NHS
Business Services Authority

Coffee & Coding

Introduction to SQL macros

Introduction to SQL macros

SQL macros are a way to create a “parameterised” view, where logic can be applied based on the parameters passed when calling the macro function.

The SQL macro generally use the same coding as a regular SQL query. However, there is an added ability to be able to reference parameter values at runtime.

Using SQL macros can help limit the reproduction of code, and in certain scenarios can help improve performance.

Introduction to SQL macros

SQL macros were introduced in Oracle in 2020.

There are two different types of SQL macros:

- **TABLE** expressions (available in 19c onwards)
- **SCALAR** expressions (available in 21c onwards)

We are focusing on the **TABLE** expressions **only**.

Introduction to SQL macros

In this session we will:

- Look at how SQL macros can be used to replace a simple query and view.
- Look at some of the syntax involved with SQL macros.
- Discuss some of the main benefits/limitations.

The demonstration requires access to an Oracle database so there may be limited opportunity to “code along”.

All code used today is available in the Coffee & Coding Git repository.

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Data Understanding: Indices of Deprivation

The Index of Multiple Deprivation (IMD) is the official measure of relative deprivation in England and is part of a suite of outputs that form the Indices of Deprivation (IoD).

IMD is reported at LSOA level (32,844 small areas in England) and can be used to rank areas from most to least deprived.

The IMD is based on a calculation across 7 domains of deprivation:

1. **Income** (proportion of population experiencing deprivation relating to low income)
2. **Employment** (proportion of working age population involuntarily excluded from the labour market)
3. **Education** (lack of attainment and skills in the local population)
4. **Health** (risk of premature death and the impairment of quality of life through poor physical or mental health)
5. **Crime** (risk of personal and material victimisation at local level)
6. **Barriers to Housing & Services** (physical and financial accessibility of housing and local services)
7. **Living Environment** (quality of the local environment)

Source: [The English Indices of Deprivation 2019 (publishing.service.gov.uk)]

Data Understanding: LSOA to other geography mapping

The Office for National Statistics (ONS) publish a variety of geographical mappings via their Open Geography Portal.

This can be very useful when data is only published at certain levels but may be required to be reported at other geographical levels.

Unpublished mappings can also sometimes be attained based on email requests.

For this analysis we are going to focus on the following mappings:

- LSOA to Sub-ICB location
- LSOA to Local Authority

Please note: Before aggregating data to various geographical levels for reporting you should consider if this is appropriate, supported by the data you have, and not already available elsewhere.

Data Understanding: How to rank IMD at higher geographies

The IMD scores are only published at LSOA level and therefore calculation is required to rank higher geographies based on IMD scores.

The approach we use is based on methodology defined by the Ministry of Housing, Communities & Local Government (MHCLG).

An average score will be calculated for each parent geography which can then be used to rank from most to least deprived.

$$\text{Average score for parent geography} = \frac{\text{Sum of population * IMD score for each LSOA}}{\text{Sum of LSOA populations within parent geography}}$$

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Data Understanding: Importing datasets

In the SQL folder there are two SQL scripts to create the table structures:

- 01_01_IMD_DATA
- 01_02_GEO_MAPPING

In the Data folder there are two CSV files that can be imported to these tables:

- CC_IMD_DATA
- CC_GEO_DATA_2023

Tip: Creating the table structure directly, rather than using the import wizard in SQL Developer can help limit actions required if import fails due to issues.

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Code Demonstration

The following scripts will be used from the SQL folder in the Git repository:

Basic analysis of the datasets:

02_01_BASIC_ANALYSIS

View to simplify queries:

03_01_BASIC_VIEW

SQL macro functions:

04_01_BASIC_FUNCTION
04_02_GEOGRAPHY_PARAMETER
04_03_COLUMN_PARAMETER
04_04_TABLE_PARAMETER

Additional example using EPD data:

05_01_EPD_DATA_TABLE_CREATION
05_02_EPD_FUNCTION

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Benefits

Reusable code

- A single stored function can be referenced without the need to copy/paste or reproduce code with minor variations.
- Including table(s)/column(s) as parameters allow generic functions to cover multiple scenarios.

Advantages over regular views

- The logic can be embedded in the code, limiting what data needs to be exposed to the end user.
- Performance improvements where data can be limited earlier in the process.

Advantages over PL/SQL

- A lot of the functionality of SQL macros is not new and could previously have been done using PL/SQL. However, the introduction of SQL macros have some benefits.
- No need to learn a new language/syntax other than the basic declaration commands to create the function.
- Using SQL macros makes the code transparent to the optimiser, removing the context switching between PL/SQL and SQL which can impact performance.

Limitations

No compatibility with Common Table Expressions (CTEs or “with” clauses)

- The code within the SQL macro function cannot include the use of CTEs and therefore may miss out on the benefits these bring such as readability and reduction of repetition.
    - SQL macro functions can however be called from within other SQL macro functions.
- A SQL macro function cannot be called as part of a CTE expression.

Errors may not always be identified until runtime

- SQL macros are just producing text, so it's easy to enter incorrect syntax and the macro will compile.
- It's only at runtime you will see an error, so test your macros carefully.

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Conclusions

We have only scratched the surface so far of how this new functionality could be used.

There are many benefits to limiting the reproduction of code and reusing consistent functions where possible.

SQL macros may not always be the best choice but could be a consideration in many scenarios.

Data Understanding

Importing Datasets

Code Demonstration

Benefits & Limitations

Conclusion

Useful Links

Useful Links

Coffee & Coding Git Repo:

- Git Repo

Datasets:

- The English Indices of Deprivation 2019 (publishing.service.gov.uk)
- Open Geography Portal
- NHSBSA Open Data Portal – English Prescribing Dataset

Videos:

- [Feuertips #24] - The power of SQL macros and more (youtube.com)