

SQL

(seconda parte)

Argomenti della lezione

- ▶ Interrogazioni in SQL:
 - ▶ selezioni, proiezioni, join

SQL, operazioni sui dati

- ▶ interrogazione:

- ▶ **SELECT**

- ▶ modifica:

- ▶ **INSERT, DELETE, UPDATE**

Istruzione SELECT

SELECT Lista Attributi
FROM Lista Tabelle
[**WHERE** Condizione]

- ▶ "target list"
- ▶ clausola **FROM**
- ▶ clausola **WHERE**

Maternità

Madre	Figlio
Luisa	Maria
Luisa	Luigi
Anna	Olga
Anna	Filippo
Maria	Andrea
Maria	Aldo

Paternità

Padre	Figlio
Sergio	Franco
Luigi	Olga
Luigi	Filippo
Franco	Andrea
Franco	Aldo

Persone

Nome	Età	Reddito
Andrea	27	21
Aldo	25	15
Maria	55	42
Anna	50	35
Filippo	26	30
Luigi	50	40
Franco	60	20
Olga	30	41
Sergio	85	35
Luisa	75	87

Selezione e proiezione

- Nome e reddito delle persone con meno di trenta anni

$PROJ_{Nome, Reddito}(SEL_{Eta < 30}(Persone))$

```
select nome, reddito  
from persone  
where eta < 30
```

Persone

Nome	Reddito
Andrea	21
Aldo	15
Filippo	30

SELECT, abbreviazioni

```
select nome, reddito  
from persone  
where eta < 30
```

```
select p.nome, p.reddito as entrate  
from persone p  
where p.eta < 30
```


Selezione, senza proiezione

- ▶ Nome, età e reddito delle persone con meno di trenta anni

$SEL_{Eta < 30}(Persone)$

```
select nome, età, reddito  
from persone  
where età < 30
```

```
select nome, età, reddito  
from persone  
where eta < 30
```

```
select *  
from persone  
where eta < 30
```

Proiezione, senza selezione

- ▶ Nome e reddito di tutte le persone

$\text{PROJ}_{\text{Nome, Reddito}}(\text{Persone})$

```
select nome, reddito  
from persone
```

SELECT, abbreviazioni

select *
from R

equivale (intuitivamente) a

select X.A AS A, X.B AS B
from R X
where true

Espressioni nella target list

```
select Reddito/2 as  
    RedditoSemestrale  
from Persone  
where Nome = 'Luigi'
```

RedditoSemestrale

20

Condizione “LIKE”

- ▶ Le persone che hanno un nome che inizia per 'A' e ha una 'd' come terza lettera

select *

from persone

where nome like 'A_d%'

Persone

Nome	Età	Reddito
Andrea	27	21
Aldo	25	15
Maria	55	42
Anna	50	35
Filippo	26	30
Luigi	50	40
Franco	60	20
Olga	30	41
Sergio	85	35
Luisa	75	87

Gestione dei valori nulli

Impiegati

Matricola	Cognome	Filiale	Età
5998	Neri	Milano	45
9553	Bruni	Milano	NULL

SEL `Età > 40 OR Età IS NULL` (**Impiegati**)

- **Gli impiegati la cui età è o potrebbe essere maggiore di 40**

SEL **Età > 40 OR Età IS NULL** **(Impiegati)**

select *

from impiegati

where eta > 40 or eta is null

SQL e algebra relazionale

```
select R1.A1, R2.A4  
from   R1, R2  
where  R1.A2 = R2.A3
```

$\text{PROJ}_{A1,A4} (\text{SEL}_{A2=A3} (R1 \text{ JOIN } R2))$

- ▶ possono essere necessarie ridenominazioni
 - ▶ nel prodotto cartesiano
 - ▶ nella target list

```
select X.A1 AS B1, ...  
from   R1 X, R2 Y, R1 Z  
where  X.A2 = Y.A3 AND ...
```

```

select X.A1 AS B1, Y.A4 AS B2
from   R1 X, R2 Y, R1 Z
where  X.A2 = Y.A3 AND Y.A4 = Z.A1

```

```

      RENB1,B2 ← A1,A4 (
    PROJA1,A4 (SELA2 = A3 AND A4 = C1 (
R1 JOIN R2 JOIN RENC1,C2 ← A1,A2 (R1))))

```

- **cognome e filiale di tutti gli impiegati**

Cognome	Filiale
Neri	Napoli
Neri	Milano
Rossi	Roma

PROJ Cognome, Filiale **(Impiegati)**

```
select  
    cognome, filiale  
from impiegati
```

Cognome	Filiale
Neri	Napoli
Neri	Milano
Rossi	Roma
Rossi	Roma

```
select distinct  
    cognome, filiale  
from impiegati
```

Cognome	Filiale
Neri	Napoli
Neri	Milano
Rossi	Roma

Maternità

Madre	Figlio
Luisa	Maria
Luisa	Luigi
Anna	Olga
Anna	Filippo
Maria	Andrea
Maria	Aldo

Paternità

Padre	Figlio
Sergio	Franco
Luigi	Olga
Luigi	Filippo
Franco	Andrea
Franco	Aldo

Persone

Nome	Età	Reddito
Andrea	27	21
Aldo	25	15
Maria	55	42
Anna	50	35
Filippo	26	30
Luigi	50	40
Franco	60	20
Olga	30	41
Sergio	85	35
Luisa	75	87

Selezione, proiezione e join

- I padri di persone che guadagnano più di venti milioni

**PROJ_{Padre}(paternita JOIN_{Figlio = Nome}
SEL_{Reddito > 20} (persone))**

**PROJ_{Padre}(paternita JOIN_{Figlio =Nome}
SEL_{Reddito>20} (persone))**

**select distinct padre
from persone, paternita
where figlio = nome and
reddito > 20**

Join naturale

- Padre e madre di ogni persona
paternita JOIN maternita

```
select paternita.figlio,padre, madre  
from maternita, paternita  
where paternita.figlio =  
      maternita.figlio
```

Join complessi

- Le persone che guadagnano più dei rispettivi padri; mostrare nome, reddito e reddito del padre

```
PROJNome, Reddito, RP (SELReddito>RP  
(RENNP,EP,RP ← Nome,Eta,Reddito (persone)  
JOINNP=Padre  
(paternita JOINFiglio = Nome persone)))
```

PROJ_{Nome, Reddito, RP} (SEL_{Reddito>RP}
 (REN_{NP,EP,RP} ← Nome, Eta, Reddito (persone)
 JOIN_{NP=Padre}
 (paternita JOIN_{Figlio = Nome} persone)))

select f.nome, f.reddito, p.reddito
 from persone p, paternita, persone f
 where p.nome = padre and
 figlio = f.nome and
 f.reddito > p.reddito

- ▶ Le persone che guadagnano più dei rispettivi padri; mostrare nome, reddito e reddito del padre

```

    PROJFiglio, Reddito, RP (SELReddito>RP (
RENNP,EP,RP ← Nome, Eta, Reddito (persone)
                                JOINNP=Padre
    (paternita JOINFiglio =Nome persone)))

```

PROJ_{Figlio, Reddito, RP} (SEL_{Reddito>RP} (
 REN_{NP,EP,RP} ← Nome, Eta, Reddito (persone)
 JOIN_{NP=Padre}
 (paternita JOIN_{Figlio = Nome} persone)))

select figlio, f.reddito as reddito,
 p.reddito as redditoPadre from
 persone p, paternita, persone f
 where p.nome = padre and figlio = f.nome
 and .reddito > p.reddito

Argomenti della lezione

- ▶ Interrogazioni in SQL:
 - ▶ selezioni, proiezioni, join