



**UNIMORE**  
UNIVERSITÀ DEGLI STUDI DI  
MODENA E REGGIO EMILIA

Dipartimento di Scienze Fisiche,  
Informatiche e Matematiche

# Basi di Dati

*Corso di Laurea in Informatica*

E' vietata la copia e la riproduzione dei contenuti e immagini in qualsiasi forma. E' inoltre vietata la redistribuzione e la pubblicazione dei contenuti e immagini non autorizzata espressamente dall'autore o dall'Università di Modena e Reggio Emilia.



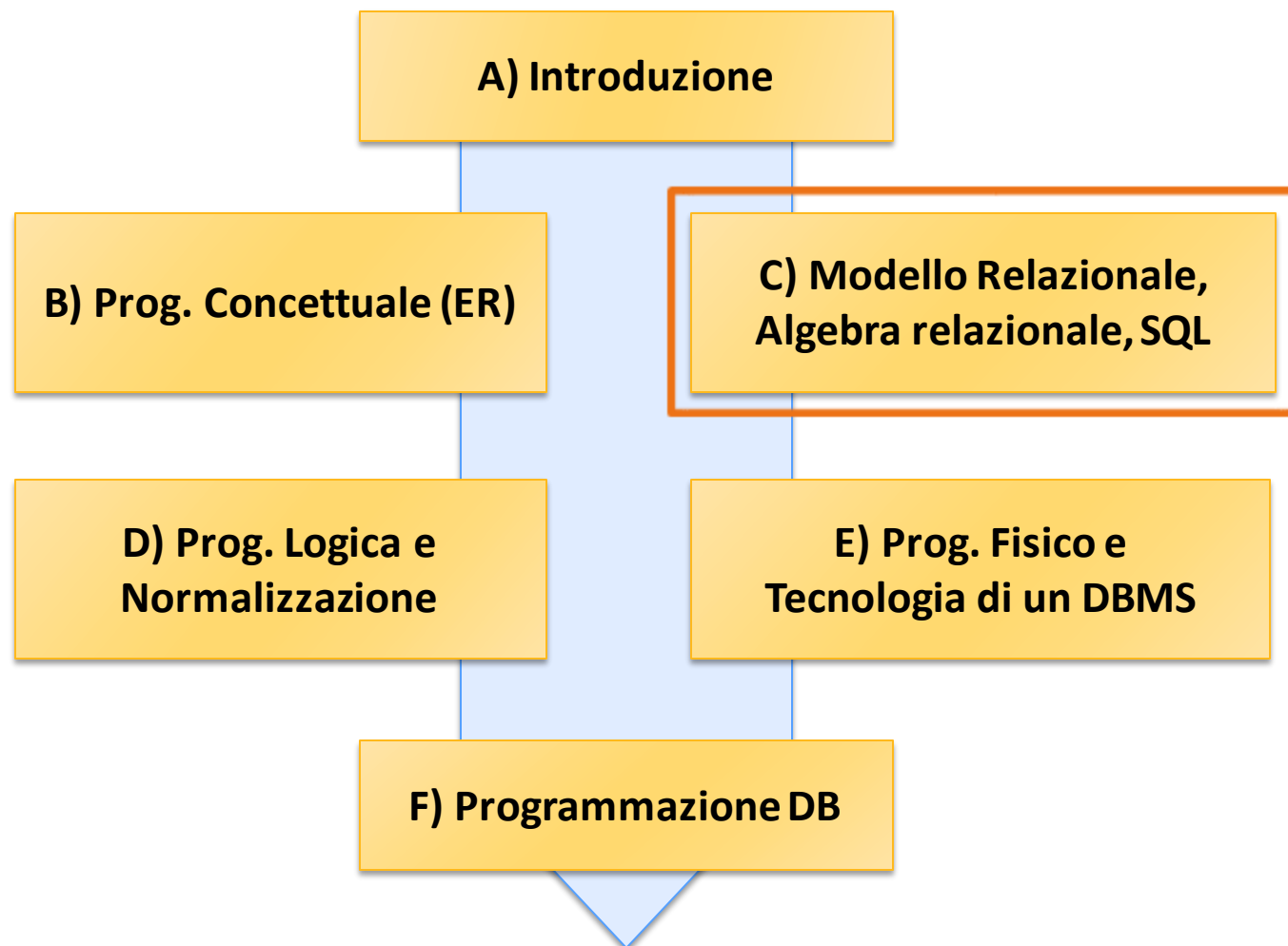
# Basi di Dati



Linguaggi di interrogazione

# Basi di Dati – Dove ci troviamo?

---



# Classificazione

---

- ▶ Linguaggi formali
  - ▶ Algebra relazionale
  - ▶ Calcolo relazionale
  - ▶ Programmazione logica
- ▶ Linguaggi programmatici
  - ▶ SQL: Structured Query Language
  - ▶ QBE: Query By Example

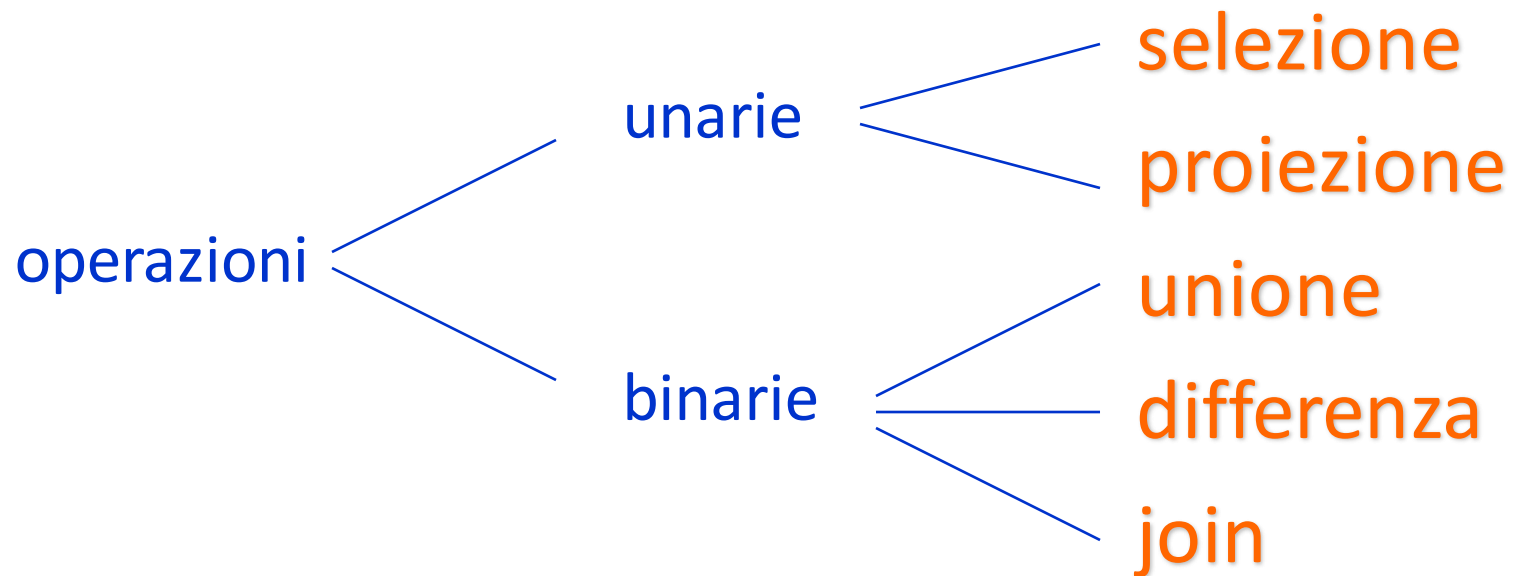
# Algebra relazionale

---

- ▶ Definita da Codd (1970)
- ▶ Molto utile per imparare a formulare query
- ▶ Insieme minimo di 5 operatori che danno l'intero potere espressivo del linguaggio

# Una visione d'insieme

---



# Esempio: gestione degli esami universitari

---

## studente

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf
415	Paola	Torino	Inf
702	Antonio	Roma	Log

## esame

MATR	COD-CORSO	DATA	VOTO
123	1	7-9-14	30
123	2	8-1-15	28
702	2	7-9-14	20

## corso

COD-CORSO	TITOLO	DOCENTE
1	matematica	Barozzi
2	informatica	Meo

# Selezione ( $\sigma$ )

---

## **SELECT [PredSel] TABELLA**

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ lo stesso schema di 'TABELLA'
- ▶ istanza :
  - ▶ le tuple di 'TABELLA' che soddisfano il predicato di selezione 'PredSel'



## Esempio di Selezione:

---

**SELECT [NOME='Paola'] STUDENTE**

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ lo stesso schema di STUDENTE
- ▶ istanza :
  - ▶ le tuple di STUDENTE che soddisfano il predicato di selezione [NOME='Paola']

MATR	NOME	CITTA'	C-DIP
415	Paola	Torino	Inf

# Sintassi del PREDICATO di SELEZIONE

---

Espressione booleana di predicati semplici

operazioni booleane :

- AND (P1 AND P2)
- OR (P1 OR P2)
- NOT (NOT P1)

predicati semplici :

- TRUE, FALSE
- termine comparatore

termine

comparatore :

- =, !=, <, <=, >, >=

termine :

- costante, attributo
- espressione aritmetica di costanti e attributi

## Esempio di selezione

---

SELECT

[(CITTA'='Torino') OR ((CITTA'='Roma')  
AND NOT (C-DIP='log'))] STUDENTE

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bolegna	Inf
415	Paola	Torino	Inf
702	Antonio	Roma	Log

# Proiezione ( $\pi$ )

---

## PROJECT [attributiProiez] TABELLA

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ gli attributi 'attributiProiez'
- ▶ istanza :
  - ▶ la restrizione delle tuple sugli attributi 'attributiProiez'

## Esempio di Proiezione:

---

### **PROJECT [NOME,C-DIP] STUDENTE**

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ gli attributi NOME e C-DIP
- ▶ istanza :
  - ▶ la restrizione delle tuple sugli attributi NOME e C-DIP

NOME	C-DIP
Carlo	Inf
Paola	Inf
Antonio	Log

# Proiezione e duplicati

---

- ▶ Nel modello formale la proiezione elimina i duplicati
- ▶ Nel modello informale (e nei sistemi) la eliminazione dei duplicati va richiesta esplicitamente

Es.

**PROJECT [C-DIP] STUDENTE**

C-DIP
Inf
Log

## Assegnamento (=)

---

- ▶ Serve per dare un nome al risultato di una espressione algebrica
- ▶ Non fa parte delle operazioni algebriche

**INFORMATICI** =

SELECT [C-DIP='Inf'] STUDENTI

**TORINESI** =

SELECT [CITTA'='Torino'] STUDENTI

# Unione (U)

---

## **TABELLA1 UNION TABELLA2**

- ▶ Si può fare se TABELLA1 e TABELLA2 sono compatibili (con domini ordinatamente dello stesso tipo)

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ lo schema di TABELLA1
- ▶ istanza :
  - ▶ l'unione delle tuple di TABELLA1 e TABELLA2

**L'unione è commutativa, associativa**



## Esempio di Unione:

---

### INFORMATICI UNION TORINESI

È una tabella (priva di nome) con

- ▶ schema :

- ▶ lo schema di INFORMATICI

- ▶ istanza :

- ▶ la unione delle tuple di INFORMATICI e TORINESI

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf
415	Paola	Torino	Inf

# Differenza (-)

---

## **TABELLA1 MINUS TABELLA2**

- ▶ Si può fare se TABELLA1 e TABELLA2 sono compatibili

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ lo schema di TABELLA1
- ▶ istanza :
  - ▶ la differenza delle tuple di TABELLA1 e TABELLA2

**ATTENZIONE: no commutativa, no associativa!**

## Esempio di Differenza:

---

### INFORMATICI MINUS TORINESI

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ lo schema di INFORMATICI
- ▶ istanza :
  - ▶ la differenza delle tuple di INFORMATICI e TORINESI

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf

ATTENZIONE:  
non e'  
commutativa !

## Join ()

---

### **TABELLA1 JOIN [PredJoin] TABELLA2**

È una tabella (priva di nome) con

- ▶ schema :
  - ▶ la concatenazione degli schemi di TABELLA1 e TABELLA2
- ▶ istanza:
  - ▶ le tuple ottenute concatenando quelle tuple di TABELLA1 e di TABELLA2 che soddisfano il predicato 'PredJoin'

# Esempio di Join

## STUDENTE JOIN [MATR=MATR] ESAME

È una tabella (priva di nome) con

► schema :

► la concatenazione degli schemi di STUDENTE e ESAME

► istanza:

► le tuple ottenute concatenando le tuple di STUDENTE e di ESAME che soddisfano [MATR=MATR]

STUDENTE. MATR	NOME	CITTA'	C-DIP	ESAME. MATR	COD- CORSO	DATA	VOTO
123	Carlo	Bologna	Inf	123	1	7-9-14	30
123	Carlo	Bologna	Inf	123	2	8-1-15	28
702	Antonio	Roma	Log	702	2	7-9-14	20

# Sintassi del predicato di join

---

- ▶ Espressione congiuntiva di predicati semplici:

**ATTR1 comp ATTR2**

Dove

- ▶ ATTR1 appartiene a TAB1
  - ▶ ATTR2 appartiene a TAB2
  - ▶ comp: =, !=, <, <=, >, >=
- 
- ▶ Attributi omonimi sono resi non ambigui usando la **notazione “puntata”**
    - ▶ es: ESAME.MATR, STUDENTE.MATR

# Equi-join e Join naturale

---

- ▶ EQUI-JOIN :
  - ▶ soli confronti di uguaglianza
- ▶ JOIN NATURALE :
  - ▶ equi-join di tutti gli attributi omonimi
- ▶ Es: **STUDENTE JOIN ESAME**

MATR	NOME	CITTA'	C-DIP	COD-CORSO	DATA	VOTO
123	Carlo	Bologna	Inf	1	7-9-14	30
123	Carlo	Bologna	Inf	2	8-1-15	28
702	Antonio	Roma	Log	2	7-9-14	20

# Join naturale di tre tabelle

---

## ► Es. STUDENTE JOIN ESAME JOIN CORSO

MATR	NOME	CITTA'	C-DIP	COD-CORSO	DATA	VOTO	TITOLO	DOCENTE
123	Carlo	Bologna	Inf	1	7-9-14	30	matem	barozzi
123	Carlo	Bologna	Inf	2	8-1-15	28	infor	meo
702	Antonio	Roma	Log	2	7-9-14	20	infor	meo



## Proprietà del join

---

- ▶ Il Join è **commutativo** e **associativo**
  - ▶  $R1 \bowtie R2 \equiv R2 \bowtie R1$
  - ▶  $R1 \bowtie (R2 \bowtie R3) \equiv (R1 \bowtie R2) \bowtie R3$
- ▶ **Join naturale senza attributi omonimi**: se le due relazioni non hanno attributi omonimi ( $R1 \cap R2 = \emptyset$ ), il Join naturale si riduce ad essere un semplice prodotto cartesiano. Cioè, concatenazione di tutte le tuple in R1 e con tutte le tuple in R2:
  - ▶  $R1 \bowtie R2 \equiv R1 \times R2$

# Priorità degli operatori

---

- ▶ E' sempre bene, specie in caso di dubbio, utilizzare le **parentesi** per specificare in esplicito la priorità desiderata tra gli operatori di un'espressione algebrica
- ▶ Ad ogni modo, in assenza di parentesi, valgono le seguenti regole di priorità:
  1. SELEZIONE, PROIEZIONE (massima priorità)
  2. JOIN
  3. UNIONE, DIFFERENZA

# Espressioni algebriche

---

- ▶ Concatenazione di più operazioni algebriche
- ▶ Esprimono interrogazioni in modo formale
- ▶ Consentono di estrarre informazioni dai dati

# Selezione e proiezione

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf
415	Paola	Torino	Inf
702	Antonio	Roma	Log

- Quali studenti (nome) sono iscritti al diploma di informatica?

```
PROJECT [NOME]  
SELECT [C-DIP='Inf']  
STUDENTE
```

NOME
Carlo
Paola

# Selezione e proiezione

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf
415	Paola	Torino	Inf
702	Antonio	Roma	Log

- Quali studenti (nome) di Logistica non sono di Milano?

```
PROJECT [NOME]  
SELECT [C-DIP='Log' AND  
        CITTA' != 'Milano']  
STUDENTE
```

NOME
Antonio

# Esempio: gestione degli esami universitari

---

## studente

MATR	NOME	CITTA'	C-DIP
123	Carlo	Bologna	Inf
415	Paola	Torino	Inf
702	Antonio	Roma	Log

## esame

MATR	COD-CORSO	DATA	VOTO
123	1	7-9-14	30
123	2	8-1-15	28
702	2	7-9-14	20

## corso

COD-CORSO	TITOLO	DOCENTE
1	matematica	Barozzi
2	informatica	Meo

# Selezione, proiezione e join

---

- Quali studenti (nome) hanno preso 30 in matematica?

MATR	NOME	CITTA'	C-DIP	COD-CORSO	DATA	VOTO	TITOLO	DOCENTE
123	Carlo	Bologna	Inf	1	7-9-14	30	matem	barozzi
123	Carlo	Bologna	Inf	2	8-1-15	28	infor	meo
702	Antonio	Roma	Log	3	7-9-14	20	infor	meo

# Selezione, proiezione e join

---

- Quali studenti (nome) hanno preso 30 in matematica?

```
PROJECT [NOME]  
  SELECT [VOTO=30 AND  
          TITOLO='matematica']  
(STUDENTE JOIN ESAME JOIN CORSO)
```

NOME
Carlo



## Equivalenza di espressioni

---

- ▶ Quali studenti (nome) hanno preso 30 in matematica?

```
PROJECT [NOME]  
  SELECT [VOTO=30 AND  
          TITOLO='matematica']  
(STUDENTE JOIN ESAME JOIN CORSO)
```

```
PROJECT [NOME]  
( STUDENTE JOIN  
  (SELECT [VOTO=30] ESAME) JOIN  
  (SELECT [TITOLO='matematica'] CORSO))
```

# Selezione, proiezione e join

---

- Quali professori hanno esaminato Antonio?

```
PROJECT [DOCENTE]  
  SELECT [NOME = 'Antonio']  
(STUDENTE JOIN ESAME JOIN CORSO)
```

DOCENTE
Meo

# Equivalenza di espressioni

---

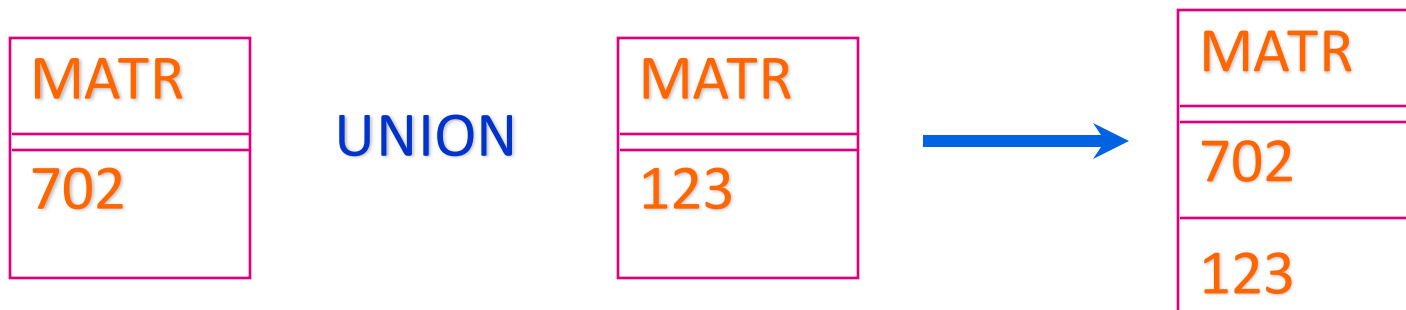
```
PROJECT [DOCENTE]  
  SELECT [NOME = 'Antonio']  
(STUDENTE JOIN ESAME JOIN CORSO)
```

```
PROJECT [DOCENTE]  
(CORSO JOIN  
  (ESAME JOIN  
    SELECT [NOME = 'Antonio'] STUDENTE))
```

## Espressioni con unione e differenza

- ▶ Estrarre la matricola degli studenti romani oppure degli studenti che hanno sostenuto un esame il giorno 8-1-15

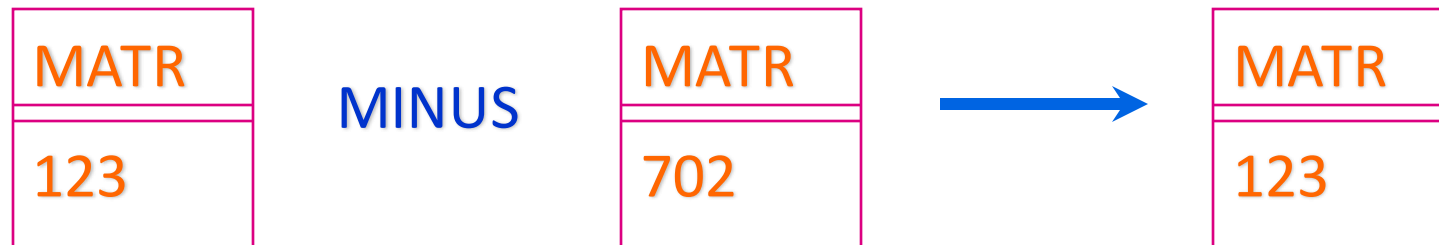
```
( PROJECT [MATR]  
  SELECT [CITTA]='Roma'] STUDENTE)  
UNION  
( PROJECT [MATR]  
  SELECT [DATA=8-1-15] ESAME)
```



## Espressioni con unione e differenza

- Estrarre la matricola degli studenti che hanno preso almeno un voto superiore a 28 e non sono mai scesi sotto il 25

```
(PROJECT [MATR]  
SELECT [VOTO>28] ESAME)  
MINUS  
(PROJECT [MATR]  
SELECT [VOTO<25] ESAME)
```



## Espressioni con unione e differenza

---

- La seconda soluzione è equivalente alla prima?

```
(PROJECT [MATR]  
SELECT [VOTO>28] ESAME)  
MINUS  
(PROJECT [MATR]  
SELECT [VOTO<25] ESAME)  
  
PROJECT [MATR]  
(SELECT [VOTO>28] ESAME  
MINUS  
SELECT [VOTO<25] ESAME)
```

# Espressioni con unione e differenza

---

- ▶ No! La seconda soluzione è errata
- ▶ Ad esempio, in questa situazione...

esame

MATR	COD-CORSO	DATA	VOTO
123	1	7-9-14	30
702	2	8-1-15	28
123	2	7-9-16	20

- ▶ ... il risultato corretto è chiaramente la relazione vuota (prodotta dalla prima soluzione)

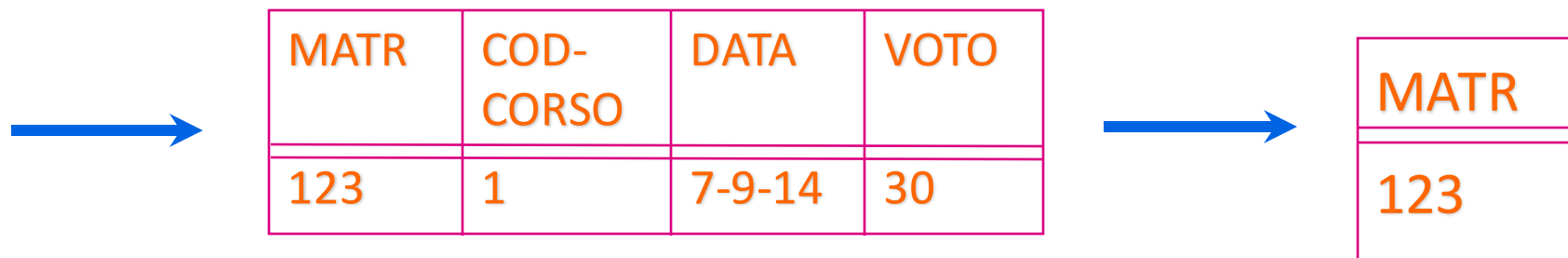
## Espressioni con unione e differenza

- La seconda soluzione (non corretta, in quanto realizza una differenza tra esami e non tra matricole) produrrebbe:

MATR	COD-CORSO	DATA	VOTO
123	1	7-9-14	30

MINUS

MATR	COD-CORSO	DATA	VOTO
123	2	7-9-16	20





# Esercizi

---

- ▶ Riprendere le basi di dati per la gestione del personale e degli ordini e esprimere in forma algebrica le interrogazioni:
  - ▶ in quali tipi di progetti lavora Giovanni?
  - ▶ chi e' il manager di Piero?
  - ▶ quale impiegato e' stato assunto per primo?
  - ▶ quali ordini ha emesso Paolo?
  - ▶ quali prodotti sono ordinati da un cliente di Milano?
  - ▶ quali prodotti hanno prezzo inferiore a 10 € e non sono presenti in nessun ordine?

# Esempio : gestione personale

---

## impiegato

MATR	NOME	DATA-ASS	SALARIO	MATR-MGR
1	Piero	1-1-12	1500 €	2
2	Giorgio	1-1-14	2000 €	null
3	Giovanni	1-7-13	1000 €	2

## assegnamento

MATR	NUM-PROG	PERC
1	3	50
1	4	50
2	3	100
3	4	100

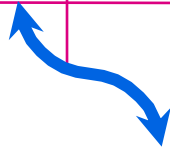
## progetto

NUM-PROG	TITOLO	TIPO
3	Idea	Esprit
4	Wide	Esprit

## Esempio : gestione ordini

COD-CLI	INDIRIZZO	P-IVA

cliente



COD-ORD	COD-CLI	DATA	IMPORTO

ordine



COD-ORD	COD-PROD	QTA

dettaglio



COD-PROD	NOME	PREZZO

prodotto