

[Home Page](#) [Novità](#) [Aiuto](#)

## *Il linguaggio SQL*

### *L'operatore UNION*

[http://www.vbsimple.fbi/database/db\\_08\\_10.htm](http://www.vbsimple.fbi/database/db_08_10.htm)

Difficoltà: 3 / 5

Si tratta di un operatore di unione verticale da utilizzare nelle selezioni, non supportato da tutti i database, che consente di specificare una selezione aggiuntiva ed un filtro opzionale per le righe duplicate. La sintassi generale è:

```
SELECT <campi>  
FROM <origine>  
UNION [ALL]  
SELECT <campi>  
FROM <origine>  
ORDER BY <campi>
```

Ogni riga della seconda selezione viene accodata alla prima nell'ordine dei campi specificati, in tal modo il risultato finale sarà la somma verticale delle righe della prima selezione con quelle della seconda; sono automaticamente escluse le righe duplicate. Il predicato opzionale **ALL** consente di includere anche quelle righe che all'unione risultano duplicate e quindi ottenere un set di risultati composto esattamente dal numero di righe della prima e della seconda selezione.

Date le due tabelle PERSONALE e DIPENDENTI esposte nell'[articolo precedente](#) l'unione verticale si espone come segue:

```
SELECT nome, cognome, ruolo FROM personale  
UNION  
SELECT nome, cognome, ruolo FROM dipendenti
```

Estrarrà 6 righe contenenti tutti i nominativi unici contenuti nelle due tabelle.

NOME	COGNOME	RUOLO
Silvana	Perugia	Consulente
Milena	Piconi	Responsabile Acq.
Marco	Tardi	Ragioniere
Pasquale	Catozzo	Manager
Maurizio	Maugeri	Usciere
Lucio	Capelli	Ragioniere

I 3 nominativi duplicati contenuti in entrambe le tabelle saranno esclusi; per includere tutte le righe, senza esclusione di quelle duplicate è necessario specificare il predicato **ALL**:

```
SELECT nome, cognome, ruolo FROM personale  
UNION ALL  
SELECT nome, cognome, ruolo FROM dipendenti
```

Estrarrà le 9 righe date dalla somma di 5 righe della tabella personale e di 4 righe della tabella dipendenti. È possibile ordinare il set di risultati finale semplicemente includendo la clausola [ORDER BY](#) in coda all'ultima selezione. Non sarà infatti possibile ordinare i risultati restituiti dalle singole selezioni ma soltanto tutti i dati restituiti dall'unione:

```
SELECT nome, cognome, ruolo FROM personale
UNION
SELECT nome, cognome, ruolo FROM dipendenti
ORDER BY ruolo
```

Viceversa è possibile raggruppare i risultati delle singole selezioni specificando la clausola [GROUP BY e HAVING](#) ma non sarà possibile applicare il raggruppamento all'insieme dato dall'unione delle due selezioni. Un'eventuale clausola GROUP BY aggiunta in fondo all'istruzione di unione sarà invece applicata all'ultima selezione soltanto:

```
SELECT ruolo FROM personale
GROUP BY ruolo
UNION ALL
SELECT ruolo FROM dipendenti
GROUP BY ruolo
```

Restituirà 7 righe corrispondenti ai 4 ruoli della tabella personale ed ai 3 ruoli della tabella dipendenti; se non avessimo incluso il predicato ALL l'unione avrebbe filtrato ed escluso automaticamente le righe duplicate. È tuttavia possibile aggirare questa limitazione [creando una vista \(VIEW\)](#) e quindi applicare il raggruppamento ai risultati della vista stessa.

Si ricorda che i campi esposti nell'unione vanno considerati in ordine anziché assunti per nome; ciò significa che una selezione del genere non restituisce i dati aspettati:

```
SELECT nome, cognome, ruolo FROM personale
UNION
SELECT cognome, nome, ruolo FROM dipendenti
```

NOME	COGNOME	RUOLO
Marco	Tardi	Ragioniere
Tardi	Marco	Ragioniere
Maugeri	Maurizio	Usciere
Maurizio	Maugeri	Usciere
Lucio	Capelli	Ragioniere
Capelli	Lucio	Ragioniere
Pasquale	Catozzo	Manager
Perugia	Silvana	Consulente
Milena	Piconi	Responsabile Acq.

Saranno infatti restituite sempre 9 righe (notare l'assenza del predicato ALL) contenenti le 5 righe della tabella personale e quattro righe composte dai dati del cognome all'interno della colonna del nome e dei dati del nome all'interno della colonna del cognome.

Sebbene non fosse stato finora accennato, è possibile unire più di due set di risultati separando ciascun insieme dal relativo operatore UNION:

```
SELECT <campi> FROM <origine>
UNION [ALL]
SELECT <campi>FROM <origine>
UNION [ALL]
SELECT <campi> FROM <origine>
...
```

[Eibia FBI](#)

25 Aprile 2004



[Torna all'indice della sezione Database](#)

---