

Basi di dati I — Prova di autovalutazione 22 ottobre 2018

Soluzioni

Domanda 1 Considerare la base di dati relazionale sul seguente schema:

- STUDENTI(Matricola, Cognome, Nome, Età)
- CORSI(Codice, Titolo, CFU)
- ESAMI(Studente, Corso, Data, Voto), con vincoli di integrità referenziale fra Studente e la relazione STUDENTI e fra Corso e la relazione CORSI

Supponendo che le relative relazioni abbiano rispettivamente le cardinalità $S = 10.000$ (STUDENTI), $C = 1.000$ (CORSI) e $E = 40.000$ (ESAMI), indicare le cardinalità minime e massime (in simboli e numeri) dei risultati delle seguenti interrogazioni:

	Min (simboli)	Max (simboli)	Min (valore)	Max (valore)
ESAMI $\bowtie_{\text{Corso=Codice}}$ CORSI	E	40.000	E	40.000
STUDENTI \bowtie CORSI	$S \times C$	10 milioni	$S \times C$	10 milioni
ESAMI $\bowtie_{\text{Studente=Matricola}}$ STUDENTI	E	40.000	E	40.000

Domanda 2 Con riferimento allo schema di base di dati mostrato nella domanda precedente, formulare in algebra relazionale e in SQL le seguenti interrogazioni:

1. trovare codici e titoli di corsi nei cui esami è stato assegnato almeno un trenta
2. trovare le matricole degli studenti che hanno sostenuto almeno due esami
3. per ogni studente, trovare la media dei voti riportati
4. trovare lo studente con la media più alta; mostrare i dati dello studente, la media in questione e il numero di esami superati
5. trovare i corsi nei cui esami ciascuno studente ha riportato un voto uguale o superiore a quello della propria media

vedi pagine seguenti

1.

σ voto \geq 30 (corsi \bowtie codice=corso esami)

```
select distinct codice, titolo
from corsi join esami on codice=corso
where voto >= 30;
```

2.

$e1 = \rho$ e1 (esami)

$e2 = \rho$ e2 (esami)

π e1.studente (σ e1.corso \neq e2.corso (e1 \bowtie e1.studente = e2
e2))

```
select distinct e1.studente as matricola
from esami e1 join esami e2 on e1.studente = e2.studente
where e1.corso <> e2.corso
```

π studente (σ numesami \geq 2 (γ studente; count(*) \rightarrow nume:
(esami)))

```
select studente as matricola
from esami
group by studente
having count(*) >= 2
```

```
select matricola
from (select studente as matricola, count(voto) as numesami
      from esami
      group by studente) conteggio
where numesami >= 2
```

3.

γ matricola, cognome, nome, eta; avg(voto) \rightarrow media (studenti \bowtie
matricola=studente esami)

```
select matricola, cognome, nome, eta, avg(voto) as media
from esami join studenti on studente=matricola
group by matricola, cognome, nome, eta
```

γ studente; avg(voto) \rightarrow media (esami)

\cup

(π matricola (studenti) - π studente (esami)) \bowtie ρ tmp
{ media:number

```
select matricola, avg(voto) as media, count(voto) as nume
from studenti left join esami on matricola=studente
group by matricola
```

```
select studente, avg(voto) as media, count(*) as numeroes:
from esami
group by studente
union
select matricola as studente, null, 0
from studenti
where matricola not in (select studente from esami)
```

4.

```
medievoti = γ studente; avg(voto) -> media, count(*) -> numesami
```

```
maxmedia = γ max(media) -> maxmedia (medievoti)
```

```
π matricola, cognome, nome, eta, media, numesami
(σ media=maxmedia ((medievoti ⋈ studente=matricola studente
maxmedia))
```

```
create view medievoti as
select studente, avg(voto) as media, count(*) as
from esami
group by studente;
```

```
select studenti.*, media, numesami
from medievoti join studenti on studente = matricola
where media = (select max(media) from medievoti)
```

5.

```
medievoti = ρ medievoti ( γ studente; avg(voto) -> media
```

```
votisottomedia = σ voto < media (esami ⋈
esami.studente=medievoti.studente medievoti)
```

```
π codice, titolo, cfu ((π corso (esami))
```

```
(π corso (votisottomedia))) ⋈ corso=codice corsi)
```

```
create view votisottomeia as
  select e.*, media
  from esami e join medievoti m on e.studente=m.studente
  where voto < media;
```

```
select distinct corsi.*
from corsi join esami on codice=corso
where codice not in (select corso from votisottomeia)
```

```
create view corsifacili as
select corso
from esami
except
select corso
from medievoti m join esami e on m.studente = e.studente
where voto < media
```

```
select corsi.*
from corsi join corsifacili on codice=corso
```