

## Prova in Itinere per l'esame di Laboratorio di Basi di Dati 6 Dicembre 2007

Sia dato il seguente schema di base di dati:

STUDENTI(Matricola, Nome, Cognome, DataNascita, Indirizzo, Telefono, Cod\_CorsodiLaurea)  
DOCENTI(Codice\_Docente, Nome, Cognome)  
CORSO\_DI\_LAUREA(Codice\_CdL, Nome, Cod\_Fac)  
INSEGNAMENTO(Codice\_Insegnamento, Nome, Cod\_Docente, Codice\_CdL)  
ESAMI(MatrStud, Voto, Data, Lode, Cod\_Ins)  
FACOLTA(Codice\_Fac, Nome, Indirizzo\_Sede, N\_Telefonico)

1. Scrivere le istruzioni SQL per creare lo schema sopra descritto, specificando i vincoli interrelazionali e intrarelazionali fra le tabelle.

Scrivere le query SQL per trovare:

2. Il numero di studenti di ingegneria che hanno sostenuto l'esame di *Analisi I* con un voto superiore al 25.
3. I cognomi, i nomi e il numero di corsi insegnati dai docenti che insegnano più di due corsi.
4. Il numero di insegnamenti di ogni corso di laurea.
5. L'elenco degli studenti di *Informatica*, ordinati per cognome, che hanno una media dei voti superiore al 28.
6. Il corso di laurea con il massimo numero di insegnamenti.
7. L'elenco degli studenti iscritti alla facoltà di *Medicina Veterinaria*, in ordine decrescente del numero degli esami sostenuti.
8. Lo studente di Ingegneria che ha sostenuto il massimo numero di esami.
9. Elencare tutti i docenti che insegnano corsi della facoltà di Ingegneria insieme a quelli che insegnano corsi nel cui nome è contenuta la parola *Programmazione*.
10. Gli studenti che hanno sostenuto sia l'esame di *Diritto penale* che quello di *Diritto civile* senza utilizzare operatori booleani.
11. L'elenco dei docenti e il numero degli esami superati nei rispettivi corsi, in ordine decrescente (di numero di esami).
12. Creare una vista che contiene l'elenco dei docenti che tengono insegnamenti del corso di laurea in *Informatica*.

## SOLUZIONI:

1. Scrivere le istruzioni SQL per creare lo schema sopra descritto, specificando i vincoli interrelazionali e intrarelazionali fra le tabelle.

```
create table facolta(  
  codice_fac          char(6)          primary key,  
  nome                varchar(20)      not null,  
  indirizzo_sede      varchar(50)      not null,  
  n_telefonico        char(12)         unique  
)  
  
create table corso_di_laurea(  
  codice_cdl          char(6)          primary key,  
  nome                varchar(20)      not null,  
  cod_fac             references facolta(codice_fac)  
)  
  
create table docenti(  
  codice_docente      char(6)          primary key,  
  nome                varchar(20)      not null,  
  cognome             varchar(20)      not null  
)  
  
create table studenti(  
  matricola           char(6)          primary key,  
  nome                varchar(20)      not null,  
  cognome             varchar(20)      not null,  
  datanascita         date,  
  indirizzo            varchar(50),  
  telefono            char(12),  
  cod_corsodilaurea   references corso_di_laurea(codice_cdl)  
)  
  
create table insegnamento(  
  codice_insegnamento char(6)          primary key,  
  nome                varchar(20)      not null unique,  
  cod_docente         references docenti(codice_docente),  
  codice_cdl          references corso_di_laurea(codice_cdl)  
)  
  
create table esami(  
  matrstud            references studenti(matricola),  
  voto                numeric(2)        check(voto>17 and voto<31),  
  data                date,  
  lode                numeric(1)        default 0,  
  cod_ins             references insegnamento(codice_insegnamento),  
  check(voto<>30 and lode=0),  
  primary key(matrstud,cod_ins)  
)
```

2. Il numero di studenti di ingegneria che hanno sostenuto l'esame di *Analisi1* con un voto superiore al 25.

```
select count(matrstud)
from esami
where cod_ins in (select codice_insegnamento
                  from insegnamento
                  where nome = 'Analisi1'
                  and codice_cdl = (select codice_cdl
                                    from corso_di_laurea
                                    where cod_fac = (select codice_fac
                                                      from facolta
                                                      where nome = 'Ingegneria'
                                                      )
                                    )
                  )
and voto > 25
```

3. I cognomi, i nomi e il numero di corsi insegnati dai docenti che insegnano più di due corsi

```
create view num_insegnamenti(codice_docente, num_corsi) as
select codice_docente, count(codice_insegnamento) as num_corsi
from docenti D join insegnamento I on (D.codice_docente = I.cod_docente)
group by codice_docente

select nome, cognome, num_corsi
from docenti join num_insegnamenti using (codice_docente)
where num_corsi > 1
```

4. Il numero di insegnamenti di ogni corso di laurea.

```
select codice_cdl, count(codice_insegnamento) as num_insegnamenti
from insegnamento
group by codice_cdl
```

5. L'elenco degli studenti di *Informatica*, ordinati per cognome, che hanno una media dei voti superiore al 28.

```
create view medie(matricola, media) as
select matrstud, avg(voto) as media
from esami
group by matrstud

select *
from studenti
where matricola in (select matricola
                    from medie
                    where media > 28)
and cod_corsodilaurea = (select codice_cdl
                         from corso_di_laurea
                         where nome = 'Informatica')
order by cognome
```

6. Il corso di laurea con il massimo numero di insegnamenti.

```
create view num_insegnamenti_cdl(cdl, num_ins) as
select codice_cdl, count(codice_insegnamento) as num_ins
from insegnamento
group by codice_cdl

select cdl
from num_insegnamenti_cdl
where num_ins >= all (select num_ins
                     from num_insegnamenti_cdl)
```

7. L'elenco degli studenti iscritti alla facoltà di *Medicina Veterinaria*, in ordine decrescente del numero degli esami sostenuti.

```
select matrstud, count(voto) as esami_sost
from esami
where matrstud in (select matricola
                  from studenti
                  where cod_corsodilaurea =
                    (select codice_cdl
                     from corso_di_laurea
                     where cod_fac =
                       (select cod_fac
                        from facolta
                        where nome = 'Medicina Veterinaria'
                       )
                    )
                 )
group by matrstud
order by count(matrstud) desc
```

8. Lo studente di Ingegneria che ha sostenuto il massimo numero di esami.

```
create view n_esami_sostenuti(matricola, n_esami) as
select matrstud, count(voto) as n_esami
from esami
group by matrstud

select matricola
from n_esami_sostenuti
where matricola in (select matricola
                  from studenti
                  where cod_corsodilaurea in
                    (select codice_cdl
                     from corso_di_laurea
                     where cod_fac in (select codice_fac
                                       from facolta
                                       where nome = 'Ingegneria')
                    )
                 )
and n_esami = (select max(n_esami)
              from n_esami_sostenuti)
```

9. Elencare tutti i docenti che insegnano corsi della facoltà di Ingegneria insieme a quelli che insegnano corsi nel cui nome è contenuta la parola *Programmazione*.

```
select cod_docente
from insegnamento
where codice_cdl in (select codice_cdl
                     from corso_di_laurea
                     where cod_fac = (select codice_fac
                                     from facolta
                                     where nome = 'Ingegneria'
                                    )
                    )
UNION
select cod_docente
from insegnamento
where lower(nome) like '%programmazione%'
```

10. Gli studenti che hanno sostenuto sia l'esame di *Diritto penale* che quello di *Diritto civile* senza utilizzare operatori booleani.

```
select E1.matrstud
from esami E1, esami E2
where E1.matrstud = E2.matrstud
and E1.cod_ins <> E2.cod_ins
and E1.cod_ins = (select codice_insegnamento
                  from insegnamento
                  where nome = 'Diritto penale')
and E2.cod_ins = (select codice_insegnamento
                  from insegnamento
                  where nome = 'Diritto civile')
```

11. L'elenco dei docenti e il numero degli esami superati nei rispettivi corsi, in ordine decrescente (di numero di esami).

```
create view n_promossi(insegnamento, promossi) as
select cod_ins, count(matrstud) as promossi
from esami
group by cod_ins

select cod_docente, promossi
from insegnamento, n_promossi
where codice_insegnamento = insegnamento
order by promossi desc
```

12. Creare una vista che contiene l'elenco dei docenti che tengono insegnamenti del corso di laurea in *Informatica*.

[illegible]