

Viste (relazioni derivate)

- **Relazioni di base:** contenuto autonomo, le relazioni nella base di dati
- **Relazioni derivate:**
 - relazioni il cui contenuto è funzione del contenuto di altre relazioni (definito per mezzo di interrogazioni)
- Le relazioni derivate possono essere definite su altre derivate, ma ...

Viste, esempio

Afferenza

Impiegato	Reparto
Rossi	A
Neri	B
Bianchi	B

Direzione

Reparto	Capo
A	Mori
B	Bruni

- una vista:

Supervisione =

$\text{PROJ}_{\text{Impiegato, Capo}} (\text{Afferenza JOIN Direzione})$

Interrogazioni sulle viste

- Sono eseguite sostituendo alla vista la sua definizione:

SEL_{Capo='Leoni'} (**Supervisione**)

viene eseguita come

PROJ_{Impiegato, Capo} (**SEL**_{Capo='Leoni'} (**Afferenza JOIN Direzione**))

Viste, motivazioni

Nota bene:

- L'utilizzo di viste non influisce sull'efficienza delle interrogazioni

Vantaggi:

- Soprattutto:
 - **Strumento di programmazione:**
 - si può semplificare la scrittura di interrogazioni:
espressioni complesse e sottoespressioni ripetute
- Ogni utente vede solo
 - ciò che gli interessa e nel modo in cui gli interessa, senza essere distratto dal resto
 - ciò che è autorizzato a vedere (autorizzazioni)
- Utilizzo di programmi esistenti su schemi ristrutturati

Viste come strumento di programmazione

- Trovare gli impiegati che hanno lo stesso capo di Rossi (vedremo meglio il concetto più avanti)
- Senza vista:

```
PROJ Impiegato ((Afferenza JOIN Direzione) JOIN  
                REN ImpR,RepR ← Imp,Reparto (  
                SEL Impiegato='Rossi' (Afferenza JOIN Direzione)))
```

- Con la vista:

```
PROJ Impiegato (Supervisione JOIN  
                REN ImpR ← Imp (  
                SEL Impiegato='Rossi' (Supervisione)))
```

Un servizio online per esercitazioni in algebra relazionale

(lo abbiamo già visto ma ripetiamo i dettagli per comodità – ora è necessario usare lo strumento)

- RelaX
 - <http://dbis-uibk.github.io/relax/calc>
- Verrà proposto un “homework” il cui svolgimento sarà necessario per partecipare alla prova parziale

RelaX

- Utilizza una sintassi molto simile a quella vista a lezione e sul libro
- L'editor aiuta nella scrittura degli operatori e dei nomi di relazione e di attributo (basta cliccare sul simbolo desiderato)
- Talvolta è utile scrivere direttamente – allora attenzione a maiuscole e minuscole (è “case-sensitive”)
- Le espressioni sono talvolta di lettura non semplice, perché tutto su una linea, senza “pedici”:
 - scriviamo $\sigma_{\text{Stipendio} > 40}(\text{Impiegati})$ invece di
 $\sigma_{\text{Stipendio} > 40}(\text{Impiegati})$
- Attenzione agli spazi (talvolta lo strumento si confonde) e spesso è utile qualche parentesi in più
- Una differenza nella “assegnazione”; serve una “ridenominazione” esplicita della relazione; invece di
 $\text{Capi} := \text{Impiegati}$
dobbiamo scrivere
 $\text{Capi} = \rho_{\text{Capi}}(\text{Impiegati})$

Rappresentazione grafica

- RelaX fornisce anche una rappresentazione grafica delle espressioni sotto forma di albero, molto espressiva
- Ogni operatore è un nodo, con uno o due nodi discendenti (a seconda che abbia uno o due operandi) e le foglie sono relazioni nella base di dati
- Nei lucidi seguenti sono mostrate le interrogazioni discusse in aula e per ciascuna è mostrata la formulazione mostrata in aula, quelle in RelaX (molto simile) e l'albero generato da RelaX

Dati

- Accedendo al servizio si possono specificare interrogazioni su una base di dati
 - fra quelle disponibili sul servizio, oppure
 - su una “caricata” dall’utente
- Per i primi esempi (in questa presentazione), le basi di dati sono state predisposte e possono essere caricate selezionando il link “Select DB ..” (in alto a sinistra) e inserendo nel campo “Load dataset stored in a gist” il relativo link
 - [1a9dc6cd0f3478388fc177dfc9b5a314](https://gist.github.com/1a9dc6cd0f3478388fc177dfc9b5a314) (prima bd)
 - [b7a8eac38317e0d6a7f0b904a9a10bd3](https://gist.github.com/b7a8eac38317e0d6a7f0b904a9a10bd3) (seconda bd)
- oppure, più semplicemente richiamando RelaX con l’url:
 - <http://dbis-uibk.github.io/relax/calc/gist/1a9dc6cd0f3478388fc177dfc9b5a314>
 - <http://dbis-uibk.github.io/relax/calc/gist/b7a8eac38317e0d6a7f0b904a9a10bd3>
- Ulteriori basi di dati (data-set nella terminologia di RelaX) possono essere predisposti con una sintassi molto semplice e caricati su github (vedere l’help)

```
Impiegati = {  
  Matricola, Nome, Eta:number, Stipendio:number  
    7309, Rossi, 34, 45  
    5998, Bianchi, 37, 38  
    9553, Neri, 42, 35  
    5698, Bruni, 43, 42  
    4076, Mori, 45, 50  
    8123, Lupi, 46, 60  
}
```

```
Supervisione = {  
  Impiegato, Capo  
    7309, 5698  
    5998, 5698  
    9553, 4076  
    5698, 4076  
    4076, 8123  
}
```

Metodo alternativo per caricare i dati

- Con il Group Editor: "Preview" e "use group in Editor"

RelaX

CalculatorLanguageFeedbackHelpImprint

13 Supervisione = {
14 Impiegato Capo
15 7309 5698
16 5998 5698
17 9553 4076
18 5698 4076
19 4076 8123
20 }
21

preview

Download

Azienda use group in editor

• Impiegati

Impiegati.MatricolaImpiegati.NomeImpiegati.EtaImpiegati.Stipendio

7309	'Rossi'	34	45
5998	'Bianchi'	37	38

group: Azienda

Impiegati = {

Matricola	Nome	Eta	Stipendio
7309	'Rossi'	34	45
5998	'Bianchi'	37	38
9553	'Neri'	42	35
5698	'Bruni'	43	42
4076	'Mori'	45	50
8123	'Lupi'	46	60

}

Supervisione = {

Impiegato	Capo
7309	5698
5998	5698
9553	4076
5698	4076
4076	8123

}

**Istruzioni per
caricare i dati
con il group editor**

Esempi

Impiegati

<u>Matricola</u>	Nome	Età	Stipendio
7309	Rossi	34	45
5998	Bianchi	37	38
9553	Neri	42	35
5698	Bruni	43	42
4076	Mori	45	50
8123	Lupi	46	60

Supervisione

<u>Impiegato</u>	Capo
7309	5698
5998	5698
9553	4076
5698	4076
4076	8123

- Trovare matricola, nome, età e stipendio degli impiegati che guadagnano più di 40

$SEL_{\text{Stipendio} > 40}(\text{Impiegati})$

$\sigma_{\text{Stipendio} > 40}(\text{Impiegati})$

$\sigma_{\text{Stipendio} > 40}$

Impiegati

- Trovare matricola, nome ed età degli impiegati che guadagnano più di 40

$\text{PROJ}_{\text{Matricola, Nome, Età}} (\text{SEL}_{\text{Stipendio} > 40} (\text{Impiegati}))$

$\pi \text{ Matricola, Nome, Eta } (\sigma \text{ Stipendio} > 40 (\text{Impiegati}))$



- Trovare le matricole dei capi degli impiegati che guadagnano più di 40

$\text{PROJ}_{\text{Capo}} (\text{Supervisione}$
 $\text{JOIN}_{\text{Impiegato=Matricola}}$
 $(\text{SEL}_{\text{Stipendio}>40}(\text{Impiegati})))$

$\pi \text{ Capo } ((\text{Supervisione})$
 $\bowtie \text{ Impiegato=Matricola}$
 $(\sigma \text{ Stipendio}>40 (\text{Impiegati})))$



- Trovare nome e stipendio dei capi degli impiegati che guadagnano più di 40

$\text{PROJ}_{\text{Nome, Stipendio}} ($
 $\text{Impiegati JOIN}_{\text{Matricola=Capo}}$
 $\text{PROJ}_{\text{Capo}}(\text{Supervisione}$
 $\text{JOIN}_{\text{Impiegato=Matricola}}$
 $(\text{SEL}_{\text{Stipendio}>40}(\text{Impiegati}))))$

$\pi_{\text{Nome, Stipendio}} ($
 $\text{Impiegati} \bowtie_{\text{Matricola = Capo}}$
 $(\pi_{\text{Capo}} ((\text{Supervisione})$
 $\bowtie_{\text{Impiegato=Matricola}}$
 $(\sigma_{\text{Stipendio}>40} (\text{Impiegati}))))$

