

# Left Joins

by Sophia

## WHAT'S COVERED

This tutorial explores using the left join and left outer join in two parts:

1. Introduction
2. Example

## 1. Introduction

In the prior tutorial, we covered the full outer join. We also have the ability to do a left join and a left outer join. The structure of the query looks like this:

```
SELECT <columnlist>
FROM <table1>
LEFT JOIN <table2> ON <table1>.<table1column1> = <table2>.<table2column1>;
```

## 2. Example

Let us revisit our data set with the representatives and departments again:

```
CREATE TABLE representative ( representative_id INT PRIMARY KEY, first_name VARCHAR (30) NOT NULL, last_name VARCHAR (30) NOT NULL );
```

```
CREATE TABLE department ( department_id INT PRIMARY KEY, department_name VARCHAR (100) NOT NULL, manager_id INT, constraint fk_manager FOREIGN KEY (manager_id,
```

```
INSERT INTO representative (representative_id, first_name, last_name)
VALUES (1, 'Bob','Evans'), (2, 'Tango','Rushmore'), (3, 'Danika','Arkane'), (4, 'Mac','Anderson');
```

```
INSERT INTO department (department_id, department_name,manager_id)
VALUES (1, 'Sales', 1), (2, 'Marketing', 3), (3, 'IT', 4), (4, 'Finance', null), (5, 'Support', null);
```

The left join clause is used to join the representative table with the department table. The first table listed in the FROM clause is considered to be the left table and the second table is considered to be the right table.

```
SELECT *
FROM representative
LEFT JOIN department ON representative.representative_id = department.manager_id;
```

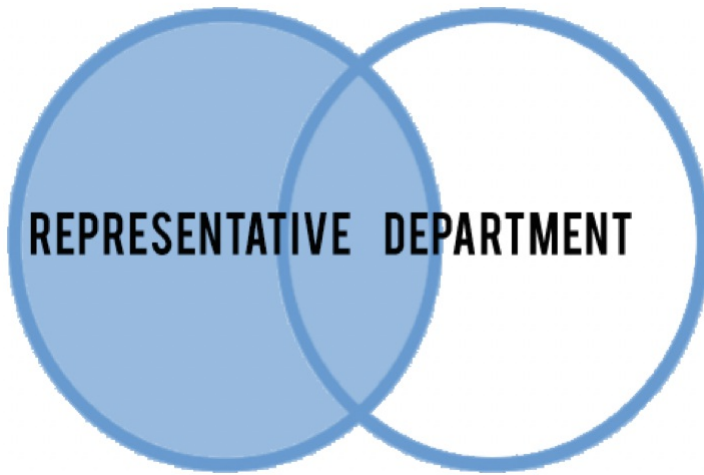
### Query Results

Row count: 4

representative_id	first_name	last_name	department_id	department_name	manager_id
1	Bob	Evans	1	Sales	1
3	Danika	Arkane	2	Marketing	3
4	Mac	Anderson	3	IT	4
2	Tango	Rushmore			

The left join starts to select data from the left table. It compares the representative\_id from the representative table with the manager\_id in the department table. If those values are equal, the left join creates a new row that contains the columns of both tables and adds the new row in the result set, as you can see in the first three rows returned in the image above. If the values are not equal, the left join also creates a new row containing columns from both tables, but fills in the columns of the right table (department) with a null value, as you can see in the 4th row.

The Venn diagram of the left join looks like the following:



We can also do a left outer join, where the left table does not have matching rows in the right table, with an added WHERE clause to check where the manager\_id is set to null. By doing so, this join does not return any rows that have matching data. The query would look like the following:

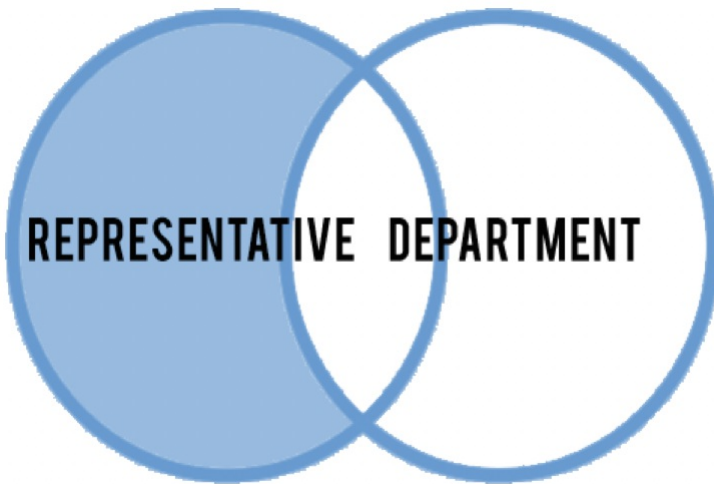
```
SELECT *
FROM representative
LEFT JOIN department ON representative.representative_id = department.manager_id
WHERE manager_id IS NULL;
```

#### Query Results

Row count: 1

representative_id	first_name	last_name	department_id	department_name	manager_id
2	Tango	Rushmore			

The following Venn diagram illustrates the left outer join:



Your turn! Open the SQL tool by clicking on the LAUNCH DATABASE button below. Then enter in one of the examples above and see how it works. Next, try your own choices for which columns you want the query to provide.



The left join and left outer join selects rows from one table that may or may not have matching rows in the second table.

Source: Authored by Vincent Tran