

## 28 - Alias in SQL

1. Alias per Colonne
2. Alias per Tabelle
3. Alias per Funzioni di Aggregazione
4. Alias per Espressioni
5. Alias con JOIN e Subquery
6. Conclusioni

Gli alias sono uno strumento potente nelle query [SQL](#) che consentono di assegnare nomi temporanei o alternativi agli elementi all'interno di una query. Questo può migliorare la leggibilità delle query, semplificare i calcoli e rendere più comprensibili i risultati. In questa lezione, esploreremo l'uso degli alias, compresi quelli per colonne, tabelle e funzioni, fornendo esempi dettagliati.

### Alias per Colonne

Gli alias di colonna sono utilizzati per assegnare nomi alternativi alle colonne restituite da una query. Questi alias sono spesso utilizzati per migliorare la leggibilità dei risultati.

#### Esempio: Alias per Colonna

```
SELECT nome AS NomeCliente, cognome AS CognomeCliente  
FROM clienti;
```

### Alias per Tabelle

Gli alias per tabelle sono utilizzati per creare nomi abbreviati o più significativi per le tabelle utilizzate in una query. Questo è particolarmente utile quando si lavora con query complesse che coinvolgono più tabelle.

#### Esempio: Alias per Tabella

```
SELECT c.nome, o.data_ordine  
FROM clienti AS c  
JOIN ordini AS o ON c.id_cliente = o.id_cliente;
```

[Copia](#)

### Alias per Funzioni di Aggregazione

Gli alias per le funzioni di aggregazione possono rendere i risultati delle query più leggibili, specialmente quando si eseguono calcoli complessi.

### Esempio: Alias per Funzione di Aggregazione

```
SELECT AVG(prezzo) AS MediaPrezzo  
FROM prodotti;
```

## Alias per Espressioni

È possibile assegnare alias anche alle espressioni calcolate all'interno di una query.

### Esempio: Alias per Espressione

```
SELECT nome, cognome, anno_corrente - anno_nascita AS Età  
FROM utenti;
```

## Alias con JOIN e Subquery

Gli alias possono essere utilizzati con le operazioni di JOIN e con le subquery per semplificare la sintassi delle query complesse.

### Esempio: Alias con JOIN e Subquery

```
SELECT c.nome, o.data_ordine, dettagli.prezzo_unitario  
FROM clienti AS c  
JOIN ordini AS o ON c.id_cliente = o.id_cliente  
JOIN (  
    SELECT id_ordine, prezzo_unitario  
    FROM dettaglio_ordini  
) AS dettagli ON o.id_ordine = dettagli.id_ordine;
```

## Conclusioni

Gli alias sono uno strumento versatile che migliora la leggibilità e la comprensione delle query SQL. L'utilizzo appropriato degli alias può semplificare le operazioni di selezione, calcolo e manipolazione dei dati, rendendo più agevole l'interazione con i database.