

Basi di Dati

Applicazioni web in Python per l'interrogazione di basi di dati

Esercitazione n. 5

Questa esercitazione ha come obiettivo quello di realizzare una semplice applicazione web basata su Python per inserire dati e aggiornare il contenuto di una base di dati.

Passi preliminari per lo svolgimento dell'esercitazione

Questa esercitazione usa il server WEB di Flask e il database MySQL disponibile per Python e da XAMPP rispettivamente. Al fine di poter svolgere l'esercitazione è necessario avviare entrambi i servizi.

Avvio del server MySQL sulla macchina locale e avvio di Apache

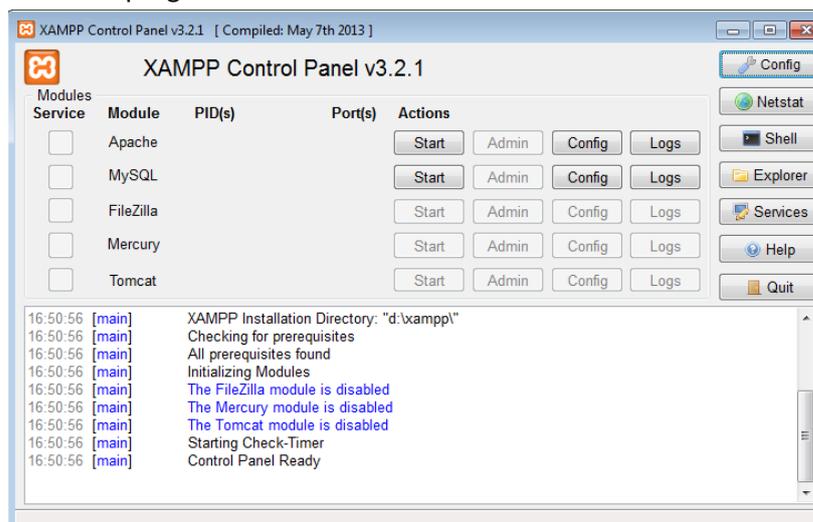
L'esecuzione degli script SQL contenenti i comandi SQL per la creazione e il popolamento della base di dati utilizzata in questa esercitazione avviene tramite l'interfaccia web di MySQL.

Prima di aprire l'interfaccia web di MySQL è necessario:

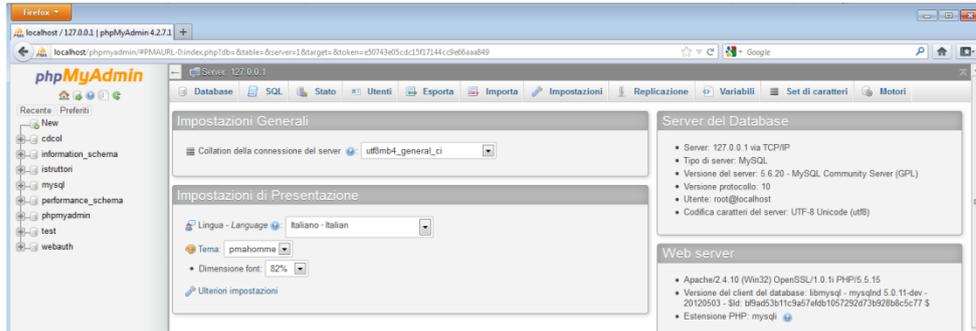
- Avviare il server locale Apache
- Avviare il server locale MySQL

In particolare, eseguire i seguenti passi:

- 1) Avviare il programma "XAMPP Control Panel"



- 2) Avviare Apache premendo il tasto Start nella riga relativa a Apache
- 3) Avviare MySQL premendo il tasto Start nella riga MySQL
- 4) Aprire l'interfaccia web di MySQL premendo il tasto Admin nella riga di MySQL (il browser si aprirà automaticamente sull'url associata alla pagina di amministrazione e interrogazione di MySQL)



- 5) Per eseguire uno script SQL dall'interfaccia Web di MySQL:
 - a. Selezionare il pannello "Importa"
 - b. Selezionare il file contenente lo script che si intende eseguire e poi cliccare su Esegui
- 6) Per rilanciare più volte lo script di creazione/popoloamento ricordarsi di cancellare eventuali istanze del database creato in precedenza dal pannello Database oppure includere all'inizio dello script i comandi per la cancellazione delle tabelle preesistenti

Creazione e popolamento della base di dati utilizzata per l'esercitazione

La base di dati utilizzata per questa esercitazione è la stessa che avete creato nell'esercitazione precedente. La base di dati si chiama PALESTRA e riguarda le attività di una palestra. Essa è caratterizzata dal seguente schema logico (le chiavi primarie sono sottolineate):

ISTRUTTORE (CodFisc, Nome, Cognome, DataNascita, Email, Telefono*)

CORSI (CodC, Nome, Tipo, Livello)

PROGRAMMA (CodFisc, Giorno, Oralnizio, Durata, CodC, Sala)

- 1) Creare la base di dati PALESTRA e popolarla usando gli script *creaDB.sql* e *popolaDB.sql* presenti sul sito del corso.

Dopo l'esecuzione degli script, le tabelle conterranno i dati seguenti:

Tabella ISTRUTTORE

<u>CodFisc</u>	Nome	Cognome	DataNascita	Email	Telefono
SMTPLA80N31B791Z	Paul	Smith	31/12/1980	p.smith@email.it	NULL
KHNJHN81E30C455Y	John	Johnson	30/5/1981	j.johnson@email.it	+2300110303444
AAAGGG83E30C445A	Peter	Johnson	30/5/1981	p.johnson@email.it	+2300110303444

Tabella CORSI

<u>CodC</u>	Nome	Tipo	Livello
CT100	Spinning principianti	Spinning	1
CT101	Ginnastica e musica	Attività musicale	2
CT104	Spinning professionisti	Spinning	4

Tabella PROGRAMMA

CodFisc	Giorno	OrarioInizio	Durata	CodC	Sala
SMTPLA80N31B791Z	Lunedì	10:00	45	CT100	S1
SMTPLA80N31B791Z	Martedì	11:00	45	CT100	S1
SMTPLA80N31B791Z	Martedì	15:00	45	CT100	S2
KHNJHN81E30C455Y	Lunedì	10:00	30	CT101	S2
KHNJHN81E30C455Y	Lunedì	11:30	30	CT104	S2
KHNJHN81E30C455Y	Mercoledì	9:00	60	CT104	S1

Publiccare/caricare una pagina Web Dinamica

Per pubblicare una pagina dinamica tramite Python che utilizza una base di dati è necessario installare Flask e SQLAlchemy e le necessarie dipendenze.

- instal flask
- install sqlalchemy
- install mysqlclient

Esercizio

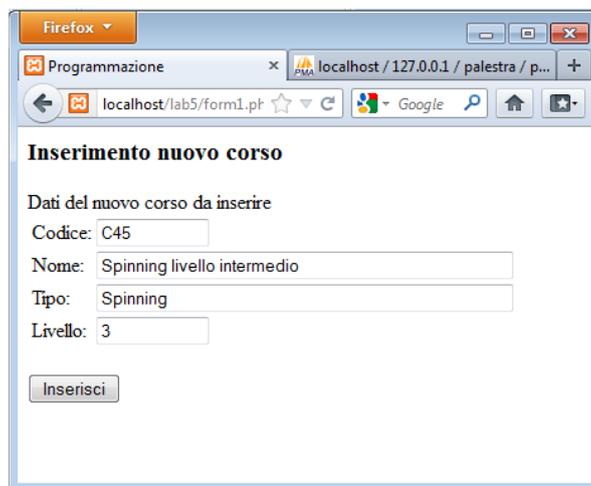
Progettare un'applicazione web in Python per l'aggiornamento online della base di dati PALESTRA.

L'applicazione deve estendere l'applicazione web in Python creata precedentemente in laboratorio fornendo le seguenti nuove funzionalità:

1. *Inserimento di nuovi corsi.* Creare una pagina web contenente una form d'inserimento che richieda tutti i dati necessari all'inserimento di un nuovo corso nella base di dati (CodC, Nome, Tipo, Livello). L'applicazione deve verificare che tutti i campi siano valorizzati e che il valore dell'attributo livello sia un numero intero compreso tra 1 e 4. In caso di dati mancanti, chiave duplicata o valori fuori dall'intervallo consentito per l'attributo livello, l'applicazione deve generare un messaggio d'errore. Se invece i dati inseriti sono corretti e l'operazione d'inserimento va a buon fine, si deve visualizzare un messaggio di corretto inserimento. Figura 1 e la Figura 2 mostrano un esempio dell'uso della presente funzionalità per l'inserimento dei corsi.
2. *Inserimento di una nuova lezione settimana nel programma.* Creare una pagina dinamica in Python e contenente una form per l'inserimento di una nuova lezione settimanale nella tabella PROGRAMMA. La form deve permettere di inserire tutti i campi necessari (CodFisc, Giorno, OrarioInizio, Durata, CodC, Sala) relativi alla programmazione di una nuova lezione. La selezione dell'istruttore deve avvenire tramite un menù a tendina contenente cognome, nome e codice fiscale dei possibili istruttori generato dal contenuto della tabella della base di dati. In modo analogo anche la selezione del corso deve avvenire tramite un menù a tendina popolato dalla base di dati. Gli altri campi sono invece campi testuali popolati manualmente dall'utente.

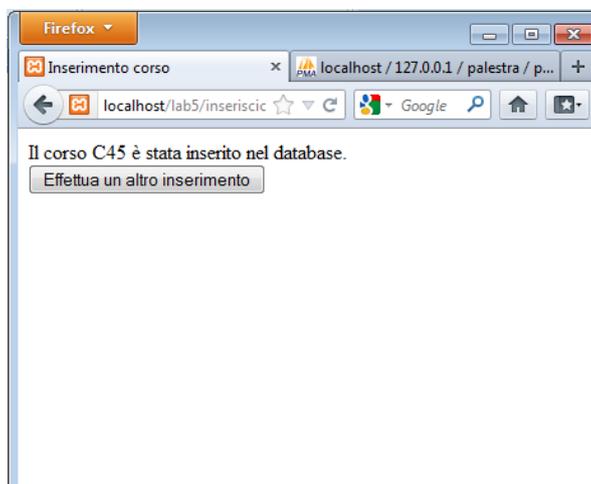
L'applicazione deve verificare che l'utente non cerchi di inserire nel programma lezioni che durino più di 60 minuti e che il giorno indicato sia un giorno compreso tra Lunedì e Venerdì. L'inserimento di una nuova lezione in programma deve essere consentito ed eseguito se e solo se non sono in programma altre lezioni per lo stesso corso nello stesso giorno della settimana. Se la richiesta di inserimento rispetta i vincoli indicati e l'inserimento termina correttamente, si deve visualizzare un messaggio di corretto inserimento, altrimenti si deve notificare un messaggio d'errore (il messaggio d'errore deve riportare il tipo di problema che ha comportato l'errore).

Attenzione: l'interrogazione SQL che serve per verificare che non sia già presente nella tabella programma, per la stessa giornata, una lezione per il corso cui si riferisce la lezione che si sta cercando di inserire, deve essere inserita all'interno di un'unica transazione insieme all'operazione di inserimento della nuova lezione in programma (onde evitare problemi dati dal tentativo di inserimento contemporaneo di nuove lezioni per lo stesso corso per la stessa giornata della settimana da parte di due utenti). Figura 3 e Figura 4 mostrano un esempio dell'uso della presente funzionalità per l'inserimento delle lezioni.



The screenshot shows a Firefox browser window with the address bar at localhost/127.0.0.1/palestra/p... and the page title 'Programmazione'. The main content area is titled 'Inserimento nuovo corso' and contains a form with the following fields: 'Codice: C45', 'Nome: Spinning livello intermedio', 'Tipo: Spinning', and 'Livello: 3'. There is an 'Inserisci' button at the bottom of the form.

Figura 1 - Inserimento nuovo corso.



The screenshot shows a Firefox browser window with the address bar at localhost/127.0.0.1/palestra/p... and the page title 'Inserimento corso'. The main content area displays the message 'Il corso C45 è stata inserito nel database.' and a button labeled 'Effettua un altro inserimento'.

Figura 2 – Messaggio inserimento riuscito con successo.

Firefox

Programmazione localhost / 127.0.0.1 / palestra / p...

localhost/lab5/form2.pt

Inserimento nuova programmazione/lezione

Programmazione della nuova lezione settimanale

Istruttore: Johnson John (KHNJHN81E30C455Y)

Giorno: Giovedì

Ora inizio: 11:30

Durata: 45

Istruttore: Spinning livello intermedio (C45)

Sala: S4

Inserisci

Figura 3 - Inserimento nuova lezione settimanale nel programma.

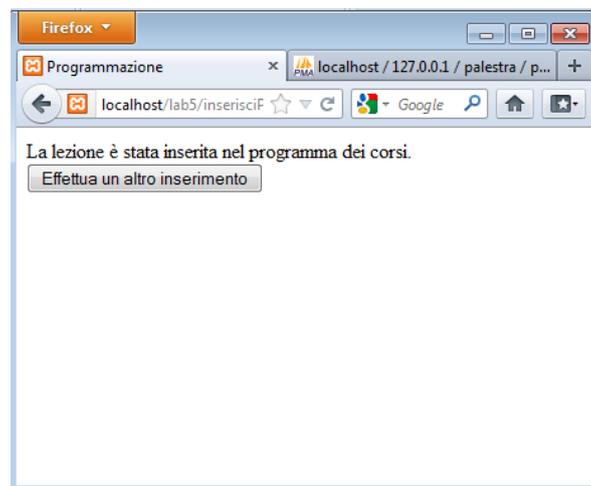


Figura 4 - Messaggio inserimento riuscito con successo.