

Prima prova di verifica del 4/11/2008

1. Una provincia P vuole usare una base di dati per raccogliere informazioni per il controllo del traffico con sistemi di rilevamento che registrano la targa dei veicoli e la loro velocità di percorrenza. Un sistema di rilevamento è identificato da un numero unico nell'ambito del comune della provincia in cui è situato. Ad ogni sistema è associato un limite di velocità, nonché il nome ed il Km della strada in cui è situato. Di ogni veicolo che passa di fronte al sistema di rilevamento viene registrata la targa, la velocità al momento del passaggio, e il giorno, l'ora, ed il minuto del passaggio. Si assuma che uno stesso veicolo non possa passare per più di una volta di fronte allo stesso sistema nella stessa ora e nello stesso minuto dello stesso giorno. Esistono due categorie di veicoli: i veicoli commerciali ed i veicoli privati. Dei veicoli commerciali interessano la targa (che li identifica), la cilindrata, il volume e gli autisti registrati per la conduzione (almeno un autista per ogni veicolo commerciale). Dei veicoli privato interessano la targa (che li identifica), la cilindrata ed il proprietario. Di ogni autista interessano il codice fiscale, l'anno di nascita, il comune di nascita e quello di residenza. Una persona può essere proprietaria di più veicoli privati, ed essere anche registrata come autista di uno o più veicoli commerciali. Di ogni proprietario di veicolo interessano il codice fiscale, l'anno di nascita, il comune di nascita e quello di residenza. Di ogni comune interessano il codice (che lo identifica), la provincia (non tutti i comuni di interesse sono situati nella provincia P), ed il numero di abitanti.

(a) Si definisca lo schema concettuale della base di dati.

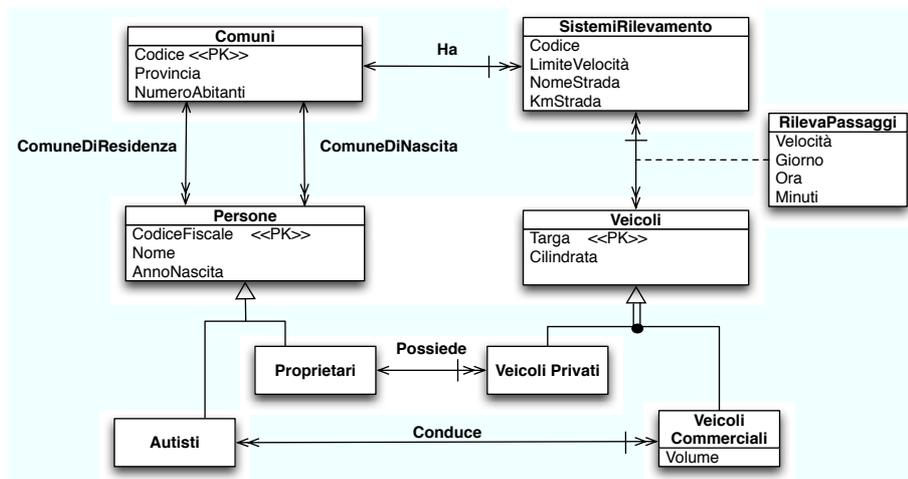


Figura 1: Schema concettuale

(b) Si traduca lo schema concettuale in uno schema relazionale grafico, definendo gli attributi delle chiavi primarie e delle chiavi esterne.

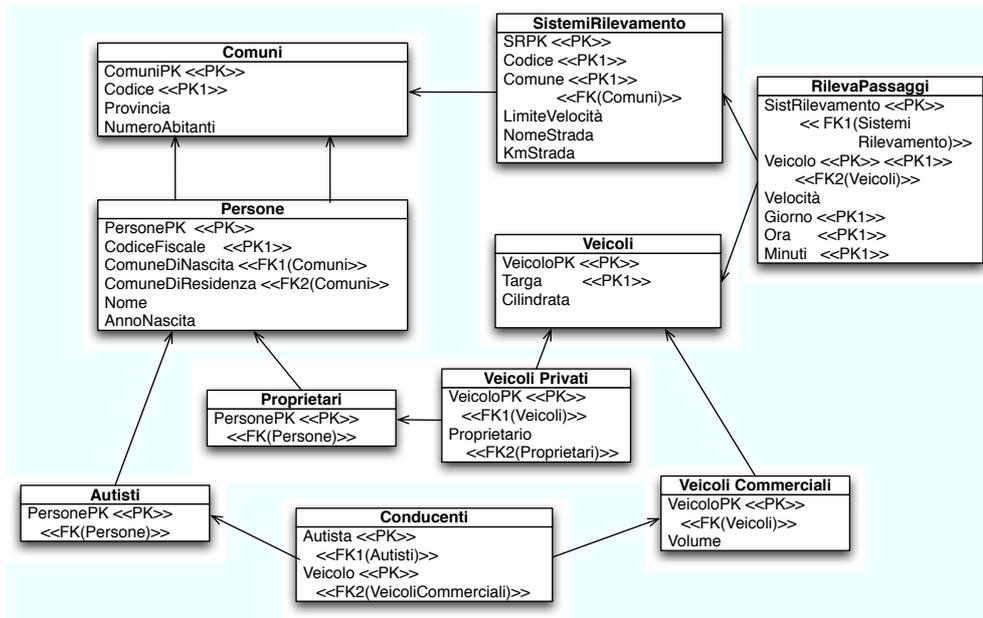


Figura 2: Schema relazionale

2. Si considerino le relazioni:

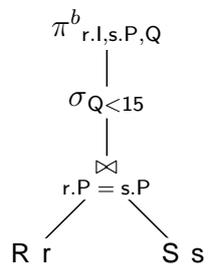
<i>R</i>		
<i>I</i>	<i>P</i>	<i>B</i>
i1	p1	10
i1	p2	20
i1	p3	30
i2	p2	20
i2	p3	30
i3	p3	30

<i>S</i>	
<i>P</i>	<i>Q</i>
p1	5
p2	10
p3	20

Si dia l'albero logico e il risultato delle seguenti interrogazioni:

```

SELECT  r.I, s.P, Q
FROM    R r, S s
WHERE   r.P = s.P AND Q < 15;
    
```

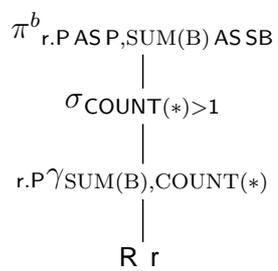


<i>r.I</i>	<i>s.P</i>	<i>Q</i>
i1	p1	5
i1	p2	10
i2	p2	10

```

SELECT    r.P AS P, SUM(B) AS SB
FROM      R r
GROUP BY  r.P
HAVING    COUNT(*) > 1;

```



<i>P</i>	<i>SB</i>
p2	40
p3	90

3. Si consideri il seguente schema relazionale:

Giocatore (CodiceG, Nome, AnnoNascita, CodiceS *)
 Squadra (CodiceS, Nome, Presidente, Allenatore, Città)
 Partita (CodiceP, Data, SquadraVincitrice*, SquadraPerdente*)
 Partecipazione(CodiceGiocatore*, CodicePartita*, GoalSegnati)

- (a) Si rappresenti graficamente lo schema relazionale della base di dati.
 (b) Si scrivano le interrogazioni SQL che restituiscono le seguenti informazioni, senza duplicazione dei risultati:

- i. Trovare il nome e l'allenatore di ogni squadra.

```
SELECT Nome, Allenatore
FROM Squadra;
```

- ii. Trovare il numero dei giocatori della squadra Pisa.

```
SELECT COUNT(*)
FROM Squadra s, Giocatore g
WHERE s.CodiceS = g.CodiceS AND s.Nome = 'Pisa';
```

- iii. Trovare il codice e il nome dei giocatori della squadra Pisa che hanno giocato almeno una partita.

```
SELECT DISTINCT g.CodiceG, g.Nome
FROM Squadra s, Giocatore g, Partecipazione p
WHERE s.CodiceS = g.CodiceS AND p.CodiceGiocatore = g.CodiceG AND s.Nome = 'Pisa';
```

- iv. Trovare il nome dei giocatori che non hanno partecipato a nessuna partita.

```
SELECT Nome
FROM Giocatore
WHERE CodiceG NOT IN (SELECT DISTINCT CodiceGiocatore FROM Partecipazione);
```

- v. Trovare i nomi dei giocatori della squadra Pisa che hanno giocato contro la squadra Empoli

```
SELECT DISTINCT g.Nome
FROM Squadra s, Giocatore g, Partecipazione p, Partita pa
WHERE s.CodiceS = g.CodiceS AND p.CodiceGiocatore = g.CodiceG
AND p.CodicePartita = pa.CodiceP AND s.Nome = 'Pisa'
AND (pa.SquadraVincitrice = (SELECT CodiceS FROM Squadra WHERE Nome = 'Empoli' )
OR pa.SquadraPerdente = (SELECT CodiceS FROM Squadra WHERE Nome = 'Empoli' ));
```

- vi. Per ogni giocatore che ha partecipato almeno ad una partita, trovare il nome, il nome della squadra, il numero totale di partite giocate e di goal segnati.

```
SELECT g.Nome, s.Nome, COUNT(*), SUM(p.GoalSegnati)
FROM Squadra s, Giocatore g, Partecipazione p
WHERE s.CodiceS = g.CodiceS AND p.CodiceGiocatore = g.CodiceG
GROUP BY g.Nome, g.CodiceG, s.Nome;
```

- vii. (Opzionale) Per ogni squadra che ha vinto almeno quattro partite, il nome della squadra ed il numero di giocatori della squadra.

```
SELECT    s.Nome, COUNT(DISTINCT CodiceG) AS NGiocatori
FROM      Squadra s, Giocatore g, Partita pa
WHERE     s.CodiceS = g.CodiceS AND pa.SquadraVincitrice = s.SquadraC
GROUP BY  s.CodiceS, s.Nome
HAVING    COUNT(DISTINCT CodiceP) >= 4;
```

```
SELECT    s.Nome, COUNT(*) AS NGiocatori
FROM      Squadra s, Giocatore g
WHERE     s.CodiceS = g.CodiceS AND
          s.CodiceS IN
          (SELECT  ss.CodiceS
           FROM    Squadra ss, Partita pa
           WHERE   pa.SquadraVincitrice = ss.CodiceS
           GROUP BY ss.CodiceS
           HAVING  COUNT(*) >= 4)
GROUP BY  s.Nome ;
```