

## Operazioni sulle Tabelle

Mostrare l'elenco delle Tabelle

```
show tables;
```

e la risposta sarà costituita da una tabella contenente l'elenco delle tabelle presenti nel db selezionato:

```
Tables_in_nome_db
+-----+
| Persone |
+-----+
| Figli   |
+-----+
```

Creare una Tabella

```
create table nome_tab (
  nome_campo1 tipo_dato [ opzioni ],
  .....
  nome_campoN tipo_dato [ opzioni ]
);
```

E' possibile usare ancora l'opzione 'IF NOT EXISTS' per creare la tabella. I valori opzionali sono anche detti **Modificatori di colonne**, ed i principali sono:

<b>auto_increment</b>	usato per il campo ID ( tipo int ) e serve per incrementare in modo automatico il valore dell'intero; per questo motivo il valore dell'ID non deve essere inserito manualmente per ogni record della tabella. Per ogni tabella può essere definito un solo campo con questo attributo.
<b>binary</b>	viene usato per il tipo char e varchar ( stringhe binarie ).
<b>default valore</b>	stabilisce un valore di default e riguarda tutti i tipi di dati eccetto text e blob.
<b>null</b>	stabilisce un valore nullo per tale campo.
<b>not null</b>	stabilisce un valore non nullo per quel campo.
<b>primary key</b>	stabilisce che quel campo funge da chiave primaria ed in genere si usa per il campo ID; inoltre assume implicitamente un valore non nullo e deve essere unica.
<b>unique</b>	stabilisce un valore unico e differente dagli altri inseriti in quel campo. E' simile ad 'autoincrement', solo che ammette valori nulli.
<b>zerofill</b>	si usa per il tipo numerico e serve per visualizzare gli zeri iniziali (es.: 00179).

Vediamo ora come creare la seguente tabella "Persone", costituita dai campi Identificativo (chiave primaria), Nome, Cognome, Sesso Data\_Nascita:

Tabella "Persone"

* Identificativo	Nome	Cognome	Sesso	Data di Nascita
1	Andrea	Rossi	M	15/12/1982
2	Piero	Russo	M	01/05/2000
3	Maria	Coletti	F	23/10/1960

```
create table Persone (
  Identificativo int auto_increment primary key,
  Nome varchar(20),
  Cognome varchar(20),
  Sesso char(1),
  Data_Nascita varchar(50)
);
```

Esiste anche un modo veloce di creare una tabella vuota copiando la struttura da una tabella già esistente:

```
CREATE TABLE nuova_tabella LIKE tabella_originale;
```

Mostrare la Struttura di una Tabella

```
describe nome_tab;
```

Per la tabella "Persone", otterrei:

```

+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key  | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| Identificativo | int           |      | PRI  | NULL     | auto_increment |
+-----+-----+-----+-----+-----+-----+
| Nome           | varchar(20)   | YES  |      | NULL     |                |
+-----+-----+-----+-----+-----+-----+
| Cognome        | varchar(20)   | YES  |      | NULL     |                |
+-----+-----+-----+-----+-----+-----+
| Sesso          | char(1)       | YES  |      | NULL     |                |
+-----+-----+-----+-----+-----+-----+
| Data_Nascita   | varchar(50)   | YES  |      | NULL     |                |
+-----+-----+-----+-----+-----+-----+

```

Per la descrizione dettagliata sui tipi di Indice (BTREE, HASH) e su come si possano definire, cancellare ed aggiungere, rimando alla guida ufficiale.