

INTERSECT OPERATOR

The INTERSECT clause in SQL is used to combine two SELECT statements but the dataset returned by the INTERSECT statement will be the intersection of the data sets of the two SELECT statements. In simple words, the INTERSECT statement will return only those rows which will be common to both of the SELECT statements.



Syntax

```
SELECT column1 , column2 ...  
FROM table_names  
WHERE condition  
INTERSECT  
SELECT column1 , column2 ...  
FROM table_names  
WHERE condition
```

Let's assume that we have two table Customer Table and the Orders Table and we will perform some operations related to INTERSECT to understand better SQL intersect.

Customers Table

```
CREATE TABLE Customer  
(  
    CustomerID INT PRIMARY KEY,  
    FirstName VARCHAR(50) NOT NULL,  
    LastName VARCHAR(50) NOT NULL,  
    Email VARCHAR(100) UNIQUE NOT NULL,  
    Phone VARCHAR(20) NOT NULL,  
    Address VARCHAR(200) NOT NULL,  
    City VARCHAR(50) NOT NULL,  
    State VARCHAR(50) NOT NULL,  
    ZipCode VARCHAR(10) NOT NULL  
);
```

Output

CustomerID	FirstName	LastName	Email	Phone	Address	City	State	ZipCode
1	Shubham	Doe	shubham@example.com	555-1234	123 Main St	Anytown	CA	12345
2	Jane	Smith	janesmith@example.com	555-5678	456 Oak St	Sometown	NY	54321
3	Bob	Johnson	bojohnson@example.com	555-9012	789 Pine St	Other town	TX	67890

Orders Table

```
CREATE TABLE Orderss
(
    OrderID INT PRIMARY KEY,
    CustomerID INT NOT NULL,
    OrderDate DATE NOT NULL,
    ShipDate DATE,
    TotalAmount DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)
);
```

Let's insert some random data in the order table.

Output

OrderID	CustomerID	OrderDate	ShipDate	TotalAmount
1001	1	2023-05-01	2023-05-03	125.99
1002	2	2023-05-02	2023-05-04	95.5
1003	3	2023-05-03		230
1004	1	2023-05-05	2023-05-07	75.25
1005	2	2023-05-06	2023-05-08	150.75

Sample Queries

```
SELECT Customer.CustomerID, Customer.FirstName,
Customer.LastName
FROM Customer
LEFT JOIN Orderss ON Customer.CustomerID =
Orderss.CustomerID
UNION
SELECT Customer.CustomerID, Customer.FirstName,
Customer.LastName
FROM Customer
LEFT JOIN Orderss ON Customer.CustomerID =
Orderss.CustomerID
WHERE Orderss.OrderID IS NULL;
```

Output

CustomerID	FirstName	LastName
1	Shubham	Doe
2	Jane	Smith
3	Bob	Johnson

INTERSECT with BETWEEN Operator

As we have already discussed in the initial syntax, we can also use the INTERSECT operator along with some conditional operators. We can use the INTERSECT operator with the BETWEEN operator in SQL to find rows that fall within a specified range.

Let's assume that we have one table name Customer and another one as orderss.

Query

```
SELECT CustomerID, FirstName, LastName
FROM Customer
WHERE CustomerID BETWEEN 100 AND 200
INTERSECT
SELECT CustomerID, FirstName, LastName
FROM Customer
WHERE LastName BETWEEN 'A' AND 'M';
```

Output

OrderID	CustomerName	OrderDate
1	John Doe	2022
2	Jane Smith	2022
3	Bob Johnson	2022
4	Alice Brown	2022

INTERSECT With LIKE Operator

To match patterns in a string, use the LIKE operator. In SQL, the INTERSECT operator and the LIKE operator can also be used to find common rows that match a given pattern.

Example: Let's use the LIKE operator and the wildcard '%' to retrieve names that begin with the letter 'v' from the common names of both tables.

```
SELECT John
FROM Customer
WHERE common_name LIKE 'v%'
INTERSECT
SELECT Bob
FROM orderss
WHERE common_name LIKE 'v%';
```

Output

No output