



# Il modello relazionale

---

Sistemi Informativi L-A

Home Page del corso:

<http://www-db.deis.unibo.it/courses/SIL-A/>

Versione elettronica: [Relazionale.pdf](#)



# Relazionale, Gerarchico e Reticolare

- Il modello relazionale è stato introdotto nel 1970 da E.F. Codd (un ricercatore dell'IBM di San Jose, CA) allo scopo di favorire l'indipendenza dei dati
- I modelli preesistenti (gerarchico e reticolare) erano fortemente influenzati da considerazioni di natura fisica, che enfatizzavano quindi aspetti di efficienza rispetto a quelli di semplicità d'uso
- La principale differenza tra relazionale e gerarchico o reticolare sta nel modo con cui si rappresentano i “legami” (associazioni) tra diverse strutture:
  - Gerarchico e reticolare usano puntatori
  - Nel modello relazionale si fa solo uso di valori
- Un'altra differenza importante è che, a differenza del gerarchico e del reticolare, il modello relazionale è formalmente definito
  - Sviluppo di una teoria relazionale utile per la progettazione di DB, per la definizione di linguaggi e per l'ottimizzazione delle richieste



# Un po' di storia...

---

**Anni '70:** definizione del modello, prima versione del linguaggio SQL (allora SEQUEL), studi fondamentali sulla tecnologia relazionale (ottimizzazione, transazioni, recovery, ...) e primi prototipi di DBMS relazionali (RDBMS):

- System R (IBM, laboratorio di ricerca di San Jose, CA, USA)
- Ingres (Università di Berkeley, CA, USA)

**Anni '80:** prima standardizzazione di SQL, primi prototipi commerciali:

- SQL/DS (derivato da System R)
- Oracle
- IBM DB2

**Anni '90:** standard ISO-ANSI SQL-2 (quello attualmente di riferimento, anche noto come SQL-92)

- Esiste già lo standard SQL:1999 (o SQL-3), ma non è ancora completamente recepito dai costruttori
- ...e sono allo studio diverse altre estensioni del linguaggio



# Sul termine “Relazione”

---

- Il termine “relazione” può essere usato con diverse accezioni, che non vanno confuse tra loro:
  - Nel **linguaggio comune** indica un “**legame**” di qualche tipo
  - Nella **teoria degli insiemi** indica una “**relazione matematica**”
  - Nel **modello relazionale** è una **generalizzazione della relazione matematica**
- e inoltre:
  - Nel **modello Entity-Relationship** (trattato in SI L-B) indica una **classe di legami tra due o più entità** (si parla anche di “**associazione**”)
  - Nei **DBMS** è usato come sinonimo di “**tabella**”
- Per introdurre il modello relazionale è quindi opportuno innanzitutto rivedere il concetto di...

# Relazione matematica

- Si considerino  $n$  insiemi  $D_1, D_2, \dots, D_n$ , non necessariamente distinti
- Il prodotto Cartesiano  $D_1 \times D_2 \times \dots \times D_n$  è l'insieme di tutte le  $n$ -ple ordinate  $(d_1, d_2, \dots, d_n)$  tali che  $d_1 \in D_1, d_2 \in D_2, \dots, d_n \in D_n$
- Una relazione (matematica) su  $D_1, D_2, \dots, D_n$ , è un qualunque sottoinsieme del prodotto Cartesiano  $D_1 \times D_2 \times \dots \times D_n$

Esempio:

$D_1 = \{a, b, c\}, D_2 = \{1, 2\}; D_1 \times D_2 = \{(a, 1), (a, 2), (b, 1), (b, 2), (c, 1), (c, 2)\}$   
 $r = \{(a, 1), (b, 2), (c, 1), (c, 2)\}$  è una relazione su  $D_1$  e  $D_2$  ( $r \subseteq D_1 \times D_2$ )

- $D_1, D_2, \dots, D_n$  sono i domini della relazione
- Il valore di  $n$  è detto grado (o “arità”) della relazione
- Il numero di  $n$ -ple di una relazione è la sua cardinalità

# Relazione matematica: proprietà

- Una relazione è un insieme di n-ple....:

- Tutte le n-ple sono distinte tra loro
- Non è definito alcun ordinamento tra le n-ple

$$\{(a,1),(a,1),(b,2),(c,1),(c,2)\} = \{(a,1),(b,2),(c,1),(c,2)\} = \{(b,2),(c,2),(c,1),(a,1)\}$$

- ... ordinate...:

- L'ordine in cui si considerano i domini è rilevante ( $D_1 \times D_2 \neq D_2 \times D_1$ )

$$\{(a,1),(c,1),(c,2)\} \neq \{(1,a),(1,c),(2,c)\}$$

- ...su domini non necessariamente distinti:

- Uno stesso dominio può essere usato in più posizioni

$$\{(2,a,1),(1,c,1),(1,c,2)\} \subseteq D_2 \times D_1 \times D_2$$

# Rappresentazione di relazioni

- La notazione insiemistica è adeguata solo per relazioni con poche n-ple
- Molto più efficace è la rappresentazione tabellare....:

a	1
b	2
c	1
c	2

- ... o anche multi-dimensionale, se il grado è  $\leq 3$ :

2	0	1	1
1	1	0	1
	a	b	c

# L'importanza della posizione

- Nel caso di domini ripetuti, l'interpretazione dei dati si complica e la **posizione** assume un ruolo determinante

$\text{partite} \subseteq \text{String} \times \text{String} \times \text{Integer} \times \text{Integer}$

Benetton TV	Poliform Cantù	100	71
Kinder BO	MontePaschi SI	90	51
Paf BO	Adr RM	62	97
Adr RM	Kinder BO	80	62

- Il primo e il terzo dominio si riferiscono alla squadra ospitante (nome e numero di punti), mentre il secondo e il quarto dominio si riferiscono alla squadra ospitata

**È SCOMODO E POCO CHIARO!!**



# Relazione nel modello relazionale

- Ad ogni occorrenza di dominio (ripetuto o meno) si associa un nome univoco nella relazione, detto **attributo**, il cui compito è specificare il **ruolo** che quel dominio svolge nella relazione (“cosa significa”)
- Nella **rappresentazione tabellare**, gli attributi sono le **intestazioni delle colonne** (e in quella **multi-dimensionale** sono i **nomi degli assi**)

TeamCasa	TeamOspite	PuntiCasa	PuntiOspite
Benetton TV	Poliform Cantù	100	71
Kinder BO	MontePaschi SI	90	51
Paf BO	Adr RM	62	97
Adr RM	Kinder BO	80	62

- La struttura non è più posizionale, ovvero l'ordine degli attributi non ha più rilevanza!

# Relazione: una definizione formale

- Si indichi con  $\text{dom}(A)$  il dominio dell'attributo  $A$  e si consideri un insieme di attributi  $X = \{A_1, A_2, \dots, A_n\}$
- Una **tupla**  $t$  su  $X$  è una **funzione** che associa ad ogni  $A_i \in X$  un valore di  $\text{dom}(A_i)$
- (L'istanza di) una **relazione** su  $X$  è un **insieme di tuple** su  $X$
- Lo **schema** di una **relazione** su  $X$  è dato da un **nome** (della relazione)  $R$  e dall'**insieme di attributi**  $X$ , scritto  $R(X)$

Partite

TeamCasa	TeamOspite	PuntiCasa	PuntiOspite
Benetton TV	Poliform Cantù	100	71
Kinder BO	MontePaschi SI	90	51
Paf BO	Adr RM	62	97
Adr RM	Kinder BO	80	62

# Notazione di base (1)

- Per denotare insiemi di attributi si usa la notazione semplificata:
  - A in luogo di {A} e XY in luogo di  $X \cup Y$
  - ...e si scrive ABC (o A,B,C) anziché {A,B,C}
  - ...e quindi R(ABC) o R(A,B,C) anziché R({A, B, C})
- Se t è una tupla su X e  $A \in X$ , allora t[A] o t.A è il valore di t su A

Partite

t	TeamCasa	TeamOspite	PuntiCasa	PuntiOspite
	Benetton TV	Poliform Cantù	100	71
	Kinder BO	MontePaschi SI	90	51
	Paf BO	Adr RM	62	97
	Adr RM	Kinder BO	80	62

t[TeamOspite] = t.TeamOspite = 'MontePaschi SI'

- La stessa notazione si usa per insiemi di attributi, e denota una tupla
  - t[TeamOspite,PuntiOspite] è una tupla su {TeamOspite,PuntiOspite}

## Notazione di base (2)

- Se necessario, per riferirsi all'istanza della relazione con schema  $R(X)$  si usa  $r$  (il nome in minuscolo della relazione)

Partite(TeamCasa,TeamOspite,PuntiCasa,PuntiOspite)

partite =

Benetton TV	Poliform Cantù	100	71
Kinder BO	MontePaschi SI	90	51
Paf BO	Adr RM	62	97
Adr RM	Kinder BO	80	62



# Data Base relazionale

- Lo schema di un DB relazionale è un insieme di schemi di relazioni con nomi distinti

$$\mathbf{R} = \{R_1(X_1), R_2(X_2), \dots, R_m(X_m)\} \quad (R_i \neq R_j \quad \forall i \neq j)$$

- (L'istanza di) un DB con schema  $\mathbf{R} = \{R_1(X_1), R_2(X_2), \dots, R_m(X_m)\}$  è un insieme di (istanze di) relazioni

$$\mathbf{r} = \{r_1, r_2, \dots, r_m\}$$

con  $r_i$  istanza su  $R_i(X_i)$

# Un semplice DB relazionale

## Studenti

Matricola	Cognome	Nome	DataNascita
29323	Bianchi	Giorgio	21/06/1978
35467	Rossi	Anna	13/04/1978
39654	Verdi	Marco	20/09/1979
42132	Neri	Lucia	15/02/1978

## Corsi

CodCorso	Titolo	Docente	Anno
483	Analisi	Biondi	1
729	Analisi	Neri	1
913	Sistemi Informativi	Castani	2

## Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	26	NO
35467	913	30	NO



# 1NF, ovvero solo domini semplici

---

- Il modello relazionale non permette di usare domini arbitrari per la definizione delle relazioni; in particolare **non è in generale possibile usare domini strutturati** (array, set, liste, ...)
  - Vi sono delle **eccezioni** notevoli (ad es. le **date** e le **stringhe**)
- Concisamente, **una relazione in cui ogni dominio è “atomico”** (non ulteriormente decomponibile) si dice che è in  
**Prima Forma Normale**, o **1NF** (1st Normal Form)
- In molti casi è pertanto richiesta un’attività di **normalizzazione dei dati** che dia luogo a relazioni in 1NF e che preservi l’informazione originale

# Normalizzazione di strutture nidificate

Ricevuta n. 231 del 12/02/2002		
Coperti	2	3,00
Antipasti	1	5,80
Primi	2	11,45
Secondi	2	22,30
Caffè	2	2,20
Vino	1	8,00
Totale (Euro)		52,75

Ricevuta n. 352 del 13/02/2002		
Coperti	1	1,50

Ricevute

Numero	Data	Totale
231	12/02/2002	52,75
352	13/02/2002	...
...	...	...

Dettaglio



Numero	Quantità	Descrizione	Prezzo
231	2	Coperti	3,00
231	1	Antipasti	5,80
231	2	Primi	11,45
231	2	Secondi	22,30
231	2	Caffè	2,20
231	1	Vino	8,00
352	1	Coperti	1,50





# Considerazioni sulla normalizzazione

---

- Il fatto che una rappresentazione normalizzata sia adeguata o meno dipende (molto) dal contesto
  - Ad es.: l'ordine delle righe nella ricevuta è rilevante o meno?
- Lo stesso dicasi per eventuali ridondanze che si possono venire ad osservare
  - Ad es.: il coperto ed il caffè hanno un prezzo che non varia da ricevuta a ricevuta?
- In generale è bene ricordare che ogni caso presenta una sua specificità, e quindi non va trattato “automaticamente”
- Normalizzare in 1NF è, a tutti gli effetti, un'attività di progettazione (logica), e in quanto tale può essere solo oggetto di “regole guida” che però non hanno validità assoluta

# Informazione incompleta

- Le informazioni che si vogliono rappresentare mediante relazioni non sempre corrispondono pienamente allo schema prescelto, in particolare per alcune tuple e alcuni attributi potrebbe non essere possibile specificare, per diversi motivi, un valore del dominio

Nome	DataMorte
Mario Rossi	29/11/1954
Paolo Bianchi	
Maria Verdi	
Giuseppe Neri	

Paolo Bianchi è ancora vivo (valore **non applicabile**)

Maria Verdi è morta, ma non conosciamo in che data (**applicabile ma ignoto**)

Giuseppe Neri non si sa se è morto o meno (**ignota l'applicabilità**)

# Cosa si fa nel modello relazionale?

- In diversi casi, in mancanza di informazione, si tende a usare un “valore speciale” del dominio (0, “”, “-1”, ecc.) che non si utilizza per altri scopi
- Questa pratica è fortemente sconsigliata, in quanto, anche dove possibile:
  - Valori inutilizzati potrebbero successivamente diventare significativi
  - Le applicazioni dovrebbero sapere “cosa significa in realtà” il valore usato allo scopo

Esempio (reale!): nel 1998, analizzando i clienti di un’assicurazione, si scoprì una strana concentrazione di ultra-novantenni... tutte le date di nascita ignote erano state codificate con “01/01/00”!!

- Nel modello relazionale si opera in maniera pragmatica: si adotta il concetto di valore nullo (NULL), che denota assenza di un valore nel dominio (e non è un valore del dominio)
- ...pertanto  $t[A] \in \text{dom}(A) \cup \{\text{NULL}\}$

# Valori nulli: considerazioni

Nome	DataMorte
Mario Rossi	29/11/1954
Paolo Bianchi	NULL
Maria Verdi	NULL
Giuseppe Neri	NULL

- La presenza di un valore nullo non fornisce alcuna informazione sull'applicabilità o meno
- È importante ricordare che NULL non è un valore del dominio; in particolare, se due tuple hanno entrambe valore NULL per un attributo, non si può inferire che esse abbiano lo stesso valore per quell'attributo, ovvero:

NULL ≠ NULL

# Valori nulli: restrizioni

- La presenza di valori nulli non può essere sempre tollerata, ovvero è necessario imporre delle **restrizioni** al loro uso

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
NULL	729	30	Sì
29323	913	NULL	NO
35467	913	30	NO

- Il valore nullo per Matricola non permette di sapere chi ha sostenuto l'esame
- Il valore nullo per Voto non è ammissibile nel contesto considerato



Istanze di questo tipo non sono accettabili!

# Vincoli di integrità

- La “correttezza sintattica” di un’istanza non è condizione sufficiente affinché i dati rappresentino informazione possibile nel contesto reale considerato

Studenti

Matricola	Cognome	Nome	DataNascita
35467	Bianchi	Giorgio	21/06/1978
35467	Rossi	Anna	13/04/1978
39654	Rossi	Anna	13/04/1978

- La prima e la seconda tupla hanno la stessa Matricola (!?)
  - La seconda e la terza tupla hanno gli stessi valori per Nome, Cognome e DataNascita, ma questo in linea di principio è possibile
- Un **vincolo di integrità** è una **proprietà che deve essere soddisfatta dalle istanze**; ogni vincolo può quindi essere descritto da una funzione booleana che associa a ogni istanza il valore VERO o FALSO

# Vincoli di dominio

- Un vincolo che si riferisce ai valori ammissibili per un singolo attributo viene detto **vincolo di dominio** (o **sui valori**)

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	31	NO
35467	913	30	FORSE

- Il Voto deve essere compreso tra 18 e 30  
( $\text{Voto} \geq 18$ ) AND ( $\text{Voto} \leq 30$ )
- La Lode può solo assumere i valori `Sì` o `NO`  
( $\text{Lode} = \text{'Sì'}$ ) OR ( $\text{Lode} = \text{'NO'}$ )

# Vincoli di tupla

- I vincoli di dominio sono un caso particolare dei **vincoli di tupla**, ovvero vincoli che esprimono condizioni su ciascuna tupla, indipendentemente dalle altre

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	26	Sì
35467	913	30	NO

- La Lode si può assegnare solo se il Voto è 30:  
(Voto = 30) OR NOT(Lode = 'Sì')
- Nello schema **Pagamenti(Data,ImportoLordo,Ritenute,Netto)** si ha:  
 $\text{ImportoLordo} = \text{Netto} + \text{Ritenute}$



# Vincoli di chiave: intuizione

- Un tipo importantissimo di vincoli sono i **vincoli di chiave**, che **vietano la presenza di tuple distinte che hanno lo stesso valore su uno o più attributi**

Studenti	Matricola	CodiceFiscale	Cognome	Nome	DataNascita
	29323	BNCGRG78F21A	Bianchi	Giorgio	21/06/1978
	35467	RSSNNA78D13A	Rossi	Anna	13/04/1978
	39654	VRDMRC79I20A	Verdi	Marco	20/09/1979
	42132	VRDMRC79I20B	Verdi	Marco	20/09/1979

- Il valore di Matricola **identifica univocamente** uno studente
- Lo stesso vale per CodiceFiscale
- ...e per ogni insieme di attributi che includa Matricola o CodiceFiscale
  - {Matricola,Cognome}, {CodiceFiscale,Nome}, ...
- Viceversa, possono esistere due tuple uguali su {Cognome,Nome,DataNascita}



# Chiavi e superchiavi

---

- Dato uno schema  $R(X)$ , un insieme di attributi  $K \subseteq X$  è
  - una **superchiave** se e solo se  
in ogni istanza ammissibile  $r$  di  $R(X)$  non esistono due tuple distinte  $t_1$  e  $t_2$  tali che  $t_1[K] = t_2[K]$
  - una **chiave** se e solo se  
è una superchiave minimale, ovvero  
non esiste  $K' \subset K$  con  $K'$  superchiave
- Una chiave è pertanto un identificatore minimale per ogni  $r$  su  $R(X)$
- Nella relazione Studenti:
  - $\{\text{Matricola}\}$  e  $\{\text{CodiceFiscale}\}$  sono due chiavi
  - $\{\text{Matricola}, \text{Cognome}\}$  e  $\{\text{CodiceFiscale}, \text{Nome}\}$  sono solo superchiavi
  - $\{\text{Cognome}, \text{Nome}, \text{DataNascita}\}$  non è superchiave

# Esistenza di chiavi e superchiavi

- Poiché ogni istanza  $r$  su  $R(X)$  è un insieme, ne segue che l'insieme  $X$  di tutti gli attributi dello schema è senz'altro una superchiave per  $R(X)$
- Poiché il numero di attributi,  $n$ , è finito, è sempre possibile arrivare ad individuare (almeno) una chiave  $K \subseteq X$

```
{ K := X;  
  For i = 1 to n do  
    { If  $K - \{A_i\}$  è superchiave then {K := K - {Ai};}}}
```



Si noti che in casi (molto) particolari il numero di chiavi può essere esponenziale in  $n$  (quando?)

# Dai vincoli alle istanze, non viceversa!

- I vincoli di chiave si esprimono a livello di schema, sulla base di un'analisi della realtà che si vuole modellare mediante relazioni, e limitano l'insieme di istanze legali (o “ammissibili”, “corrette” “valide”, ecc.)
- Una specifica istanza può soddisfare altri vincoli (di chiave), ma ciò non autorizza a generalizzare

Esami

Matricola	CodCorso	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	26	NO
35467	913	30	NO

- La (sola) chiave è {Matricola,CodiceCorso}
- L'istanza soddisfa anche altri vincoli, ad es. {Matricola,Voto} è un identificatore, ma ciò è puramente casuale

# Importanza delle chiavi

- L'esistenza delle chiavi garantisce l'**accessibilità a ciascun dato del DB**, in quanto ogni singolo valore è univocamente individuato da:
  - nome della relazione      ➡      individua una relazione del DB
  - valore della chiave      ➡      individua una tupla della relazione
  - nome dell'attributo      ➡      individua il valore desiderato
- Le chiavi sono lo strumento principale attraverso il quale vengono **correlati i dati in relazioni diverse** ("il modello relazionale è basato su valori")

# Chiavi e valori nulli

- In presenza di valori nulli entrambe le due funzioni svolte dalle chiavi (identificazione e correlazione) possono venire a mancare

Studenti	Matricola	CodiceFiscale	Cognome	Nome	DataNascita
	NULL	NULL	Bianchi	Giorgio	21/06/1978
	35467	RSSNNA78D13A	Rossi	Anna	13/04/1978
	NULL	VRDMRC79I20A	Verdi	Marco	20/09/1979
	42132	NULL	Verdi	Marco	20/09/1979

- La **prima tupla** non è identificabile in alcun modo, pertanto:  
È necessario specificare il valore di almeno una chiave!
- La **terza e quarta tupla** non sappiamo se si riferiscano o meno allo stesso studente, pertanto:  
Non è sufficiente specificare il valore di una chiave!

# Chiave primaria

- Per evitare i problemi visti è necessario scegliere una chiave, detta **chiave primaria**, su cui non si ammettono valori nulli
- Convenzionalmente, gli attributi della chiave primaria vengono sottolineati

Studenti	<u>Matricola</u>	CodiceFiscale	Cognome	Nome	DataNascita
	29323	NULL	Bianchi	Giorgio	21/06/1978
	35467	RSSNNA78D13A	Rossi	Anna	13/04/1978
	39654	VRDMRC79I20A	Verdi	Marco	20/09/1979
	42132	NULL	Verdi	Marco	20/09/1979



Nei casi in cui per nessuna chiave si possa garantire la disponibilità di valori, è necessario introdurre un nuovo attributo (un “**codice**”) che svolga le funzioni di chiave primaria

# Vincoli di integrità referenziale

- I vincoli sinora visti sono tutti di tipo **intra-relazionale**, in quanto interessano una relazione alla volta
- Viceversa, i **vincoli di integrità referenziale** sono importanti tipi di vincoli **inter-relazionali**, che enfatizzano come le correlazioni tra le tuple siano spesso ottenute usando i valori delle chiavi
- Si considerino due schemi  $R_1(X_1)$  e  $R_2(X_2)$  di un DB  $R$ , e sia  $Y$  un insieme di attributi in  $X_2$
- Un **vincolo di integrità referenziale su  $Y$**  impone che in ogni istanza  $r = \{r_1, r_2, \dots\}$  del DB  
l'insieme dei valori di  $Y$  in  $r_2$  sia un sottoinsieme dell'insieme dei valori della chiave primaria di  $R_1(X_1)$  presenti nell'istanza  $r_1$
- L'insieme  $Y$  viene detto una **foreign key** (o “**chiave importata**”)



# Esempi di foreign key

Studenti

<u>Matricola</u>	Cognome	Nome	DataNascita
29323	Bianchi	Giorgio	21/06/1978
35467	Rossi	Anna	13/04/1978
39654	Verdi	Marco	20/09/1979
42132	Neri	Lucia	15/02/1978

Corsi

<u>CodCorso</u>	Titolo	Docente	Anno
483	Analisi	Biondi	1
729	Analisi	Neri	1
913	Sistemi Informativi	Castani	2

Esami

<u>Matricola</u>	<u>CodCorso</u>	Voto	Lode
29323	483	28	NO
39654	729	30	Sì
29323	913	26	NO
35467	913	30	NO

In Esami, {Matricola} è una foreign key, così come {CodCorso}

# Foreign key: alcune precisazioni (1)

- In generale la foreign key  $Y$  e la primary key  $K$  di  $R_1(X_1)$  possono includere attributi con nomi diversi

Corsi

<u>Codice</u>		Titolo	Docente	Anno
483		Analisi	Biondi	1
729		Analisi	Neri	1

Esami

<u>NumMatricola</u>	<u>CodCorso</u>	Voto	Lode
29323	483	28	NO

- Foreign key e primary key possono far parte della stessa relazione, ovviamente con  $Y \neq K$


Personale

<u>Codice</u>	Nome	...	CodResponsabile
123	Mario Rossi	...	325
134	Gino Verdi	...	325
325	Anna Neri	...	...

## Foreign key: alcune precisazioni (2)

- In presenza di **valori nulli**, i vincoli di integrità referenziale si possono parzialmente rilassare


Personale



<u>Codice</u>	Nome	...	CodResponsabile
123	Mario Rossi	...	325
134	Gino Verdi	...	325
325	Anna Neri	...	NULL

- Nei DBMS un vincolo di integrità referenziale può anche esprimersi con riferimento ad una generica chiave (quindi anche non primaria)

Studenti



<u>Matricola</u>	CodiceFiscale	Cognome	Nome	DataNascita
29323	BNCGRG78F21A	Bianchi	Giorgio	21/06/1978
35467	RSSNNA78D13A	Rossi	Anna	13/04/1978

Redditi

<u>CF</u>	Imponibile
BNCGRG78F21A	10000



# Riassumiamo:

---

- Il **modello relazionale** è basato sul concetto di **relazione**, che estende quello di relazione matematica tra  $n$  domini associando a ciascuna occorrenza di dominio un nome, detto **attributo**
- Lo **schema** di una relazione consiste di un **nome** e di un **insieme di attributi**; l'**istanza** di una relazione è un **insieme di tuple**, ovvero funzioni che associano a ogni attributo dello schema un valore del corrispondente dominio
- In **assenza di informazioni** si fa uso di un particolare valore, detto **valore nullo (NULL)**, che non appartiene a nessun dominio
- Per garantire l'**integrità dei dati** si possono specificare diversi tipi di **vincoli**, che definiscono quali sono le **istanze legali** (ammissibili)
- I **vincoli intra-relazionali** includono quelli sui **domini**, sulle **tuple** e i vincoli **di chiave**; i **vincoli inter-relazionali** quelli di **integrità referenziale**. Questi ultimi permettono di stabilire le principali correlazioni tra i dati di diverse relazioni