

Esercizi:

1) Utilizzo di DISTINCT e WHERE

Elencare, senza ripetizioni, tutte le regioni (Region) dei paesi che appartengono al continente (Continent) 'Europe'.

2) Combinazione di WHERE, ORDER BY

Elencare i nomi (Name) e la popolazione (Population) delle città (City) degli Stati Uniti (CountryCode = 'USA') che hanno una popolazione superiore a 1.000.000 abitanti, ordinando i risultati dalla città più popolosa alla meno popolosa.

3) GROUP BY con funzioni di aggregazione

Mostrare per ogni continente (Continent) presente nella tabella Country:

- **Il numero totale di paesi appartenenti a ciascun continente.**
- **La popolazione totale del continente.**

Ordinare il risultato per popolazione totale in ordine decrescente.

Esercizi:

Esercizio sui comandi SQL: GROUP BY, ORDER BY e INSERT INTO

Si consideri un database che contiene informazioni su una libreria. Nel database è presente una tabella chiamata Libri con la seguente struttura:

Libri (
 id INT PRIMARY KEY,
 titolo VARCHAR(100),
 autore VARCHAR(100),
 genere VARCHAR(50),
 prezzo DECIMAL(5,2),
 anno_publicazione INT
)

1) Inserimento dati (INSERT INTO)

Inserire almeno 6 nuovi libri nella tabella Libri usando il comando SQL INSERT INTO.

I libri devono appartenere a generi e autori diversi, ed essere pubblicati in anni differenti.

2) Aggregazione e raggruppamento (GROUP BY)

Scrivere una query che, usando il comando GROUP BY, mostri per ogni genere:

- il numero totale di libri presenti;
- il prezzo medio dei libri appartenenti a quel genere.

La query dovrà restituire il risultato ordinato alfabeticamente per genere.

3) Ordinamento risultati (ORDER BY)

Scrivere una query che elenchi tutti i libri pubblicati dopo l'anno 2010 ordinati in modo decrescente per anno di pubblicazione e, in caso di anno uguale, in ordine crescente per prezzo.

Esercizio

Si consideri una tabella chiamata Vendite con la seguente struttura, almeno 20 elementi generati:

- 1. Vendite (**
- 2. id INT,**
- 3. prodotto VARCHAR(100),**
- 4. categoria VARCHAR(50),**
- 5. quantita INT,**
- 6. prezzo_unitario DECIMAL(6,2),**
- 7. data_vendita DATE**
- 8.)**

Scrivi le query SQL per rispondere alle seguenti richieste:

- **Totale vendite per categoria**

Visualizza, per ogni categoria, il numero totale di vendite effettuate.

- **Prezzo medio per categoria**

Mostra, per ogni categoria, il prezzo medio dei prodotti venduti.

- **Quantità totale venduta per ogni prodotto**

Mostra il totale delle quantità vendute (SUM) per ciascun prodotto.

- **Prezzo massimo e minimo venduto nella tabella**

Mostra il prezzo massimo e il prezzo minimo tra tutti i prodotti venduti.

- **Numero totale di righe nella tabella**

Conta quante vendite sono state registrate nella tabella Vendite.

- **I 5 prodotti più costosi (in base al prezzo_unitario)**

Elenca i 5 prodotti più costosi ordinati in modo decrescente rispetto al prezzo.

- **I 3 prodotti meno venduti per quantità totale**

Mostra i nomi dei 3 prodotti con la quantità totale più bassa venduta (usa SUM e LIMIT).

SQL: WILDCARD CHARACTERS

In italiano metacaratteri, sono dei caratteri che servono per sostituire uno o più caratteri in una stringa.

Vengono utilizzati nelle condizioni come WHERE e LIKE sono:

- **"%" - zero o più caratteri ("bl%" = bl, black, blue e blob);**
- **"_" - un singolo carattere ("h_t" = hot, hat e hit);**

Il resto dei metacaratteri hanno bisogno della funzione "REGEXP_LIKE(colonna, pattern)"

- **"[]" - ogni singolo carattere all'interno delle parentesi ("h[oa]t" = hot e hat);**
- **"^" - qualsiasi carattere non compreso tra parentesi ("h[^oa]t" = hit);**
- **"-" - qualsiasi carattere all'interno del range specificato ("c[a-b]t" = cat e cbt);**

I metacaratteri possono essere usati in combinazione come ad esempio '_r%' che rappresenta valori con almeno una r al secondo carattere.