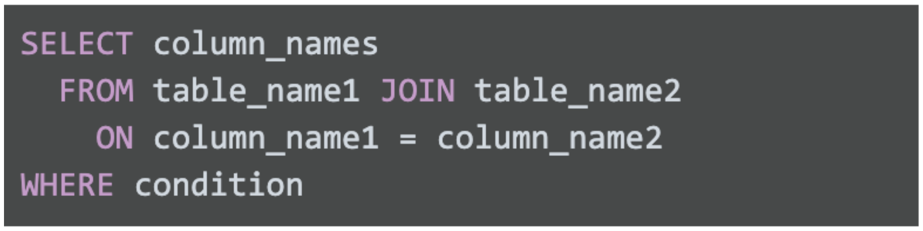
Using JOINS effectively

In this reading, you will review JOINs and find some resources that you can use to learn more about them. **JOINs** help you combine tables by using their matching or related values. JOINs use these values to identify relationships and corresponding values between tables.

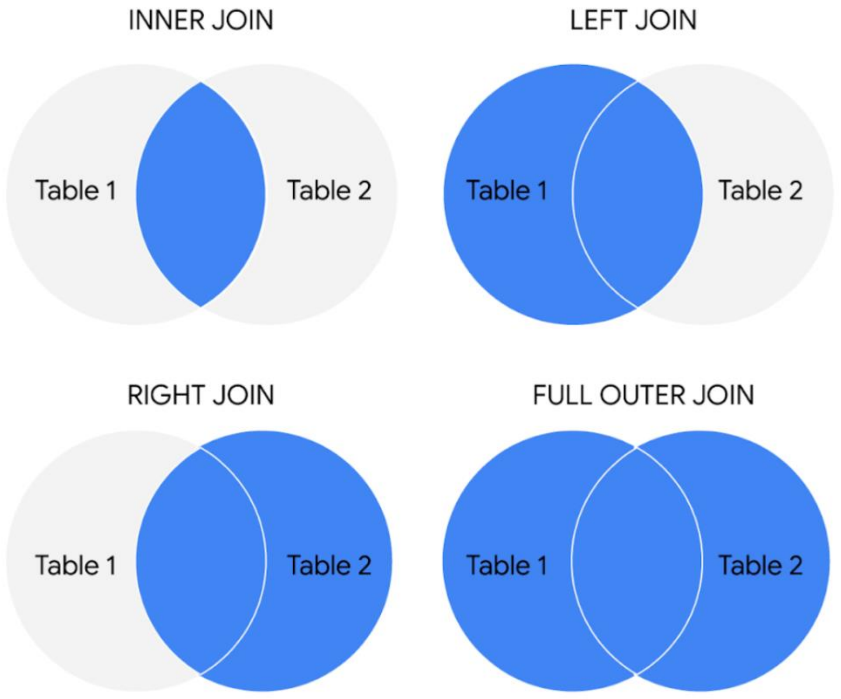
The JOIN syntax



```
SELECT column_names
  FROM table_name1 JOIN table_name2
    ON column_name1 = column_name2
WHERE condition
```

SELECT column_names FROM table_name1 JOIN table_name2 ON column_name1 = column_name2 WHERE condition

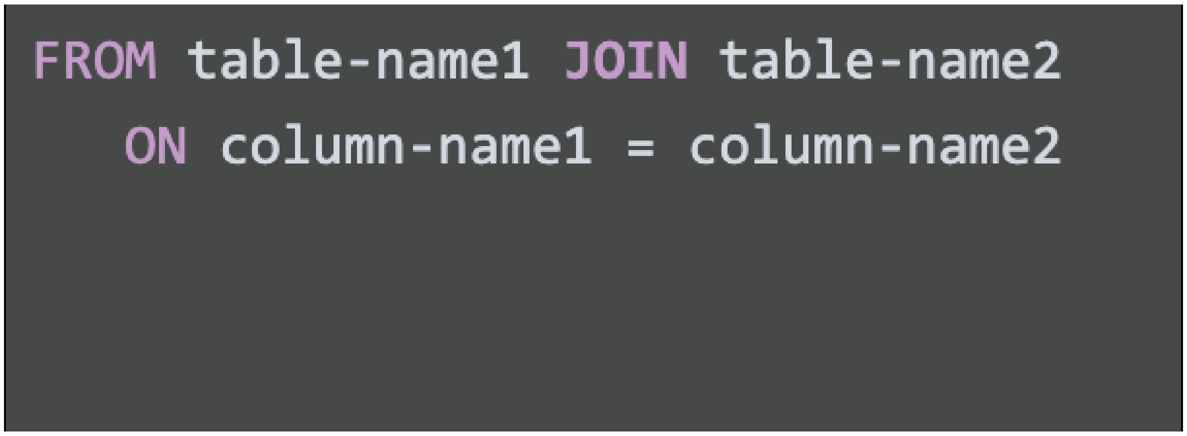
As you can see from the syntax, the JOIN command is included as a part of the FROM clause of the query. The first part of the statement identifies the two tables to be joined: “table-name1” and “table-name2”. The second part of the command identifies the column name in the first table that will be used to match the column name in the second table. There are four general ways in which we can conduct JOINs in SQL queries: INNER, LEFT, RIGHT, FULL OUTER. The INNER keyword is optional in your SQL query because it is the default as well as the most commonly used JOINs operation. Let’s review what these different JOINs queries do:



The circles represent left and right tables, and where they are joined is highlighted in blue

Type of JOIN

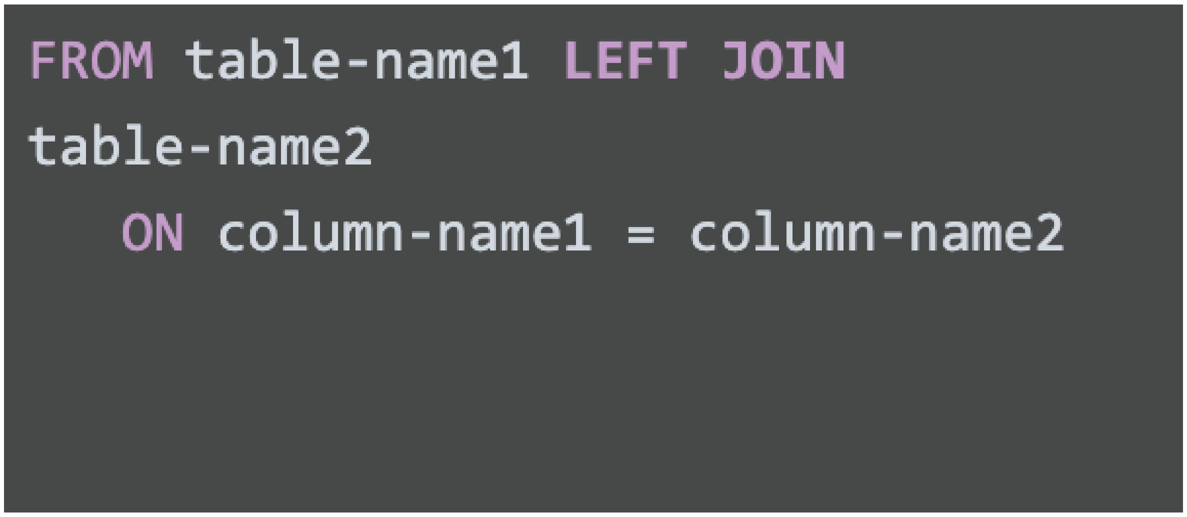
INNER JOIN: Returns records with matching values in both tables. You will use inner JOINs whenever you want to work with only matching values from both of the two tables. For example, you could use an INNER JOIN if you wanted to combine the data in a customer table with data from an order table.



```
FROM table-name1 JOIN table-name2
   ON column-name1 = column-name2
```

FROM table-name1 JOIN table-name2 ON column-name1 = column-name2

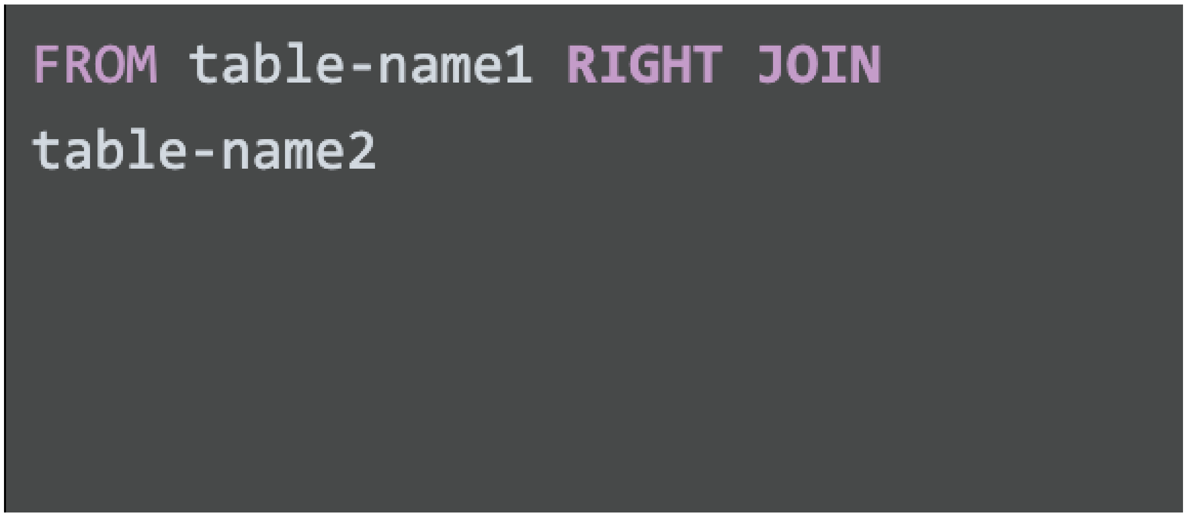
LEFT JOIN: Returns all the records from the left table and only the matching records from the right table. You will use LEFT JOINs whenever you need the entire first table and just the matching values from the second table. For example, if you wanted all of the values from the customer data, but only the matching order_id values from the Orders table.



```
FROM table-name1 LEFT JOIN
table-name2
   ON column-name1 = column-name2
```

FROM table-name1 LEFT JOIN table-name2 ON column-name1 = column-name2

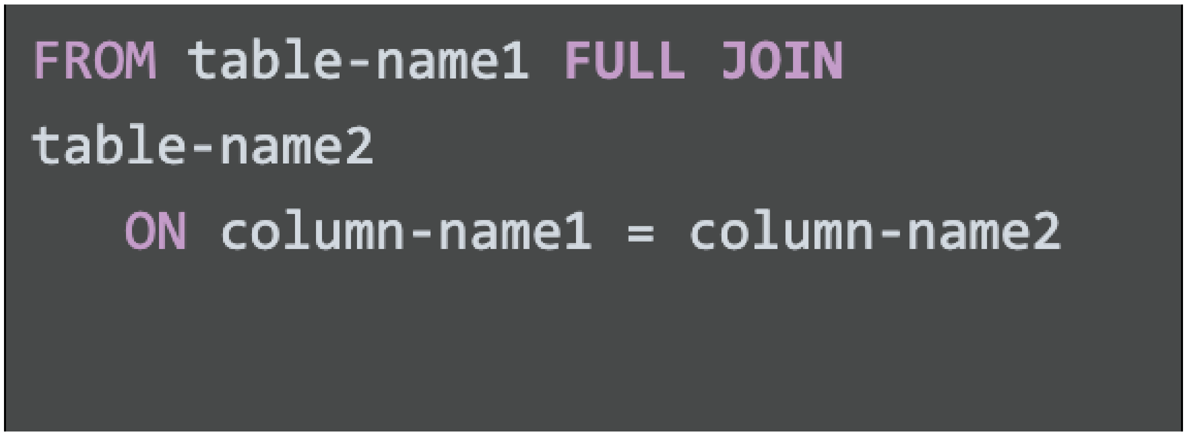
RIGHT JOIN: Returns all records from the right table and the matched records from the left table. You will use RIGHT JOINS when you want to pull all of the values from the second table, but only the matching values from the first. For example, if you wanted all of the order data from the Order table and the customer_id values from the Customer table.



```
FROM table-name1 RIGHT JOIN
table-name2
```

FROM table-name1 RIGHT JOIN table-name2

FULL OUTER JOIN: Returns all records when there is a match in left or right table records. You can combine tables this way, but remember: this can potentially pull a really large result-set, and any values that don't have corresponding values in both the left and right table will be NULL.



```
FROM table-name1 FULL JOIN
table-name2
   ON column-name1 = column-name2
```

FROM table-name1 FULL JOIN table-name2 ON column-name1 = column-name2

For more information

JOINs are going to be useful for working with relational databases and SQL—and you will have plenty of opportunities to practice them on your own. But, here are a few resources that can give you more information about JOINs and how to use them:

- **SQL JOINs:** This is a good basic explanation of JOINs with examples. If you need a quick reminder of what the different JOINs do, this is a great resource to bookmark and come back to later.

- **Database JOINs - Introduction to JOIN Types and Concepts:** This is a really thorough introduction to JOINs. Not only does this article explain what JOINs are and how to use them, but it also explains the various scenarios in more detail of when and why you would use the different JOINs. This is a great resource if you are interested in learning more about the logic behind JOINing.
- **SQL JOIN Types Explained in Visuals:** This resource has a visual representation of the different JOINs. This is a really useful way to think about JOINs if you are a visual learner, and it can be a really useful way to remember the different JOINs.
- **SQL JOINs: Bringing Data Together One Join at a Time:** Not only does this resource have a detailed explanation of JOINs with examples, but it also provides example data that you can use to follow along with their step-by-step guide. This is a useful way to practice JOINs with some real data.
- **SQL JOIN:** This is another resource that provides a clear explanation of JOINs and uses examples to demonstrate how they work. The examples also combine JOINs with aliasing. This is a great opportunity to see how JOINs can be combined with other SQL concepts that you have been learning about in this course.