

# Sistemi per la gestione di database:

MySQL (♥)

# Relational Database e Relational Database Management System

Un **database** è una raccolta di dati organizzata in modo da consentire l'accesso, il reperimento e l'utilizzo di tali dati.

Un **sistema per la gestione di database** (DataBase Management System – DBMS), come Access, permette di usare un computer per

- creare un database,
- aggiungere, modificare ed eliminare dati,
- ordinare e reperire dati nel database,
- creare report che usano i dati del database.

# Database relazionali

I **database relazionali** (Edgar Codd, 1969) sono database in cui i dati sono immagazzinati in relazioni che all'utente appaiono come tabelle.

Ciascuna relazione (tabella) è composta da **record** (righe) e **campi** (attributi, colonne)

| <b>cognome</b> | <b>nome</b> | <b>città</b> | <b>telefono</b> |
|----------------|-------------|--------------|-----------------|
| Rossi          | Fabio       | Venezia      | 5221345         |
| Storchi        | Gianni      | Milano       | 34678905        |
| Valenti        | Sara        | Roma         | 99223455        |
| ...            | ...         | ...          | ...             |

**Tabella/relazione Clienti (cognome, nome, città, telefono)**

# **Relational DataBase Management System (RDBMS)**

- **permette agli utenti di creare nuovi database specificandone la struttura logica**
- **permette di interrogare e modificare i dati**
- **permette di memorizzare una grossa quantità di dati**
- **permette di controllare che l'accesso contemporaneo di più utenti agli stessi dati non li danneggino**

## **Query**

**domande poste al database per ottenere informazioni dai dati in esso inseriti**

## **Tabelle (o relazioni)**

- **Ogni tabella rappresenta un soggetto unico e specifico**
- **L'ordine dei record e dei campi di una tabella non è rilevante**
- **Ogni tabella deve contenere almeno un campo (chiave primaria) che identifica ogni record in modo univoco**

| <b>codice<br/>cliente</b> | <b>cognome</b> | <b>nome</b> | <b>città</b> | <b>telefono</b> |
|---------------------------|----------------|-------------|--------------|-----------------|
| <b>C1</b>                 | Rossi          | Fabio       | Venezia      | 5221345         |
| <b>C2</b>                 | Storchi        | Gianni      | Milano       | 34678905        |
| <b>C3</b>                 | Valenti        | Sara        | Roma         | 99223455        |
| ...                       | ...            | ...         | ...          | ...             |

**Tabella/relazione Clienti (codice, cognome, nome, città, telefono)**

# Chiavi

- **Campi speciali all'interno di una tabella**
- **Chiave primaria:** campo che identifica in modo univoco ogni record all'interno della tabella. Il campo di una chiave primaria identifica una tabella all'interno del database (rafforza l'integrità dei dati a livello di tabella ed aiuta a stabilire delle relazioni con le altre tabelle)
- **Chiave esterna:** campo di una tabella che rappresenta la copia della chiave primaria di un'altra tabella (permette di stabilire relazioni tra tabelle, aiuta ad assicurare l'integrità a livello di relazione, aiuta ad evitare i record orfani)

# Elementi di un Database

## Chiavi - esempio

Chiave primaria  
tabella clienti



| <b>codice cliente</b> | <b>cognome</b> | <b>nome</b> | <b>città</b> | <b>telefono</b> |
|-----------------------|----------------|-------------|--------------|-----------------|
| <b>C1</b>             | Rossi          | Fabio       | Venezia      | 5221345         |
| <b>C2</b>             | Storchi        | Gianni      | Milano       | 34678905        |
| <b>C3</b>             | Valenti        | Sara        | Roma         | 99223455        |
| ...                   | ...            | ...         | ...          | ...             |

**Tabella Clienti**

Chiave esterna tabella  
transazioni

Chiave primaria  
tabella transazioni



| <b>codice transaz.</b> | <b>codice cliente</b> | <b>data</b> | <b>Importo (Euro)</b> |
|------------------------|-----------------------|-------------|-----------------------|
| <b>T1</b>              | <b>C1</b>             | 23/03/2001  | 1000,00               |
| <b>T2</b>              | <b>C3</b>             | 26/02/2001  | 790,90                |
| <b>T3</b>              | <b>C1</b>             | 10/06/2001  | 335,50                |
| ...                    | ...                   | ...         | ...                   |

**Tabella Transazioni**

# Elementi di un Database

## Relazioni

- **Indicano le associazioni tra i record di una tabella con quelli di un'altra tabella**
- **Relazione “uno a uno”**
- **Relazione “uno a molti”**
- **Relazione “molti a molti”**



# Elementi di un Database

## Relazione “uno a uno”

- Tra due tabelle c'è una relazione “uno a uno” se ciascun record della prima tabella è associato a un solo record della seconda e ciascun record della seconda tabella è associato a un solo record della prima tabella

| <b>codice cliente</b> | <b>città</b> |
|-----------------------|--------------|
| <b>C1</b>             | Venezia      |
| <b>C2</b>             | Milano       |
| <b>C3</b>             | Roma         |
| ...                   | ...          |

| <b>codice cliente</b> | <b>cognome</b> | <b>nome</b> | <b>telefono</b> |
|-----------------------|----------------|-------------|-----------------|
| <b>C1</b>             | Rossi          | Fabio       | 5221345         |
| <b>C2</b>             | Storchi        | Gianni      | 34678905        |
| <b>C3</b>             | Valenti        | Sara        | 99223455        |
| ...                   | ...            | ...         | ...             |

# Elementi di un Database

## Relazione “uno a molti”

- Tra due tabelle c'è una relazione “uno a molti” se ciascun record della prima tabella può essere associato a molti record della seconda ma ciascun record della seconda tabella può essere associato ad un solo record della prima tabella

| codice cliente | città   |
|----------------|---------|
| C1             | Venezia |
| C2             | Milano  |
| C3             | Roma    |
| ...            | ...     |

| codice transaz. | codice cliente | data       | Importo (Euro) |
|-----------------|----------------|------------|----------------|
| T1              | C1             | 23/03/2001 | 1000,00        |
| T2              | C3             | 26/02/2001 | 790,90         |
| T3              | C1             | 10/06/2001 | 335,50         |
| ...             | ...            | ...        | ...            |

Tabella Clienti

Tabella Transazioni

# Elementi di un Database

## Relazione “molti a molti”

- Tra due tabelle c'è una relazione “molti a molti” se ciascun record della prima tabella può essere associato a molti record della seconda e ciascun record della seconda tabella può essere associato a molti record della prima tabella
- Occorre una tabella di collegamento che viene definita prendendo una copia delle chiavi primarie di ciascuna tabella della relazione

| codice cliente | città   | codice transaz. | codice prodotto | codice cliente | codice prodotto | descrizione           |
|----------------|---------|-----------------|-----------------|----------------|-----------------|-----------------------|
| C1             | Venezia | T1              | P1              | C1             | P1              | Cappotto taglia 42    |
| C2             | Milano  | T2              | P1              | C2             | P2              | Completo donna tg 44  |
| C3             | Roma    | T3              | P3              | C1             | P3              | Pantalone uomo tg. 48 |
| ...            | ...     | ...             | ...             | ...            | ...             |                       |

Tabella Clienti

Tab. Transaz. (di collegamento)

Tab. prodotti

# Come interrogare e modificare un database: SQL

Una **query** è una domanda posta al database per estrarre dei dati. Il linguaggio più comunemente usato per formulare query è **SQL** (Structured Query Language, linguaggio d'interrogazione strutturato). SQL è un insieme di 36 comandi che, opportunamente combinati permettono di estrarre qualunque dato da un database.

Alcuni di questi comandi sono: *Select, Distinct, From, Where, And, Or, Group by, Order by.*

## Esempio di query in SQL

```
SELECT DISTINCT cognome, indirizzo, città, codice postale;  
FROM clienti;  
WHERE codice postale = "66208";  
ORDER BY cognome
```

# **Esempio di utilizzo di un RBDMS: MySQL**

**Tralasciamo tutti gli aspetti relativi alla creazione, gestione e manutenzione di un database, ponendoci dal punto di vista dell'utilizzatore finale: colui che vuole trasformare i dati in informazioni.**

**Impariamo ad usare l'istruzione SELECT di MySQL.**

**MySQL è scaricabile gratuitamente dal sito [www.mysql.com](http://www.mysql.com).**

# MySQL: clauseole dell'istruzione SELECT

**select**: (indispensabile) si usa per specificare le colonne che si vogliono utilizzare nella query.

**from**: (indispensabile) si usa per specificare le tabelle a cui appartengono le colonne specificate nella clausola select.

**where**: (opzionale) si usa per filtrare le righe provenienti dalla clausola from. Where è seguita da un'espressione, nota come *predicato*, che valuta se un'informazione è vera, falsa o sconosciuta.

Collegamento a MySQL tramite **Telnet** o **Putty**:

Host: **matlab.cicsd.unive.it**

Utente: **mapleusr**

Password: **mpl7#\$**

**digitare**

***mysql -u mapleusr -p***

**per poter lavorare**

# MySQL: struttura database dispensa (database: prova, tabella: ricavi)

| <b>tempo</b> | <b>ricavi1</b> | <b>ricavi2</b> |
|--------------|----------------|----------------|
| <b>1</b>     | <b>100.00</b>  | <b>100.00</b>  |
| <b>2</b>     | <b>98.37</b>   | <b>100.40</b>  |
| <b>3</b>     | <b>101.13</b>  | <b>99.77</b>   |
| <b>4</b>     | <b>101.67</b>  | <b>100.79</b>  |
| <b>5</b>     | <b>101.97</b>  | <b>102.55</b>  |
| <b>6</b>     | <b>102.51</b>  | <b>104.70</b>  |
| <b>7</b>     | <b>101.59</b>  | <b>107.01</b>  |
| <b>...</b>   | <b>...</b>     | <b>...</b>     |

**CHIAVE  
PRIMARIA**





# MySQL: struttura database prova online (database: info, tabella: aziende)

| codice | data       | A   | B   | C   | D   | E   | ... |
|--------|------------|-----|-----|-----|-----|-----|-----|
| 1      | 1995-01-01 | ... | ... | ... | ... | ... | ... |
| 2      | ...        |     |     |     |     |     |     |
| 3      |            |     |     |     |     |     |     |
| 4      |            |     |     |     |     |     |     |
| 5      |            |     |     |     |     |     |     |
| ...    |            |     |     |     |     |     |     |

**CHIAVE PRIMARIA**

**AAAA-MM-GG**

# Query

Dalla tabella *ricavi* del database *prova* trovare

- il massimo dei ricavi della prima azienda

select max(ricavi1) from ricavi;

- il range dei ricavi della seconda azienda

select max(ricavi2)-min(ricavi2) as RANGE from ricavi;

- la differenza media tra i ricavi della prima e della seconda azienda

select avg(ricavi1-ricavi2) from ricavi;

- evidenziare tutte le colonne della tabella ricavi

select \* from ricavi;

# Query

- contare quante volte i ricavi della prima azienda sono superiori a quelli della seconda

```
select count(ricavi1-ricavi2) from ricavi  
where ricavi1-ricavi2>0;
```

- contare quante volte i ricavi della prima azienda sono superiori a quelli della seconda per più del 50%

```
select count(ricavi1/ricavi2) from ricavi  
where ricavi1/ricavi2>1.5;
```

# Query

Dalla tabella *aziende* del database *info* trovare

- il primo istante in cui i ricavi dell'azienda A superano quelli dell'azienda C

select min(data) from aziende where A>C;

- mettere in evidenza il fatturato dell'azienda E alla data 1996-11-01;

select E from aziende where data="1996-11-01";

# Query

Dalla tabella *aziende* del database *info* trovare

- mettere in evidenza il fatturato delle aziende F, H, T nel mese di marzo 1998

**select F,H,T from aziende where data="1998-03-01";**

- calcolare il fatturato medio dell'azienda B nel periodo dal 1 gennaio 1996 al 1 ottobre 1997

**select avg(B) from aziende where data>="1996-01-01" and data<="1997-10-01";**

- ...