

# Lab 4: Using Set Operators

## Retrieving Customer Addresses

---

Customers can have two kinds of address: a main office address and a shipping address. The accounts department wants to ensure that the main office address is always used for billing, and have asked you to write a query that clearly identifies the different types of address for each customer.

### Instructions

Write a query that retrieves the company name, first line of the street address, city, and a column named `AddressType` with the value 'Billing' for customers where the address type in the `SalesLT.CustomerAddress` table is 'Main Office'. Make sure to use the aliases provided, and default column names elsewhere.

```
SELECT c.CompanyName, a.AddressLine1, a.City, 'Billing' AS AddressType
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON ca.AddressID = a.AddressID
WHERE ca.AddressType = 'Main Office';
```

## Retrieving Customer Addresses (2)

---

The ideal solution to the previous exercise has been included in the sample code on the right. Can you adapt it slightly to generate a very similar result?

### Instructions

Adapt the query to retrieve the company name, first line of the street address, city, and a column named `AddressType` with the value 'Shipping' for customers where the address type in the `SalesLT.CustomerAddress` table is 'Shipping'. Make sure to use the aliases provided, and default column names elsewhere.

```
SELECT c.CompanyName, a.AddressLine1, a.City, 'Shipping' AS AddressType
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
```

```
ON ca.AddressID = a.AddressID
```

```
WHERE ca.AddressType = 'Shipping';
```

## Retrieving Customer Addresses (3)

---

The code on the right contains the two queries from the previous exercises.

### Instructions

Use `UNION ALL` to combine the results returned by the two queries to create a list of all customer addresses that is sorted by company name and then address type.

```
SELECT c.CompanyName, a.AddressLine1, a.City, 'Billing' AS AddressType
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON ca.AddressID = a.AddressID
WHERE ca.AddressType = 'Main Office'

UNION ALL

SELECT c.CompanyName, a.AddressLine1, a.City, 'Shipping' AS AddressType
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON ca.AddressID = a.AddressID
WHERE ca.AddressType = 'Shipping'

ORDER BY c.CompanyName, AddressType;
```

## Filtering Customer Addresses (1)

---

You have created a master list of all customer addresses, but now you have been asked to create filtered lists that show which customers have only a main office address, and which customers have both a main office and a shipping address.

## Instructions

Write a query that returns the company name of each company that appears in a table of customers with a 'Main Office' address, but not in a table of customers with a 'Shipping' address.

```
SELECT c.CompanyName
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON a.AddressID = ca.AddressID
WHERE ca.AddressType = 'Main Office'
EXCEPT
SELECT c.CompanyName
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON ca.AddressID = a.AddressID
WHERE ca.AddressType = 'Shipping'
ORDER BY c.CompanyName;
```

## Filtering Customer Addresses (2)

---

This exercise builds upon your work in the previous exercise.

## Instructions

Write a query that returns the company name of each company that appears in a table of customers with a 'Main Office' address, and also in a table of customers with a 'Shipping' address. Make sure to use the aliases provided, and default column names elsewhere.

```
SELECT c.CompanyName
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
```

```
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON a.AddressID = ca.AddressID
WHERE ca.AddressType = 'Main Office'
INTERSECT
SELECT c.CompanyName
FROM SalesLT.Customer AS c
JOIN SalesLT.CustomerAddress AS ca
ON c.CustomerID = ca.CustomerID
JOIN SalesLT.Address AS a
ON a.AddressID = ca.AddressID
WHERE ca.AddressType = 'Shipping'
ORDER BY c.CompanyName;
```