a lesson already exists in the SCHEDULE table for the same day and course specified in the form, should be part of the same transaction of the query used to insert the new lesson (this is to avoid problems in case of simultaneous insertions by two users of two lessons for the same course and the same day).

Fig. 9 and Fig. 10 show two example pages used to perform the request and to show the result.



← → ↻ 🏠 localhost/gym/form\_1.php

### Search course lessons

Course code:

Figure 1 – Search by course code.

localhost/gym/search\_course\_lessons.php?course\_code=CT100

### Lessons scheduled for course CT100

Day	Start time	Duration	Room	Trainer name	Instructor surname
Monday	10:00:00	45	R1	Paul	Smith
Tuesday	11:00:00	45	R1	Paul	Smith
Tuesday	15:00:00	45	R2	Paul	Smith

Search again

Figure 2 – Result of search by course code.

localhost/gym/form\_2.php

### Search lessons by instructor

Surname:

Day:

Search

Figure 3 – Search by trainer surname and day of week.

localhost/gym/seaech\_trainer\_lessons.php?surname=Johnson&day=Monday

### Scheduled lessons

The lessons scheduled for trainer Johnson on Monday are the following.

Day	Start time	Duration	Room	Course name	Level	Trainer SSN	Trainer Name	Trainer Surname
Monday	11:30:00	30	R2	Advanced spinning	4	KHNJHN81E30C455Y	John	Johnson
Monday	10:00:00	30	R2	Fitdancing	2	KHNJHN81E30C455Y	John	Johnson

Search again

Figure 4 – Result of search by trainer surname and day of week.

localhost/gym/form\_3.php

### Search course lessons

Course code:

Search

Figure 5 – Search by course code and name.

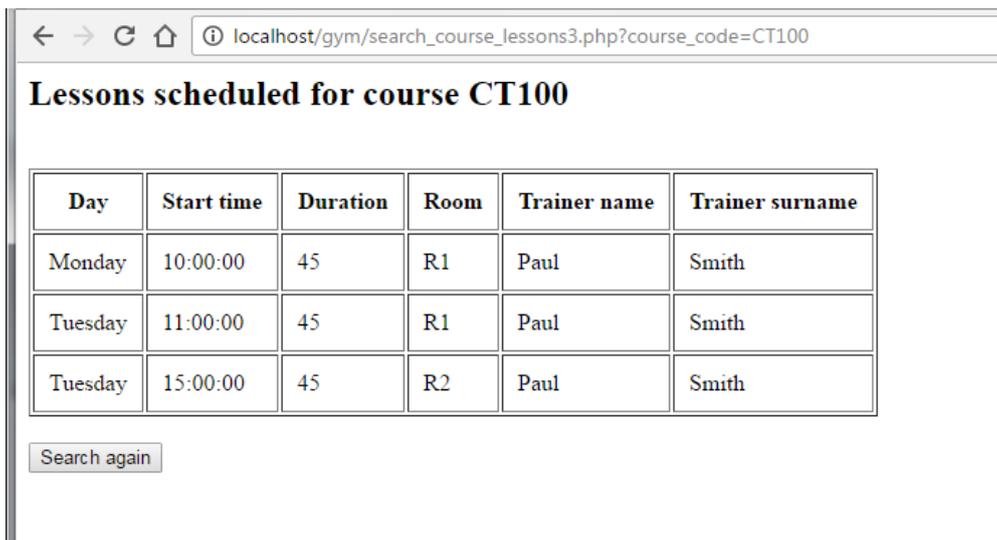


Figure 6 – Result of search by course code and name.

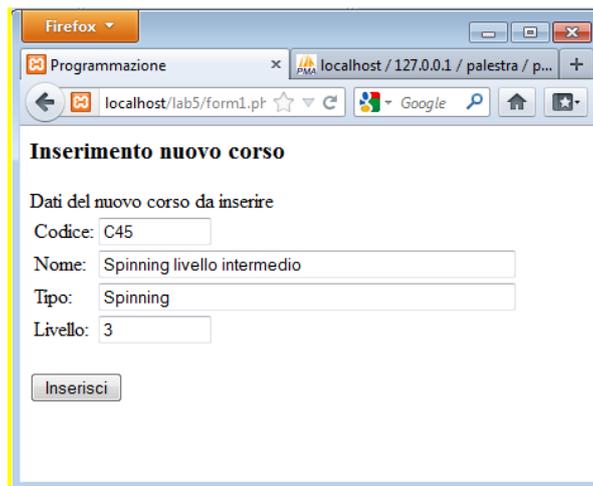


Figure 7 – insert new course

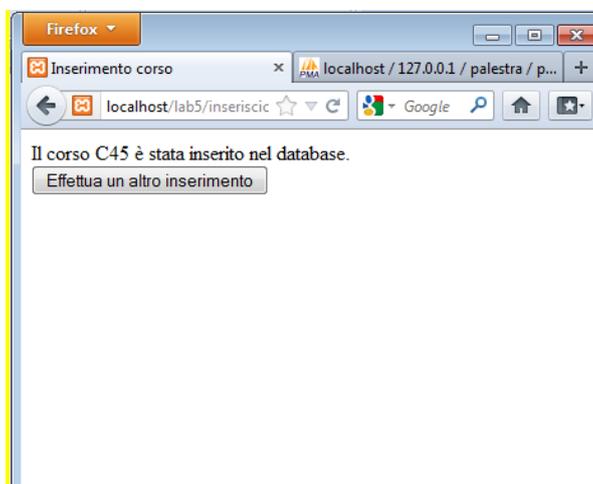


Figure 8 – Successful insert

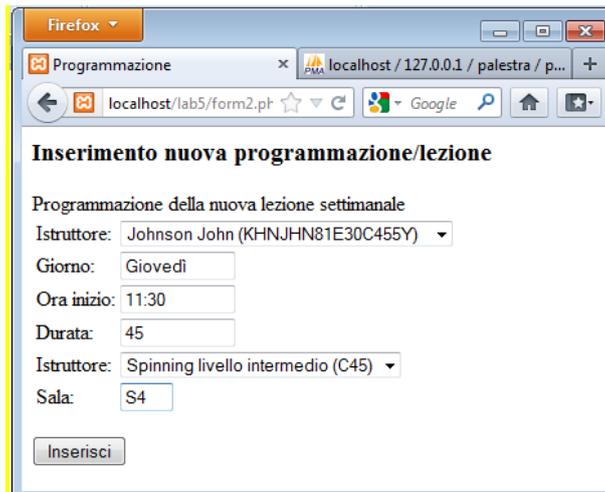


Figure 9 – Insert a new weekly lesson in the schedule.

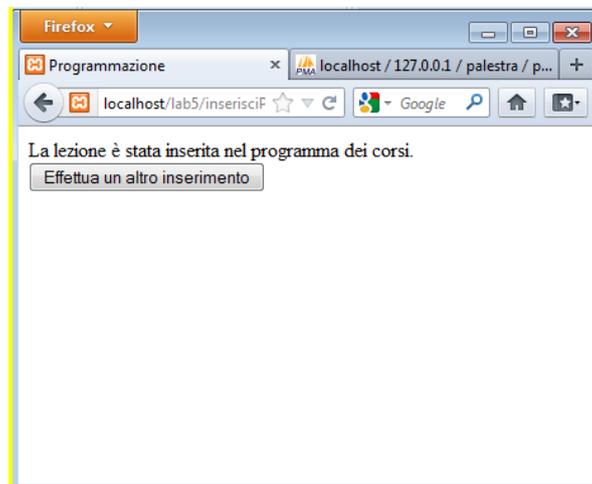


Figure 10 – succesfull insert