<https://pub.towardsai.net/a-complete-guide-to-sql-for-data-science-35743e73fd>

# A Complete Guide to SQL for Data Science

## Querying Techniques Every Data Scientist Should Know

[](https://mahadeveaswar.medium.com/?source=post_page-----35743e73fd--------------------------------)

[Mahadev Easwar](https://mahadeveaswar.medium.com/?source=post_page-----35743e73fd--------------------------------)

[Feb 19](https://pub.towardsai.net/a-complete-guide-to-sql-for-data-science-35743e73fd?source=post_page-----35743e73fd--------------------------------) · 13 min read

Structured Query Language (SQL) is a language designed for storing, managing, and retrieving data from a relational database. All Relational DB Systems like MySQL, MS Access, SQL Server, Oracle, and Postgres use SQL as their standard database language for performing **CRUD** (Create, Read, Update & Delete) operations.

Filtering and performing certain manipulations or computations at the Database level can significantly improve the performance of the application that utilizes the data. SQL is a language used by Data Scientists almost on a day-to-day basis but not spoken about as much as R or Python. Working knowledge of **SQL** and **databases** is imperative for **Data Science**.

<https://www.w3resource.com/sql/joins/perform-an-inner-join.php>

<https://mode.com/sql-tutorial/sql-left-join/>

**SQL INNER JOIN**

Last update on February 26 2020 08:07:43 (UTC/GMT +8 hours)

**What is Inner Join in SQL?**

The INNER JOIN selects all rows from both participating tables as long as there is a match between the columns. An SQL INNER JOIN is same as JOIN clause, combining rows from two or more tables.

**Syntax:**

SELECT \*

FROM table1 INNER JOIN table2

ON table1.column\_name = table2.column\_name;

OR

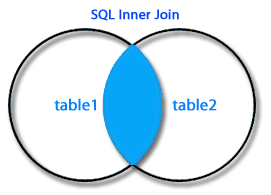
SELECT \*

FROM table1

JOIN table2

ON table1.column\_name = table2.column\_name;

**Pictorial Presentation:**



# <https://mode.com/sql-tutorial/sql-left-join/>

# SQL LEFT JOIN

**Starting here?** This lesson is part of a full-length tutorial in using SQL for Data Analysis. [Check out the beginning](https://mode.com/sql-tutorial/introduction-to-sql).

**In this lesson we'll cover:**

* [The LEFT JOIN command](https://mode.com/sql-tutorial/sql-left-join/#the-left-join-command)
* [Practice problems](https://mode.com/sql-tutorial/sql-left-join/#sharpen-your-sql-skills)

## The LEFT JOIN command



Let's start by running an [INNER JOIN](https://mode.com/sql-tutorial/sql-inner-join) on the [Crunchbase dataset](https://mode.com/sql-tutorial/sql-outer-joins#the-crunchbase-dataset) and taking a look at the results. We'll just look at company-permalink in each table, as well as a couple other fields, to get a sense of what's actually being joined.

SELECT companies.permalink AS companies\_permalink,

Practice :

<https://app.mode.com/tutorial/reports/e6cde36b3e4a/details/queries/c011d8e443a5>

<https://mode.com/sql-tutorial/sql-outer-joins/#the-crunchbase-dataset>

# SQL Outer Joins

**Starting here?** This lesson is part of a full-length tutorial in using SQL for Data Analysis. [Check out the beginning](https://mode.com/sql-tutorial/introduction-to-sql).

**In this lesson we'll cover:**

* [Outer joins](https://mode.com/sql-tutorial/sql-outer-joins/#outer-joins)
* [The Crunchbase dataset](https://mode.com/sql-tutorial/sql-outer-joins/#the-crunchbase-dataset)

## Outer joins

When performing an [inner join](https://mode.com/sql-tutorial/sql-inner-join), rows from either table that are unmatched in the other table are not returned. In an outer join, unmatched rows in one or both tables can be returned. There are a few types of outer joins:

* [LEFT JOIN returns only unmatched rows from the left table.](https://mode.com/sql-tutorial/sql-left-join)
* [RIGHT JOIN returns only unmatched rows from the right table.](https://mode.com/sql-tutorial/sql-right-join)
* [FULL OUTER JOIN returns unmatched rows from both tables.](https://mode.com/sql-tutorial/sql-full-outer-join)
* As you work through the following lessons about outer joins, it might be helpful to refer to [this JOIN visualization](http://joins.spathon.com/) by [Patrik Spathon](https://twitter.com/Spathon).
* [Graphical user interface

  Description automatically generated](https://joins.spathon.com/)

## The Crunchbase dataset

* The data for the following lessons was pulled from [Crunchbase](http://info.crunchbase.com/about/crunchbase-data-exports/), a crowdsourced index of startups, founders, investors, and the activities of all three. It was collected Feb. 5, 2014, and large portions of both tables were randomly dropped for the sake of this lesson. The first table lists a large portion of companies in the database; one row per company. The permalink field is a unique identifier for each row, and also shows the web address. For each company in the table, you can view its online Crunchbase profile by copying/pasting its permalink after Crunchbase’s web domain. For example, the third company in the table, “.Club Domains,” has the permalink “/company/club-domains,” so its profile address would be <http://www.crunchbase.com/company/club-domains>. The fields with "funding" in the name have to do with how much outside investment (in USD) each company has taken on. The rest of the fields are self-explanatory.
* SELECT \*
* FROM tutorial.crunchbase\_companies
* The second table lists acquisitions—one row per acquisition. company\_permalink in this table maps to the permalink field in tutorial.crunchbase\_companies as described in the previous lesson. Joining these two fields will add information about the company being acquired.
* You'll notice that there is a separate field called acquirer\_permalink as well. This can also be mapped to the permalink field tutorial.crunchbase\_companies to add additional information about the acquiring company.
* SELECT \*
* FROM tutorial.crunchbase\_acquisitions
* The foreign key you use to join these two tables will depend entirely on whether you're looking to add information about the acquiring company or the company that was acquired.
* It's worth noting that this sort of structure is common. For example, a table showing a list of emails sent might include a sender\_email\_address and a recipient\_email\_address, both of which map to a table listing email addresses and the names of their owners.
* [Next Lesson](https://mode.com/sql-tutorial/sql-left-join/)
* [SQL LEFT JOIN](https://mode.com/sql-tutorial/sql-left-join/)

# Get our weekly data newsletter

* Work-related distractions for every data enthusiast.
* Contact
* [hi@modeanalytics.com](mailto:hi@modeanalytics.com)
* 208 Utah Street, Suite 400  
  San Francisco CA 94103
* © Mode Analytics, Inc.  2021
* [Terms of Service](https://mode.com/tos/)[Privacy Policy](https://mode.com/privacy/)
* Product
* [Compare plans](https://mode.com/compare-plans/)[SQL Editor](https://mode.com/online-sql-editor/)[Notebooks](https://mode.com/notebooks/)[Reports & Dashboards](https://mode.com/reports-and-dashboards/)[Embedded Analytics](https://mode.com/embedded-analytics/)[Mode for Slack](https://mode.com/slack/)[Customers](https://mode.com/customers/)[Integrations](https://mode.com/data-sources/)[Security](https://mode.com/security/)
* Resources
* [Resource center](https://mode.com/resources/)[Learn SQL](https://mode.com/sql-tutorial/)[Learn Python](https://mode.com/python-tutorial/)[Data newsletter](https://mode.com/analytics-dispatch/)[ROI Calculator](https://mode.com/roi-calculator/)[Help & support](https://mode.com/help)[Developer](https://mode.com/developer/api-reference/introduction/)[Gallery](https://mode.com/example-gallery/)[Blog](https://mode.com/blog/)[System status](https://status.modeanalytics.com/)
* Company
* [About us](https://mode.com/about-us/)[Team](https://mode.com/team/)[Values](https://mode.com/values/)[Careers](https://mode.com/careers/)[Contact us](https://mode.com/contact-us/)

### This Website Uses Cookies

* We use our own and third-party cookies to provide you with a great online experience. We also use thes