



Informatica**Umanistica**

Basi di dati

Il modello relazionale

Anna Monreale
Università di Pisa

Sommario

- **Introduzione**
- **Base di dati, tabella, ennupla, attributo, dominio**
- **Valori nulli**
- **Vincoli di integrità**
 - di chiave
 - di ennupla
 - di riferimento
- **Caratteristiche del modello**

Introduzione

- **Modello logico dei DBMS commerciali**
- **Storia:**
 - Basato sul “**Modello Relazionale**” [E. F. Codd, 1970]
 - Centrato sull'indipendenza dei dati
 - Disponibile in DBMS reali dal 1981
 - ANSI/ISO SQL-92

Attenzione

- L'obiettivo di questa lezione è descrivere le **caratteristiche del modello**
- Non ci occupiamo, per ora di:
 - **Linguaggio**: delle tecniche per manipolare i dati
 - **Metodologia di progetto**: dei metodi per costruire la base di dati
 - come scegliere le tabelle e valutare la qualità

Intuizione

- **Base di dati**
 - insieme di **tabelle**
- **Tabella**
 - collezione di **ennuple**
- **Ennupla**
 - insieme di coppie (**attributo, valore**)
 - analogo nei linguaggi di programmazione: “**struttura**”
o “**record**”

Esempio: Studenti, Corsi, Esami

- **Base di dati universitari**
- **Studenti**
 - nome, cognome, matricola, data di nascita
- **Corsi**
 - codice, nome del corso, nome del docente
- **Esami sostenuti**
 - corso, studente, voto

Esempio: Studenti, Corsi, Esami

- **Studenti**
 - **nome**: stringa
 - **cognome**: stringa
 - **matricola**: intero
 - **data di nascita**: data
- **Corsi**
 - **codice**: stringa
 - **nome**: stringa
 - **nome del docente**: stringa
- **Esami Sostenuti**
 - **corso**: “riferimento” ad un corso
 - **studente**: “riferimento” ad uno studente
 - **voto**: intero
 - **lode**: sì/no

Studenti

Tabella: relazione
(istanza)

Ennupla

Attributo: proprietà di
interesse

valore

Studenti	Matricola	Cognome	Nome	Data di nascita
	276545	Rossi	Maria	25/11/1991
	485745	Neri	Anna	23/04/1992
	200768	Verdi	Fabio	12/02/1992
	587614	Rossi	Luca	10/10/1991
	937653	Bruni	Mario	01/12/1991

```
TABLE Studenti(  
  matricola integer,  
  cognome char(20),  
  nome char(20),  
  dataNascita date);
```

Schema Relazione
esempio di sintassi

Dominio
Tipo

Attributi

Studenti	Matricola	Cognome	Nome	Data di nascita
	276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992	
200768	Verdi	Fabio	12/02/1992	
587614	Rossi	Luca	10/10/1991	
937653	Bruni	Mario	01/12/1991	

```
TABLE Studenti(  
  matricola integer,  
  cognome char(20),  
  nome char(20),  
  dataNascita date);
```

- Ogni attributo dispone di un **dominio** che definisce **l'insieme di valori validi** per quell'attributo.
 - Es. $\text{dom}(\text{matricola}) = \textit{integer}$
- E' possibile avere **domini ripetuti** nella stessa relazione!

Vincoli sull'ordine dei dati

Studenti



Matricola	Cognome	Nome	Data di nascita
276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
587614	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991

- L'ordinamento delle righe è irrilevante
- L'ordinamento delle colonne è irrilevante

Vincoli sui dati della relazione

Studenti



Matricola	Cognome	Nome	Data di nascita
20/11/1991	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
587614	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991

- Non possono esistere **attributi uguali** (1)
- Non possono esistere **righe uguali** (2)
- I dati di una **colonna** devono essere **omogenei** (3)

Corsi

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli


```
TABLE Corsi (  
  codice char(3),  
  titolo char(50),  
  docente char(20));
```

Esami


ESAMI

Studente	Voto	Lode	Corso
276545	28	0	01
276545	27	0	04
937653	25	0	01
200768	30	1	04

Matricola
di uno
studente



codice di
un corso



```
TABLE Esami (  
  studente integer,  
  voto integer,  
  corso char(3),  
  lode bool);
```

Tabelle

STUDENTI

Matricola	Cognome	Nome	Data di nascita
276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
587614	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991

ESAMI

Studente	Voto	Lode	Corso
276545	28	0	01
276545	27	0	04
937653	25	0	01
200768	30	1	04

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

Schema della Base di Dati

```
TABLE Studenti(  
    matricola integer,  
    cognome char(20),  
    nome char(20),  
    dataNascita date);
```

```
TABLE Esami(  
    studente integer,  
    voto integer,  
    corso char(3),  
    lode bool);
```

```
TABLE Corsi(  
    codice char(3),  
    titolo char(50),  
    docente char(20));
```

Valori Nulli

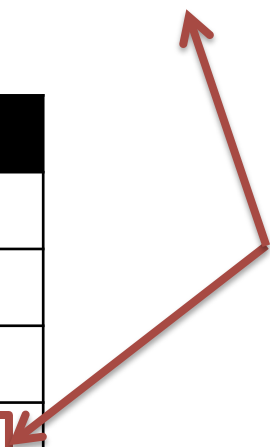
STUDENTI

Matricola	Cognome	Nome	Data di nascita
276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
587614	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991
993354	Gialli	Lucia	null

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli
05	Basi Dati	null

Valore
nullo



Vincoli sui Dati

- **Regole della realtà di interesse**
- **Unicità degli identificatori (chiavi)**
 - codici di corso e matricole
- **Condizioni sui valori di ciascuna tupla**
 - Voti degli studenti
 - Da 18 a 30
 - lode solo se il voto è 30
- **Correttezza dei riferimenti**

Chiavi

- **Chiave:** insieme minimale di attributi che **identifica univocamente** le tuple di una relazione

STUDENTI

	Matricola	Cognome	Nome	Data di nascita
Chiave 	276545	Rossi	Maria	25/11/1991
	485745	Neri	Anna	23/04/1992
	200768	Verdi	Fabio	12/02/1992
	587614	Rossi	Luca	10/10/1991
	937653	Bruni	Mario	01/12/1991

- $\{\text{Cognome}, \text{Nome}\}$: causalmente chiave!
- Una chiave (*primaria*) non può avere valore *null*

Una Base di Dati Scorretta

STUDENTI

Matricola	Cognome	Nome	Data di nascita
276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
937653	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991

Unicità
della
matricola



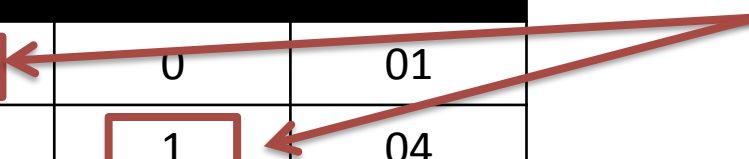
ESAMI

Studente	Voto	Lode	Corso
276545	32	0	01
276545	27	1	04
937653	25	0	01
300300	30	1	04

riferimento
scorretto



Voti
scorretti



Vincoli di Integrità

- **Regole imposte sui valori della base di dati**
- Vincoli sulle singole tabelle
 - **vincoli di chiave**
 - **vincoli di ennupla**
- Vincoli tra tabelle diverse
 - **vincoli di riferimento** o di integrità referenziale

Vincoli di Integrità

- **Vincoli di chiave**
 - chiave: identificatore per le ennuple
 - es: “matricola” è una chiave per “Studenti”
- **Vincoli di ennupla**
 - predicati sui valori delle ennuple
 - es: $(\text{voto} \geq 18 \text{ and } \text{voto} \leq 30)$
- **Vincoli di Riferimento**
 - assenza di riferimenti inesistenti
 - es: esistono esami solo per gli studenti della bd

Vincoli di chiave

Studenti	Matricola	Cognome	Nome	Data di nascita
	276545	Rossi	Maria	25/11/1991
	485745	Neri	Anna	23/04/1992
	200768	Verdi	Fabio	12/02/1992
	587614	Rossi	Luca	10/10/1991
	937653	Bruni	Mario	01/12/1991

```
TABLE Studenti(  
  matricola integer,  
  cognome char(20),  
  nome char(20),  
  dataNascita date,  
  UNIQUE(matricola) );
```

Vincoli di ennupla

ESAMI

Studente	Voto	Lode	Corso
276545	28	0	01
276545	27	0	04
937653	25	0	01
200768	30	1	04

```
TABLE Esami (  
  studente integer,  
  voto integer,  
  corso char(3),  
  lode bool,  
  CHECK (voto>=18 and voto<=30),  
  CHECK (not lode or voto=30) );
```

Vincoli di Riferimento

ESAMI

Studente	Voto	Lode	Corso
276545	28	0	01
276545	27	0	04
937653	25	0	01
200768	30	1	04

```
TABLE Esami (  
  studente integer,  
  voto integer,  
  corso char(3),  
  lode bool,  
  CHECK (voto>=18 and voto<=30),  
  CHECK (not lode or voto=30),  
  FOREIGN KEY(studente) REFERENCES  
    Studenti(matricola),  
  FOREIGN KEY(corso) REFERENCES Corsi(codice));
```


Schema con vincoli di Integrità

```
TABLE Studenti(  
  matricola integer,  
  cognome char(20),  
  nome char(20),  
  dataNascita date,  
  UNIQUE(matricola));
```

```
TABLE Esami(  
  studente integer,  
  voto integer,  
  corso char(3),  
  lode bool,  
  UNIQUE(studente, corso),  
  FOREIGN KEY(studente) REFERENCES Studenti(matricola),  
  FOREIGN KEY(corso) REFERENCES Corsi(codice)  
);
```

```
TABLE Corsi(  
  codice char(3),  
  titolo char(50),  
  docente char(20),  
  UNIQUE(codice));
```

Caratteristiche del Modello

- **Legami tra i dati basati sui valori**
 - **assenza di puntatori**
- **I valori devono essere semplici**
 - valori **atomici** : numeri, caratteri, stringhe, booleani, date ecc.
 - non sono consentite “nidificazioni” (base di dati in “1 forma normale”)
 - differenza con altri modelli (es: strutture)

Puntatori

In molti modelli (es. *gerarchico*, *reticolare*, *oggetti*), i riferimenti tra tabelle sono definiti mediante **puntatori**

STUDENTI

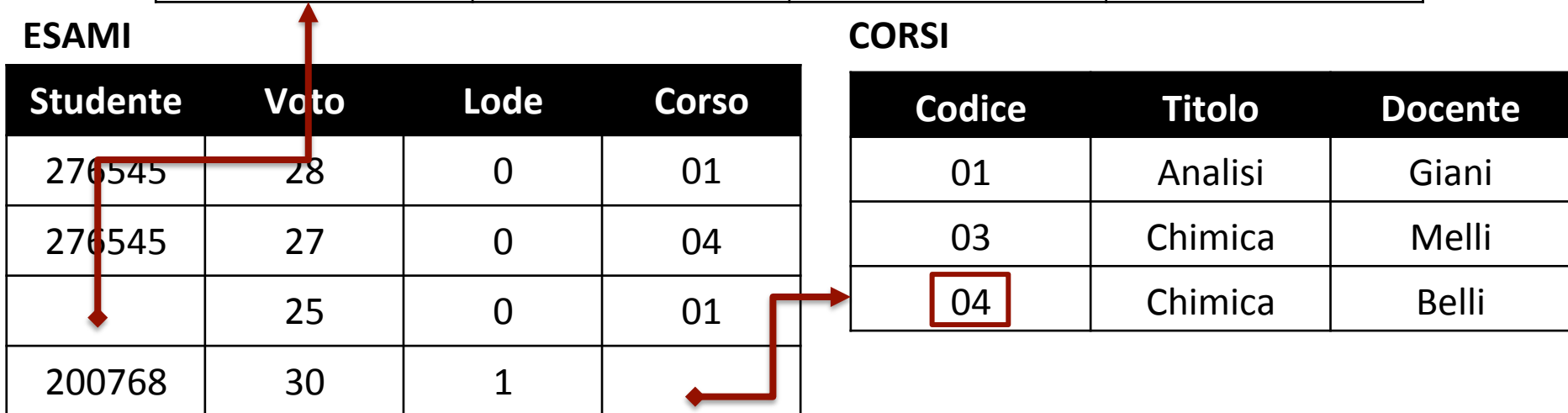
Matricola	Cognome	Nome	Data di nascita
276545	Rossi	Maria	25/11/1991
485745	Neri	Anna	23/04/1992
200768	Verdi	Fabio	12/02/1992
587614	Rossi	Luca	10/10/1991
937653	Bruni	Mario	01/12/1991

ESAMI

Studente	Voto	Lode	Corso
276545	28	0	01
276545	27	0	04
	25	0	01
200768	30	1	

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli



Un Esempio di Informazione Nidificata

<i>Dal Sudicio Via Buia, Pisa</i>		
<i>Ricevuta Fiscale 1235 del 12/10/2001</i>		
3	Coperti	3,00
2	Antipasti	6,20
3	Primi	12,00
2	Bistecche	18,00
<i>Totale</i>		39,20

<i>Dal Sudicio Via Buia, Pisa</i>		
<i>Ricevuta Fiscale 1240 del 13/10/2001</i>		
2	Coperti	2,00
2	Antipasti	7,00
2	Primi	8,00
2	Orate	20,00
2	Caffè	2,00
<i>Totale</i>		39,00

Una Possibile Rappresentazione

Ricevute	numero	data	totale
	1235	12/10/2000	39,20
	1240	13/10/2000	39,00

Dettaglio	numero	qta	portata	prezzo
	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

Modello Relazionale: Relazione

Q₁: Perché si chiama **modello relazionale**?

A₁: Una **relazione sui dati** può essere vista come una **relazione matematica**! (con una leggera variazione).

Q₂:... Com'è definita una relazione matematica nella *teoria degli insiemi*?

Prodotto cartesiano vs relazione

DEF. Dati n insiemi D_1, D_2, \dots, D_n (*domini*), una **relazione matematica** sugli insiemi D_1, D_2, \dots, D_n è definita come un **sottoinsieme del prodotto cartesiano** $D_1 \times D_2 \times \dots \times D_n$.

DEF. Il **prodotto cartesiano** degli insiemi D_1, D_2, \dots, D_n è definito come l'insieme delle tuple ordinate (d_1, d_2, \dots, d_n) , con $d_i \in D_i, \forall i = 1, 2, \dots, n$

Insiemi:

$A = \{1, 2, 4\}$,

$B = \{a, b\}$

$A \times B$

1	a
1	b
2	a
2	b
4	a
4	b

**Prodotto
Cartesiano**

1	a
1	b
4	b

Relazione r_1
 $r_1 \subseteq A \times B$

1	a
2	b

Relazione r_2
 $r_2 \subseteq A \times B$

Relazione vs Relazione matematica

Se usassimo la definizione classica di **relazione matematica** nel modello relazionale dei dati:

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

$$CORSI \subseteq \text{int} \times \text{string} \times \text{string}$$

Problema: *Ordinamento* in una relazione ...

In generale: $A \times B \neq B \times A$

Rappresentazione mediante attributi

Dal punto di vista dei dati, i due schemi sono uguali, ma **non lo sono se consideriamo la definizione di relazione matematica!**

CORSI

Titolo	Codice	Docente
Analisi	01	Giani
Chimica	03	Melli
Chimica	04	Belli

$CORSI \subseteq string \times int \times string$

CORSI

Codice	Titolo	Docente
01	Analisi	Giani
03	Chimica	Melli
04	Chimica	Belli

$CORSI \subseteq int \times string \times string$

Soluzione: Usare rappresentazione **NON posizionale**, mediante gli attributi ...