

# Databases

## Web applications in PHP to query a database

### Practice n. 5

The purpose of this practice is to build a simple web application in PHP to insert and update data in 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

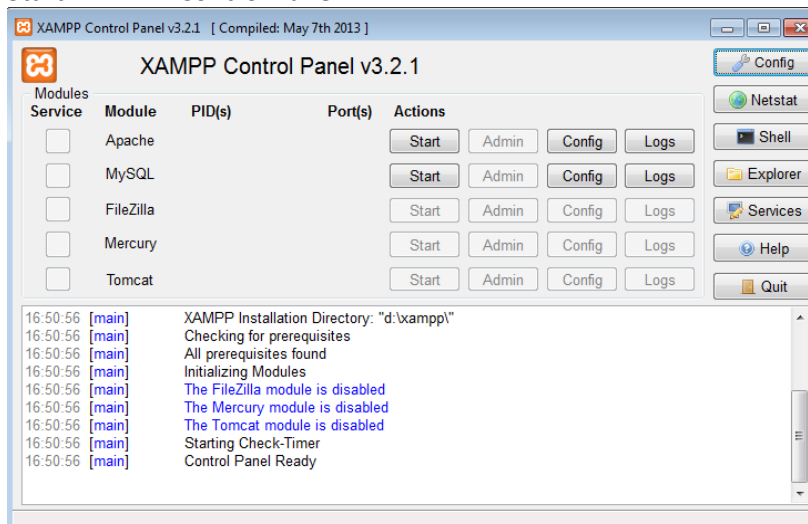
The execution of scripts with SQL commands for the creation and population of the database will be performed through the Web interface of MySQL.

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

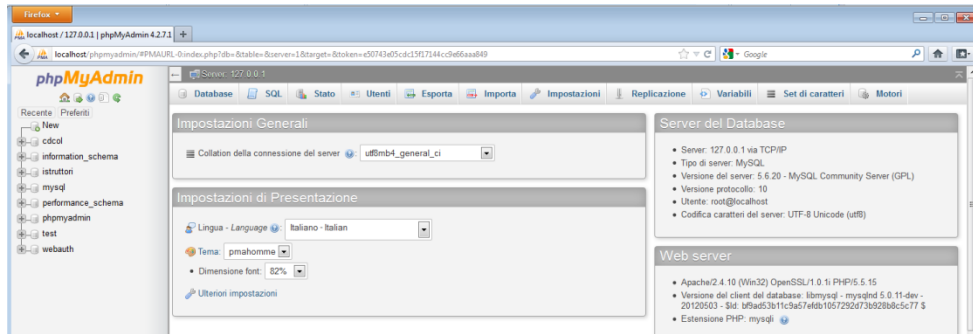
- Start the local Apache server
- Start the local MySQL server

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
SMTPLA80N31B791Z	Paul	Smith	31/12/1980	p.smith@gym.it	NULL
KHNJHN81E30C455Y	John	Johnson	30/5/1981	j.johnson@gym.it	+2300110303444
AAAGGG83E30C445A	Peter	Johnson	30/5/1981	p.johnson@gym.it	+2300110303444

*COURSE table*

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

		activity	
CT104	Advanced spinning	Spinning	4

*SCHEDULE table*

<u>SSN</u>	<u>WeekDay</u>	<u>StartTime</u>	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*.

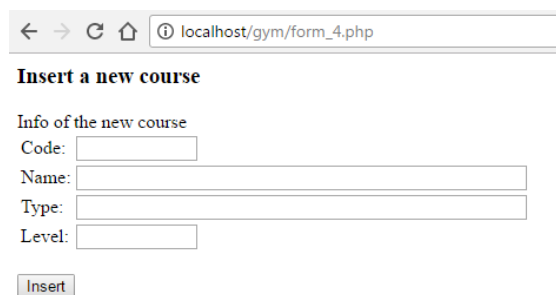
## **Exercise**

Design a web application in PHP for the update of the Gym database. The application has to extend the one created during the previous lab, providing the following functionalities:

1. *Insert a new course.* Create a PHP 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) or values outside the allowed range for CLevel, 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.1 and Fig. 2 show two example pages used to perform the request and to show the result.
2. *Insert a new weekly lesson in the schedule.* Create a PHP 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

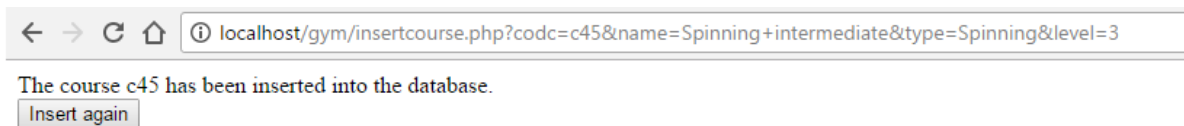
lasting more than 60 minutes and that the day is between Monday and Friday. The insertion of a new lesson should be allowed and executed only if no other lesson is scheduled for the same course in the same day of week. If the request fulfils such constraints and the insert operation is successful the 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. 3 and Fig. 4 show two example pages used to perform the request and to show the result.



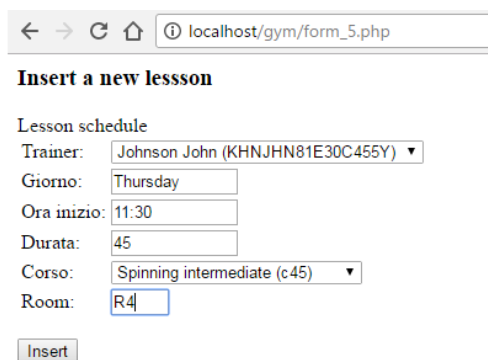
The screenshot shows a web browser window with the address bar containing 'localhost/gym/form\_4.php'. The page title is 'Insert a new course'. Below the title, there is a section 'Info of the new course' with four input fields: 'Code:', 'Name:', 'Type:', and 'Level:'. An 'Insert' button is located at the bottom of the form.

Figure 1 – Insertion of a new course



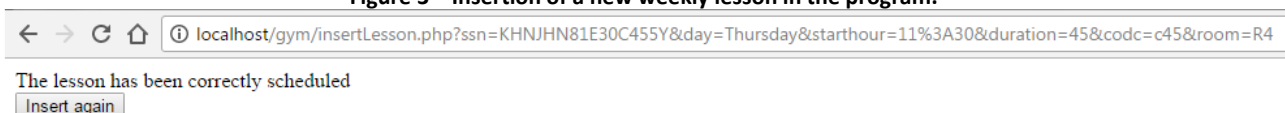
The screenshot shows a web browser window with the address bar containing 'localhost/gym/insertcourse.php?codc=c45&name=Spinning+intermediate&type=Spinning&level=3'. The page content displays the message 'The course c45 has been inserted into the database.' and an 'Insert again' button.

Figure 2 – Message for a successful course insertion



The screenshot shows a web browser window with the address bar containing 'localhost/gym/form\_5.php'. The page title is 'Insert a new lesson'. Below the title, there is a section 'Lesson schedule' with several input fields: 'Trainer:' (a dropdown menu showing 'Johnson John (KHNJHN81E30C455Y)'), 'Giorno:' (a dropdown menu showing 'Thursday'), 'Ora inizio:' (a text input field showing '11:30'), 'Durata:' (a text input field showing '45'), 'Corso:' (a dropdown menu showing 'Spinning intermediate (c45)'), and 'Room:' (a text input field showing 'R4'). An 'Insert' button is located at the bottom of the form.

Figure 3 – Insertion of a new weekly lesson in the program.



The screenshot shows a web browser window with the address bar containing 'localhost/gym/insertLesson.php?ssn=KHNJHN81E30C455Y&day=Thursday&tstarthour=11%3A30&duration=45&codc=c45&room=R4'. The page content displays the message 'The lesson has been correctly scheduled' and an 'Insert again' button.

Figure 4 - Message for a successful lesson insertion