

**BASI DI DATI 1 - Laboratorio**  
**ESERCITAZIONE 5 - Soluzione**

**1) Definizione dello schema e delle tabelle**

```
create schema prestiti;
```

```
create table prestiti.cliente  
( IDcliente char(3) primary key,  
  cognome varchar(20) not null,  
  nome varchar(20) not null,  
  citta_residenza varchar(20)  
);
```

```
create table prestiti.filiale  
( IDfiliale char(3) primary key,  
  importo_max decimal(10,2) not null,  
  citta varchar(20)  
);
```

```
create table prestiti.prestito  
( IDprestito char(3) primary key,  
  IDfiliale char(3) not null references prestiti.filiale on update cascade,  
  importo decimal(8,2) not null,  
  data_accensione date not null,  
  data_scadenza date not null,  
  check(data_scadenza > data_accensione)  
);
```

```
create table prestiti.accordato_a  
( IDprestito char(3) references prestiti.prestito on delete cascade on update cascade,  
  IDcliente char(3) references prestiti.cliente on delete cascade on update cascade,  
  primary key(IDprestito, IDcliente)  
);
```

**2) Inserimento dei dati**

```
insert into prestiti.cliente values('C01', 'Rossi', 'Andrea', 'Cagliari');
```

.....

```
insert into prestiti.filiale values('F01', 2000000.00, 'Milano');
```

.....

```
insert into prestiti.prestito values('P45', 'F03', 80000.00, '1992-02-01', '2012-01-31');
```

.....

```
insert into prestiti.accordato_a values('P45', 'C03');
```

.....

### 3) Interrogazioni

- a) *Selezionare i dati dei clienti a cui è stato accordato almeno un prestito da una filiale non situata nella città in cui risiedono, ordinati in senso crescente rispetto al cognome (prima) e al nome (poi).*

```
select IDcliente, cognome, nome, citta_residenza
  from prestiti.cliente natural join prestiti.accordato_a
        natural join prestiti.prestito
        natural join prestiti.filiale
 where citta_residenza <> citta
 order by cognome, nome;
```

- b) *Selezionare il cognome e il nome dei clienti a cui sono stati accordati almeno due prestiti.*

```
select cognome, nome
  from prestiti.cliente
 where IDcliente IN (select A1.IDcliente
                    from prestiti.accordato_a A1, prestiti.accordato_a A2
                    where A1.IDcliente = A2.IDcliente
                    and A1.IDprestito <> A2.IDprestito);
```

```
select cognome, nome
  from prestiti.cliente
 where IDcliente IN (select IDcliente
                    from prestiti.accordato_a
                    group by IDcliente
                    having count(*) > 1);
```

- c) *Selezionare i dati delle filiali che hanno concesso almeno due prestiti di importo superiore a 50000 euro.*

```
select * from prestiti.filiale
 where IDfiliale IN (select P1.IDfiliale
                    from prestiti.prestito P1, prestiti.prestito P2
                    where P1.importo > 50000 and P2.importo > 50000
                    and P1.IDfiliale = P2.IDfiliale
                    and P1.IDprestito <> P2.IDprestito);
```

```
select * from prestiti.filiale
 where IDfiliale IN (select IDfiliale from prestiti.prestito
                    where importo > 50000
                    group by IDfiliale
                    having count(*) > 1);
```

- d) *Selezionare i dati delle filiali che non hanno concesso alcun prestito tra il 01/01/2000 e il 31/12/2005.*

```
select * from prestiti.filiale
  where IDfiliale NOT IN
    (select IDfiliale from prestiti.prestito
     where data_accensione between '2000-01-01' and '2005-12-31');
```

- e) *Selezionare i dati delle filiali che hanno concesso prestiti esclusivamente a clienti residenti nella propria città.*

```
select * from prestiti.filiale
  where IDfiliale IN (select IDfiliale from prestiti.prestito)
    and IDfiliale NOT IN (select IDfiliale
                        from prestiti.filiale natural join prestiti.prestito
                        natural join prestiti.accordato_a
                        natural join prestiti.cliente
                        where citta <> citta_residenza);
```

- f) *Selezionare l'identificativo della filiale che complessivamente ha concesso in prestito la somma più elevata.*

```
select IDfiliale from prestiti.prestito
  group by IDfiliale
  having SUM(importo) >= ALL (select SUM(importo)
                            from prestiti.prestito
                            group by IDfiliale);
```

- g) *Selezionare gli identificativi delle filiali per cui il totale dei prestiti accordati supera il 50% dell'importo massimo che può essere concesso in prestito dalla filiale.*

```
select IDfiliale from prestiti.filiale F
  where 0.5*importo_max < (select SUM(importo)
                          from prestiti.prestito
                          where IDfiliale = F.IDfiliale);
```

#### **4) Aggiornamento**

- h) *Incrementare del 20% l'importo massimo (importo\_max) relativo alle filiali che hanno concesso almeno due prestiti.*

```
update prestiti.filiale
  set importo_max = 1.2 * importo_max
  where IDfiliale in (select IDfiliale
                    from prestiti.prestito
                    group by IDfiliale having count(*) > 1);
```

i) *Cancellare le filiali che non hanno concesso alcun prestito.*

delete from prestiti.filiale

where IDfiliale not in (select IDfiliale from prestiti.prestito);