

Esercizio 1:

Data la tabella:

Clienti				
Cognome	Nome	Città	Salario	Eta
Bianchi	Mario	Rimini	1000	20
Bianchi	Ettore	Milano	0	15
Casadei	Mario	Rimini	3000	35
Rossi	Mario	Bologna	1500	50
Rossi	Fabio	Firenze	8000	40
Bianchi	Ettore	Rimini	4500	25
Neri	Fabio	Arezzo	3500	35

Scrivere il codice SQL che permetta di:

1. Visualizzare il salario massimo dei clienti che abitano a Rimini.
2. Per ogni città, visualizzare il nome della città, il numero di abitanti e la loro età media.
3. Visualizzare il nome della città e l'età media, delle città con età media maggiore di 30 anni.

- Visualizzare il salario massimo dei clienti che abitano a Rimini.

```
SELECT MAX(Salario) AS Salario_massimo
FROM Clienti
WHERE Città = "Rimini";
```

- Per ogni città, visualizzare il nome della città, il numero di abitanti e la loro età media.

```
SELECT Città, COUNT(*) AS N_abitanti, AVG(Eta) as Eta_media,
FROM Clienti
GROUP BY Città; /* Per ogni città */
```

Esercizio 3:

Dato il seguente schema relazionale:

Dischi (CodDisco, Titolo, CodAutore*, dataPubblicazione, genere, ncopieVendute)

Autori (CodAutore, Nome)

Scrivere le seguenti query:

1. Visualizzare il nome dell'autore con il numero massimo di dischi venduti.
2. Visualizzare il numero di copie vendute per tutti i dischi (o il disco) pubblicato il 06/05/2024.

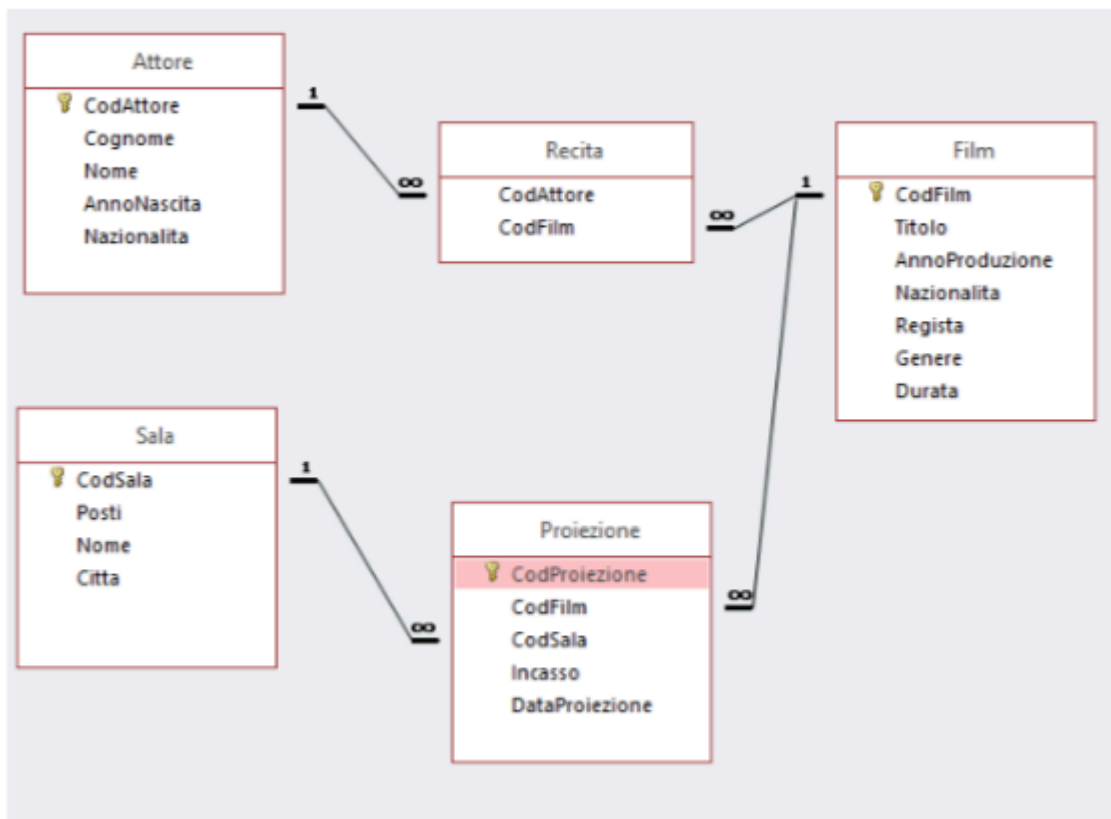
```
SELECT Nome, SUM(NCopieVendute) AS Numerocopie
FROM Autore A
JOIN Dischi D ON A.CodAutore = D.CodAutore;
GROUP BY Nome;
```

Per prendere tutto:

```
SELECT a.Nome
FROM Autori a
JOIN (
    SELECT CodAutore, SUM(ncopieVendute) AS TotaleVendite
    FROM Dischi
    GROUP BY CodAutore
) d ON a.CodAutore = d.CodAutore
WHERE d.TotaleVendite = (
    SELECT MAX(TotaleVendite)
    FROM (
        SELECT CodAutore, SUM(ncopieVendute) AS TotaleVendite
        FROM Dischi
        GROUP BY CodAutore
    ) x
)
```

- Visualizzare il numero di copie vendute per tutti i dischi (o il disco) pubblicato/i il 06/05/2024

```
SELECT NcopieVendute, Titolo
FROM Dischi
WHERE dataPubblicazione = "06/05/2024";
```



Dato lo schema relazionale, scrivi la definizione SQL per la creazione della base di dati.

```
CREATE TABLE Attori(  
    CodAttore INT AUTOINCREMENT PRIMARY KEY,  
    Cognome VARCHAR(50),  
    Nome VARCHAR(50),  
    AnnoNascita INT,  
    Nazionalita VARCHAR(50);  
);  
  
CREATE TABLE Recita(  
    FOREIGN KEY (CodAttore) REFERENCES Attore(CodAttore),  
    FOREIGN KEY (CodFilm) REFERENCES Film(CodFilm),  
);  
...
```

Date le seguenti tabelle:

```
STUDENTI (Matricola, Nome, Cognome, DataNascita, CorsoDiLaurea)  
ESAMI (CodEsame, NomeEsame, CFU, DocenteResponsabile)  
VOTI (Matricola*, CodEsame*, Data, Voto)
```

Scrivere le seguenti query SQL:

1. Elencare tutti gli studenti iscritti al corso di laurea "Informatica".
 2. Trovare il voto medio ottenuto in ogni esame.
 3. Visualizzare nome e cognome degli studenti che hanno superato l'esame di "Basi di Dati" con un voto superiore a 27.
 4. Contare quanti esami ha sostenuto ogni studente.
 5. Trovare gli studenti che hanno sostenuto tutti gli esami previsti.
- Elencare tutti gli studenti iscritti al corso di laurea "Informatica".

```
SELECT Nome, Cognome, Matricola  
FROM Studenti  
WHERE CorsoDiLaurea = "Informatica";
```

- Trovare il voto medio ottenuto in ogni esame.

```
SELECT AVG(Voto) AS Voto_Medio, CodEsame, NomeEsame  
FROM Voti V  
JOIN Esami E ON V.CodEsame = E.CodEsame  
GROUP BY CodEsame, NomeEsame;
```

- Trovare gli studenti che hanno sostenuto tutti gli esami previsti.

```
SELECT Nome, Cognome, Matricola
FROM Studenti S
JOIN Voti V ON S.Matricola = V.Matricola
WHERE VOTO IS NOT NULL AND VOTO >= 18 AND VOYO <= 32;
```