



Modulo 2 Data Base 2

Università degli Studi di Salerno

Corso di Laurea in Scienze della comunicazione

Informatica generale

Docente: Angela Peduto

A.A. 2004/2005



Relazioni: riepilogo

- **Relazione** : concetto mutuato dalla definizione di relazione matematica della teoria degli insiemi, come sottoinsieme del prodotto cartesiano fra n insiemi. Nel modello relazionale corrisponde ad una struttura dati tabellare.
- **Proprietà** :
 - Ogni n -pla è internamente ordinata: l' i -esimo valore proviene dall' i -esimo dominio (struttura **posizionale**)
 - Non esiste ordinamento intrinseco fra le n -ple, per la natura insiemistica della relazione
 - Non sono ammesse n -ple uguali (ogni elemento di un insieme è unico)
- **Conseguenze** :
 - Lo scambio fra righe di una tabella non modifica la relazione
 - Lo scambio fra colonne di una tabella può portare alla sua inconsistenza con lo schema



Relazioni

In informatica si tende ad utilizzare rappresentazioni **non posizionali** dei dati, utilizzando l'ordinamento solo quando ha utilità pratica (es. parametri di funzioni, elementi di vettori, matrici).

Si utilizzano i tipi strutturati (record) nei casi in cui si debbano raccogliere dati di natura differente in una stessa entità logica.

Una relazione è un **insieme di record omogenei**, cioè definiti sugli stessi campi.

Come ogni campo di un record è associato ad un **nome**, così si associa ad ogni colonna della relazione un **attributo**.

Relazioni

Esempio di relazione con attributi

Casa	Ospiti	RetiCasa	RetiOspiti
Parma	Inter	3	2
Palermo	Lazio	2	0
Milan	Juventus	1	1

Ogni attributo ha un suo *dominio* su cui è definito.

Ogni riga è detta convenzionalmente *tupla* (n-pla).

Quindi una tupla è un insieme di valori, uno per attributo, ordinati secondo lo schema della relazione e definiti ciascuno su un proprio dominio.

Una relazione è una serie di tuple definite sul dominio della relazione (insieme ordinato dei domini dei singoli attributi).



Relazioni

- ✦ Una Base di Dati è generalmente costituita da più di una Relazione
- ✦ **Le corrispondenze fra dati presenti in relazioni diverse sono rappresentate per mezzo di valori comuni** che compaiono in tuple appartenenti a tali relazioni.
- ✦ In altri modelli (Gerarchico e Reticolare) le corrispondenze tra i dati presenti in strutture diverse è fatta tramite riferimenti espliciti (i puntatori)



Relazioni - notazioni

Se t è una tupla definita sullo schema X (insieme ordinato di domini) della relazione e A è uno dei domini di X

$t[A]$ (o $t.A$) è il valore di t relativo al dominio A

Es. (relazione **Partite**)

se t è la prima tupla della relazione

$t.Casa = \text{Parma}$



Basi di dati e Relazioni

Consideriamo il seguente schema di basi di dati:

Studenti (Matricola, Cognome, Nome, DataNascita)

Corsi (Codice, Titolo, Docente)

Esami (Studente, Voto, Corso)

Studenti contiene dati su un insieme di studenti

Corsi contiene dati su un insieme di corsi

Esami contiene dati su un insieme di esami e **fa riferimento** alle altre due attraverso i numeri di matricola e il nome del corso.

Quindi *Matricola* e *Studente*, come anche *Corso* e *Titolo*, sono definiti sullo stesso dominio e possono (in alcuni casi devono) assumere gli stessi valori.

studenti

Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1978
8765	Neri	Paolo	03/11/1976
9283	Verdi	Luisa	12/11/1979
3456	Rossi	Maria	01/02/1978

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

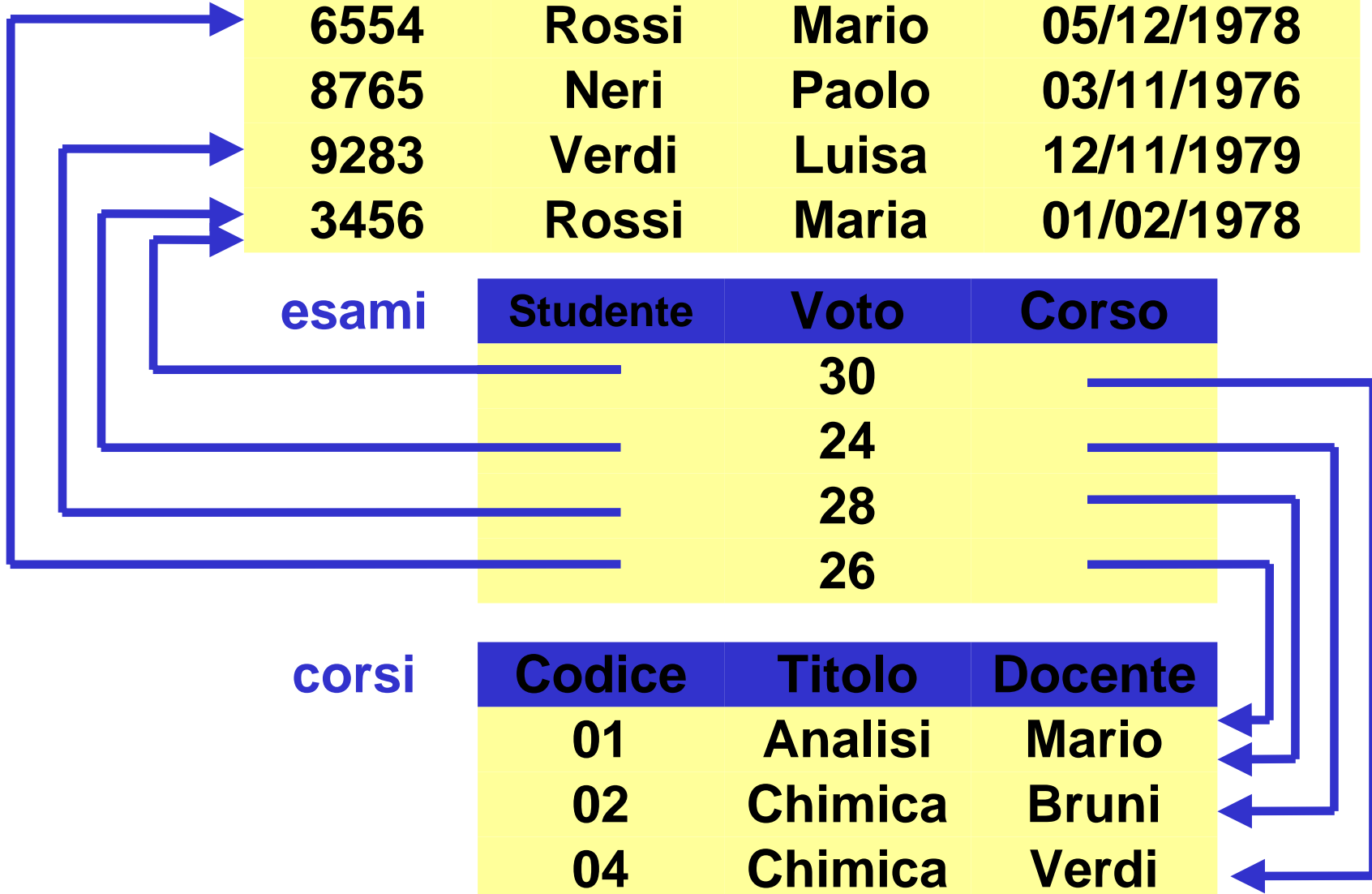
Matricola	Cognome	Nome	Data di nascita
6554	Rossi	Mario	05/12/1978
8765	Neri	Paolo	03/11/1976
9283	Verdi	Luisa	12/11/1979
3456	Rossi	Maria	01/02/1978

esami

Studente	Voto	Corso
	30	
	24	
	28	
	26	

corsi

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





Basi di dati e Relazioni

Il modello relazionale è *basato su valori*.

I riferimenti fra dati in relazioni diverse avvengono attraverso la corrispondenza dei valori con cui nelle tuple che sono logicamente collegate si instanziano domini corrispondenti.

Gli altri modelli (gerarchico, reticolare) utilizzano puntatori per le corrispondenze e sono *basati su record e puntatori*.



Basi di dati e Relazioni

Vantaggi dell'approccio basato su valori

- Si inseriscono nella base di dati solo valori significativi per l'applicazione (i puntatori sono dati aggiuntivi relativi alla sola implementazione).
- Il trasferimento dei dati da un ambiente ad un altro è più semplice (i puntatori hanno validità solo locale)
- la rappresentazione logica dei dati non fa riferimento a quella fisica e quindi si ottiene l'indipendenza dei dati

Basi di dati e Relazioni

- Uno **schema di relazione** $R(X)$ è costituito da un simbolo (*nome della relazione*) R e da una serie di *attributi* $X = \{A_1, A_2, \dots, A_n\}$

Corsi (Codice, NomeCorso, Docente)

- Una **relazione** su uno schema $R(X)$ è un insieme r di tuple su X .
- Uno **schema di base di dati** è un insieme di schemi di relazione con nomi diversi

$\mathbf{R} = \{R_1(X_1), R_2(X_2), \dots, R_n(X_n)\}$

- Una **base di dati** su uno schema $\mathbf{R} = \{R_1(X_1), R_2(X_2), \dots, R_n(X_n)\}$ è un insieme di relazioni $\mathbf{r} = \{r_1, r_2, \dots, r_n\}$ dove ogni r_i è una relazione sullo schema $R_i(X_i)$

Informazione incompleta

Le tuple che compongono la base di dati devono essere omogenee. Quindi ad ogni attributo deve essere associato un valore in ogni tupla. Non sempre questo è possibile.

Es. **Persone**(Cognome, Nome, Indirizzo, Telefono)

Potrebbe esistere una persona che non ha telefono, o di cui non conosciamo l'indirizzo.

Oppure:

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



Informazione incompleta: soluzioni?

- non conviene (anche se è un espediente di uso comune) usare valori del dominio normalmente non utilizzati (0, stringa nulla, "99", ...), come spesso accade nella programmazione:
 - potrebbero non esistere valori "non utilizzati"
 - valori "non utilizzati" potrebbero diventare significativi
 - in fase di utilizzo (nei programmi) sarebbe necessario ogni volta tener conto del "significato" di questi valori



Informazione incompleta

Nel modello relazionale è definito un valore convenzionale, detto *valore nullo*, che indica la non disponibilità dell'informazione.

Non è un valore del dominio, ma è **un valore aggiuntivo rispetto a quelli del dominio**

Il valore nullo può rappresentare 3 tipi di informazione:

- sconosciuta
- inesistente
- indeterminata (nei DBMS disponibili sul mercato si considera in genere questo caso)

Tipi di valore nullo

- ✦ **valore sconosciuto** (Firenze certamente ha una Prefettura. Se non compare nella Relazione significa che al momento non ci è nota)
- ✦ **valore inesistente** (Tivoli non è capoluogo di provincia e quindi non ha Prefettura. In questo caso il valore non esisterà mai)
- ✦ **valore senza informazione** (Prato è stata da poco istituita come Provincia per cui non si sa se già esiste una Prefettura. Siamo in uno dei due casi precedenti, ma non sappiamo quale)

Città	Indirizzo Prefettura
Roma	Via Quattro Novembre
Firenze	NULL
Tivoli	NULL
Prato	NULL



Vincoli di Integrità

Non tutte le combinazioni possibili di valori dei domini su cui è definita una relazione sono accettabili.

Esistono istanze di basi di dati che, pur sintatticamente corrette ...

... non rappresentano informazioni possibili per l'applicazione di interesse



Vincoli di Integrità

- Alcuni attributi possono assumere valori in un certo intervallo
- Alcuni attributi **devono** essere diversi in ogni tupla della stessa relazione

Es. valori dell'attributo Matricola in una relazione del tipo

Studenti(Matricola, Cognome, Nome,
DataNascita)

Vincoli di Integrità

- Alcuni valori possono essere incompatibili con altri all'interno della stessa relazione

Es. data la relazione

Esami(Matricola, Voto, Lode, CodCorso)

1. una stessa coppia Matricola, Corso può apparire una sola volta
2. Il valore Vero per l'attributo Lode è corretto solo se Voto=30

- Alcuni valori possono essere incompatibili con i valori di un'altra relazione

Es. data la relazione **Esami** e la relazione

Corsi(CodCorso, Titolo, Docente)

ogni valore di CodCorso in **Esami** deve essere un valore esistente di CodCorso in **Corsi**



Vincoli di Integrità

Sono condizioni, espresse come *predicati logici*, che sono inserite nella base di dati per garantirne la consistenza.

Ogni istanza della base di dati *deve soddisfare* i vincoli di integrità (il predicato corrispondente al vincolo deve assumere valore *vero* in ogni istante).

Una istanza che soddisfi tutti i vincoli è detta *corretta* (o *lecita* o *ammissibile*)



Vincoli di Integrità

Possono essere di vario tipo:

- Intrarelazionale se coinvolge attributi della stessa relazione
 - Vincoli di tupla
possono essere valutati su ciascuna tupla indipendentemente dalle altre
 - Vincoli di dominio
sono definiti su singoli valori
- Interrelazionale se coinvolge più relazioni

Vincoli di tupla

- ✦ I **vincoli di tupla** sono vincoli che vengono valutati su ciascuna tupla, indipendentemente dalle altre
- ✦ Come caso particolare dei vincoli di tupla ci sono i **vincoli su valori (o di dominio)** definiti con riferimento a singoli valori; ciò corrisponde a limitazioni sul dominio dell'attributo (il primo caso dell'esempio precedente)

Vincoli di tupla

Possono essere definiti attraverso operatori booleani

Es.

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

Questo vincolo esprime il fatto che un voto deve essere compreso tra 18 e 30 (in effetti siamo in presenza di un vincolo di dominio)

$(\text{NOT } (\text{lode} = \text{Vero})) \text{ OR } (\text{Voto} = 30)$

Questo vincolo esprime il fatto che "O non c'è la lode O il voto è 30", cioè che la lode è ammessa solo se il voto è pari a 30.

Vincoli di tupla

- ✦ La definizione data ammette anche espressioni più complesse, purché definite sui valori delle singole tuple.

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

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

Identificazione delle tuple

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/78
78763	Rossi	Mario	Ing Inf	3/11/76
65432	Neri	Piero	Ing Mecc	10/7/79
87654	Neri	Mario	Ing Inf	3/11/76
67653	Rossi	Piero	Ing Mecc	5/12/78

- non ci sono due tuple con lo stesso valore sull'attributo Matricola
- non ci sono due tuple uguali su tutti e tre gli attributi Cognome, Nome e Nascita



Chiavi

Chiave = insieme di attributi che identificano univocamente le tuple di una relazione

Una *chiave* è un insieme *minimale* di attributi utilizzato per identificare univocamente le tuple di una relazione.

Formalmente:

Un insieme di attributi K è *superchiave* per una relazione r se r non contiene due tuple t_1 e t_2 tali che

$$t_1[K] = t_2[K]$$

Un insieme di attributi K è *chiave* per r se è superchiave minimale, cioè se non esiste un'altra superchiave K' che sia sottoinsieme di K o, comunque, di dimensioni inferiori.

Una chiave

Matricola	Cognome	Nome	Corso	Nascita
27655	Rossi	Mario	Ing Inf	5/12/78
78763	Rossi	Mario	Ing Inf	3/11/76
65432	Neri	Piero	Ing Mecc	10/7/79
87654	Neri	Mario	Ing Inf	3/11/76
67653	Rossi	Piero	Ing Mecc	5/12/78

- Matricola è una chiave:
 - è superchiave
 - contiene un solo attributo e quindi è minimale



Chiavi

- Una chiave è tale se soddisfa la definizione per tutte le possibili tuple appartenenti alla relazione, e non solo per quelle che effettivamente appaiono come istanze della relazione stessa.
- Quindi la chiave è legata allo schema della relazione e non ai valori effettivamente assunti dalle istanze dello schema.
- Ogni relazione, per definizione, possiede una chiave.

Infatti, poiché una relazione non ammette due tuple uguali, l'intero insieme X su cui la relazione è definita è sicuramente superchiave per essa.

Esempi

✦ **Cognome, Nome, Nascita** è un'altra chiave, in quanto

- è superchiave
- E' minimale (Da soli Cognome, Nome o Nascita non sono sufficienti, come non sono sufficienti Cognome+Nome, Cognome+Nascita o Nome+Nascita)

✦ **Matricola, Corso**

- è superchiave (visto che da sola lo è Matricola)
- Non è minimale perchè un suo sottoinsieme proprio (Matricola) è esso stesso una superchiave.

Per cui Matricola+Corso è superchiave ma non chiave.



Esempi

- ✦ Non ci sono tuple uguali su Cognome e Corso:
 - Per questa ISTANZA della relazione, Cognome e Corso formerebbero una superchiave minimale, cioè una chiave.
- ✦ Ma è sempre vero? NO, in quanto possono benissimo esistere studenti con lo stesso cognome iscritti allo stesso corso di studio!



Importanza delle chiavi

- ✦ Una relazione, in quanto insieme matematico, per definizione non può contenere ennuple uguali
- ✦ Ogni relazione ha almeno una superchiave: l'insieme di tutti gli attributi su cui è definita
- ✦ e quindi ha (almeno) una chiave



Chiavi

La presenza di valori nulli in una chiave può vanificare la proprietà di unicità delle tuple che identifica.

Si impone quindi che almeno una chiave non contenga valori nulli. Tale chiave è detta *chiave primaria*.

Di solito la chiave primaria compare sottolineata nello schema di una relazione.

Es.

Studenti(Matricola, Cognome, Nome, Nascita, Corso)



Vincoli di Integrità Referenziale

In alcuni casi (corrispondenze fra relazioni) è necessario che i valori degli attributi di una relazione R_1 si trovino anche in attributi corrispondenti di un'altra relazione R_2 .

Un *vincolo di integrità referenziale* (o *foreign key*) fra un insieme di attributi X di R_1 e un'altra relazione R_2 è soddisfatto se i valori su X di ciascuna tupla di R_1 compaiono come valori della chiave (primaria) di R_2 .

Non tutti i DBMS consentono di definire una chiave come primaria, sia quando questa è unica, sia in presenza di più chiavi possibili.

Quindi in questi casi sarà necessario (ed è comunque sempre consigliabile farlo) esprimere il vincolo di integrità referenziale per esteso, specificando esplicitamente le corrispondenze fra gli attributi dell'una e dell'altra relazione collegate dal vincolo stesso.