

Contents

1	Schema	2
2	Prima query	2
3	Seconda Query	3
4	Terza query	4
5	Quarta query	4
5.1	Primo passaggio	4
5.2	Secondo passaggio	4
6	Quinta query	5
6.1	Prima con l'all	5
6.2	Rivediamola con la sotto sotto query	6
6.2.1	count(impiegati)	6
6.2.2	max(count(impiegati))	6
6.2.3	having count = max(count(impiegati))	6
7	Sesta query	7
7.1	Il Pala	7
7.2	La mia cazzata	8
8	Settima query	8
9	Ottava query	9
9.1	Primo passo	9
9.2	Secondo passo	9
10	Nona query	9
10.1	Primo passo	10
10.2	Secondo passo	10
11	Decima query	10
12	Undicesima query	11
13	Dodicesima query	12
13.1	Trova conto dipendenti per progetto	12
13.2	Applico metriche	12

(se si prendesse una tabella possibilmente infinita di tutti gli stati di una macchina di moore a due registri forse si potrebbe fare una versione completa dell'sql)

1 Schema

- **Impiegato** : matricola, nome, cognome, dipartimento(come codice), stipendio
- **Progetto** : codice, nome, budget, scadenza, responsabile(come matricola)
- **PP** : impiegato(come matricola) <-> progetto(come codice)
- **Dipartimento** : codice, nome, direttore(come matricola)

2 Prima query

estrarre matricola, nome, e cognome di ciascun impiegato ed il numero di progetti a cui partecipa, facendo comparire anche chi non partecipa a progetto

- facendo comparire anche ... -> join esterno
- matricola <-> nome, cognome -> serve tabella impiegato
- numero progetti a cui ... -> serve tabella pp, e serve group by impiegato

leghiamo i progetti all'impiegato con

```
pp join impiegato i on pp.impiegato = i.matricola
```

poi da questo impiegato aumentato si estrae nome, cognome, e matricola

```
select i.nome, i.cognome, i.matricola
from pp join impiegato i on (pp.impiegato = i.matricola)
```

e visto che vogliamo il numero di progetti (ricordandoci che la `select` e la `group by`...)

```
select i.matricola ,i.nome, i.cognome, count(progetto) as count_progetto
from pp join impiegato i on (pp.impiegato = i.matricola)
group by i.matricola, i.nome, i.cognome;
```

vogliamo far comparire anche gli impiegati sfaticati, quindi tutte le righe di impiegato, quindi qui

```
select i.matricola ,i.nome, i.cognome, count(progetto) as count_progetto
from pp right join impiegato i on (pp.impiegato = i.matricola)
group by i.matricola, i.nome, i.cognome;
```

e si vede che Giuliano Casini è uno stronzo

3 Seconda Query

estrarre per ciascun dipartimento il nome del dipartimento ed il numero dei suoi affiliati che risultano essere responsabili di progetto

- nome dipartimento -> serve tabella dipartimento
- impiegato affiliati a dipartimento -> serve tabella impiegati
- responsabili di progetto -> serve la tabella progetto

si isolano innanzitutto i responsabili di progetto facendo

```
progetto p join impiegato i on (p.responsabile = i.matricola)
```

questo lo leggiamo a dipartimenti in modo abbastanza ovvio

```
(progetto p join impiegato i on (p.responsabile = i.matricola))
join dipartimento d on (d.codice = i.dipartimento)
```

però serve che ci siano TUTTI i dipartimenti, anche quelli sficati senza responsabili tra gli impiegati, quindi

```
(progetto p join impiegato i on (p.responsabile = i.matricola))
right join dipartimento d on (d.codice = i.dipartimento)
```

con le solite `select` e `group by` si fa

```
select d.nome ,count(i.matricola) as n_responsabili
from (progetto p join impiegato i on (p.responsabile = i.matricola))
right join dipartimento d on (d.codice = i.dipartimento)
group by d.codice;
```

4 Terza query

estrarre i codici e nomi dei dipartimenti tra i cui affiliati **non** ci sono responsabili di progetto

possiamo fare gli stronzi e copiare la query precedente, aggiungendo un'ultima riga in fondo

```
select d.nome ,count(i.matricola) as n_responsabili
from (progetto p join impiegato i on (p.responsabile = i.matricola))
right join dipartimento d on (d.codice = i.dipartimento)
group by d.codice
having count(i.matricola) = 0;
```

cambiando il select per rispettare la traccia

```
select d.codice, d.nome
from (progetto p join impiegato i on (p.responsabile = i.matricola))
right join dipartimento d on (d.codice = i.dipartimento)
group by d.codice
having count(i.matricola) = 0;
```

5 Quarta query

Estrarre i codici dei progetti che hanno budget massimo

Questa query si divide in due passaggi

1. Trova il budget massimo
2. `Select codice from progetto where budget = quello`

divisa così i due passaggi non sono molto complicati, quindi

5.1 Primo passaggio

```
select max(budget) as max_bud
from progetto
```

5.2 Secondo passaggio

```
select codice
from progetto
where budget =
```

```
(select max(budget) as max_bud
from progetto);
```

visto che la *sottoquery* al primo passaggio si capiva che ritornava un solo valore, postgresql ci ha fatto il favore di capire che era uno scalare e di trattarne il risultato come tale.

6 Quinta query

- nome dipartimento -> serve tabella dipartimento
- impiegato affiliati -> serve tabella impiegato

estrarre il massimo numero di impiegati affiliati ad un dipartimento insieme al codice e nome del dipartimento

questa si può strutturare similmente alla precedente con

1. trova il massimo numero di impiegati affiliato
2. vedi quale dipartimento aveva quel numero, e prendine i dati

6.1 Prima con l'all

visto che a paragonare con `max(count(qualcosa))` va fatta una query per trovare il count, e un'altra query per trovarne il max, e solo dopo quelle puoi paragonare con `max(count())`, facciamo prima con la clausola `all`

per riformulare con la clausola `all` diciamo che vogliamo i dipartimenti con numero di impiegato `>=` a quello di tutti gli altri dipartimenti, alla sql, `>= all (select count(t'ha capito)...)`

quindi prima troviamo i conti di tutti i dipartimenti

```
select count(matricola) from impiegato group by dipartimento;
-- oppure, che tanto matricola è chiave primaria
select count(*) from impiegato group by dipartimento;
```

ora paragoniamo i conti di tutti i dipartimenti a quelli appena ottenuti

```
select ...
from impiegato, dipartimento
group by dipartimento
having count(matricola) >= all
(select count(matricola)
from impiegato
group by dipartimento);
```

la traccia chiedeva il numero di impiegati, il codice del dipartimento, e il nome

e mi sono ricordato adesso che esisteva la tabella dipartimento

```
select count(matricola) as n_impiegati, d.codice, d.nome
from impiegato i join dipartimento d on (i.dipartimento = d.codice)
group by i.dipartimento, d.codice, d.nome
having count(matricola) >= all
(select count(matricola)
from impiegato
group by dipartimento);
```

6.2 Rivediamola con la sotto sotto query

ci servono tre passaggi

- count(impiegati)
- max(count(impiegati))
- having count = max(count(impiegati))

6.2.1 count(impiegati)

```
select count(matricola) as n_impiegati
from impiegato
group by dipartimento
```

6.2.2 max(count(impiegati))

```
select max(conti.n_impiegati)
from (select count(matricola) as n_impiegati
from impiegato
group by dipartimento) as conti;
```

6.2.3 having count = max(count(impiegati))

```
select d.nome, d.codice, count(i.matricola)
from dipartimento d join impiegato i on (i.dipartimento = d.codice)
group by d.nome, d.codice
-- ricordati di non mettere la matrice nel group by, visto che è stupido
having count(i.matricola)=
```

```
(select max(conti.n_impiegati)
from (select count(matricola) as n_impiegati
from impiegato
group by dipartimento) as conti);
```

come potete vedere, ho creato un mostro

7 Sesta query

Estrarre matricola, nome e cognome degli impiegati che non sono direttori di dipartimento o responsabili di progetto

ci sono vari modi per risolvere questo problema, qui metterò

- quello che ha fatto il Pala
- la cazzata che ho provato a fare io

7.1 Il Pala

I'll select you the fuck out with precision and elegance the likes
of which has never been seen before

~ il Pala, suppongo

la soluzione qui ha una forma del tipo

```
select quello
where
matricola <> tutte (quelle di responsabili) and
matricola <> tutte (quelle di direttori)
```

questo si fa col (sacro)costrutto all nel seguente modo

```
select i.matricola, i.nome, i.cognome
from impiegato i
where
i.matricola <> all (select direttore from dipartimento) and
i.matricola <> all (select responsabile from progetto);
```

gli astuti tra i lettori noteranno il ritorno di quello stronzo di Giuliano Casini tra i risultati

7.2 La mia cazzata

a fare da cattivo esempio ecco come l'aveva risolta il sovrascritto

- `select roba`
`except(`
 - `select direttori`
 - `union`
 - `select responsabili)`

visto che la cosa richiedeva anche il nome e cognome, e quindi una tabella impiegati in tutte e 3 le query, e, per l'appunto, 3 query, potete immaginare come questo sia più un concetto per fare un ddos che qualcosa che ho avuto lo sbatti di scrivere

8 Settima query

Estrarre codice, nome, e budget dei progetti che hanno come responsabile l'impiegato che tra tutti i responsabili ha lo stipendio massimo

- stipendio di impiegato -> serve tabella impiegato
- progetti che... -> serve tabella progetti

possiamo isolare i responsabili come prima facendo

```
impiegato i join progetto p on (i.matricola = p.responsabile)
```

di questa tabella di serve lo stipendio massimo, quindi

```
select max(i.stipendio)
from impiegato i join progetto p on (i.matricola = p.responsabile)
```

adesso, armati nuovamente di questa tabella impiegati, ci serve un `with stipendio = quello`, e visto che la query di sopra è scalare

```
select p.codice, p.nome, p.budget
from progetto p join impiegato i on (p.responsabile = i.matricola)
where i.stipendio =
(select max(i.stipendio)
from impiegato i join progetto p on (i.matricola = p.responsabile));
```

la query in questione rende un progetto dal nome molto divertente

9 Ottava query

(da qui in poi sto facendo sta roba durante la lezione di statistica, potrebbe essere meno ordinato)

(qui si potrebbe fare tipo

```
select distinct pp.progetto
from pp join impiegato i on (pp.impiegato = i.matricola)
group by i.dipartimento, pp.progetto
having count(*) = 1;
```

?) (serve la distinct visto che a uno stesso progetto può lavorare più gente dallo stesso dipartimento) (sembra tornare)

estrarre il codice dei progetti a cui partecipano solo impiegati dello stesso dipartimento

si va per negativo, vale a dire vedi tutti i progetti a cui partecipano impiegati da 2 o più dipartimenti, poi fai l'**except** del caso e voilà

9.1 Primo passo

quindi per gli impiegati che partecipano a 2 o più progetti

- legame tra impiegati e progetti -> serve la tabella pp
- legame tra impiegati e dipartimenti -> serve tabella impiegato

9.2 Secondo passo

ok lascia fare, per ora

10 Nona query

Estrarre il codice di ogni dipartimento insieme alla matricola, nome, cognome, e stipendio dell-impiegato che ha in quel dipartimento lo stipendio massimo.

- serve la tabella impiegato
- mi sa che serve solo quella

questa query va divisa in due passi

1. trova lo stipendio massimo per ogni dipartimento
2. usa la query precedente per prendere anche dati sull'impiegato

10.1 Primo passo

```
select max(stipendio) as big_money
from impiegato
group by dipartimento;
```

questa andrebbe anche bene se volessimo solo gli stipendi massimi ma oltre a questo va legato lo stipendio massimo al dipartimento, quindi

```
select dipartimento, max(stipendio) as big_money
from impiegato
group by dipartimento;
```

10.2 Secondo passo

```
select i.dipartimento, i.matricola, i.nome, i.cognome, i.stipendio
from impiegato i
where (i.dipartimento, i.stipendio) = any(
select dipartimento, max(stipendio) as big_money
from impiegato
group by dipartimento);
```

11 Decima query

Estrarre matricola, nome, e cognome dell'impiegato responsabile per il maggior numero di progetti insieme al numero di progetti di cui è responsabile (al momento lo faccio senza creare la vista)

qui la dividiamo in due passi

```
select p.responsabile as resp, count(p.codice) as numero
from progetto p
group by p.responsabile;
```

per la seconda parte ci servirà estrarre il massimo da questa, così poi si joina con la tabella impiegato

troviamo ora il responsabile con più progetti

```
select cnt.resp, cnt.numero
from (select p.responsabile as resp, count(p.codice) as numero
from progetto p
group by p.responsabile) as cnt
where cnt.numero >= all
(select count(p.codice) as numero from progetto p group by p.responsabile);
```

per questa terza parte ci servirà anche la tabella impiegato, per ottenere nome, cognome, et al a partire dalla matricola

```
select i.nome, i.cognome, cntcnt.numero
from impiegato i join
(select cnt.resp as resp, cnt.numero as numero
from (select p.responsabile as resp, count(p.codice) as numero
from progetto p
group by p.responsabile) as cnt
where cnt.numero >= all
(select count(p.codice) as numero from progetto p group by p.responsabile)) as cntcnt
on i.matricola = cntcnt.resp;
```

cosa cazzo ho creato, mi sa che boccio

12 Undicesima query

Estrarre il numero medio degli impiegati affiliati ad un dipartimento
prima associamo i dipartimenti alla conta di quanti ci sono affiliati

```
select i.dipartimento, count(*) as numero
from impiegato i
group by dipartimento;
```

ora facciamo la media del campo numero (il campo dipartimento non andava estratto, ma vabbe')

```
select avg(counter.numero)
from (select i.dipartimento, count(*) as numero
from impiegato i
group by dipartimento) as counter;
```

(la query senza i.dipartimento per ridere era

```
select avg(counter.numero)
from (select count(*) as numero
from impiegato
group by dipartimento) as counter;
```

13 Dodicesima query

Estrarre numero minimo, medio, e massimo di impiegati che partecipano a un progetto.

si fa nello stesso stile della undicesima query, ma questa volta dobbiamo usare il collegamento tra l'impiegato e il progetto, vale a dire la tabella pp

13.1 Trova conto dipendenti per progetto

```
select pp.progetto, count(*) as numero
from pp
group by pp.progetto;
```

13.2 Applico metriche

```
select
min(counter.numero) as minimo,
avg(counter.numero) as media,
max(counter.numero) as massimo
from (select pp.progetto, count(*) as numero
from pp
group by pp.progetto) as counter;
```