

# Basi di Dati

## Applicazioni web in PHP per l'interrogazione di basi di dati - Esercitazione n. 5

### Introduzione

Questa esercitazione ha come obiettivo quello di realizzare una semplice applicazione web basata su PHP per interrogare una base di dati.

### 1 Passi preliminari per lo svolgimento dell'esercitazione

Questa esercitazione usa il server WEB Apache e il database MySQL disponibile nel prodotto XAMPP. Al fine di poter svolgere l'esercitazione è necessario avviare entrambi i servizi.

#### 1.1 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 ad Apache
3. Avviare MySQL premendo il tasto Start nella riga MySQL
4. Aprire l'interfaccia web di MySQL (phpMyAdmin) 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:
  - Selezionare il pannello Importa
  - Selezionare il file contenente lo script che si intende eseguire e poi cliccare su Esegui

## 1.2 Pubblicazione/caricamento di una pagina PHP in XAMPP

Il server web Apache presente in XAMPP, quando è attivo, usa come cartella principale la cartella *htdocs*. Questo significa che per richiedere il file *index.php*, si deve navigare all'indirizzo **http://localhost/index.php** e il file deve risiedere nel proprio filesystem al percorso **[XAMPP]/htdocs/index.php**. La cartella *htdocs* si trova nella cartella principale delle risorse di XAMPP. In Windows, di default, è salvata al percorso *C:\XAMPP\htdocs*.

## 2 Creazione e popolamento della base di dati

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):

```
1 ISTRUTTORE (CodFisc, Nome, Cognome, DataNascita, Email, Telefono*)
2 CORSI (CodC, Nome, Tipo, Livello)
3 PROGRAMMA (CodFisc, Giorno, OraInizio, Durata, CodC, Sala)
```

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 del database conterranno i dati riportati nelle Tabelle 1, 2 e 3.

**Info:** 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.

CodFisc	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 1: Tabella ISTRUTTORE

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

Tabella 2: Tabella CORSI

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

Tabella 3: Tabella PROGRAMMA

## 3 Esercizi

**Assicurarsi di aver seguito gli step preliminari: la base dati deve essere presente e popolata correttamente, mentre i file PHP su cui si lavora devono trovarsi nella cartella htdocs.**

Creare un'applicazione web in PHP che consenta sia l'inserimento di un nuovo corso sia l'inserimento di una nuova programmazione per uno tra quelli esistenti. I passi per lo sviluppo sono elencati di seguito.

**Info:** Il metodo PHP `mysqli_connect()` richiede un username e una password come argomenti. Per semplicità, ai fini dell'esercitazione, si possono utilizzare quelli di default:

- **username:** 'root'
- **password:** " (stringa vuota)

### 3.1 Inserimento di un nuovo corso

Creare un form che richiede tutti i dati necessari all'inserimento di un nuovo corso nella base dati. La pagina di risultato deve valutare il corretto inserimento dei dati da parte dell'utente verificando:

- se sono stati inseriti tutti i campi;
  - se il corso inserito non è già presente;
  - se i tipi di dato non sono coerenti (ad esempio il livello alfanumerico anziché numerico).
- Questi controlli sono da effettuare tramite l'utilizzo di funzioni apposite.**

Se l'inserimento termina correttamente, si deve visualizzare un messaggio di corretto inserimento, altrimenti si deve notificare l'errore.

### 3.2 Aggiunta delle classi CSS

Per migliorare l'esperienza dell'utente, l'esito dell'inserimento del corso deve essere stilizzato appropriatamente. Utilizzare le classi CSS per creare un banner colorato in verde/rosso nella pagina restituita dal server. Nelle Figure 1 e 2 è presente un esempio di applicazione.



**Figura 1:** Un esempio di inserimento riuscito.




**Figura 2:** Un esempio di inserimento fallito.

### 3.3 Inserimento di una nuova programmazione

Creare un form che permetta di inserire la programmazione per uno dei corsi esistenti. Il form deve permettere di scegliere, tramite due menu a tendina, il codice fiscale dell'istruttore incaricato e il codice del corso per cui si sta inserendo la programmazione. I menu devono proporre gli istruttori e i corsi già presenti nella base dati. La pagina di risultato deve valutare il corretto inserimento dei dati da parte dell'utente verificando:

- se sono stati inseriti tutti i campi;
- se si verificano inconsistenze tra i valori indicati come orario d'inizio e di fine attività (ad esempio, ora d'inizio 18:30 e ora di fine 15:00<sup>1</sup>),
- se i tipi di dato non sono coerenti (ad esempio il giorno indicato non è uno dei giorni della settimana).
- se si verificano problemi di compatibilità con le programmazioni già presenti. In particolare, verificare se, nel giorno e nell'ora indicate, la sala selezionata non è occupata e se l'istruttore selezionato non è occupato in un altro corso.

**Questi controlli sono da effettuare tramite l'utilizzo di funzioni apposite.** Se l'inserimento non va a buon fine, la pagina deve notificare l'errore.



**Inserimento nuova programmazione**

Giorno: Lunedì

Ora inizio: 18:00

Ora fine: 19:00

Sala: 3

Codice istruttore: MTTSRG41L23G273B

Codice corso: CT105

**Figura 3:** Un esempio del form richiesto per l'inserimento di una nuova programmazione.

---

<sup>1</sup>In PHP, le date o le ore possono essere verificate (controllo del tipo) e confrontate (controllo di inconsistenze) usando la funzione `strtotime($stringa)`, che permette di convertire un orario fornito in formato stringa in uno in formato `time` e quindi confrontabile. Se la stringa inserita NON è in un formato corretto, la funzione ritorna `FALSE`, altrimenti la stringa convertita in formato `time`.

## 4 Gestione dei caratteri accentati e della codifica

La scelta della codifica dei dati testuali specifica con quale formato (ovvero, con quali e quanti byte) vengono salvati i caratteri nella base di dati. Tale scelta è fondamentale per gestire casi di caratteri non presenti nell'alfabeto inglese, come gli accenti o altri segni diacritici. In questa esercitazione verrà utilizzata la codifica **utfmb4**, che rispetta lo standard 4-Byte UTF-8 Unicode Encoding. Il file fornito con l'esercitazione crea un base di dati che utilizza questa codifica:

```
creaDB.sql
...
CREATE DATABASE IF NOT EXISTS PALESTRA;
SET NAMES = 'utf8mb4';
...
```

A questo punto, per poter interrogare correttamente la base dati è necessario specificare, anche nel codice PHP, la codifica che si intende utilizzare per comunicare con il server MySQL. Nella versione procedurale delle API, si usa il metodo **mysqli\_set\_charset()**. Il metodo può essere invocato:

```
process_request.php
...
mysqli_set_charset($conn, 'utf8mb4');
...
```

Infine, poichè sia il database che lo script PHP lavorano in codifica UTF-8, è **consigliabile che anche il testo sorgente HTML sia nello stesso formato**. Può succedere, infatti, che definendo dei valori staticamente che contengono, ad esempio, caratteri accentati, questi vengano codificati erroneamente. È quindi consigliabile utilizzare un editor di testo moderno (Visual Studio Code, Atom, Sublime Text ed altri) in cui è possibile specificare facilmente la codifica desiderata.

## 5 Soluzioni

### 5.1 Creazione della base dati

```
1  -- create an empty database. Name of the database:
2  SET storage_engine=InnoDB;
3  SET FOREIGN_KEY_CHECKS=1;
4  CREATE DATABASE IF NOT EXISTS palestra;
5  SET NAMES 'utf8mb4';
6
7  -- use palestra
8  use palestra;
9
10
11 -- drop tables if they already exist
12 DROP TABLE IF EXISTS PROGRAMMA;
13 DROP TABLE IF EXISTS CORSI;
14 DROP TABLE IF EXISTS ISTRUTTORE;
15
16 -- create tables
17
18 CREATE TABLE Istruttore (
19     CodFisc CHAR(20) ,
20     Nome CHAR(50) NOT NULL ,
21     Cognome CHAR(50) NOT NULL ,
22     DataNascita DATE NOT NULL ,
23     Email CHAR(50) NOT NULL ,
24     Telefono CHAR(20) NULL ,
25     PRIMARY KEY (CodFisc)
26 );
27
28 CREATE TABLE Corsi (
29     CodC CHAR(10) ,
30     Nome CHAR(50) NOT NULL ,
31     Tipo CHAR(50) NOT NULL ,
32     Livello SMALLINT NOT NULL,
33     PRIMARY KEY (CodC),
34     CONSTRAINT chk_Livello CHECK (Livello>=1 and Livello<=4)
35 );
36
37 CREATE TABLE Programma (
38     CodFisc CHAR(20) NOT NULL ,
39     Giorno CHAR(15) NOT NULL ,
40     OraInizio TIME NOT NULL ,
41     Durata SMALLINT NOT NULL ,
42     Sala CHAR(5) NOT NULL,
43     CodC CHAR(10) NOT NULL,
44     PRIMARY KEY (CodFisc,Giorno,OraInizio),
45     FOREIGN KEY (CodFisc)
46         REFERENCES Istruttore(CodFisc)
47         ON DELETE CASCADE
48         ON UPDATE CASCADE,
49     FOREIGN KEY (CodC)
50         REFERENCES Corsi(CodC)
51         ON DELETE CASCADE
52         ON UPDATE CASCADE
53 );
```

## 5.2 Popolamento della base dati

```
1 SET storage_engine=InnoDB;
2 SET FOREIGN_KEY_CHECKS=1;
3 use palestra;
4
5 -- Insert data
6 INSERT INTO Istruttore (CodFisc, Nome, Cognome, DataNascita, Email, Telefono)
7 VALUES ('SMTPLA80N31B791Z', 'Paul', 'Smith', '1980-12-31', 'p.smith@email.it', NULL);
8 INSERT INTO Istruttore (CodFisc, Nome, Cognome, DataNascita, Email, Telefono)
9 VALUES ('KHNJHN81E30C455Y', 'John', 'Johnson', '1981-05-30',
10         'j.johnson@email.it', '+2300110303444');
11 INSERT INTO Istruttore (CodFisc, Nome, Cognome, DataNascita, Email, Telefono)
12 VALUES ('AAAGGG83E30C445A', 'Peter', 'Johnson', '1981-05-30',
13         'p.johnson@email.it', '+2300110303444');
14 INSERT INTO Corsi (CodC, Nome, Tipo, Livello)
15 VALUES ('CT100', 'Spinning principianti', 'Spinning ', 1);
16 INSERT INTO Corsi (CodC, Nome, Tipo, Livello)
17 VALUES ('CT101', 'Ginnastica e musica', 'Attivit`a musicale', 2);
18 INSERT INTO Corsi (CodC, Nome, Tipo, Livello)
19 VALUES ('CT104', 'Spinning professionisti', 'Spinning', 4);
20 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
21 VALUES ('SMTPLA80N31B791Z', 'Lunedì', '10:00', 45, 'S1', 'CT100');
22 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
23 VALUES ('SMTPLA80N31B791Z', 'Martedì', '11:00', 45, 'S1', 'CT100');
24 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
25 VALUES ('SMTPLA80N31B791Z', 'Martedì', '15:00', 45, 'S2', 'CT100');
26 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
27 VALUES ('KHNJHN81E30C455Y', 'Lunedì', '10:00', 30, 'S2', 'CT101');
28 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
29 VALUES ('KHNJHN81E30C455Y', 'Lunedì', '11:30', 30, 'S2', 'CT104');
30 INSERT INTO Programma (CodFisc, Giorno, OraInizio, Durata, Sala, CodC)
31 VALUES ('KHNJHN81E30C455Y', 'Mercoledì', '9:00', 60, 'S1', 'CT104');
```

## 5.3 Inserimento di un nuovo corso

### 1. Front-end

```
1 <html>
2 <head>
3 <title>Programmazione</title>
4 </head>
5
6 <body>
7 <h3>Inserimento nuovo corso</h3>
8 <form method="get" action="inseriscicorso.php">
9
10 <p>Dati del nuovo corso da inserire<br>
11
12 <table>
13 <tr>
14 <td> Codice: </td>
15 <td> <input type="text" size="10" maxlength="10" name="codc"> </td>
16 </tr>
17 <tr>
18 <td> Nome: </td>
19 <td> <input type="text" size="50" maxlength="50" name="nome"> </td>
```

```

20 </tr>
21 <tr>
22 <td> Tipo: </td>
23 <td> <input type="text" size="50" maxlength="50" name="tipo"> </td>
24 </tr>
25 <tr>
26 <td> Livello: </td>
27 <td> <input type="text" size="10" maxlength="10" name="livello"> </td>
28 </tr>
29 </table>
30
31 <br>
32 <input type="submit" value="Inserisci">
33
34 </form>
35 </body>
36 </html>

```

## 2. Back-end

```

1 <html>
2 <head>
3 <title>Inserimento corso</title>
4 </head>
5
6 <body>
7 <?php
8
9 /* CONTROLLO E LETTURA PARAMETRI*/
10
11 if( !isset($_REQUEST["codc"]) or
12 !isset($_REQUEST["nome"]) or
13 !isset($_REQUEST["tipo"]) or
14 !isset($_REQUEST["livello"])){
15     echo "<div class = 'w3-red w3-text-white'>
16         <h2>Errore di inserimento!</h2>
17         <p>Errore: inserire tutti i dati richiesti</p>
18         </div>";
19     exit;
20 }
21
22 $codc = $_REQUEST["codc"];
23 $nome = $_REQUEST["nome"];
24 $tipo = $_REQUEST["tipo"];
25 $livello = $_REQUEST["livello"];
26
27 if ($codc=="")
28     echo "<div class = 'w3-red w3-text-white'>
29         <h2>Errore di inserimento!</h2>
30         <p>Errore: Codice corso non valorizzato</p>
31         </div>";
32
33 if ($nome=="")
34     echo "<div class = 'w3-red w3-text-white'>
35         <h2>Errore di inserimento!</h2>
36         <p>Errore: Nome corso non valorizzato</p>
37         </div>";
38

```



```

39 if ($tipo=="")
40     echo "<div class = 'w3-red w3-text-white'>
41         <h2>Errore di inserimento!</h2>
42         <p>Errore: Tipo corso non valorizzato</p>
43     </div>";
44
45 if ($livello=="")
46     echo "<div class = 'w3-red w3-text-white'>
47         <h2>Errore di inserimento!</h2>
48         <p>Errore: Livello corso non valorizzato</p>
49     </div>";
50
51
52 if ($codc=="" || $nome=="" || $tipo=="" || $livello=="")
53     die();
54
55 if (!is_numeric($livello))
56 {
57     echo "<div class = 'w3-red w3-text-white'>
58         <h2>Errore di inserimento!</h2>
59         <p>Errore: Il livello deve essere un valore intero
60         compreso tra 1 e 4</p>
61     </div>";
62     exit;
63 }
64
65 if ($livello<1 || $livello>4)
66 {
67     echo "<div class = 'w3-red w3-text-white'>
68         <h2>Errore di inserimento!</h2>
69         <p>Errore: Il livello deve essere un valore intero
70         compreso tra 1 e 4</p>
71     </div>";
72     exit;
73 }
74
75
76 /* CONNESSIONE AL DB*/
77
78 $con = mysqli_connect('localhost','root','','palestra');
79
80 if (mysqli_connect_errno())
81 {
82     echo "Failed to connect to MySQL: " . mysqli_connect_error();
83 }
84
85 mysqli_set_charset($con, 'utf8mb4');
86 /* ESECUZIONE INSERIMENTO*/
87
88 if(mysqli_query($con,"INSERT INTO CORSI(CodC,Nome,Tipo,Livello)
89 VALUES('$codc', '$nome', '$tipo', '$livello');"))
90 {
91     echo "<div class = 'w3-green w3-text-white'>
92         <h2>Inserimento avvenuto!</h2>
93         <p>Il corso $codc è stato inserito nel database.</p>
94     </div>";
95 }
96 else

```

```

97 {
98     echo "<div class = 'w3-red w3-text-white'>
99         <h2>Errore di inserimento!</h2>
100         <p>Errore: Non è stato possibile inserire i dati
101         perchè si è verificato un errore:</p>
102         </div>";
103     echo mysqli_error($con);
104     exit;
105 }
106
107 mysqli_close($con);
108 ?>
109
110 </br>
111     <form method="get" action="form1.php">
112         <input type="submit" value="Effettua un altro inserimento">
113     </form>
114
115 </body>
116 </html>

```

## 5.4 Inserimento programmazione

### 1. Front-end

```

1 <html>
2 <head>
3 <title>Programmazione</title>
4 </head>
5
6 <body>
7 <h3> Inserimento nuova programmazione/lezione </h3>
8 <form method="get" action="inserisciProgrammazione.php">
9
10 Programmazione della nuova lezione settimanale<br>
11
12 <table>
13     <tr>
14         <td> Istruttore: </td>
15         <td> <select name="codfisc">
16             <?php
17                 /* CONNESSIONE AL DB*/
18                 $con = mysqli_connect('localhost','root','','palestra');
19
20                 if (mysqli_connect_errno())
21                 {
22                     echo "Failed to connect to MySQL: " . mysqli_connect_error();
23                 }
24
25                 mysqli_set_charset($con, 'utf8mb4');
26
27                 /* ESECUZIONE QUERY */
28                 $sql = "SELECT CodFisc, Cognome, Nome
29                 FROM istruttore
30                 ORDER BY Cognome, Nome";
31
32                 $result = mysqli_query($con,$sql);

```

```

33
34     if( !$result )
35         die('Query error: ' . mysqli_error($con));
36
37
38     /* Caricamento dei risultati nel menù a tendina */
39
40     if( mysqli_num_rows($result) > 0 ){
41
42         while ($row = mysqli_fetch_row($result)) {
43             echo "<option value=$row[0]>$row[1] $row[2] ($row[0])</option>";
44         }
45
46     }
47
48     ?>
49     </select>
50 </td>
51 </tr>
52
53 <tr>
54     <td> Giorno: </td>
55     <td> <input type="text" size="10" maxlength="10" name="giorno"> </td>
56 </tr>
57 <tr>
58     <td> Ora inizio: </td>
59     <td> <input type="text" size="10" maxlength="10" name="orainizio"> </td>
60 </tr>
61 <tr>
62     <td> Durata: </td>
63     <td> <input type="text" size="10" maxlength="10" name="durata"> </td>
64 </tr>
65 <tr>
66     <td> Corso: </td>
67     <td>
68         <select name="codc">
69 <?php
70
71     /* CONNESSIONE AL DB*/
72
73     $con = mysqli_connect('localhost','root','','palestra');
74
75     if (mysqli_connect_errno())
76     {
77         echo "Failed to connect to MySQL: " . mysqli_connect_error();
78     }
79
80     /* ESECUZIONE QUERY */
81     $sql = "SELECT CodC, Nome
82     FROM corsi
83     ORDER BY Nome";
84
85     $result = mysqli_query($con,$sql);
86
87     if( !$result )
88         die('Query error: ' . mysqli_error($con));
89
90

```

```

91      /* Caricamento dei risultati nel menù a tendina */
92
93      if( mysqli_num_rows($result) > 0 ){
94
95          while ($row = mysqli_fetch_row($result)) {
96              echo "<option value=$row[0]>$row[1] ($row[0])</option>";
97          }
98
99      }
100
101      ?>
102      </select>
103      </td>
104  </tr>
105
106  <tr>
107      <td> Sala: </td>
108      <td> <input type="text" size="2" maxlength="2" name="sala"> </td>
109  </tr>
110
111 </table>
112
113 <p> <input type="submit" value="Inserisci">
114 </form>
115 </body>
116 </html>

```

## 2. Back-end

```

1 <html>
2 <head>
3 <title>Programmazione</title>
4 <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
5 </head>
6
7 <body>
8
9
10
11 <?php
12
13 /* CONTROLLO E LETTURA PARAMETRI*/
14
15 if( !isset($_REQUEST["codfisc"]) or
16     !isset($_REQUEST["giorno"]) or
17     !isset($_REQUEST["orainizio"]) or
18     !isset($_REQUEST["durata"]) or
19     !isset($_REQUEST["codc"]) or
20     !isset($_REQUEST["sala"])) {
21     echo "<div class = 'w3-red w3-text-white">
22         <h2>Errore di inserimento!</h2>
23         <p>Errore: inserire tutti i dati richiesti</p>
24     </div>";
25     exit;
26 }
27
28 $codfisc = $_REQUEST["codfisc"];
29 $giorno = $_REQUEST["giorno"];

```

```

30 if($giorno!="Lunedì" and $giorno!="Martedì" and $giorno!="Mercoledì" and
    $giorno!="Giovedì" and $giorno!="Venerdì")
31 {
32     echo "<div class = 'w3-red w3-text-white'>
33         <h2>Errore di inserimento!</h2>
34         <p>Errore: Il giorno specificato non è valido</p>
35     </div>";
36     exit;
37 }
38
39 $orainizio = $_REQUEST["orainizio"];
40 $durata = $_REQUEST["durata"];
41
42 if ($durata>60)
43 {
44     echo "<div class = 'w3-red w3-text-white'>
45         <h2>Errore di inserimento!</h2>
46         <p>Errore: Le lezioni non possono durare più di 60 minuti</p>
47     </div>";
48     exit;
49 }
50
51 $codc = $_REQUEST["codc"];
52
53 $sala = $_REQUEST["sala"];
54
55
56 /* CONNESSIONE AL DB*/
57
58 $con = mysqli_connect('localhost','root','','palestra');
59
60 if (mysqli_connect_errno())
61 {
62     echo "Failed to connect to MySQL: " . mysqli_connect_error();
63 }
64
65 mysqli_set_charset($con, 'utf8mb4');
66
67 /* ESECUZIONE QUERY PER LA VERIFICA DELLA CONDIZIONE DA VERIFICARE E POI INSERIMENTO
    LEZIONE.
68     Fanno parte di una singola transazione */
69 // Imposto a FALSE l'autocommit e inizio la transazione
70 mysqli_query($con,"SET autocommit=0;");
71 mysqli_query($con,"START TRANSACTION;");
72
73 /* ESECUZIONE QUERY */
74 $sql = "SELECT count(*)
75 FROM programma
76 where CodC='$codc' and Giorno='$giorno' and OraInizio='$orainizio'";
77
78 $result = mysqli_query($con,$sql);
79
80 if( !$result )
81     die('Query error: ' . mysqli_error($con));
82
83
84 // Test sul numero di lezioni del corso nel giorno indicato alla stessa ora
85 $row = mysqli_fetch_row($result);

```

```

86 if ($row[0]!=0) {
87     echo "<div class = 'w3-red w3-text-white'>
88         <h2>Errore di inserimento!</h2>
89         <p>Non è
90             consentito inserire due lezioni per lo stesso corso nello stesso
91             giorno e alla stessa ora</p>
92         </div>";
93     mysqli_query($con,"ROLLBACK;");
94 }
95 else
96 {
97     //Test sull'occupazione della sala
98     $sql = "SELECT count(*)
99     FROM programma
100     where Sala='$sala' and Giorno='$giorno' and OraInizio='$orainizio'";
101
102     $result = mysqli_query($con,$sql);
103
104     if( !$result )
105         die('Query error: ' . mysqli_error($con));
106
107     $row = mysqli_fetch_row($result);
108     if ($row[0]!=0) {
109         echo "<div class = 'w3-red w3-text-white'>
110             <h2>Errore di inserimento!</h2>
111             <p>Non è consentito inserire due lezioni per la stessa sala nello
112             stesso giorno e alla stessa ora</p>
113             </div>";
114         mysqli_query($con,"ROLLBACK;");
115     }
116
117     else{
118         //Test sulla disponibilità dell'istruttore
119         $sql = "SELECT count(*)
120         FROM programma
121         where CodFisc='$codfisc' and Giorno='$giorno' and OraInizio='$orainizio'";
122
123         $result = mysqli_query($con,$sql);
124
125         if( !$result )
126             die('Query error: ' . mysqli_error($con));
127
128
129         $row = mysqli_fetch_row($result);
130         if ($row[0]!=0) {
131             echo "<div class = 'w3-red w3-text-white'>
132                 <h2>Errore di inserimento!</h2>
133                 <p>Non è consentito inserire due lezioni per lo stesso insegnate
134                 nello stesso giorno e alla stessa ora</p>
135                 </div>";
136             mysqli_query($con,"ROLLBACK;");
137         }
138         else{
139             // Inserimento nuova lezione
140             if(mysqli_query($con,"INSERT INTO Programma
141                 (CodFisc,Giorno,OraInizio,Durata,CodC,Sala)

```

```

140     VALUES ('$codfisc','$giorno','$sorainizio','$durata','$codc','$sala');"))
141 {
142     mysqli_query($con,"COMMIT;");
143     echo "<div class = 'w3-green w3-text-white'>
144         <h2>Inserimento avvenuto!</h2>
145         <p>La lezione è stata inserita nel programma dei corsi.</p>
146     </div>";
147 }
148 else
149 {
150     echo "<div class = 'w3-red w3-text-white'>
151         <h2>Errore di inserimento!</h2>
152         <p>Non è stato possibile inserire i dati perchè si è verificato
153             un errore</p>
154     </div>";
155     echo mysqli_error($con);
156     mysqli_query($con,"ROLLBACK;");
157 }
158 }
159 }
160
161 mysqli_close($con);
162
163 ?>
164
165 </br>
166 <form method="get" action="form2.php">
167     <input type="submit" value="Effettua un altro inserimento">
168 </form>
169
170
171 </body>
172 </html>

```