

*Corso di Laurea in Ingegneria Gestionale  
Sapienza Università di Roma*

# ***Corso di Basi di Dati***

## ***A.A. 2018/2019***

**MySQL: Creazione di Schemi ed  
Interrogazioni**

Francesco Leotta

Ultimo aggiornamento : 21/10/2018

# Il DBMS che utilizzeremo: MySQL

---

- ▶ MySQL è un *RDBMS* open-source disponibile per il download su <http://dev.mysql.com/downloads/mysql/>
- ▶ E' composto da un client con interfaccia a caratteri e un server.
- ▶ Nelle esercitazioni si farà riferimento alla versione MySQL 5.X per Windows.
  - È possibile utilizzare MySQL anche su Linux e MacOS.
  - MySQL 5 presenta sostanziali cambiamenti rispetto alle versioni precedenti (es. introduce la possibilità di definire viste).
  - **Le versioni precedenti alla 5 di MySQL non sono da considerarsi adatte.**
    - *Per l'installazione del DBMS, si prega di fare riferimento al PDF contenente la procedura autoguidata e disponibile per il download sulla pagina web del docente.*

# Lanciare il client MySQL

---

- ▶ Per lanciare il client MySQL è sufficiente richiamarlo dal prompt dei comandi indicando **nome utente** e **password** (la stessa indicata durante la procedura di installazione):

```
shell> mysql --user=root --password=xxxx
```

- ▶ oppure:

```
shell> mysql -uroot -p
```

- ▶ Accedere al DBMS come utente *root* significa disporre dei permessi di Amministratore di Sistema.

# Chiusura del client MySQL

---

- ▶ Una volta connessi alla base di dati appare il prompt di MySQL:

```
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 5  
Server version: 5.1.32-community MySQL Community Server (GPL)  
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.  
mysql>
```

- ▶ Per chiudere il client MySQL digitiamo:

```
mysql> quit
```

# Note sull'utilizzo di MySQL

---

- ▶ MySQL **NON E' case sensitive**.
- ▶ Ogni istruzione **DEVE** essere seguita dal ‘;’ per essere interpretata correttamente.
- ▶ Un DBMS permette solo alle istanze legali (quelle che soddisfano tutti i vincoli di integrità) di essere memorizzate in una base di dati.

# I permessi in MySQL

---

- ▶ Una volta connessi al server, un utente deve possedere i permessi necessari per lavorare sui vari database.
- ▶ Per chiedere quali basi di dati gestite dal DBMS sono accessibili dall'utente **root** si può utilizzare il comando *show databases*

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| test |
+-----+
3 rows in set (0.00 sec)
```

# Creazione e cancellazione di una base di dati 1\2

---

- ▶ Per poter creare un nuovo database su cui lavorare, si può utilizzare il comando:

***CREATE DATABASE [IF NOT EXISTS] nome\_database***

- ▶ Con l'opzione *IF NOT EXISTS* possiamo evitare la segnalazione di errore nel caso esista già un database con lo stesso nome.

- ▶ Per eliminare un database si utilizza l'istruzione:

***DROP DATABASE [IF EXISTS] nome\_database***

- ▶ Con l'opzione *IF EXISTS* possiamo evitare la segnalazione di errore nel caso non esista un database chiamato *nome\_database*.

- ▶ **ATTENZIONE**

- ▶ Le istruzioni contenute tra parentesi quadre [...] sono facoltative
- ▶ **Le parentesi [ e ] non vanno MAI inserite in un comando!!!!**

# Creazione e cancellazione di una base di dati 2\2

---

- ▶ Creiamo un database denominato “*ProgEuropei*” utilizzando il client MySQL.
- ▶ L’istruzione da utilizzare è:

```
mysql> CREATE DATABASE PROGEUROPEI;  
Query OK, 1 row affected (0.00 sec)
```

- ▶ Ora i database gestiti dall’utente **root** sono:

```
mysql> SHOW DATABASES;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| progeuropei |  
+-----+  
4 rows in set (0.00 sec)
```



# Uso e Creazione di una tabella

---

- ▶ Per selezionare il database su cui effettuare le modifiche si usi il comando *USE nome\_database*

```
mysql> USE PROGEUROPEI;  
Database changed
```

- ▶ Per visualizzare la tabelle contenute in un database, si può utilizzare il comando *show tables*

```
mysql> show tables;  
Empty set (0.00 sec)
```

- ▶ Si cominci creando la la seguente tabella :
  - ▶ **Professore(codiceProfessore, nome, cognome, dipartimento)**
    - ▶ *codiceProfessore* è una stringa di 3 caratteri ;
    - ▶ *nome* è una stringa di massimo 20 caratteri ;
    - ▶ *cognome* è una stringa di massimo 20 caratteri ;
    - ▶ *dipartimento* è una stringa di 1 carattere.

# Creazione di una tabella

```
mysql> CREATE TABLE Professore(  
-> CodiceProfessore CHAR(3) PRIMARY KEY,  
-> Nome VARCHAR(20),  
-> Cognome VARCHAR(20),  
-> Dipartimento CHAR  
-> );  
Query OK, 0 rows affected (0.06 sec)
```

- ▶ Per visualizzare lo schema e i vincoli definiti su di una tabella (il suo livello intensionale) si può utilizzare l'istruzione :

***DESCRIBE nome\_tabella***

```
mysql> DESCRIBE Professore;  
+-----+-----+-----+-----+-----+-----+  
| Field          | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| CodiceProfessore | char(3)       | NO   | PRI | NULL    |       |  
| Nome           | varchar(20)   | YES  |     | NULL    |       |  
| Cognome        | varchar(20)   | YES  |     | NULL    |       |  
| Dipartimento    | char(1)       | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.05 sec)
```

# Vincoli di Integrità

- ▶ Aggiungere alla tabella Professore un vincolo di DEFAULT per l'attributo Codice Professore.

```
mysql> ALTER TABLE Professore  
-> ALTER COLUMN CodiceProfessore  
-> SET DEFAULT 'AAA';  
Query OK, 0 rows affected (0.03 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE PROFESSORE;
```

Field	Type	Null	Key	Default	Extra
CodiceProfessore	char(3)	NO	PRI	AAA	
Nome	varchar(20)	YES		NULL	
Cognome	varchar(20)	YES		NULL	
Dipartimento	char(1)	YES		NULL	

```
4 rows in set (0.02 sec)
```

# Vincoli di Integrità

- ▶ Eliminare alla tabella Professore il vincolo di DEFAULT per l'attributo Codice Professore.

```
mysql> ALTER TABLE Professore  
-> ALTER COLUMN CodiceProfessore  
-> DROP DEFAULT;  
Query OK, 0 rows affected (0.00 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Professore;
```

Field	Type	Null	Key	Default	Extra
CodiceProfessore	char(3)	NO	PRI	NULL	
Nome	varchar(20)	YES		NULL	
Cognome	varchar(20)	YES		NULL	
Dipartimento	char(1)	YES		NULL	

```
4 rows in set (0.01 sec)
```

# Creazione di una tabella

---

- ▶ Si crei adesso la seguente tabella :
  - ▶ **Progetto (codiceProgetto, nome, enteFinanziatore)**
    - ▶ *codiceProgetto* è una stringa di 3 caratteri ;
    - ▶ *nome* è una stringa di massimo 20 caratteri ;
    - ▶ *EnteFinanziatore* è una stringa di massimo 20 caratteri.

```
mysql> CREATE TABLE Progetto(  
-> CodiceProgetto CHAR(3) PRIMARY KEY,  
-> Nome VARCHAR(20),  
-> EnteFinanziatore VARCHAR(20)  
-> );  
Query OK, 0 rows affected (0.03 sec)
```

# Vincoli di Integrità

- ▶ Eliminare dalla tabella Progetto il vincolo di chiave primaria.

```
mysql> ALTER TABLE Progetto
-> DROP PRIMARY KEY;
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> DESCRIBE Progetto;
+-----+-----+-----+-----+-----+-----+
| Field                | Type          | Null  | Key  | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| CodiceProgetto       | char(3)       | NO    |     | NULL    |      |
| Nome                 | varchar(20)   | YES   |     | NULL    |      |
| EnteFinanziatore     | varchar(20)   | YES   |     | NULL    |      |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

# Vincoli di Integrità

- ▶ Inserire nuovamente nella tabella Progetto il vincolo di chiave primaria sull'attributo CodiceProgetto.

```
mysql> ALTER TABLE progetto  
-> add PRIMARY KEY(codiceprogetto);  
Query OK, 0 rows affected (0.00 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> DESCRIBE Progetto;
```

Field	Type	Null	Key	Default	Extra
CodiceProgetto	char(3)	NO	PRI	NULL	
Nome	varchar(20)	YES		NULL	
EnteFinanziatore	varchar(20)	YES		NULL	

3 rows in set (0.00 sec)

# Creazione di una tabella

- ▶ Si crei la seguente tabella :
  - ▶ **Partecipa** (codiceProfessore, codiceProgetto, dataInizio, dataFine)
    - ▶ *Partecipa.codiceProfessore* è legato all'attributo *codiceProfessore* della tabella Professore;
    - ▶ *Partecipa.codiceProgetto* è legato all'attributo *codiceProgetto* della tabella Progetto.

```
mysql> CREATE TABLE Partecipa(  
-> CodiceProfessore CHAR(3),  
-> CodiceProgetto CHAR(3),  
-> DataInizio DATE,  
-> DataFine DATE,  
-> PRIMARY KEY(CodiceProfessore, CodiceProgetto),  
-> FOREIGN KEY(CodiceProfessore) REFERENCES Professore(CodiceProfessore),  
-> FOREIGN KEY(CodiceProgetto) REFERENCES Progetto(CodiceProgetto)  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> DESCRIBE Partecipa;
```

Field	Type	Null	Key	Default	Extra
CodiceProfessore	char(3)	NO	PRI		
CodiceProgetto	char(3)	NO	PRI		
DataInizio	date	YES		NULL	
DataFine	date	YES		NULL	

4 rows in set (0.00 sec)



# Inserimento dei dati nelle tabelle

---

- ▶ Per inserire dei dati in una tabella si utilizza l'istruzione:

***INSERT INTO nome\_tabella [(nome\_attributo1,nome\_attributo2,...)]  
VALUES (valore1,valore2,...)***

## **Attenzione:**

- ▶ L'ordinamento degli attributi (se presente) e dei valori è significativo;
- ▶ Le due liste di attributi e di valori devono avere lo stesso numero di elementi;
- ▶ Se la lista di attributi è omessa, si fa riferimento a tutti gli attributi della relazione secondo l'ordine con cui sono stati definiti;
- ▶ Se la lista di attributi non contiene tutti gli attributi della relazione, per gli altri viene inserito un valore nullo (che deve essere permesso) o un valore di default.

# Inserimento dei dati nelle tabelle

- ▶ Inserire nella tabella Professore le tuple necessarie affinché la tabella stessa risulti popolata nel seguente modo :

CodiceProfessore	Nome	Cognome	Dipartimento
000	Andrea	Marrella	A
001	Massimo	Mecella	A
002	Massimiliano	De Leoni	A
003	Alessandro	Russo	A
004	Roberto	Palmieri	B
005	Roberto	Vitali	B
006	Francesco	Quaglia	B
007	Roberto	Baldoni	C
008	Silvia	Bonomi	C
009	Stefano	Leonardi	D
010	Maurizio	Lenzerini	E
011	Paolo	Liberatore	F

12 rows in set (0.02 sec)

# Inserimento dei dati nelle tabelle

---

## ► Soluzione...

```
mysql> INSERT INTO PROFESSORE VALUES('000','Andrea','Marrella','A');
Query OK, 1 row affected (0.09 sec)

mysql> INSERT INTO PROFESSORE VALUES('001','Massimo','Mecella','A');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('002','Massimiliano','De Leoni','A');
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO PROFESSORE VALUES('003','Alessandro','Russo','A');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO PROFESSORE VALUES('004','Roberto','Palmieri','B');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('005','Roberto','Vitali','B');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('006','Francesco','Quaglia','B');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('007','Roberto','Baldoni','C');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('008','Silvia','Bonomi','C');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('009','Stefano','Leonardi','D');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('010','Maurizio','Lenzerini','E');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROFESSORE VALUES('011','Paolo','Liberatore','F');
Query OK, 1 row affected (0.00 sec)
```

# Inserimento dei dati nelle tabelle

- ▶ Inserire nella tabella Progetto le tuple necessarie affinché la tabella stessa risulti popolata nel seguente modo :

```
+-----+-----+-----+
| CodiceProgetto | Nome          | EnteFinanziatore |
+-----+-----+-----+
| ACS            | ACSI         | Commissione Europea |
| NET           | NETWORK     | Commissione Europea |
| P32           | PD32        | Microsoft           |
| SM4           | SM4ALL      | Commissione Europea |
| WP1           | WORKPAD     | Commissione Europea |
+-----+-----+-----+
5 rows in set (0.02 sec)
```

# Inserimento dei dati nelle tabelle

---

## ► Soluzione...

```
mysql>
mysql> INSERT INTO PROGETTO VALUES('WP1','WORKPAD','Commissione Europea');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROGETTO VALUES('SM4','SM4ALL','Commissione Europea');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PROGETTO VALUES('ACS','ACSI','Commissione Europea');
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO PROGETTO VALUES('P32','PD32','Microsoft');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO PROGETTO VALUES('NET','NETWORK','Commissione Europea');
Query OK, 1 row affected (0.02 sec)
```

# Inserimento dei dati nelle tabelle

- ▶ Inserire nella tabella Partecipa le tuple necessarie affinché la tabella stessa risulti popolata nel seguente modo :

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
000	WP1	2006-11-01	2009-11-30
001	SM4	2008-09-01	2011-09-30
001	WP1	2006-11-01	2009-11-30
002	WP1	2006-11-01	2009-11-30
003	WP1	2008-03-01	2009-11-30
004	P32	2008-09-01	2011-09-30
005	P32	2008-09-01	2011-09-30
006	P32	2008-09-01	2011-09-30
007	SM4	2008-09-01	2011-09-30
008	SM4	2008-09-01	2011-09-30
009	NET	2008-09-01	2011-09-30
009	P32	2008-09-01	2011-09-30
010	ACS	2010-09-01	2013-09-30

13 rows in set (0.02 sec)

# Inserimento dei dati nelle tabelle

---

## ► Soluzione...

```
mysql> INSERT INTO PARTECIPA VALUES('000','WP1','2006-11-01','2009-11-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('001','WP1','2006-11-01','2009-11-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('002','WP1','2006-11-01','2009-11-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('003','WP1','2008-03-01','2009-11-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('001','SM4','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('007','SM4','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('008','SM4','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('004','P32','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('005','P32','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('006','P32','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('009','P32','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO PARTECIPA VALUES('009','NET','2008-09-01','2011-09-30');
Query OK, 1 row affected (0.02 sec)

► mysql> INSERT INTO PARTECIPA VALUES('010','ACS','2010-09-01','2013-09-30');
Query OK, 1 row affected (0.05 sec)
```

# Interrogazioni (soluzioni – slide 35)

---

**Professore**

CodiceProfessore	Nome	Cognome	Dipartimento
------------------	------	---------	--------------

**Progetto**

CodiceProgetto	Nome	EnteFinanziatore
----------------	------	------------------

**Partecipa**

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	----------------	------------	----------

- ▶ Selezionare tutti quei progetti non finanziati dalla Microsoft.
- ▶ Selezionare tutti quei progetti finanziati dall'unione europea il cui nome ha come ultima lettera una 'd'.
- ▶ Selezionare il numero di progetti finanziati dall'Unione Europea.
- ▶ Selezionare per ogni ente finanziatore, il numero di progetti che finanzia.
- ▶ Selezionare nome e cognome di tutti i professori che partecipano a qualche progetto finanziato dall'Unione Europea.
- ▶ Restituire i nomi di quei i progetti finanziati dalla Comunità Europea in ordine di data di partenza ed aventi almeno un professore del dipartimento 'A' che vi partecipa.



# Esercizio 1\10

---

- ▶ Sia dato il seguente schema relazionale:

**Medici** (**Matricola:String**, **Cognome:String**, **Nome:String**, **Anni di servizio:Int**, **Reparto:String**)

Eseguire le seguenti operazioni:

- 1) **Accedere al DBMS MySQL installato sul sistema operativo**
  - a) Aprire il *Prompt dei comandi* di Windows
  - b) Digitare :
    - **mysql -uroot -p**
  - c) Alla richiesta della password digitare la password scelta in fase di installazione
- 2) **Creare una base di dati (evitando la segnalazione di errore se già ne esiste uno con lo stesso nome) chiamata Ospedale**

Digitare il comando:

  - **create database if not exists Ospedale;**
- 3) **Verificare se la base di dati Ospedale è stata effettivamente creata**

Digitare il comando:

  - **show databases;**

# Esercizio 2\10

```
C:\> Prompt dei comandi - mysql -uroot -p
Microsoft Windows XP [Versione 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Andrea>mysql -uroot -p
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4
Server version: 5.1.32-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> create database if not exists ospedale;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| ciao |
| mysql |
| ospedale |
| test |
+-----+
5 rows in set (0.00 sec)
```

# Esercizio 3\10

---

4) Creare la tabella **Medici** così come specificata nello schema relazionale

## Medici

Matricola	Cognome	Nome	AnniServizio	Reparto
203	Neri	Piero	3	A
574	Bisi	Mario	4	B
461	Bargio	Sergio	10	B
530	Belli	Nicola	3	C
405	Mizzi	Nicola	20	A
501	Monti	Mario	12	A

Digitare il comando:

- **use Ospedale;**

che ci permette di operare nel database Ospedale creato in precedenza.

# Esercizio 4\10

---

Digitare il comando:

- **Create table Medici(  
Matricola char(3),  
Cognome char(20),  
Nome char(20),  
anniServizio numeric(2),  
Reparto char  
) ;**

che ci consente la creazione della tabella Medici con gli attributi richiesti.

## **ATTENZIONE :**

- Scrivere char o char(1) è equivalente.
- La Matricola, anche se numerica, viene sempre rappresentata come una stringa (a meno che non sia esplicitamente specificato).

# Esercizio 5\10

---

```
mysql> use ospedale;
Database changed
mysql> create table Medici(
  -> Matricola char(3),
  -> Cognome char(20),
  -> Nome char(20),
  -> AnniServizio numeric(2),
  -> Reparto char(1)
  -> );
Query OK, 0 rows affected (0.17 sec)

mysql> show tables;
+-----+
| Tables_in_ospedale |
+-----+
| medici              |
+-----+
1 row in set (0.00 sec)
```

# Esercizio 6\10

---

## 5) Popolare la tabella medici inserendo le tuple mostrate nell'istanza della relazione *Medici*

Digitare il comando:

- **INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)  
VALUES('203','Neri','Piero',3,'A');**

tante volte quante sono le tuple da inserire, cambiando il valore dei vari campi a seconda della tupla che si sta inserendo.

Se si rispetta l'ordine degli attributi con cui è stata definita la tabella Medici, si può utilizzare per l'inserimento anche il seguente comando

- **INSERT INTO Medici VALUES('203','Neri','Piero',3,'A');**

### **ATTENZIONE :**

- In MySQL una Stringa viene rappresentata tramite *caratteri* (per l'esempio si utilizzi il dominio *CHAR(x)*, dove *x* è il numero massimo di caratteri ammessi)
- In MySQL un intero può essere rappresentato in molti modo (per l'esempio si utilizzi il dominio *NUMERIC(x)*, dove *x* è il numero massimo di cifre ammesse)
- *Es : Numeric(2) -> rappresenta tutti gli interi da due cifre (da -99 a +99)*

```

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('203','Neri','Piero',3,'A');
Query OK, 1 row affected (0.06 sec)

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('574','Bisi','Mario',4,'B');
Query OK, 1 row affected (0.02 sec)

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('461','Bargio','Sergio',10,'B');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('530','Belli','Nicola',3,'C');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('405','Mizzi','Nicola',20,'A');
Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO Medici(Matricola,Cognome,Nome,AnniServizio,Reparto)
-> VALUES('501','Monti','Mario',12,'A');
Query OK, 1 row affected (0.00 sec)

```

```
mysql> select * from Medici;
```

Matricola	Cognome	Nome	AnniServizio	Reparto
203	Neri	Piero	3	A
574	Bisi	Mario	4	B
461	Bargio	Sergio	10	B
530	Belli	Nicola	3	C
405	Mizzi	Nicola	20	A
501	Monti	Mario	12	A

```
6 rows in set (0.00 sec)
```

# Esercizio 8\10

**6) Cosa accade se inserisco due volte la tupla  
(405,'Mizzi','Nicola',20,'A') ?**

Il DBMS inserisce regolarmente la tupla.

```
mysql> insert into Medici values('405','Mizzi','Nicola',20,'A');  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select * from Medici;
```

Matricola	Cognome	Nome	AnniServizio	Reparto
203	Neri	Piero	3	A
574	Bisi	Mario	4	B
461	Bargio	Sergio	10	B
530	Belli	Nicola	3	C
→ 405	Mizzi	Nicola	20	A
→ 405	Monti	Mario	12	A
→ 405	Mizzi	Nicola	20	A

```
7 rows in set (0.00 sec)
```



# Esercizio 9\10

---

## 7) La tupla inserita è legale? Perché?

Formalmente la tupla inserita **non è legale**, perché una relazione non è altro che un insieme i cui elementi sono appunto le tuple. **Ma un insieme non può contenere due elementi identici**, perciò dal punto di vista formale **non è possibile avere due tuple identiche**.

**I DBMS non si preoccupano di quest'aspetto**  
**(perciò si presti molta attenzione)**

# Esercizio 10\10

---

8) Cosa dovrei fare se volessi fare in modo che tutti i medici siano identificati da una *Matricola* differente?

Dovrei settare *Matricola* come una **chiave primaria**.

9) Cosa dovrei fare se volessi fare in modo che tutti i medici abbiano la coppia *nome e cognome* differente?

Dovrei settare *nome e cognome* come una chiave.

# Soluzioni esercizio slide 24

---

<b>Professore</b>	CodiceProfessore	Nome	Cognome	Dipartimento
-------------------	------------------	------	---------	--------------

<b>Progetto</b>	CodiceProgetto	Nome	EnteFinanziatore
-----------------	----------------	------	------------------

<b>Partecipa</b>	CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	------------------	----------------	------------	----------

- ▶ Selezionare tutti quei progetti non finanziati dalla Microsoft.
- ▶ Selezionare tutti quei progetti finanziati dall'unione europea il cui nome ha come ultima lettera una 'd'.
- ▶ Selezionare il numero di progetti finanziati dall'Unione Europea.
- ▶ Selezionare per ogni ente finanziatore, il numero di progetti che finanzia.
- ▶ Selezionare nome e cognome di tutti i professori che partecipano a qualche progetto finanziato dall'Unione Europea.
- ▶ Restituire i nomi di quei i progetti finanziati dalla Comunità Europea in ordine di data di partenza ed aventi almeno un professore del dipartimento 'A' che vi partecipa.

# Interrogazioni

**Professore**

CodiceProfessore	Nome	Cognome	Dipartimento
------------------	------	---------	--------------

**Progetto**

CodiceProgetto	Nome	EnteFinanziatore
----------------	------	------------------

**Partecipa**

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	----------------	------------	----------

- ▶ Selezionare tutti quei progetti non finanziati dalla Microsoft.

```
mysql> SELECT *
-> FROM Progetto
-> WHERE EnteFinanziatore<>'Microsoft';
```

CodiceProgetto	Nome	EnteFinanziatore
ACS	ACSI	Commissione Europea
NET	NETWORK	Commissione Europea
SM4	SM4ALL	Commissione Europea
WP1	WORKPAD	Commissione Europea

```
4 rows in set (0.00 sec)
```

# Interrogazioni

**Professore**

CodiceProfessore	Nome	Cognome	Dipartimento
------------------	------	---------	--------------

**Progetto**

CodiceProgetto	Nome	EnteFinanziatore
----------------	------	------------------

**Partecipa**

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	----------------	------------	----------

- ▶ Selezionare tutti quei progetti finanziati dall'unione europea il cui nome ha come ultima lettera una 'd'

```
mysql> SELECT *
-> FROM Progetto
-> WHERE Nome LIKE '%D' AND EnteFinanziatore='Commissione Europea';
+-----+-----+-----+
| CodiceProgetto | Nome      | EnteFinanziatore |
+-----+-----+-----+
| WP1            | WORKPAD  | Commissione Europea |
+-----+-----+-----+
1 row in set (0.00 sec)
```

# Interrogazioni

---

**Professore**

CodiceProfessore	Nome	Cognome	Dipartimento
------------------	------	---------	--------------

**Progetto**

CodiceProgetto	Nome	EnteFinanziatore
----------------	------	------------------

**Partecipa**

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	----------------	------------	----------

- ▶ Selezionare il numero di progetti finanziati dall'Unione Europea

```
mysql> SELECT count(*)
-> FROM Progetto
-> WHERE EnteFinanziatore='Commissione Europea';
+-----+
| count(*) |
+-----+
|         4 |
+-----+
1 row in set (0.36 sec)
```

# Interrogazioni

**Professore**

CodiceProfessore	Nome	Cognome	Dipartimento
------------------	------	---------	--------------

**Progetto**

CodiceProgetto	Nome	EnteFinanziatore
----------------	------	------------------

**Partecipa**

CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	----------------	------------	----------

- ▶ Selezionare per ogni ente finanziatore, il numero di progetti che finanzia

```
mysql> SELECT EnteFinanziatore, count(*)
-> FROM Progetto
-> GROUP BY EnteFinanziatore;
+-----+-----+
| EnteFinanziatore | count(*) |
+-----+-----+
| Commissione Europea | 4 |
| Microsoft | 1 |
+-----+-----+
2 rows in set (0.03 sec)
```

# Interrogazioni

<b>Professore</b>	CodiceProfessore	Nome	Cognome	Dipartimento
-------------------	------------------	------	---------	--------------

<b>Progetto</b>	CodiceProgetto	Nome	EnteFinanziatore
-----------------	----------------	------	------------------

<b>Partecipa</b>	CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	------------------	----------------	------------	----------

- ▶ Selezionare nome e cognome di tutti i professori che partecipano a qualche progetto finanziato dall'Unione Europea

```
mysql> select distinct P.Nome AS NomeProfessore, P.Cognome AS CognomeProfessore
-> from Professore P, Partecipa Q, Progetto E
-> where P.codiceProfessore=Q.codiceProfessore AND Q.codiceProgetto=E.codiceProgetto
-> AND EnteFinanziatore='Commissione Europea';
```

NomeProfessore	CognomeProfessore
Maurizio	Lenzerini
Stefano	Leonardi
Massimo	Mecella
Roberto	Baldoni
Silvia	Bonomi
Andrea	Marrella
Massimiliano	De Leoni
Alessandro	Russo

8 rows in set (0.01 sec)



# Interrogazioni

<b>Professore</b>	CodiceProfessore	Nome	Cognome	Dipartimento
-------------------	------------------	------	---------	--------------

<b>Progetto</b>	CodiceProgetto	Nome	EnteFinanziatore
-----------------	----------------	------	------------------

<b>Partecipa</b>	CodiceProfessore	CodiceProgetto	DataInizio	DataFine
------------------	------------------	----------------	------------	----------

- ▶ Restituire i nomi di quei i progetti finanziati dalla Comunità Europea in ordine di data di partenza ed aventi almeno un professore del dipartimento 'A' che vi partecipa.

```
mysql> SELECT DISTINCT PR.Nome
-> FROM Professore P0, Progetto PR, Partecipa PA
-> WHERE PR.EnteFinanziatore = 'Commissione Europea' AND PA.CodiceProgetto=PR.CodiceProgetto
-> AND PA.CodiceProfessore=P0.CodiceProfessore AND P0.Dipartimento='A'
-> ORDER BY DataInizio;
+-----+
| Nome |
+-----+
| WORKPAD |
| SM4ALL |
+-----+
2 rows in set (0.00 sec)
```