

# SQL SELECT – UNION

Оператор SQL UNION використовується для об'єднання результатів двох або більше операторів SELECT без повернення дублікатів рядків.

Для коректного використання оператора UNION, кожен оператор SELECT має відповідати умовам:

- однакова кількість стовпців;
- однакова кількість виразів;
- однакові типи даних;
- однаковий порядок стовпців та виразів.

# SQL SELECT – UNION

The SQL UNION clause/operator is used to combine the results of two or more SELECT statements without returning any duplicate rows.

To use this UNION clause, each SELECT statement must have:

- The same number of columns selected.
- The same number of column expressions.
- The same data type.
- Have them in the same order.

# UNION syntax

```
SELECT column1 [, column2 ]
```

```
    FROM table1 [, table2 ]
```

```
    [WHERE condition]
```

```
UNION [ALL]
```

```
SELECT column1 [, column2 ]
```

```
    FROM table1 [, table2 ]
```

```
    [WHERE condition]
```

# SQL SELECT – UNION ALL

Оператор UNION ALL використовується для об'єднання результатів двох операторів SELECT, включаючи дублікати рядків.

Ті самі правила, які застосовуються до оператору UNION, застосовуються і до оператора UNION ALL.

The UNION ALL operator is used to combine the results of two SELECT statements including duplicate rows.

The same rules that apply to the UNION clause will apply to the UNION ALL operator.

# SQL SELECT – INTERSECT

Оператор SQL INTERSECT використовується для об'єднання двох операторів SELECT, але повертає рядки тільки з першого оператора SELECT, які ідентичні рядку у другому операторі SELECT. Це означає, що INTERSECT повертає лише загальні рядки, які повертаються двома операторами SELECT.

Як і оператор UNION, застосовуються ті ж правила, коли використовується оператор INTERSECT.

# SQL SELECT – INTERSECT

The SQL INTERSECT clause/operator is used to combine two SELECT statements, but returns rows only from the first SELECT statement that are identical to a row in the second SELECT statement. This means INTERSECT returns only common rows returned by the two SELECT statements.

Just as with the UNION operator, the same rules apply when using the INTERSECT operator.

# INTERSECT syntax

```
SELECT column1 [, column2 ]  
    FROM table1 [, table2 ]  
    [WHERE condition]
```

INTERSECT

```
SELECT column1 [, column2 ]  
    FROM table1 [, table2 ]  
    [WHERE condition]
```

# SQL SELECT – EXCEPT

Оператор SQL EXCEPT використовується для об'єднання двох операторів SELECT і повертає рядки з першого оператора SELECT, які не повертаються другим оператором SELECT. Це означає, що EXCEPT повертає лише рядки, які відсутні у результаті другого оператора SELECT.

Як і оператор UNION, застосовуються ті ж правила, коли використовується оператор EXCEPT.



# SQL SELECT – EXCEPT

The SQL EXCEPT clause/operator is used to combine two SELECT statements and returns rows from the first SELECT statement that are not returned by the second SELECT statement. This means EXCEPT returns only rows, which are not available in the second SELECT statement.

Just as with the UNION operator, the same rules apply when using the EXCEPT operator.

# EXCEPT syntax

```
SELECT column1 [, column2 ]  
    FROM table1 [, table2 ]  
    [WHERE condition]
```

EXCEPT

```
SELECT column1 [, column2 ]  
    FROM table1 [, table2 ]  
    [WHERE condition]
```

# SQL – INSERT

Вираз SQL INSERT INTO використовується для додавання нових рядків даних до таблиці в базі даних.

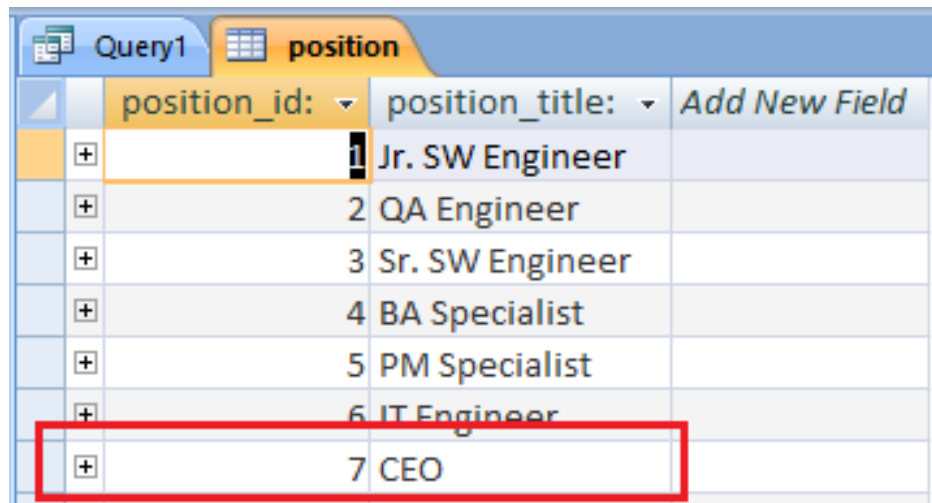
The SQL INSERT INTO Statement is used to add new rows of data to a table in the database.

```
INSERT INTO TABLE_NAME [(column1, column2,  
    column3, ... columnN)]
```

```
VALUES (value1, value2, value3, ... valueN);
```

# Query 07

INSERT INTO position VALUES (7, 'CEO');



	position_id: ▾	position_title: ▾	Add New Field
+	1	Jr. SW Engineer	
+	2	QA Engineer	
+	3	Sr. SW Engineer	
+	4	BA Specialist	
+	5	PM Specialist	
+	6	IT Engineer	
+	7	CEO	

INSERT INTO position (position\_id,  
position\_title) VALUES (7, 'CEO');

## Наповнення таблиці з використанням іншої таблиці / Populate one table using another table

Можна заповнити дані в таблиці за допомогою оператора `select`, який звертається до іншої таблиці; якщо інша таблиця має набір полів, які необхідні для заповнення першої таблиці.

You can populate the data into a table through the `select` statement over another table; provided the other table has a set of fields, which are required to populate the first table.

# Table population syntax

```
INSERT INTO first_table_name [(column1,  
    column2, ... columnN)]  
SELECT column1, column2, ...columnN  
FROM second_table_name  
[WHERE condition];
```

# SQL – UPDATE

Запит SQL UPDATE використовується для зміни існуючих записів у таблиці. Вираз WHERE можна використовувати з запитом UPDATE, щоб оновити вибрані рядки, інакше всі рядки будуть також змінені.

The SQL UPDATE Query is used to modify the existing records in a table. You can use the WHERE clause with the UPDATE query to update the selected rows, otherwise all the rows would be affected.

# SQL UPDATE syntax

```
UPDATE table_name  
SET  column1 = value1,  
      column2 = value2,  
      ...,  
      columnN = valueN  
WHERE [condition];
```

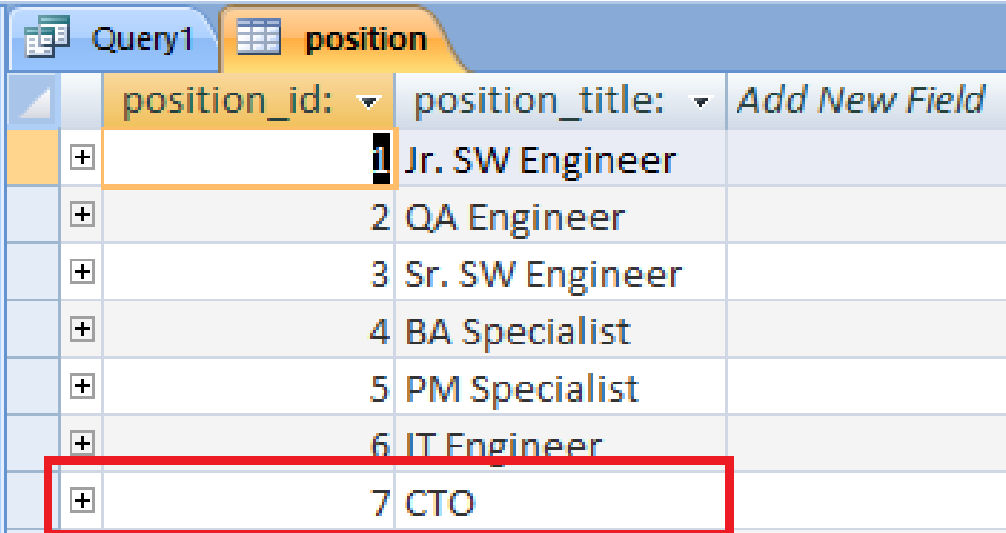


# Query 08

UPDATE position

SET position.position\_title = 'CTO'

WHERE position.position\_id = 7;



	position_id: ▾	position_title: ▾	Add New Field
+	1	Jr. SW Engineer	
+	2	QA Engineer	
+	3	Sr. SW Engineer	
+	4	BA Specialist	
+	5	PM Specialist	
+	6	IT Engineer	
+	7	CTO	

# SQL – DELETE

Запит SQL DELETE використовується для видалення існуючих записів з таблиці.

Вираз WHERE можна використовувати у запиті DELETE щоб видалити необхідні рядки, інакше всі записи будуть видалені.

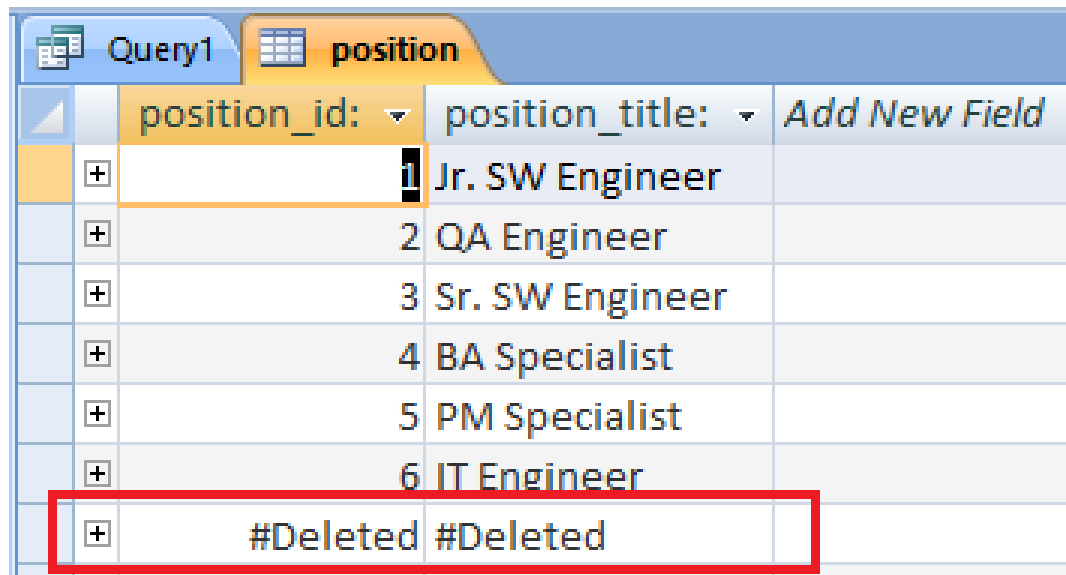
The SQL DELETE Query is used to delete the existing records from a table.

You can use the WHERE clause with a DELETE query to delete the selected rows, otherwise all the records would be deleted.

# SQL DELETE syntax + Query 09

DELETE FROM table\_name WHERE [condition];

DELETE FROM position WHERE position\_id = 7;



	position_id:	position_title:	Add New Field
+	1	Jr. SW Engineer	
+	2	QA Engineer	
+	3	Sr. SW Engineer	
+	4	BA Specialist	
+	5	PM Specialist	
+	6	IT Engineer	
+	#Deleted	#Deleted	