

Raggruppare i record in output
Filtrare i raggruppamenti

AGENDA

- ☐ Operatori di aggregazione
- ☐ Group by
- ☐ Having

Molto spesso occorre restituire delle interrogazioni aggregate al fine di esporre valori sintetici.

Molto spesso occorre analizzare i dati ed esporre dati sintetici secondo diversi livelli di dettaglio.

Alla fine di questo modulo sarai in grado di:

- ✓ Utilizzare correttamente gli operatori di aggregazione
- ✓ Raggruppare i record
- ✓ Ottenere aggregazioni

FUNZIONI DI AGGREGAZIONE

Le funzioni di aggregazione esegue una determinata operazione matematica (conteggio, somma, media, ...) su più valori (ad esempio una colonna) e ne restituisce un valore singolo o sintetico.

In questo modulo, tratteremo gli operatori SUM, COUNT, AVG, MIN, MAX

GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione

Cosa significa??

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	UnitPrice
605	313.19
605	323.99
605	323.99



```
SELECT  
ProductID  
, AVG(UnitPrice) AS AvgUnitPrice  
FROM Sales  
GROUP BY ProductID
```



ProductID	AvgUnitPrice
605	320.39

GROUP BY: crea un gruppo per ogni valore univoco della group by list. In questo caso, la group by list contiene un solo campo cioè ProductID.

Di conseguenza, si ottiene un gruppo (codice prodotto 605) rispetto al quale viene aggregato con l'operatore matematico media, il campo UnitPrice. Lo scopo è ottenere il prezzo medio di vendita di ciascun prodotto (uno stesso prodotto può essere venduto n volte a prezzi differenti...)

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	UnitPrice
605	313.19
605	323.99
605	323.99
711	450.99
711	452.50



```
SELECT  
ProductID  
, AVG(UnitPrice) AS AvgUnitPrice  
FROM Sales  
GROUP BY ProductID
```



ProductID	AvgUnitPrice
605	320.39
711	451.75

Qual è il prezzo medio di vendita di ciascun prodotto?

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	UnitPrice
605	313.19
605	323.99
605	323.99
711	450.99
711	452.50



```
SELECT  
ProductID  
, MAX(UnitPrice) AS MaxUnitPrice  
FROM Sales  
GROUP BY ProductID
```



ProductID	MaxUnitPrice
605	323.99
711	452.50

Qual è/quale è stato il prezzo di vendita più alto per ciascun prodotto?

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	UnitPrice
605	313.19
605	323.99
605	323.99
711	450.99
711	452.50



```
SELECT  
ProductID  
, MIN(UnitPrice) AS MinUnitPrice  
FROM Sales  
GROUP BY ProductID
```



ProductID	MinUnitPrice
605	319.19
711	450.99

Qual è/quale è stato il minor prezzo a cui è stato venduto ciascun prodotto?

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	UnitPrice
605	313.19
605	323.99
605	323.99
711	450.99
711	452.50



```
SELECT  
ProductID  
, Count(ProductID) AS NoOrder  
FROM Sales  
GROUP BY ProductID
```



ProductID	NoOrder
605	3
711	2

Quante volte è stato venduto ciascun prodotto?

SQL: GROUP BY

La clausola **GROUP BY** consente di creare dei gruppi (o raggruppamenti) per ogni combinazione univoca dei campi indicati nella *group by list*.

La clausola **GROUP BY** è spesso utilizzata in combinazione alla funzioni di aggregazione.

ProductID	OrderDate	Sales
605	25/03/2024	626,38
605	25/03/2024	939,57
605	26/03/2024	313,19
711	26/03/2024	450,99
711	26/03/2024	905

```
SELECT
ProductID
, OrderDate
, SUM(Sales) AS TotalSales
FROM Sales
GROUP BY
ProductID
, OrderDate
```

ProductID	OrderDate	TotalSales
605	25/03/2024	1565,95
605	26/03/2024	313,19
711	26/03/2024	905

Qual è l'importo totale di vendita (Sales) giorno per giorno per ciascun prodotto?

SQL: GROUP BY

Ordine di esecuzione logica	Clausola	Espressione	Obiettivo
5	SELECT	<select list>	Definisce i campi che devono essere restituiti nel result set cioè l'output della query
1	FROM	<table source>	Consente di indicare la tabella o le tabelle (join, union) delle quali occorre interrogare i record
2	WHERE	<search condition>	Consente di filtrare le righe restituite dalla clausola FROM in base a uno o più condizioni di ricerca
3	GROUP BY	<group by list>	Consente di raggruppare i record restituiti dagli step precedenti per ogni combinazione univoca dei campi indicati nella group by list
4	HAVING	<search condition>	Consente di filtrare i gruppi restituiti dalla GROUP BY in base a una o più condizioni di ricerca
6	ORDER BY	<order by list>	Consente di applicare ordinamenti

SQL
LET'S
TAKE A
LOOK!



GRAZIE
Epicode