Use EXISTS in a WHERE clause to indicate that a subquery should return at least one row. You can also preface EXISTS with NOT, to indicate that a subquery should not return any rows.

For example, the following query returns a list of products that are found in at least one existing order:

SELECT \*

FROM [Products]

WHERE EXISTS

(SELECT \* FROM [Order Details]

WHERE [Order Details].[Product ID]=[Products].[ID]);

Using NOT EXISTS, the query returns a list of products that are not found in at least one existing order:

SELECT \*

FROM [Products]

WHERE NOT EXISTS

(SELECT \* FROM [Order Details]

WHERE [Order Details].[Product ID]=[Products].[ID]);

IN Use IN in a WHERE clause to verify that a value in the current row of the main query is part of the set that the subquery returns. You can also preface IN with NOT, to verify that a value in the current row of the main query is not part of the set that the subquery returns.

For example, the following query returns a list of orders (with order dates) that were processed by employees who are not sales representatives:

SELECT [Order ID], [Order Date]

FROM [Orders]

WHERE [Employee ID] IN

(SELECT [ID] FROM [Employees]

WHERE [Job Title]<>'Sales Representative');

By using NOT IN, you could write the same query this way:

SELECT [Order ID], [Order Date]

FROM [Orders]

WHERE [Employee ID] NOT IN

(SELECT [ID] FROM [Employees]

WHERE [Job Title]='Sales Representative');