

# Basi di dati e Sistemi Informativi A.A. 2018-2019

## I Esercitazione in laboratorio

### Prendere confidenza con gli account e con MySQL Query Browser

Lanciare MySQL Query Browser seguendo il percorso:

- Start → Programmi → MySQL → MySQL Query Browser

Inserire i dati:

- Server host: 192.168.0.10
- Username: cognome dello studente
- Password: matricola dello studente
- Default schema: lasciare vuoto

Ogni studente ha tutti i diritti sulla base di dati DB+matricola ma non può creare basi di dati con nomi diversi né ricreare la propria, nel caso venga cancellata per sbaglio (la base di dati dovrà essere ricreata dall'amministratore).

### Primi passi: l'istruzione SELECT come operatore di calcolo

Nell'area per l'immissione delle query di MySQL Query Browser inserire i seguenti comandi e osservare il risultato nella finestra Resultset. Vengono mostrati esempi di stringhe, numeri (anche in formato esadecimale), valori booleani e il valore NULL. Si osservi che nella finestra in basso a destra del MySQL Query Browser si trova un elenco di funzioni predefinite che possono essere utilizzate all'interno dell'istruzione SELECT.

```
/* Stringhe: sequenze di caratteri racchiuse tra apici o doppi apici.  
Fare attenzione se si vuole inserire il carattere apice (o doppio apice)  
nella stringa. */
```

```
select 'ciao','"ciao"',""ciao""','cia'o','\'ciao';  
select "ciao","'ciao'",''ciao'', "cia""o","\"ciao";
```

```
/* In questo esempio la stringa contiene il carattere speciale \n.  
Nella Resultset selezionare la stringa e "View field in PopUp Editor"  
col tasto destro del mouse. */
```

```
select 'Una\nfrase\nsu quattro\nlinee' as Prova;
```

```
select -1234.5678e+2;  
select hex(255);  
select hex('MySQL');
```

```
select x'4D7953514C';
```

```
select true, false;  
select NULL,\N;
```

```
select 1+1; #Commento  
select 1+1; -- Commento  
select 1/*Commento*/+1;  
select sqrt(256);
```

## Primi passi: l'istruzione CREATE

Riprendere gli esempi visti a lezione creando le seguenti tabelle e immettendo i comandi nell'area delle query. Per fare questo è necessario selezionare prima la propria base di dati di riferimento con un doppio click sull'icona che compare nella finestra **Schemata**. Dopo la creazione osservare le icone delle tabelle all'interno della base di dati **DB+matricola** (eventualmente effettuare un **Refresh** con il tasto destro del mouse). L'istruzione **INSERT** (che poi vedremo in modo approfondito) permette di inserire tuple in una tabella. L'istruzione **DESCRIBE tabella** permette di controllare lo schema della tabella creata, in alternativa si può selezionare **Edit Table** con il tasto destro del mouse dall'icona della tabella.

```
CREATE TABLE Dipartimento(  
Nome CHAR(20) PRIMARY KEY,  
Indirizzo CHAR(50),  
Sede CHAR(20)  
);
```

```
CREATE TABLE Prova(  
i INT(3) ZEROFILL,  
j INT(6) ZEROFILL,  
k INT(11) ZEROFILL  
);  
INSERT INTO Prova(i, j, k) VALUES (123, 456, 789);  
SELECT * FROM Prova;
```

```
CREATE TABLE Studente(  
Matricola INT,  
Cognome VARCHAR(20),  
Nome VARCHAR(20),  
Nascita DATE,  
Corso ENUM('Informatica','Matematica','Fisica')  
);  
DESCRIBE Studente;
```

## Creazione di tabelle con indici

Gli indici sulle tabelle costituiscono un aspetto fondamentale nella progettazione di un database: un indice infatti consente di velocizzare l'accesso ai dati in fase di lettura. Gli indici possono essere di diverso tipo: le **primary key** (chiavi primarie) consentono di identificare univocamente un record all'interno di una tabella; gli indici **unique**, come le chiavi primarie, non possono avere valori duplicati, ma, al contrario di queste, possono contenere valori NULL.

- Per prendere confidenza con gli indici, creare le seguenti tabelle immettendo i comandi nell'area delle query ed osservando le differenze degli schemi creati con **Edit Table**. L'engine **InnoDB** permette di inserire vincoli di integrità referenziale.

```
create table Studente(  
matricola int,  
cognome varchar(20),  
nome varchar(20),  
nascita date,  
corso varchar(20) default null,  
unique(corso)) engine=InnoDB;
```

```
create table Studente1 like Studente;
```

```
create table Studente2(  
matricola int primary key,  
cognome varchar(20) not null,  
nome varchar(20),  
nascita date,  
corso varchar(20),  
unique(cognome, nome, nascita));
```

```
insert into Studente2 values  
(10,'Paperino','Paolino','1934-06-09','DolceDormire');
```

```
# Il seguente inserimento non e' permesso perche' matricola e' chiave  
# e il valore 10 e' gia' presente.
```

```
insert into Studente2 values  
(10,'De Paperoni','Paperon','1947-12-15','DiventareRicchi');
```

```
# Il seguente inserimento non e' permesso per la presenza del vincolo unique  
insert into Studente2 values  
(20,'Paperino','Paolino','1934-06-09','Divertirsi');
```

```
# Il seguente inserimento non e' possibile per il vincolo not null su cognome  
insert into Studente2 values(20,\N,\N,\N,'Divertirsi');
```

```

# Il seguente inserimento invece e' possibile
insert into Studente2 values(20,'Paperino',\N,\N,'Divertirsi');

create table Studente3(
matricola int auto_increment primary key,
cognome varchar(20) not null,
nome varchar(20),
nascita date,
corso varchar(20),
unique(cognome, nome, nascita));

insert into Studente3(cognome, nome, nascita, corso)
values('Paperino','Paolino','1934-06-09','DolceDormire');
insert into Studente3 values
(\N,'De Paperoni','Paperon','1947-12-15','DiventareRicchi');
select * from Studente3;

create table Corsodilaurea(
codice int auto_increment primary key,
nome varchar(20));

create table Studente4(
matricola int auto_increment primary key,
cognome varchar(20) not null,
nome varchar(20),
nascita date,
corso int, #Cambiato dominio rispetto a Studente3
foreign key(corso) references Corsodilaurea(codice) on delete cascade,
unique(cognome, nome, nascita));

```

## Esercizio: il database Prodotti Premiere

Il database di riferimento è descritto in

- P. J. Pratt, *Guida a SQL*, Apogeo, 2001.

La **Prodotti Premiere** è un'impresa che distribuisce casalinghi, elettrodomestici e articoli sportivi. Ha bisogno di gestire le informazioni sui suoi rappresentanti commerciali, i suoi clienti, gli ordini effettuati e le scorte di prodotti. Lo schema logico della **Prodotti Premiere** è il seguente:

- Rappresentanti(**CodR**, Cognome, Nome, Via, Città, Pr, CAP, TotProv, PerProv)
- Clienti( **CodC**, Cognome, Nome, Via, Città, Pr, CAP, Saldo, Fido, CodR)
- Ordini(**NumOrdine**, Data, CodC )

- Articoli(**NumARt**, Descrizione, Giacenza, Categoria, Magazzino, PrzUnitario)
- DettagliOrdini(**NumOrdine**, **NumArt**, QtaOrd, Prezzo)

Si considerano i seguenti vincoli di integrità referenziale:

- Clienti.CodR  $\rightarrow$  Rappresentanti.CodR
- Ordini.CodC  $\rightarrow$  Clienti.CodC
- DettagliOrdini.NumOrdine  $\rightarrow$  Ordini.NumOrdine
- DettagliOrdini.NumArt  $\rightarrow$  Articoli.NumArt
- Creare le tabelle del data base **Premiere** utilizzando le seguenti specifiche per il dominio degli attributi. In un primo momento non preoccuparsi dei vincoli di integrità referenziale.

**Tabella Rappresentanti**

Attributo	Dominio	Lunghezza
<u>CodR</u>	Carattere	2
Cognome	Carattere	10
Nome	Carattere	8
Via	Carattere	15
Citta	Carattere	15
Prov	Carattere	2
Cap	Carattere	5
TotProvv	Decimale	6 cifre di cui 2 dopo la virgola
PerProvv	Decimale	6 cifre di cui 2 dopo la virgola

**Tabella Clienti**

Attributo	Dominio	Lunghezza
<u>CodC</u>	Carattere	3
Cognome	Carattere	10
Nome	Carattere	8
Via	Carattere	15
Citta	Carattere	15
Prov	Carattere	2
Cap	Carattere	5
Saldo	Decimale	6 cifre di cui 2 dopo la virgola
Fido	Decimale	6 cifre di cui 2 dopo la virgola
CodR	Carattere	2

**Tabella Ordini**

Attributo	Dominio	Lunghezza
<u>NumOrdine</u>	Carattere	6
Data	Data	
CodC	Carattere	3

**Tabella Articoli**

<b>Attributo</b>	<b>Dominio</b>	<b>Lunghezza</b>
<u>NumArt</u>	Carattere	4
Descrizione	Carattere	20
Giacenza	Intero	4
Categoria	Carattere	2
Magazzino	Carattere	1
PrzUnitario	Decimale	6 cifre di cui 2 dopo la virgola

**Tabella DettagliOrdini**

<b>Attributo</b>	<b>Dominio</b>	<b>Lunghezza</b>
<u>NumOrdine</u>	Carattere	6
<u>NumArt</u>	Carattere	4
QtaOrd	Intero	4
Prezzo	Decimale	6 cifre di cui 2 dopo la virgola