

Basi di dati

2

IL MODELLO RELAZIONALE

Organizzazione dei dati

- Come sono organizzati i dati in una base di dati?

Modello dei dati

- Insieme di costrutti secondo cui i dati di interesse sono organizzati e utilizzati
- Esempio:
 - il **modello relazionale** prevede il costrutto **relazione** (o **tabella**), che permette di definire insiemi di record (righe) omogenei

I modelli logici dei dati

- “Tradizionali”
 - ... relazionale
- Altri più recenti:
 - a oggetti, XML, "NoSQL"

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
8765	Neri	Paolo	03/11/1996
9283	Verdi	Luisa	12/11/1999
3456	Rossi	Maria	01/02/1998

esami

Studente	Voto	Corso
3456	30	04
3456	24	02
9283	28	01
6554	26	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

Modello "relazionale"

- relazione matematica ... (alla lavagna)

$$D_1 = \{x, y\}$$

$$D_2 = \{1, 2, 3\}$$

DOMINI

PRODOTTO
CARTESIANO

$\{(x, 1),$
 $(x, 2),$
 $(x, 3),$
 $(y, 1),$
 $(y, 2),$
 $(y, 3)\}$

$D_1 \times D_2$

RELAZIONE

$$R \subseteq D_1 \times D_2$$

$\{(x, 1),$
 $(x, 3),$
 $(y, 1)\}$

x	1
x	3
y	1

o

$D_1 D_2 D_3$

DOMINI

$D_1 \times D_2 \times D_3$

PRODOTTO CARTESIANO

$R \subseteq D_1 \times D_2 \times D_3$

RELAZIONE

insieme di terne

(d_1, d_2, d_3)

$d_1 \in D_1$

$d_2 \in D_2$

$d_3 \in D_3$

Due osservazioni importanti

- Struttura non posizionale
- Modello basato su valori

Struttura "posizionale" o "non posizionale"

Partite* \subseteq *string* \times *string* \times *int* \times *int

Manchester	Arsenal	3	1
Arsenal	Liverpool	2	0
Manchester	Chelsea	0	2
Chelsea	Liverpool	0	1

Struttura non posizionale

- A ciascun dominio si associa un nome:
attributo (o **colonna**)

Casa	Fuori	RetiCasa	RetiFuori
Manchester	Arsenal	3	1
Arsenal	Liverpool	2	0
Manchester	Chelsea	0	2
Chelsea	Liverpool	0	1

Il modello è basato su valori

- I riferimenti fra dati in relazioni diverse sono rappresentati per mezzo di valori dei domini che compaiono nelle ennuple

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
8765	Neri	Paolo	03/11/1996
9283	Verdi	Luisa	12/11/1999
3456	Rossi	Maria	01/02/1998

esami

Studente	Voto	Corso
3456	30	04
3456	24	02
9283	28	01
6554	26	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
8765	Neri	Paolo	03/11/1996
9283	Verdi	Luisa	12/11/1999
3456	Rossi	Maria	01/02/1998

esami

Studente	Voto	Corso
	30	
	24	
	28	
	26	

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

Basi di dati: schema e istanza

Orario

Insegnamento	Docente	Aula	Ora
Analisi matem. I	Luigi Neri	N1	8:00
Basi di dati	Piero Rossi	N2	10:00
Chimica	Nicola Mori	N1	10:00
Fisica I	Mario Bruni	N1	12:00
Fisica II	Mario Bruni	N3	10:00
Sistemi inform.	Piero Rossi	N3	8:00

Basi di dati: schema e istanza

Lo **schema** della relazione

Orario

Insegnamento	Docente	Aula	Ora
--------------	---------	------	-----

Basi di dati: schema e istanza

Lo **schema** della relazione

Orario

Insegnamento	Docente	Aula	Ora
Analisi matem. I	Luigi Neri	N1	8:00
Basi di dati	Piero Rossi	N2	10:00
Chimica	Nicola Mori	N1	10:00
Fisica I	Mario Bruni	N1	12:00
Fisica II	Mario Bruni	N3	10:00
Sistemi inform.	Piero Rossi	N3	8:00

L'**istanza** della relazione

Definizioni e notazioni (alla lavagna)

- Schema di relazione
- Ennupla su uno schema di relazione
 - Valore di una ennupla su un attributo
- Relazione (“istanza di relazione“)

- Schema di base di dati
- (Istanza di) base di dati

Scheme di relazione

R nome della relazione

A_1, A_2, \dots, A_n insieme di attributi

$\downarrow \quad \downarrow \quad \downarrow$
 $D_1 \quad D_2 \quad D_n$ domini

$R(A_1, A_2, \dots, A_n)$

STUDENTI (Matricola, Cognome, ...)

Scheme di base di dati

$\mathcal{R} = \{ R_1(X_1), \dots, R_n(X_n) \}$

UNIVERSITÀ = {

studenti (Matricola, ...),

esami (Studento, Voto, Corso),

corsi (Codice, ...) }

ISTANZA

(istanza di) relazione su $R(A, B, C)$
insieme di esempio (tupli,
righe)
con
un valore per ciascun
attributo

(istanza di) base di dati
insieme di relazioni,
una per ciascuno schema
di relazione nello schema della BD

ennupla (o tupla) t

	A	B	C
$t \rightarrow$	a_1	b_2	c_1

$$t[A] = a_1$$

(oppure
 $t.A = a_1$)

VALORE DELLA
ENNUPLA t
SULL'ATTRIBUTO
A

- Usiamo $t[A]$ per indicare il valore della ennupla t sull'attributo A

Altra osservazione

- La struttura è rigida e non sempre corrisponde ai dati:
 - "informazione incompleta"

Informazione incompleta

Nome	SecondoNome	Cognome
Franklin	Delano	Roosevelt
Winston		Churchill
Charles		De Gaulle
Josip		Stalin

Informazione incompleta

Nome	Ufficio	Età
Mario	Roma	34
Luca	Milano	
Piero	Firenze	42

Informazione incompleta: soluzioni?

- Potremmo pensare:
 - stringa nulla
 - "0"
 - "99"
- Si può fare?
 - Meglio di no!!

Informazione incompleta nel modello relazionale

- Tecnica semplice ma efficace:
 - **valore nullo** (un valore **diverso** dai valori del dominio, indicato qui con *NULL*)
 - $t[A]$, per ogni attributo A , è
 - un valore del dominio $\text{dom}(A)$
 - oppure il valore nullo *NULL*
- Commenti
 - tecnica semplice
 - attenzione ai troppi valori nulli

Troppi valori nulli

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
9283	Verdi	Luisa	12/11/1999
<i>NULL</i>	Rossi	Maria	01/02/1998

esami

Studente	Voto	Corso
<i>NULL</i>	30	<i>NULL</i>
<i>NULL</i>	24	02
9283	28	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	<i>NULL</i>	<i>NULL</i>
04	Chimica	Verdi

Un'anticipazione (parziale)

- Riassumendo, quanto visto finora:
 - La definizione dello schema di una relazione in SQL (rivedremo più avanti)

```
postgres on paolo@PostgreSQL 11
```

```
create table esami(  
  studente integer not null,  
  corso integer not null,  
  voto integer not null  
)
```

Seconda parte

- Abbiamo visto la struttura
- Per modellare bene non basta

Vincoli di integrità

- Non sempre le nostre basi di dati sono "corrette" ..

Una base di dati "scorretta"

Esami	Studente	Voto	Lode	Corso
	276545	32	no	01
	276545	30	sì	02
	787643	27	sì	03
	739430	24	no	04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca

Una base di dati "scorretta"

Esami	Studente	Voto	Lode	Corso
	276545	32	no	01
	276545	30	sì	02
	787643	27	sì	03
	739430	24	no	04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca

Vincolo di integrità

- Proprietà che deve essere soddisfatta dalle istanze che rappresentano informazioni corrette per l'applicazione
- Un vincolo è una funzione booleana (un **predicato**):
 - associa ad ogni istanza il valore **vero** o **falso**

Vincoli di integrità, nota

- solo alcuni tipi di vincoli sono "supportati" dai DBMS:
 - ...il DBMS ne impedisce la violazione,
 - per i vincoli "non supportati", la responsabilità della verifica è dell'utente o del programmatore
- Vediamo i vincoli su un DBMS (oggi lo mostro io, prossimamente dovreste usarlo)

Verifica dei vincoli

- Vediamo fra poco con un esempio sul tipo più semplice di vincoli

Una base di dati "scorretta"

Esami	Studente	Voto	Lode	Corso
	276545	32	no	01
	276545	30	sì	02
	787643	27	sì	03
	739430	24	no	04

Studenti	Matricola	Cognome	Nome
	276545	Rossi	Mario
	787643	Neri	Piero
	787643	Bianchi	Luca

Tipi di vincoli

- vincoli **intrarelazionali**
 - vincoli su valori (o di **dominio**)
 - vincoli di **ennupla**
 - vincoli di **chiave**
- vincoli **interrelazionali**

Vincolo di dominio

- Pone condizioni sui valori ammissibili di un singolo attributo, ad esempio
 - il voto ha come tipo "intero"
 - ma deve essere compreso fra 18 e 30 (un sottoinsieme del dominio)

Vincolo di dominio, in concreto

- Una possibile sintassi:
 - combinazione booleana (AND, OR e NOT)
 - di condizioni semplici che confrontano un attributo con una costante

$(\text{Voto} \geq 18) \text{ AND } (\text{Voto} \leq 30)$

Verifica dei vincoli

- Vediamo su un sistema (è importante il concetto non i dettagli del linguaggio)
 - utilizziamo il servizio <https://sqliteonline.com/>
 - in particolare il sistema postgresQL offerto su di esso

Semplici passi

- <https://sqliteonline.com/>
- Prima creiamo e popoliamo le tabelle
- Poi vediamo i vincoli

SQL OnLine IDE

sqliteonline.com

File Owner DB Run Export Import Sign in

SQLite

Table

demo

```
1 SELECT * FROM demo;
```

ID	Name	Hint
1	SQL Online	for Data Science
2	Kirill N.	https://www.linkedin.com/in/sqliteonlinecom
3	Twitter	https://twitter.com/SQLiteOnlineCom
4	Chart	LINE-SELECT name, cos(id), sin(id) FR
5	Short CODE	s* tableName => SELECT * FROM table
6	SqLite 3.36.0	SQL OnLine on JavaScript
7	[RightClick] mouse	Opens many addit

SQL OnLine IDE

sqliteonline.com

File Owner DB Run Export Import Sign in

SQLite

Table

demo

MariaDB

PostgreSQL

MS SQL

Oracle

Docker

Syntax

Business

```
1 SELECT * FROM demo;
```

ID	Name	Hint
1	SQL Online	for Data Science
2	Kirill N.	https://www.linkedin.com/in/sqliteonlinecom
3	Twitter	https://twitter.com/SQLiteOnlineCom
4	Chart	LINE-SELECT name, cos(id), sin(id) FR
5	Short CODE	s* tableName => SELECT * FROM table
6	SqLite 3.36.0	SQL OnLine on JavaScript
7	[RightClick] mouse	Opens many addit

The screenshot shows the SQLite Online IDE interface. The browser address bar displays 'sqliteonline.com'. The top navigation bar includes buttons for 'File', 'Owner DB', 'Run', 'Export', and 'Import'. On the left sidebar, the 'PostgreSQL' option is highlighted with a red circle and contains a 'Click to connect' button. The main area shows a PostgreSQL editor with a single line of code '1 |'. Below the editor is a table with columns 'ID', 'Name', and 'Hint'.

ID	Name	Hint
1	SQL Online	for Data Science
2	Kirill N.	https://www.linkedin.com/in/sqliteonlineec
3	Twitter	https://twitter.com/SQLiteOnlineCom
4	Chart	LINE-SELECT name, cos(id), sin(id) FR
5	Short CODE	s* tableName => SELECT * FROM table
6	SQLite 3.36.0	SQL OnLine on JavaScript
7	[RightClick] mouse	Opens many additional

SQLite Online IDE

sqliteonline.com

File Owner DB Run Export Import Sign in

SQLite PostgreSQL

MariaDB

PostgreSQL

TEST - v.4.2.180

Table

demo

MS SQL

Oracle

Docker

Syntax

Business

Inseriamo qui la definizione della base di dati (lucido seguente)

SQLite (1).csv SQLite.csv PostgreSQL (1).csv PostgreSQL.json Mostra tutto

```
DROP TABLE IF EXISTS Esami;
CREATE TABLE Esami (
  Studente INTEGER,
  Voto INTEGER,
  Lode BOOLEAN,
  Corso INTEGER);
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
DROP TABLE IF EXISTS Studenti;
CREATE TABLE Studenti (
  Matricola INTEGER,
  Cognome TEXT,
  Nome TEXT);
INSERT INTO Studenti VALUES (276545, 'Rossi','Mario');
INSERT INTO Studenti VALUES (787643, 'Neri','Piero');
INSERT INTO Studenti VALUES (787643, 'Bianchi','Luca');
```

The screenshot shows the SQLite Online IDE interface. The browser address bar displays 'sqliteonline.com'. The interface includes a top navigation bar with 'File', 'Owner DB', 'Run', 'Export', and 'Import' buttons, along with a 'Sign in' link. A left sidebar lists various database engines: SQLite, MariaDB, PostgreSQL (selected), MS SQL, Oracle, Docker, Syntax, and Business. Under PostgreSQL, a 'TEST - v.4.2.180' instance is shown with a 'demo' table. The main editor area contains the following SQL code:

```
1 DROP TABLE IF EXISTS Esami;
2 CREATE TABLE Esami (
3     Studente INTEGER,
4     Voto INTEGER,
5     Lode BOOLEAN,
6     Corso INTEGER);
7 INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
8 INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
9 INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
10 INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
11 DROP TABLE IF EXISTS Studenti;
12 CREATE TABLE Studenti (
13     Matricola INTEGER,
14     Cognome TEXT,
15     Nome TEXT);
16 INSERT INTO Studenti VALUES (276545, 'Rossi', 'Mario');
17 INSERT INTO Studenti VALUES (787643, 'Neri', 'Piero');
18 INSERT INTO Studenti VALUES (787643, 'Bianchi', 'Luca');
```

At the bottom, a taskbar shows several files: 'SQLite (1).csv', 'SQLite.csv', 'PostgreSQL (1).csv', and 'PostgreSQL.json'. A 'Mostra tutto' button is visible on the right side of the taskbar.

The screenshot shows the SQLite Online IDE interface. The browser address bar is `sqliteonline.com`. The navigation menu includes `File`, `Owner DB`, `Run` (circled in red), `Export`, and `Import`. The left sidebar shows a tree view with `SQLite`, `MariaDB`, `PostgreSQL` (selected), `MS SQL`, `Oracle`, `Docker`, `Syntax`, and `Business`. Under `PostgreSQL`, there is a database `TEST - v.4.2.180` with a table `demo`. The main editor contains the following SQL code:

```
1 DROP TABLE IF EXISTS Esami;
2 CREATE TABLE Esami (
3     Studente INTEGER,
4     Voto INTEGER,
5     Lode BOOLEAN,
6     Corso INTEGER);
7 INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
8 INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
9 INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
10 INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
11 DROP TABLE IF EXISTS Studenti;
12 CREATE TABLE Studenti (
13     Matricola INTEGER,
14     Cognome TEXT,
15     Nome TEXT);
16 INSERT INTO Studenti VALUES (276545, 'Rossi', 'Mario');
17 INSERT INTO Studenti VALUES (787643, 'Neri', 'Piero');
18 INSERT INTO Studenti VALUES (787643, 'Bianchi', 'Luca');
```

The bottom of the interface shows a file explorer with `SQLite (1).csv`, `SQLite.csv`, `PostgreSQL (1).csv`, and `PostgreSQL.json`. A `Mostra tutto` button is visible on the right.

SQL OnLine IDE | sql online ide export - Google

sqliteonline.com

File | Owner DB | Run | Export | Import | Sign in

SQLite | PostgreSQL

MariaDB

PostgreSQL

TEST - v.4.2.180

Table

- demo
- esami**
- studenti

MS SQL

Oracle

Docker

Syntax

```
1 DROP TABLE IF EXISTS Esami;
2 CREATE TABLE Esami (
3     Studente INTEGER,
4     Voto INTEGER,
5     Lode BOOLEAN,
6     Corso INTEGER);
7 INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
8 INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
9 INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
10 INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
11 DROP TABLE IF EXISTS Studenti;
12 CREATE TABLE Studenti (
13     Matricola INTEGER,
14     Cognome TEXT,
15     Nome TEXT);
16 INSERT INTO Studenti VALUES (276545, 'Rossi', 'Mario');
17 INSERT INTO Studenti VALUES (787643, 'Neri', 'Piero');
18 INSERT INTO Studenti VALUES (787643, 'Bianchi', 'Luca');
```

SQLite (1).csv | SQLite.csv | PostgreSQL (1).csv | PostgreSQL.json | Mostra tutto

Per vedere il contenuto: tasto destro sul nome della tabella e poi 'SELECT'

```
CREATE TABLE Esami (  
  Studente INTEGER,  
  Voto INTEGER,  
  Lode BOOLEAN,  
  Corso INTEGER);  
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);  
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);  
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);  
INSERT INTO Esami VALUES (276545, 32, FALSE, 01);  
  
EXISTS Studenti;  
Studenti (  
  INTEGER,  
  XT,  
  );  
tudenti VALUES (276545, 'Rossi', 'Mario');  
tudenti VALUES (787643, 'Neri', 'Piero');  
tudenti VALUES (787643, 'Bianchi', 'Luca');
```

SQL OnLine IDE | sql online ide export - Google | +

sqliteonline.com

File | Owner DB | Run | Export | Import | Sign in

SQLite | PostgreSQL | MariaDB

PostgreSQL

- TEST - v.4.2.180
- Table
 - demo
 - esami
- Column
 - studente integer
 - voto integer
 - lode boolean
 - corso integer
- studenti

```
1 SELECT * FROM esami
```

studente	voto	lode	corso
276545	30	true	2
787643	27	true	3
739430	24	false	4
276545	32	false	1

SQLite (1).csv | SQLite.csv | PostgreSQL (1).csv | PostgreSQL.json | Mostra tutto

Semplici passi

- <https://sqliteonline.com/>
- Prima creiamo e popoliamo le tabelle
- Poi vediamo i vincoli

Vincolo di dominio, in concreto

- Una possibile sintassi:
 - combinazione booleana (AND, OR e NOT)
 - di condizioni semplici che confrontano un attributo con una costante

$(\text{Voto} \geq 18) \text{ AND } (\text{Voto} \leq 30)$

Una sintassi reale (SQL)

```
ALTER TABLE Esami ADD CONSTRAINT IntervalloVoto  
CHECK (Voto >= 18 AND Voto <= 30);
```

- Proviamo a definire il vincolo sulla nostra base di dati

The screenshot shows the SQLiteonline.com web interface. The browser address bar displays `https://sqliteonline.com`. The page title is "sqliteonline.com". The interface includes a navigation menu with options like "File", "Owner DB", "Run", and "Export". A sidebar on the left lists various database systems: SQLite, MariaDB, PostgreSQL (selected), MS SQL, Oracle, Docker, Syntax, and Business. Under PostgreSQL, there is a sub-menu for "TEST - v.4.2.180" and a list of tables: "demo", "esami", and "studenti". The main content area displays a SQL query: `ALTER TABLE Esami ADD CONSTRAINT IntervalloVoto CHECK (Voto >= 18 AND Voto <= 30);`. A red error message box is overlaid on the query, stating: "PostgreSQL 23514 check constraint 'IntervalloVoto' of relation 'esami' is violated by some row". The bottom of the interface shows a "History" section with tabs for "Table", "Syntax", and "History". The browser's taskbar at the bottom displays several open files: "SQLite (1).csv", "SQLite.csv", "PostgreSQL (1).csv", and "PostgreSQL.json".

Il vincolo funziona

- Se la base di dati non avesse avuto la ennupla incriminata, allora non avremmo avuto problemi

```
DROP TABLE IF EXISTS Esami;
```

```
CREATE TABLE Esami (
```

```
  Studente INTEGER,
```

```
  Voto INTEGER,
```

```
  Lode BOOLEAN,
```

```
  Corso INTEGER);
```

```
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
```

```
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
```

```
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
```

```
INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
```

Browser window showing the SQLite online IDE interface. The URL is <https://sqliteonline.com>. The interface includes a menu bar with options like File, Owner DB, Run, Export, and Import. The main area displays SQL code for creating and inserting data into a table named 'Esami' in a PostgreSQL database. The code includes a CHECK constraint named 'IntervalloVoto'.

```
1 DROP TABLE IF EXISTS Esami;
2 CREATE TABLE Esami (
3     Studente INTEGER,
4     Voto INTEGER,
5     Lode BOOLEAN,
6     Corso INTEGER);
7 INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
8 INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
9 INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
10 ALTER TABLE Esami ADD CONSTRAINT IntervalloVoto CHECK (Voto >= 18 AND Voto <= 30);
11
```

The interface also shows a sidebar with database connections (SQLite, MariaDB, PostgreSQL, MS SQL, Oracle, Docker, Syntax, Business) and a table list for the selected PostgreSQL database (demo, esami, studenti). A 'History' section at the bottom shows the executed SQL command. A red 'OK' message is displayed on the right side of the screen.

SQLite (1).csv SQLite.csv PostgreSQL (1).csv PostgreSQL.json Mostra tutto

Verifica dei vincoli

- Per vedere il concetto abbiamo visto
 - prima i dati, poi i vincoli
 - su una base di dati esistente
 - proviamo a definire un vincolo, il sistema controlla il soddisfacimento
 - se è violato, non permette di definirlo

Verifica dei vincoli

- Per vedere il concetto abbiamo visto
 - prima i dati, poi i vincoli
 - su una base di dati esistente
 - proviamo a definire un vincolo, il sistema controlla il soddisfacimento
 - se è violato, non permette di definirlo
- In realtà, di solito
 - prima i vincoli, poi i dati
 - definiamo lo schema, con i vincoli
 - modifichiamo i dati (inserimenti, eliminazioni, ...), il sistema controlla i vincoli
 - se un vincolo è violato, la modifica viene rifiutata

Il vincolo funziona

- A partire dalla base di dati corretta

```
DROP TABLE IF EXISTS Esami;  
CREATE TABLE Esami (  
  Studente INTEGER,  
  Voto INTEGER,  
  Lode BOOLEAN,  
  Corso INTEGER);  
ALTER TABLE Esami ADD CONSTRAINT IntervalloVoto  
  CHECK (Voto >= 18 AND Voto <= 30);  
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);  
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);  
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
```

- Proviamo ad inserire la ennupla scorretta

```
INSERT INTO Esami VALUES (276545, 32, FALSE, 01);
```

The screenshot shows the SQLiteOnline web interface. The browser address bar displays `https://sqliteonline.com`. The interface includes a navigation menu with options like 'File', 'Owner DB', 'Run', 'Export', and 'Import'. On the left, a sidebar lists database engines: SQLite, MariaDB, PostgreSQL (selected), MS SQL, Oracle, Docker, and Syntax. Under PostgreSQL, a database instance 'TEST - v.4.2.180' is shown with tables 'demo', 'esami', and 'studenti'. The main editor area contains the following SQL code:

```
1 DROP TABLE IF EXISTS Esami;
2 CREATE TABLE Esami (
3     Studente INTEGER,
4     Voto INTEGER,
5     Lode BOOLEAN,
6     Corso INTEGER);
7 ALTER TABLE Esami ADD CONSTRAINT IntervalloVoto
8     CHECK (Voto >= 18 AND Voto <= 30);
9 INSERT INTO Esami VALUES (276545, 30, TRUE, 02);
10 INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
11 INSERT INTO Esami VALUES (739430, 24, FALSE, 04);
12
13
```

Below the code editor is a 'History' section with tabs for 'Table', 'Syntax', and 'History'. A preview of the executed SQL code is visible. At the bottom, a file manager shows several CSV and JSON files: 'SQLite (1).csv', 'SQLite.csv', 'PostgreSQL (1).csv', and 'PostgreSQL.json'. A 'Mostra tutto' button is also present.

OK

The screenshot shows the SQLiteonline.com web interface. The browser address bar displays `https://sqliteonline.com`. The interface includes a top navigation bar with options like 'File', 'Owner DB', 'Run', and 'Export'. A sidebar on the left lists various database engines: SQLite, MariaDB, PostgreSQL (selected), MS SQL, Oracle, Docker, and Syntax. Under PostgreSQL, a database instance 'TEST - v.4.2.180' is shown with tables 'demo', 'esami', and 'studenti'. The main editor area contains the following SQL code:

```
1 INSERT INTO Esami VALUES (276545, 32, FALSE, 01);  
2  
3  
4
```

A red error message box is overlaid on the code, stating: "PostgreSQL: 23514 new row for relation "esami" violates check constraint "intervallovoto"". Below the editor, a 'History' section shows a previous execution of the same SQL statement. At the bottom, a file manager shows several files: 'SQLite (1).csv', 'SQLite.csv', 'PostgreSQL (1).csv', and 'PostgreSQL.json'. A 'Mostra tutto' button is visible on the right side of the file manager.

Tipi di vincoli

- vincoli intrarelazionali
 - vincoli su valori (o di dominio)
 - vincoli di **ennupla**
 - vincoli di chiave
- vincoli interrelazionali

Vincolo di ennupla

- Esprime condizioni sui valori di ciascuna ennupla, indipendentemente dalle altre ennuple
- Il vincolo di dominio è un caso particolare

Sintassi ed esempi

- La sintassi: come il vincolo di dominio, ma in generale su più attributi
 - combinazione booleana
 - di condizioni semplici che confrontano un attributo con una costante o due attributi (o anche espressioni su di essi)

NOT ((Voto <> 30) AND (Lode = "sì"))

Nella sintassi reale (SQL)

```
ALTER TABLE Esami ADD CONSTRAINT LodeSoloConTrenta  
CHECK (NOT (Voto<>30 AND Lode=True));
```

- Verifichiamo anche questo sul sistema
 - prima con la violazione
 - poi senza

Prima verifica

```
DROP TABLE IF EXISTS Esami;  
CREATE TABLE Esami (  
  Studente INTEGER,  
  Voto INTEGER,  
  Lode BOOLEAN,  
  Corso INTEGER);  
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);  
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);  
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);  
ALTER TABLE Esami ADD CONSTRAINT LodeSoloConTrenta  
  CHECK (NOT (Voto<>30 AND Lode=True));
```

Seconda verifica

```
DROP TABLE IF EXISTS Esami;  
CREATE TABLE Esami (  
    Studente INTEGER,  
    Voto INTEGER,  
    Lode BOOLEAN,  
    Corso INTEGER);  
ALTER TABLE Esami ADD CONSTRAINT LodeSoloConTrenta  
    CHECK (NOT (Voto<>30 AND Lode=True));  
INSERT INTO Esami VALUES (276545, 30, TRUE, 02);  
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);  
INSERT INTO Esami VALUES (739430, 24, FALSE, 04);  
  
INSERT INTO Esami VALUES (787643, 27, TRUE, 03);
```

Vincoli di ennupla, altro esempio

Stipendi	Impiegato	Lordo	Ritenute	Netto
	Rossi	55.000	12.500	42.500
	Neri	45.000	10.000	35.000
	Bruni	47.000	11.000	36.000

Ritenute ≥ 0 AND (Lordo = (Ritenute + Netto))

Vincoli di ennupla, violazione

Stipendi	Impiegato	Lordo	Ritenute	Netto
	Rossi	55.000	12.500	42.500
	Neri	45.000	10.000	35.000
	Bruni	50.000	11.000	36.000

$$\text{Lordo} = (\text{Ritenute} + \text{Netto})$$

Tipi di vincoli

- vincoli intrarelazionali
 - vincoli su valori (o di dominio)
 - vincoli di enunzia
 - vincoli di **chiave**
- vincoli interrelazionali

Il tipo di vincolo più importante

- Matricola
- Codice fiscale
- Codice del corso
- ...

Identificazione delle ennuple

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/98
78763	Rossi	Mario	Ing Inf	3/11/96
65432	Neri	Piero	Ing Mecc	10/7/99
87654	Neri	Mario	Ing Inf	3/11/96
67653	Rossi	Piero	Ing Mecc	5/12/98

- non ci sono due ennuple con lo stesso valore sull'attributo Matricola
- Matricola è la **chiave** della relazione

Vincolo di chiave

- Se un insieme K è chiave,
 - imponiamo che non ci siano due ennuple uguali fra loro su K

Chiavi, in SQL

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/98
78763	Rossi	Mario	Ing Inf	3/11/96
65432	Neri	Piero	Ing Mecc	10/7/99
87654	Neri	Mario	Ing Inf	3/11/96
67653	Rossi	Piero	Ing Mecc	5/12/98

```
CREATE TABLE Studenti (  
  Matricola INTEGER UNIQUE,  
  Cognome TEXT,  
  Nome TEXT,  
  ....);
```

Più chiavi

Matricola	Cognome	Nome	Corso	CF
27655	Rossi	Mario	Ing Inf	RSSMRI98...
78763	Rossi	Mario	Ing Inf	RSSMRI99...
65432	Neri	Piero	Ing Mecc	NRIPTR95...
87654	Neri	Mario	Ing Inf	NRIMRI99...
67653	Rossi	Piero	Ing Mecc	RSSPRI99...

Più chiavi

Matricola	Cognome	Nome	Corso	CF
27655	Rossi	Mario	Ing Inf	RSSMRI98...
78763	Rossi	Mario	Ing Inf	RSSMRI99...
65432	Neri	Piero	Ing Mecc	NRIPTR95...
87654	Neri	Mario	Ing Inf	NRIMRI99...
67653	Rossi	Piero	Ing Mecc	RSSPRI99...

```
CREATE TABLE Studenti (  
  Matricola INTEGER UNIQUE,  
  Cognome TEXT,  
  Nome TEXT,  
  Corso TEXT,  
  CF TEXT UNIQUE);
```

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
8765	Neri	Paolo	03/11/1996
9283	Verdi	Luisa	12/11/1999
3456	Rossi	Maria	01/02/1998

esami

Studente	Voto	Corso
3456	30	04
3456	30	02
9283	26	01
6554	26	01

corsi

Codice	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

Chiave su più attributi

esami

Studente	Voto	Corso
3456	30	04
3456	30	02
9283	26	01
6554	26	01

Chiave su più attributi

esami

Studente	Voto	Corso
3456	30	04
3456	30	02
9283	26	01
6554	26	01

```
CREATE TABLE Esami (  
  Studente INTEGER,  
  Voto INTEGER,  
  Corso TEXT,  
  UNIQUE (Studente, Corso)  
);
```

"Minimalità" delle chiavi

- Una chiave di una relazione è un insieme di attributi che, insieme, permettono di identificare le ennuple e **che è minimale rispetto a questa proprietà**
 - Anche Matricola e Cognome identificano le ennuple della relazione studenti, ma non formano una chiave, perché l'insieme non è minimale:
 - Matricola identifica ed è sottoinsieme di Matricola, Cognome
 - Matricola, Cognome formano una **superchiave**
 - Una **chiave** è una **superchiave minimale**

Esistenza delle chiavi

- Ogni relazione è un insieme:
 - non può contenere due ennuple uguali
- Ogni relazione ha come superchiave l'insieme degli attributi su cui è definita
- e quindi ha (almeno) una chiave

Importanza delle chiavi

- L'esistenza delle chiavi garantisce l'accessibilità a ciascun dato della base di dati
- le chiavi permettono di correlare i dati in relazioni diverse:
 - il modello relazionale è basato su valori

Chiavi e valori nulli

- In presenza di valori nulli, i valori della chiave non permettono
 - di identificare le ennuple
 - di realizzare facilmente i riferimenti da altre relazioni

Matricola	Cognome	Nome	Corso	CF
<i>NULL</i>	Rossi	Mario	Ing Inf	RSSMRI93...
78763	Rossi	Mario	Ing Inf	<i>NULL</i>
65432	Neri	<i>NULL</i>	Ing Mecc	NRIPTR95...
<i>NULL</i>	Neri	Mario	Ing Inf	NRIMRI99...
67653	Rossi	Piero	Ing Mecc	RSSPRI98...

- La presenza di valori nulli, soprattutto nelle chiavi, deve essere limitata o almeno controllata

Chiave primaria

- Chiave su cui non sono ammessi nulli
- Notazione: sottolineatura

<u>Matricola</u>	Cognome	Nome	Corso	CF
27655	Rossi	Mario	Ing Inf	RSSMRI93...
78763	Rossi	Mario	Ing Inf	<i>NULL</i>
65432	Neri	<i>NULL</i>	Ing Mecc	NRIPTR95...
87654	Neri	Mario	Ing Inf	NRIMRI99...
67653	Rossi	Piero	Ing Mecc	RSSPRI98...

Notazione SQL

<u>Matricola</u>	Cognome	Nome	Corso	CF
27655	Rossi	Mario	Ing Inf	RSSMRI93...
78763	Rossi	Mario	Ing Inf	<i>NULL</i>
65432	Neri	<i>NULL</i>	Ing Mecc	NRIPTR95...
87654	Neri	Mario	Ing Inf	NRIMRI99...
67653	Rossi	Piero	Ing Mecc	RSSPRI98...

```
CREATE TABLE Studenti (  
  Matricola INTEGER PRIMARY KEY,  
  Cognome TEXT,  
  Nome TEXT,  
  Corso TEXT,  
  CF TEXT UNIQUE);
```

studenti

<u>Matricola</u>	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1998
8765	Neri	Paolo	03/11/1996
9283	Verdi	Luisa	12/11/1999
3456	Rossi	Maria	01/02/1998

esami

<u>Studente</u>	Voto	<u>Corso</u>
3456	30	04
3456	30	02
9283	26	01
6554	26	01

corsi

<u>Codice</u>	Titolo	Docente
01	Analisi	Mario
02	Chimica	Bruni
04	Chimica	Verdi

esami

<u>Studente</u>	Voto	<u>Corso</u>
3456	30	04
3456	30	02
9283	26	01
6554	26	01

```
CREATE TABLE esami (  
  Studente INTEGER,  
  Voto INTEGER,  
  Corso INTEGER,  
  PRIMARY KEY (Studente, Corso));
```

Tipi di vincoli

- vincoli intrarelazionali
 - vincoli su valori (o di dominio)
 - vincoli di enunzia
 - vincoli di chiave
- vincoli interrelazionali
 - vincoli di **integrità referenziale**

Integrità referenziale

Infrazioni

<u>Codice</u>	Data	Vigile	Stato	Targa
34321	1/2/2018	3987	I	AA954KK
53524	4/3/2020	3295	F	BB395BB
64521	5/4/2020	3295	E	BB395BB
73321	5/2/2020	9345	E	BB395BB

Vigili

<u>Matricola</u>	Cognome	Nome
3987	Rossi	Luca
3295	Neri	Piero
9345	Neri	Mario
7543	Mori	Gino

Infrazioni

<u>Codice</u>	Data	Vigile	Stato	Targa
34321	1/2/2018	3987	I	AA954KK
53524	4/3/2020	3295	F	BB395BB
64521	5/4/2020	3295	E	BB395BB
73321	5/2/2020	9345	E	BB395BB

Auto

<u>Stato</u>	<u>Targa</u>	Cognome	Nome
I	AA954KK	Rossi	Mario
F	BB395BB	Rossi	Mario
E	BB395BB	Neri	Luca

Vincolo di integrità referenziale

- Vincolo di integrità referenziale (“foreign key”) fra X di R_1 e R_2
 - impone ai valori su X in R_1 di comparire come valori della chiave primaria di R_2

- vincoli di integrità referenziale fra:
 - l'attributo Vigile della relazione INFRAZIONI e la relazione VIGILI
 - gli attributi Stato e Targa di INFRAZIONI e la relazione AUTO

Violazione di vincolo di integrità referenziale

Infrazioni

<u>Codice</u>	Data	Vigile	Stato	Targa
34321	1/2/2018	3987	I	AA954KK
53524	4/3/2020	3295	F	EE395BB
64521	5/4/2020	3295	E	BB395BB
73321	5/2/2020	9345	E	BB395BB

Auto

<u>Stato</u>	<u>Targa</u>	Cognome	Nome
I	EE395BB	Rossi	Mario
F	FF342BB	Rossi	Mario
E	BB395BB	Neri	Luca

Definizioni in SQL

```
CREATE TABLE Auto (  
  Stato TEXT,  
  Targa INTEGER,  
  PRIMARY KEY (Stato,Targa),  
  Cognome TEXT,  
  Nome TEXT  
);  
CREATE TABLE Vigili (  
  Matricola INTEGER PRIMARY KEY,  
  Cognome TEXT,  
  Nome TEXT  
);  
CREATE TABLE Infrazioni (  
  Codice INTEGER PRIMARY KEY,  
  Data DATE,  
  Vigile INTEGER REFERENCES Vigili(Matricola),  
  Stato TEXT,  
  Targa INTEGER,  
  FOREIGN KEY (Stato, Targa) REFERENCES Auto (Stato, Targa)  
);
```

Integrità referenziale e valori nulli

Impiegati

<u>Matricola</u>	Cognome	Progetto
34321	Rossi	IDEA
53524	Neri	XYZ
64521	Verdi	<i>NULL</i>
73032	Bianchi	IDEA

Progetti

<u>Codice</u>	Inizio	Durata	Costo
IDEA	01/2019	36	200
XYZ	07/2020	24	120
BOH	09/2019	24	150

Vincoli multipli su più attributi

Incidenti

<u>Codice</u>	Data	StatoA	TargaA	StatoB	TargaB
34321	1/2/19	I	EE954AB	F	AA954AK
64521	5/4/20	E	BB395AB	I	EE395AB

Auto

<u>Stato</u>	<u>Targa</u>	Cognome	Nome
F	AA954AK	Rossi	Mario
I	EE395AB	Rossi	Mario
E	BB395AB	Neri	Luca

Vincoli multipli su più attributi, 2

- vincoli di integrità referenziale fra:
 - gli attributi StatoA e TargaA di INCIDENTI e la relazione AUTO
 - gli attributi StatoB e TargaB di INCIDENTI e la relazione AUTO
- L'ordine degli attributi è significativo

Ordine degli attributi: notazione posizionale

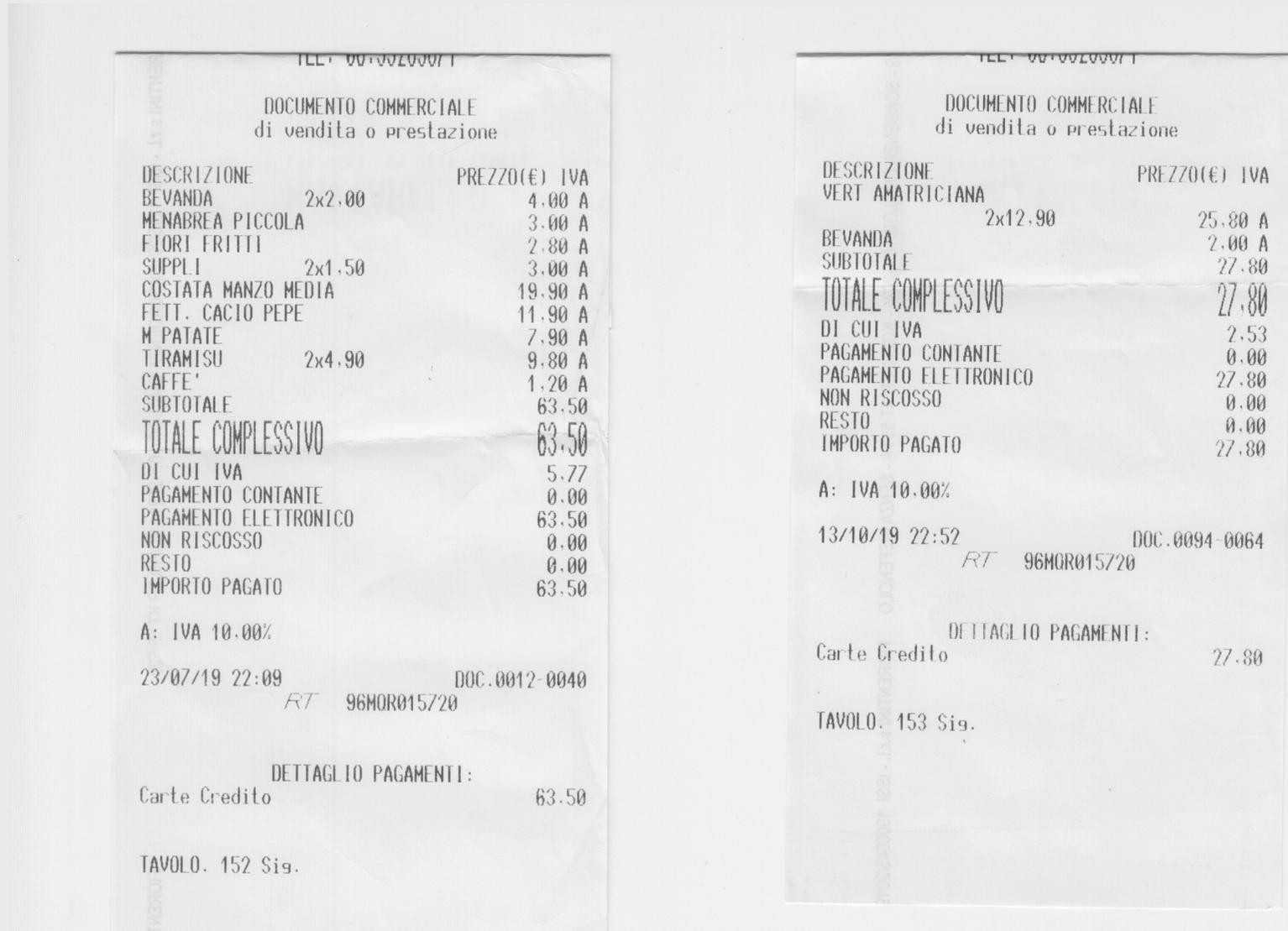
```
CREATE TABLE Auto (  
  Stato TEXT,  
  Targa INTEGER,  
  PRIMARY KEY (Stato,Targa),  
  Cognome TEXT,  
  Nome TEXT  
);  
CREATE TABLE Incidenti (  
  Codice INTEGER PRIMARY KEY,  
  Data DATE,  
  StatoA TEXT,  
  TargaA INTEGER,  
  FOREIGN KEY (StatoA, TargaA) REFERENCES Auto (Stato, Targa),  
  StatoB TEXT,  
  TargaB INTEGER,  
  FOREIGN KEY (StatoB, TargaB) REFERENCES Auto (Stato, Targa)  
);  
• OK
```

Ordine degli attributi: notazione posizionale

```
CREATE TABLE Auto (  
  Stato TEXT,  
  Targa INTEGER,  
  PRIMARY KEY (Stato,Targa),  
  Cognome TEXT,  
  Nome TEXT  
);  
CREATE TABLE Incidenti (  
  Codice INTEGER PRIMARY KEY,  
  Data DATE,  
  StatoA TEXT,  
  TargaA INTEGER,  
  FOREIGN KEY (StatoA, TargaA) REFERENCES Auto (Stato, Targa),  
  StatoB TEXT,  
  TargaB INTEGER,  
  FOREIGN KEY (StatoB, TargaB) REFERENCES Auto (Targa, Stato)  
);  
• ????
```

Esempi

Strutture nidificate



Strutture nidificate

<i>DA FILIPPO VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE 1235 DEL 12/10/2019</i>		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
TOTALE		39,20

<i>DA FILIPPO VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE 1240 DEL 13/10/2019</i>		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
TOTALE		39,00

Esercizio

- Rappresentare in una base di dati relazionale i dati di interesse delle ricevute fiscali di un ristorante

Strutture nidificate

<i>DA FILIPPO VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE 1235 DEL 12/10/2019</i>		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
TOTALE		39,20

<i>DA FILIPPO VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE 1240 DEL 13/10/2019</i>		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
TOTALE		39,00

Strutture nidificate

<i>DA FILIPPO</i> <i>VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE</i> <i>1235 DEL 12/10/2019</i>		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
		.
TOTALE		39,20

<i>DA FILIPPO</i> <i>VIA ROMA 2, ROMA</i>		
<i>RICEVUTA FISCALE</i> <i>1240 DEL 13/10/2019</i>		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
TOTALE		39,00

"Soluzione"

- alla lavagna

NUMERO	DATA	DETTAGLIO	TOTALE
1235	12/10/19	<div style="border: 1px solid black; padding: 5px;"> 3 COPERTI 3,20 2 ANTIPASTI 6,20 3 PRIMI 17,00 . . - </div>	39,20
1240	13/10/19		39,00

Strutture nidificate

Ricevute

<u>Numero</u>	Data	Qtà	Descrizione	Importo	Totale
1235	12/10/2019	3	Coperti	3,00	39,20
		2	Antipasti	6,20	
		3	Primi	12,00	
		2	Bistecche	18,00	
1240	13/10/2019	2	Coperti	2,00	39,00
		

- Ma i valori debbono essere semplici, non relazioni!

RICEVUTE

<u>NUMERO</u>	DATA	TOTALE
1235	12/10/2019	39,20
1240	13/10/2019	39,00

DETTAGLIO

<u>RICEVUTA</u>	QTA	<u>DESC</u>	IMPORTO
1235	3	COPERTI	3,80
1235	2	ANTIPASTI	6,20
...			
1240	2	COPERTI	2,00

Relazioni che rappresentano strutture nidificate

Ricevute

<u>Numero</u>	Data	Totale
1235	12/10/2019	39,20
1240	13/10/2019	39,00

Dettaglio

<u>Numero</u>	Qtà	<u>Descrizione</u>	Importo
1235	3	Coperti	3,00
1235	2	Antipasti	6,20
1235	3	Primi	12,00
1235	2	Bistecche	18,00
1240	2	Coperti	2,00
...

Commento

- Morale: le tabelle hanno struttura piatta, i dati di interesse spesso sono più articolati ("struttura nidificata")
 - Nel modello relazionale dobbiamo utilizzare più relazioni (tabelle) per rappresentare gli aspetti a diverso livello di dettaglio
 - In effetti, si tratta di concetti diversi
 - Approfondiremo il problema, con respiro maggiore, quando parleremo dei passi iniziali della progettazione

Secondo commento

- Una considerazione che potremmo fare in tanti casi, forse tutti
 - Abbiamo rappresentato veramente tutti gli aspetti delle ricevute?
 - Dipende da che cosa ci interessa!
 - l'ordine delle righe e' rilevante?
 - linee ripetute
 - al bar ... "ditemi le cose insieme ..."

"Soluzione"

- alla lavagna

Rappresentazione alternativa per strutture nidificate

Ricevute

<u>Numero</u>	Data	Totale
1235	12/10/2019	39,20
1240	13/10/2019	39,00

Dettaglio

<u>Numero</u>	<u>Riga</u>	Qtà	Descrizione	Importo
1235	1	3	Coperti	3,00
1235	2	2	Antipasti	6,20
1235	3	3	Primi	12,00
1235	4	2	Bistecche	18,00
1240	1	2	Coperti	2,00
...

10/10/2022

Atzeni e Marialdo

155

Rappresentazione alternativa per strutture nidificate, altra possibilità

Ricevute

<u>Numero</u>	Data	Totale
1235	12/10/2019	39,20
1240	13/10/2019	39,00

Dettaglio

<u>ID</u>	Numero	Riga	Qtà	Descrizione	Importo
11	1235	1	3	Coperti	3,00
12	1235	2	2	Antipasti	6,20
13	1235	3	3	Primi	12,00
14	1235	4	2	Bistecche	18,00
15	1240	1	2	Coperti	2,00
...

Riflessione

- Le basi di dati contengono dati che rappresentano le informazioni di interesse
- "All models are wrong, but some are useful" (G. Box e altri)
- La mappa dell'impero in scala 1:1 (Borges, Eco)

La mappa dell'impero (Borges)

- *“...In quell'Impero, l'Arte della Cartografia giunse a una tal Perfezione che la Mappa di una sola Provincia occupava tutta una Città, e la Mappa dell'Impero tutta una Provincia. Col tempo, queste Mappe smisurate non bastarono più. I Collegi dei Cartografi fecero una Mappa dell'Impero che aveva l'Immensità dell'Impero e coincideva perfettamente con esso. Ma le Generazioni Seguenti, meno portate allo Studio della Cartografia, pensarono che questa Mappa enorme era inutile e non senza Empietà la abbandonarono alle Inclemenze del Sole e degli Inverni. Nei Deserti dell'Ovest sopravvivono lacerate Rovine della Mappa, abitate da Animali e Mendichi; in tutto il Paese non c'è altra Reliquia delle Discipline Geografiche.”*
(Da *Viajes de Varones Prudentes di Suàrez Miranda, libro IV, cap. XIV, Lérida, 1658. Citato da Jorge Luis Borges, Storia universale dell'infamia, “Etc.”*)