



Politecnico  
di Torino



# SQL: Esercizi Parte 3

---

# SQL: Esercizi Parte 3

---

# Esercizio #1a

---

GUIDA (CodGuida, Nome, Cognome, Nazionalità)

TIPO-VISITA (CodTipoVisita, Monumento, Durata, Città)

GRUPPO (CodGR, NumeroPartecipanti, Lingua)

VISITA-GUIDATA-EFFETTUATA (CodGR, Data, Oral, CodTipoVisita, CodGuida)

- a) Tra i monumenti per cui sono state effettuate almeno 10 visite guidate, visualizzare il monumento che è stato visitato complessivamente dal maggior numero di persone

# Esercizio #1a - Tabelle derivate

- a) Tra i monumenti per cui sono state effettuate almeno 10 visite guidate, visualizzare il monumento che è stato visitato complessivamente dal maggior numero di persone

```
SELECT Monumento
FROM TIPO-VISITA TP, VISITA-GUIDATA-EFFETTUATA VGE, GRUPPO G
WHERE TP.CodTipoVisita=VGE.CodTipoVisita AND
G.CodGR=VGE.CodGR
GROUP BY Monumento
HAVING COUNT(*) >=10 AND SUM(NumeroPartecipanti) =
    (SELECT MAX(TOTPart)
     FROM (SELECT SUM(NumeroPartecipanti) AS TOTPart
           FROM TIPO-VISITA TP, VISITA-GUIDATA-EFFETTUATA VGE, GRUPPO G
           WHERE TP.CodTipoVisita=VGE.CodTipoVisita
                AND G.CodGR=VGE.CodGR
           GROUP BY Monumento
           HAVING COUNT(*) >= 10) AS TOTM)
```

# Esercizio #1a – CTE Alternativa 1

- a) Tra i monumenti per cui sono state effettuate almeno 10 visite guidate, visualizzare il monumento che è stato visitato complessivamente dal maggior numero di persone

WITH

VISITE-MONUMENTO AS (

```
    SELECT Monumento, SUM(NumeroPartecipanti) As TotPart
    FROM VISITA-GUIDATA-EFFETTUATA V, TIPO-VISITA T, GRUPPO G
    WHERE V.CodTipoVisita=T.CodTipoVisita
    AND V.CodGR=G.CodGR
    GROUP BY Monumento
    HAVING COUNT(*) >= 10),
```

MAXVISITE AS (

```
    SELECT MAX(TotPart) AS MaxTotPart
    FROM VISITE-MONUMENTO )
```

SELECT Monumento

FROM VISITE-MONUMENTO, MAXVISITE

WHERE TotPart = MaxTotPart;

# Esercizio #1a – CTE Alternativa 2

- a) Tra i monumenti per cui sono state effettuate almeno 10 visite guidate, visualizzare il monumento che è stato visitato complessivamente dal maggior numero di persone

WITH

VISITE-MONUMENTO AS (

```
    SELECT Monumento, SUM(NumeroPartecipanti) As TotPart
    FROM VISITA-GUIDATA-EFFETTUATA V, TIPO-VISITA T, GRUPPO G
    WHERE V.CodTipoVisita=T.CodTipoVisita
    AND V.CodGR=G.CodGR
    GROUP BY Monumento
    HAVING COUNT(*) >= 10)
```

SELECT Monumento

FROM VISITE-MONUMENTO

WHERE TotPart = (

```
    SELECT MAX(TotPart) AS MaxTotPart
    FROM VISITE-MONUMENTO);
```

# Esercizio #2b

---

RAGAZZO(CodFiscale, Nome, Cognome, DataNascita, CittàResidenza)

ATTIVITA'(CodAttività, NomeA, Descrizione, Categoria)

CAMPO-ESTIVO(CodCampo, NomeCampo, Città)

ISCRIZIONE-PER-ATTIVITA'-IN-CAMPO-ESTIVO(CodFiscale, CodAttività, CodCampo, DataIscrizione)

b) Visualizzare il nome e cognome del ragazzo che ha partecipato al maggior numero di campi estivi per l'attività della categoria «Tennis».

# Esercizio #2b

RAGAZZO(CodFiscale, Nome, Cognome, DataNascita, CittàResidenza)

ATTIVITA'(CodAttività, NomeA, Descrizione, Categoria)

CAMPO-ESTIVO(CodCampo, NomeCampo, Città)

ISCRIZIONE-PER-ATTIVITA'-IN-CAMPO-ESTIVO(CodFiscale, CodAttività, CodCampo, DataIscrizione)

b) Visualizzare il nome e cognome del ragazzo che ha partecipato al maggior numero di campi estivi per l'attività della categoria «Tennis».

```
SELECT Nome,Cognome
```

```
FROM R, A, IPA
```

```
WHERE Categoria='Tennis'AND R.CodFiscale=IPA.CodFiscale
```

```
AND A.CodAttività=IPA.CodAttività
```

```
GROUP BY R.CodFiscale, Nome, Cognome
```

```
HAVING COUNT(DISTINCT CodCampo) = (SELECT MAX(NUMCAMPI)
```

```
FROM (SELECT COUNT(DISTINCT CodCampo) AS NUMCAMPI
```

```
FROM IPA, A
```

```
WHERE Categoria='Tennis' AND
```

```
A.CodAttività=IPA.CodAttività
```

```
GROUP BYCodFiscale) AS NUMCAMPIPERRAGAZZO)
```



## Esercizio #2b - CTE

b) Visualizzare il nome e cognome del ragazzo che ha partecipato al maggior numero di campi estivi per l'attività della categoria «Tennis».

```
WITH CAMPI_RAGAZZO AS (  
    SELECT CodFiscale, Nome, Cognome,  
    COUNT(DISTINCT CodCampo) AS NumCampi  
    FROM ISCRIZIONE I, ATTIVITA A, RAGAZZO R  
    WHERE I.CodAttivita=A.CodAttivita  
    AND R.CodFiscale=I.CodFiscale  
    AND Categoria='Tennis'  
    GROUP BY CodFiscale, Nome, Cognome )
```

```
SELECT Nome, Cognome  
FROM CAMPI_RAGAZZO  
WHERE NumCampi = (SELECT MAX(NumCampi)  
    FROM CAMPI_RAGAZZO)
```

# Esercizio #3

---

OFFICINA (OID, Nome, Indirizzo, Città)

VEICOLO (Targa, Modello, Marca, Categoria, Alimentazione, AnnoImmatricolazione, CodFiscale)

CLIENTE (CodFiscale, Nome, Cognome, DataNascita, Indirizzo, Città)

REVISIONE (Targa, OID, Data, Costo)

Per le officine che hanno effettuato revisioni di almeno 200 veicoli diversi intestati a persone nate tra il 1970 e il 1980, visualizzare il nome e l'indirizzo dell'officina che ha eseguito il maggior numero di revisioni (considerando tutte le revisioni effettuate) tra le officine ubicate nella stessa città. Visualizzare anche il costo totale delle revisioni effettuate dall'officina e il numero di modelli di veicoli diversi revisionati.

# Esercizio #3 – Tabelle derivate

```
SELECT (O.OID), Nome, Indirizzo, Città, SUM(Costo), COUNT(DISTINCT Modello)
FROM OFFICINA O, REVISIONE R, VEICOLO V
WHERE O.OID = R.OID AND V.Targa = R.Targa
AND OID IN (SELECT OID
            FROM REVISIONE R, VEICOLO V, CLIENTE C
            WHERE R.Targa = V.Targa AND V.CodFiscale = C.CodFiscale
              AND DataNascita >= 01/01/1970 AND DataNascita <= 31/12/1980
            GROUP BY OID
            HAVING COUNT(DISTINCT Targa) >= 200)
GROUP BY O.OID, Nome, Indirizzo, O.Città
HAVING COUNT(*) = (SELECT MAX (Nrev)
                  FROM (SELECT Città, COUNT(*) as Nrev
                        FROM REVISIONE R3, OFFICINA O3
                        WHERE O3.OID = R3.OID
                        GROUP BY OID, Città) AS citta_rev
                  WHERE citta_rev.Città = O.Città)
```

# Esercizio #3 – CTE Alternativa 1

```
WITH Info_Officina AS (  
    SELECT Oid, Nome, Indirizzo, Citta, COUNT(*) AS NumRevisioni,  
           SUM(costo) CostoTot, COUNT (DISTINCT Modelli) NModelli  
    FROM REVISIONE R, OFFICINA O, VEICOLO V  
    WHERE R.Oid=O.Oid AND R.Targa=V.Targa  
    GROUP BY Oid, Nome, Indirizzo, Citta),  
  
    revCitta AS (  
        SELECT Citta, MAX(NumRevisioni) AS NumRevCitta  
        FROM Info_Officina  
        GROUP BY Citta)  
  
SELECT *  
FROM Info_Officina I, revCitta  
WHERE I.Citta=revCitta.Citta  
AND I.NumRevisioni=revCitta.NumRevCitta  
AND Oid IN (  
    SELECT Oid  
    FROM REVISIONE R, VEICOLO V, CLIENTE C  
    WHERE R.Targa= V.Targa AND V.CodFiscale=C.CodFiscale  
    AND DataNascita >= '01/01/1970' AND DataNascita <= '31/12/1980'  
    GROUP BY Oid  
    HAVING COUNT (DISTINCT Targa) >= 200 )
```

# Esercizio #3 – CTE Alternativa 2

```
WITH Info_Officina AS (  
  SELECT Oid, Nome, Indirizzo, Citta, COUNT(*) AS NumRevisioni,  
         SUM(costo) CostoTot, COUNT (DISTINCT Modelli) NModelli  
  FROM REVISIONE R, OFFICINA O, VEICOLO V  
  WHERE R.Oid=O.Oid AND R.Targa=V.Targa  
  GROUP BY Oid, Nome, Indirizzo, Citta),  
  
  SELECT *  
  FROM Info_Officina I1  
  WHERE Oid IN (  
    SELECT Oid  
    FROM REVISIONE R, VEICOLO V, CLIENTE C  
    WHERE R.Targa= V.Targa AND V.CodFiscale=C.CodFiscale  
    AND DataNascita >= '01/01/1970' AND DataNascita <= '31/12/1980'  
    GROUP BY Oid  
    HAVING COUNT (DISTINCT Targa) >= 200 )  
  AND NumRevisioni = (  
    SELECT MAX(NumRevisioni)  
    FROM Info_Officina I2  
    WHERE I2.Citta=I1.Citta)
```

# Esercizio #3 – CTE Alternativa 3

```
WITH Info_Officina AS (  
  SELECT Oid, Nome, Indirizzo, Citta, COUNT(*) AS NumRevisioni,  
         SUM(costo) CostoTot, COUNT (DISTINCT Modelli) NModelli  
  FROM REVISIONE R, OFFICINA O, VEICOLO V  
  WHERE R.Oid=O.Oid AND R.Targa=V.Targa  
  GROUP BY Oid, Nome, Indirizzo, Citta),  
  
  SELECT *  
  FROM Info_Officina  
  WHERE Oid IN (  
    SELECT Oid  
    FROM REVISIONE R, VEICOLO V, CLIENTE C  
    WHERE R.Targa= V.Targa AND V.CodFiscale=C.CodFiscale  
    AND DataNascita >= '01/01/1970' AND DataNascita <= '31/12/1980'  
    GROUP BY Oid  
    HAVING COUNT (DISTINCT Targa) >= 200 )  
  AND (Citta, NumRevisioni) = (  
    SELECT Citta, MAX(NumRevisioni)  
    FROM Info_Officina  
    GROUP BY Citta)
```

# Esercizio #4

---

TECNICO (Matricola, Nome, Cognome, DataNascita, Sesso, Tipo)

INTERVENTO (CodI, Nome, Descrizione, Costo\_orario)

STRUTTURA (CodS, Indirizzo, Città, Provincia, Regione, Tipologia)

EFFETTUA\_INTERVENTO (Matricola, CodI, Data, CodS, Durata)

Considerando solo le strutture situate nella provincia di Torino, visualizzare la data nel mese di Marzo 2022 in cui è stato effettuato complessivamente il maggior numero di interventi nelle strutture considerate.

# Esercizio #4 – Tabelle derivate

Considerando solo le strutture situate nella provincia di Torino, visualizzare la data nel mese di Marzo 2022 in cui è stato effettuato complessivamente il maggior numero di interventi nelle strutture considerate.

```
SELECT Data
FROM EFFETTUA_INTERVENTO E, STRUTTURA S
WHERE E.Data >= 01/03/2022 AND E.Data <= 31/03/2022 AND
      E.CodS = S.CodS AND S.Provincia = "Torino"
GROUP BY Data
HAVING COUNT (*) =
(SELECT MAX(NumeroInterventi)
 FROM (SELECT COUNT(*) as NumeroInterventi
       FROM EFFETTUA_INTERVENTO E1, STRUTTURA S1
       WHERE E1.CodS = S1.CodS AND S1.Provincia = "Torino"
 AND T1.Data >= 01/03/2022 AND T1.Data <= 31/03/2022
       GROUP BY Data)
);
```



# Esercizio #4 - CTE

Considerando solo le strutture situate nella provincia di Torino, visualizzare la data nel mese di Marzo 2022 in cui è stato effettuato complessivamente il maggior numero di interventi nelle strutture considerate.

```
WITH INTERV-GIORNO AS (  
  SELECT Data, COUNT(*) as NumeroInterventi  
    FROM EFFETTUA_INTERVENTO E1, STRUTTURA S1  
   WHERE E1.CodS = S1.CodS AND S1. Provincia = "Torino"  
   AND T1.Data >= 01/03/2022 AND T1.Data <= 31/03/2022  
  GROUP BY Data)
```

```
SELECT Data  
FROM IntervGiorno  
WHERE NumInterventi = (SELECT MAX(NumInterventi)  
                      FROM IntervGiorno)
```

# Esercizio #5

---

LUOGO (CodL, Nome, Città, Regione, CapienzaMax)

EVENTO (CodE, Titolo, Tipo)

EDIZIONE (CodE, Data, CodL, NumeroPartecipanti)

Tra gli eventi per cui sono state organizzate edizioni in almeno 3 città diverse, visualizzare il titolo dell'evento a cui ha complessivamente (considerando tutte le edizioni dell'evento) partecipato il maggior numero di persone.

# Esercizio #5 – Tabelle Derivate

```
SELECT E.CodE, Titolo
FROM EDIZIONE ED, EVENTO E
WHERE ED.CodE = E.CodE
AND E.CodE IN
(SELECT CodE
FROM EDIZIONE ED, LUOGO L
WHERE ED.CodL = L.CodL
GROUP BY E.CodE
HAVING COUNT (DISTINCT Città) >=3)
GROUP BY E.CodE, Titolo
HAVING SUM(NumeroPartecipanti)=
SELECT MAX(TotPart)
FROM (SELECT SUM(NumeroPartecipanti) As TotPart
      FROM EDIZIONE ED
      WHERE E.CodE IN
      (SELECT CodE
      FROM EDIZIONE ED, LUOGO L
      WHERE ED.CodL = L.CodL
      GROUP BY E.CodE
      HAVING COUNT (DISTINCT Città) >=3)
      GROUP BY E.CodE)AS PartEvento;
```

# Esercizio #5 – CTE Alternativa 1

---

```
WITH PART-EVENTO AS
(SELECT E.CodE, Titolo, SUM(NumeroPartecipanti) As TotPart
    FROM EDIZIONE ED, EVENTO E
    WHERE EE.CodE=E.CodE
    E.CodE IN (SELECT CodE
    FROM EDIZIONE ED, LUOGO L
    WHERE ED.CodL = L.CodL
    GROUP BY E.CodE
    HAVING COUNT (DISTINCT Città) >=3)
    GROUP BY E.CodE, Titolo)
```

```
SELECT CodE, Titolo
FROM PART-EVENTO
WHERE TotPart=
(SELECT MAX(TotPart)
FROM PART-EVENTO)
```

# Esercizio #5 – CTE Alternativa 2

```
WITH EDIZIONI-CITTA AS
(SELECT CodE
 FROM EDIZIONE ED, LUOGO L
 WHERE ED.CodL = L.CodL
 GROUP BY E.CodE
 HAVING COUNT (DISTINCT Città) >=3),

PART-EVENTO AS
(SELECT E.CodE, Titolo, SUM(NumeroPartecipanti) As TotPart
      FROM EDIZIONE ED, EVENTO E
      WHERE EE.CodE=E.CodE
      E.CodE IN (SELECT CodE FROM EDIZIONI-CITTA)
      GROUP BY E.CodE, Titolo)

SELECT CodE, Titolo
FROM PART-EVENTO
WHERE TotPart=
      (SELECT MAX(TotPart)
       FROM PART-EVENTO)
```

# Esercizio #6

---

FILM (CodF, Titolo, Data\_uscita, Genere, DurataMinuti)

CINEMA (CodC, Nome, Indirizzo, Città)

SALA (CodC, NumeroSala, Capienza)

PROIEZIONE (CodC, NumeroSala, Data, OraInizio, OraFine, CodF)

Visualizzare il titolo di ciascun film che ha una durata inferiore della durata media dei film appartenenti allo stesso genere, e che è stato proiettato un numero di volte maggiore del numero medio di proiezioni dei film appartenenti allo stesso genere.

# Esercizio #6 – Tabelle Derivate

```
SELECT Titolo
FROM FILM F, PROIEZIONE P
WHERE P.CodF=F.CodF
AND Durata <
    (SELECT AVG(Durata)
     FROM FILM F2
     WHERE F2.Genere=F.Genere)
GROUP BY F.CodF, Titolo, Genere
HAVING COUNT(*) > (SELECT AVG(N)
                   FROM (SELECT Genere, COUNT (*) AS N
                        FROM PROIEZIONE P2, FILM F3
                        WHERE P2.CodF=F3.CodF
                        GROUP BY F3.CodF, Genere) AS PG
                   WHERE PG.Genere=F.Genere)
```

# Esercizio #6 – CTE

---

WITH PROIEZIONI-FILM AS

```
(SELECT F.CodF, Titolo, Genere, COUNT (*) AS N
FROM PROIEZIONE P, FILM F
WHERE P.CodF=F.CodF
GROUP BY F.CodF, Genere)
```

DURATA-GENERE AS

```
(SELECT Genere, AVG(Durata) AS DurataMedia
FROM FILM F
GROUP BY Genere)
```

PROIEZIONI-GENERE AS (

```
SELECT Genere, AVG(N) AS MediaGenere
FROM PROIEZIONI-FILM
GROUP BY Genere)
```

SELECT Titolo

```
FROM PROIEZIONI-FILM P, DURATA-GENERE D, PROIEZIONI-GENERE G
WHERE D.Genere=F.Genere AND G.Genere=P.Genere AND Durata < DurataMedia AND N > MediaGenere
```