Boletim da Sociedade Astronômica Brasileira, **35**, no. 1, 244-245


# Tutorial on accessing public astrophysical data using SQL and Python languages.

Raphael Gomes Sousa[1], Sérgio Scarano Jr.[2], & Ramachrisna Teixeira[3]

[1] Universidade Federal de Santa Maria, Rio Grande do Sul e-mail: raphael.gomes@acad.ufsm.br
[2] Universidade Federal de Sergipe, Sergipe e-mail: scaranojr@academico.ufs.br
[3] Instituto Astronômico e Geofísico da USP, São Paulo, e-mail: rama.teixeira@iag.usp.br

**Abstract.** The Structured Query Language or (SQL) is an important tool for querying and manipulating data present in databases. The main astrophysical data are published in databases that support queries using the SQL language. Therefore, this work shows in a simple way the functions of the main query tools, with practical examples using data from the latest Data Release of the Gaia telescope, the DR3. Finally, structures are shown on how to query Gaia's DR3 data using SQL within the TOPCAT software and within another programming language, the Python language.

**Resumo.** A linguagem de pesquisa estruturada ou *Structured Query Language* (SQL) é uma ferramenta importante para a consulta e manipulação de dados presentes em bancos de dados. Os principais dados astrofísicos são divulgados em bancos de dados que possuem suporte para a consulta utilizando a linguagem SQL. Com isso, neste trabalho é mostrado de forma simples as funções das principais ferramentas de consultas, com exemplos práticos utilizando dados do último *Data Release* do telescópio Gaia, o DR3. Por fim, são mostradas estruturações de como consultar os dados do DR3 do Gaia utilizando o SQL dentro do software TOPCAT e dentro de outra linguagem de programação, a linguagem Python.

**Keywords.** Methods: data analysis – Virtual observatory tools

## 1. Introduction

The Structured Query Language (SQL) is a widespread tool for cleaning and filtering data in the area of Astronomy that allows the user to store and process information in a relational database. With specific knowledge of the tool, it is possible to access several databases from multiple public repositories from telescopes and observatories. The GAIA space mission, which with its most recent Data Release, DR3 Vallenari, et al (2023), provides data from around 2 billion sources in table format and provides public access in multiple ways, its accessible base with the SQL language. From DR3, it is possible to filter data related to morphology, spectroscopy and kinematics from sources that can range from stars to black holes. Benefiting from its integration with the SQL language, in this work we present the details of how to explore two interfaces for accessing public astrophysical data: the library package *Astropy* Price-Whelan & Adrian M., et al (2018) of the Python programming, which makes searching, filtering and storing data possible and the *software* Tool for OPerations on Catalogs And Tables (TOPCAT) Taylor Mark B. (2005), which makes accessing and manipulating data from the DR3 is more didactic, objective and faster, without the need for Python. Through these tools, we will demonstrate how to gain access to GAIA's public DR3 data, in the form of a short tutorial.

## 2. Objectives

This work aims to help those interested in accessing data in virtual observatories, especially high school, undergraduate or postgraduate students, in accessing virtual catalogs published in the field of Astronomy, using the structured language SQL, through tools such as the Astropy library, in Python and the TOPCAT interface. We illustrate this in a simplified way by presenting access to the Gaia telescope's *Data Release 3* public data repository.

## 3. Main SQL Inputs

There is a range of functions and inputs that can be explored in the SQL language, ranging from creating a new table to organizing them. All of them are possible for those looking for a deeper understanding of the language and its structure, but the focus in this work will be on functions and inputs that execute and search tables in large data sets. The focus here will be on `SELECT, FROM, WHERE` entries, where the `SELECT` function must be followed by the full name of the columns that will be used. The `FROM` function is accompanied by the name of the table that contains the data. The `WHERE` function, in turn, starts us reducing the data, which can be followed by any column, whether previously selected or not, to which a selection criterion will be applied. An example of a simple input code that selects right ascension (RA), declination (Dec) from a Gaia DR3 table, only searching for data with modulus of proper motion (pm) greater than 0, can be presented as below:

```
SELECT ra, dec
FROM gaiadr3.gaia_source
WHERE pm > 0
```

The return will be through a table with 2 columns, straight sensibility (RA) and declination (Dec) where all objects shown there must have a modulus of proper motion greater than 0. Figure 1 shows how each works. one of the functions used.

## 4. Using TOPCAT software

TOPCAT is an interactive graphics visualization and editing software for tabular data. It aims to provide simplified analysis and manipulation of astronomical source catalogs and other

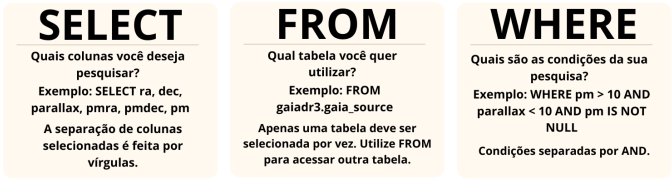


**Figure 1.** An explanation of how each of the `SELECT`, `FROM`, `WHERE` functions works. And an example of how to use these functions.

tables. It comprises several astronomically important formats (including FITS, VOTable, and CDF) and more formats can be added. It is very well used in the interactive exploration of large tables (with several million rows and many columns).

*software* operates using a Java-based expression language, where new columns can be defined as well as new subsets of rows selected for separate analysis. Table data and metadata can be edited and the resulting table can be written to a wide variety of output formats. Your *download* is available at: https://www.star.bris.ac.uk / mbt/topcat/

## 5. TOPCAT step by step - Access to Gaia's DR3 data

Basic knowledge of the SQL language is important because it is what, in a simplified way, accesses the catalogs available in TOPCAT.
In the figure 2 you can see a step-by-step guide on how to obtain your first Gaia DR3 data table using the TOPCAT software.

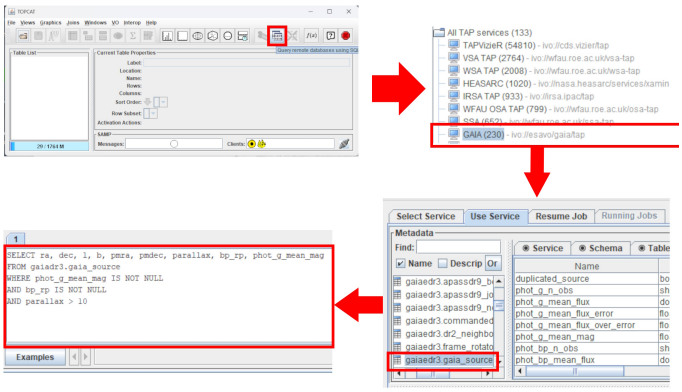


**Figure 2.** Step by step on how to access Gaia's DR3 data in *software* TOPCAT.

## 6. Astropy step by step - Access to Gaia's DR3 data

A more complete way to perform data analysis is using the Python programming language. The *Astropy* library package provides a very wide variety of functions that can be accessed with any existing Python compiler. There is a *Astroquery* library that allows data from any Gaia *Data Release* to be accessed within Python in tables in the formats voteable, fits, csv, json. To access Gaia's DR3 data using the *Astroquery* library, it is necessary to install the libraries in your compiler, as well as import the `astroquery.gaia` library. Here's a step-by-step guide:

```
!pip install astropy
!pip install astroquery

from astroquery import Gaia

job = Gaia.launch_job("SELECT top 100000 "
"ra, dec, bp_rp, parallax, phot_g_mean_mag "
"from gaiadr3.gaia_source "
"where parallax > 10 "
"and bp_rp IS NOT NULL",
dump_to_file=True, output_format='csv')
print (job.outputFile)
```

Where the functions that appear right after the code in SQL `dump_to_file` and `output_format` are library functions that download the query to a file and give us the option to select the output file type, which in this example was the CSV file type.

## 7. Conclusion

The SQL language allows simplified access to catalogs with large amounts of data. Astrophysical data are periodically released publicly available in table formats such as "CSV", "VOTable" and others. The SQL language, together with the use of functions such as `SELECT, FROM, WHERE`, helps in filtering the data that will be worked on. *software* TOPCAT also offers a series of alternatives and graphs that can be used for analysis, always using the SQL language as a form of data selection. With the *astroquery* library from the *astropy* suite, there is a way to combine the SQL and Python languages to access data from public catalogs, such as the Gaia catalog shown here in this work. With the access methods presented in this work, the reader will have the possibility of simply accessing the latest data published by the Gaia telescope *Data Release 3* and still understanding how each of the base functions of the SQL language work for applications diverse.

## References

Vallenari, A., et al, 2023, Astronomy & Astrophysics, 674
Price-Whelan, Adrian M., et al, 2018, The Astronomical Journal, 156.3
Taylor, M. B., 2005, Astronomical data analysis software and systems XIV, Vol. 347.