

Paper CT13

Next generation collaboration: using tasks and snippets in SAS Studio

Fatima Kassim, OCS Life Sciences, 's-Hertogenbosch, The Netherlands

ABSTRACT

SAS Studio is a web-based programming environment that is the successor of PC-SAS and Enterprise Guide. It runs on your computer as well as on a central server or in the cloud. One key advantage this programming platform offers is the availability of so-called tasks and snippets.

Tasks are wizard-like utilities that generate SAS code for common tasks such as generating boxplots or performing statistical analyses. Snippets are short pieces of code that can be quickly inserted into a program.

Users can customise tasks and snippets to their own requirements. Tasks and snippets are portable and, if the environment is server or cloud based, can be shared with co-workers through a repository, which allows all employees to easily share and apply useful code and further improve uniformity.

This paper will demonstrate how to create, use, and manage tasks and snippets and explain how you can benefit from them the most.

INTRODUCTION

When first switching from PC-SAS (SAS® Base) or Enterprise Guide (EG) to SAS Studio, the interface, though familiar through its simple point-and-click interface can be quite intimidating. The first things to figure out are how to program in this new environment and what adjustments need to be made to current programs to work efficiently in SAS Studio.

After the initial familiarisation process is done, such as programming, editing the Autoexec file, making use of the different perspectives (SAS or Visual programmer), there is some room to explore the many features that SAS Studio offers.

Foremost, there are two helpful features that can aid any novice or seasoned programmer in their day-to-day activities. These features are Tasks and Snippets. They can be customised for specific project purposes or general company requirements and there are various ways of creating and implementing them either by shortcuts, point-and-click and/or drag-and-drop methods.

The standard Tasks and Snippets are simple and easy to use, but for more complex, unique or repetitive activities one can create custom Tasks and Snippets which are easily generated and shared. In the following sections, both standard and custom versions of each feature are explained with handy illustrations detailing how to create, store and use them.

SNIPPETS

Snippets are, as the term implies, short pieces of (relevant) SAS code. They can be quickly implemented when writing a program and can therefore increase the efficiency and speed with which to complete a program. Snippets can be found in the 'Snippets bin' in the 'Navigation Pane' (i.e. the left-side panel) in SAS Studio.

STANDARD SNIPPETS

SAS Studio already comes with a set of its own standard snippets. Refer to Figure 1 for an overview of the categories for which SAS Studio offers standard snippets. Some of these standard snippets do not require any modification before using, see Figure 2 for an example. Other standard snippets require minimal modification, the instructions of which are indicated in purple font, see Figure 3 for an example.

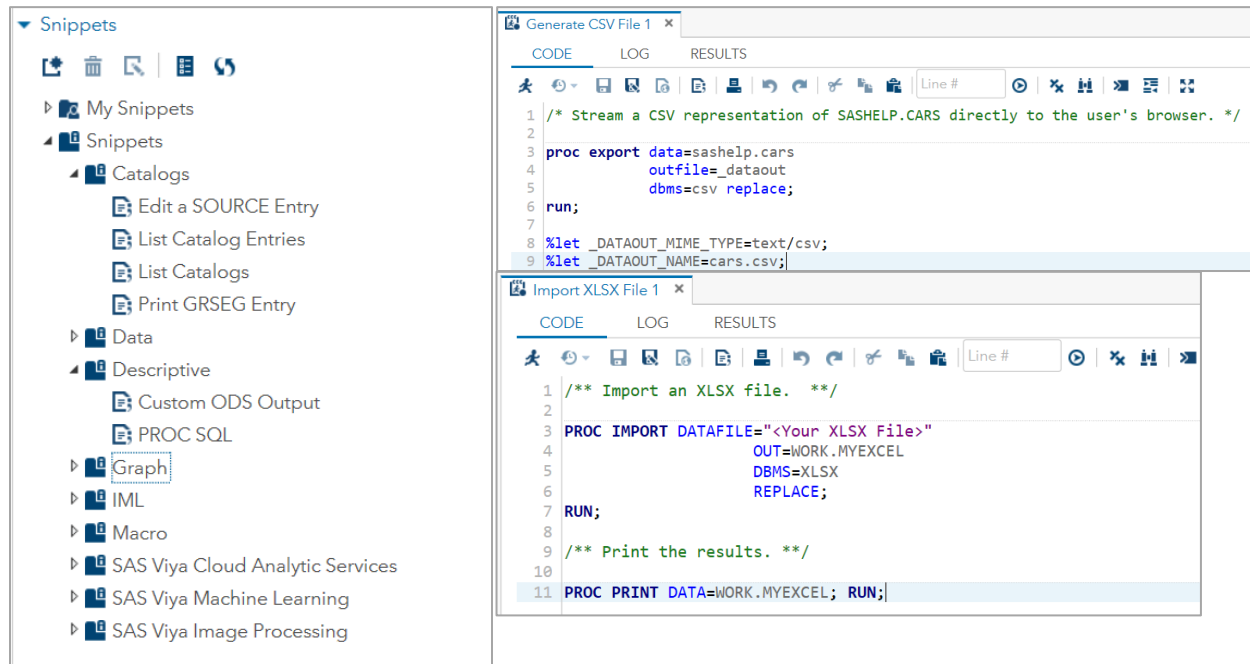


Figure 1 (left)– Snippets bin, with the standard snippets. Note that the drop-down list is shown for snippets related to Catalogs.
Figure 2 (top right) – Illustration of standard snippet 'Generate CSV File' in SAS studio. Note that the user does not need to modify the code before submitting.
Figure 3 (bottom right) – Illustration of standard snippet 'Import XLSX File' in SAS studio. Note that the user does need to modify the code before submitting, indicated by the **purple** font.

Standard snippets are thus easy to locate, manage and use as they are available to any user regardless of account rights and provide the same functionality across browsers and servers.

CUSTOM SNIPPETS

A very useful feature of snippets is that you can easily create your own custom snippets either by modifying standard snippets to suit your needs or by simply selecting code that you want to have as a custom snippet. For example, a very useful custom snippet to have within a company is a custom 'HEADER' snippet (see Figure 7), that can be shared across the company via a repository (see section '**REPOSITORY**'). This snippet can be easily implemented which allows for uniformity across programs, projects and programmers.

There are several ways of creating custom snippets:

1. Right-click and select 'Add to my snippets'. Assign a name to the custom snippet and click 'Save'.
2. Use shortcut Alt + A. Assign a name to the custom snippet and click 'Save'.
3. Click on the 'Add to my snippets' icon at the top of the work area, see Figure 4 below.
4. Click on the 'New snippet' icon in the Snippets bin in the navigation pane, see Figure 5 below
5. Use shortcuts: Shift+Alt+A for Windows; Cmd+Option+A for Mac OS; see Figure 6 below.



Figure 4 – Work area icons: The 'Add to My Snippets' icon is highlighted in red.

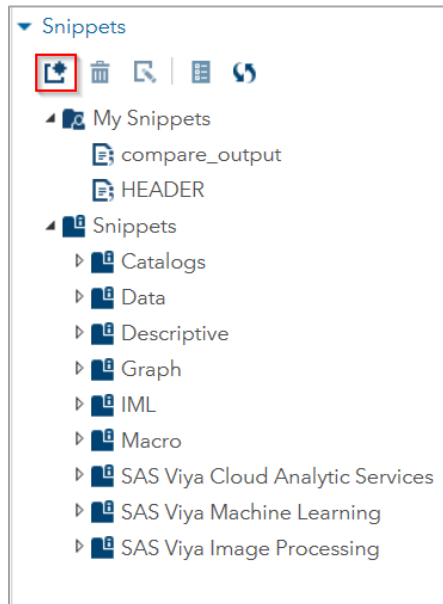


Figure 5 – Snippets bin with the 'New snippet' icon highlighted in red.

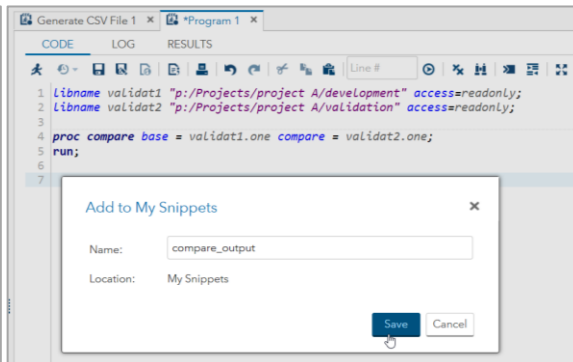


Figure 6 – Using the shortcut (Shift-Alt-A for Windows, or Cmd+Option+A for Mac OS) the panel 'Add to My Snippets' appears. Name your custom snippet, i.e. `compare_output` and click save to add to the 'My Snippets' bin on the right.

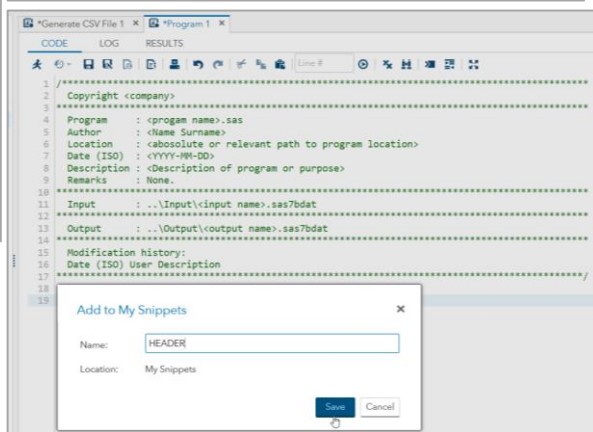


Figure 7 – Having a standard 'HEADER' snippet can be a great tool towards uniformity across company owned programs.

One can apply the previously mentioned methods of creating custom snippets by either selecting pieces of code or applying it to all the written code in the work area. Additionally, for easier access, one can select a standard snippet, right-click and then choose 'Add to My Snippets' to have it in their 'My Snippets' folder.

There are several ways of quickly inserting standard and custom snippets in a program:

1. Double-click on a particular snippet.
2. Click on a snippet and drag it to the work area.
3. Right-click on a snippet and select 'Insert' from drop-down list.
4. Hold down the Ctrl key and select several snippets and then right-click and select 'Insert' from the drop-down list to have all selected (multiple) snippets inserted simultaneously in the order of selection.
5. Use a shortcut: Alt + I for Windows; or Cmd+Option+I for Mac OS.

Note that any inserted code will be inserted from the location of the mouse pointer in the work area.

As snippets are short pieces of SAS code, one can run the SAS code by pressing F3 on their keyboard or selecting the 'Running Man' icon. As such, the use of snippets is truly as simple as adding them to your program and saving them. Hence, snippets can serve as valuable building blocks that are quickly and easily implemented when programming and can help reduce time-consuming processes.

TASKS

As mentioned above, Tasks are another useful feature available in SAS Studio. As with snippets, SAS Studio is also stocked with a set of predefined tasks that are easily used. Refer to Figure 8 for an overview of the categories for which SAS Studio offers standard tasks. A task provides a user with a point-and-click interface (of which the default view is the split setting, see Figure 9) that can guide the user through analytical, data-preparation (i.e. data cleaning) and/or data-conversion (e.g. SDTM standard) processes. A task will generate SAS code in the background and format the garnered results of the SAS procedure. The user simply selects the dataset, specifies the roles and options then executes or 'runs' a task.

Tasks are based on the common task model (CTM), which consists of XML (metalanguage that allows for custom mark-up language) and Velocity Template Language (VTL or Apache Velocity language).

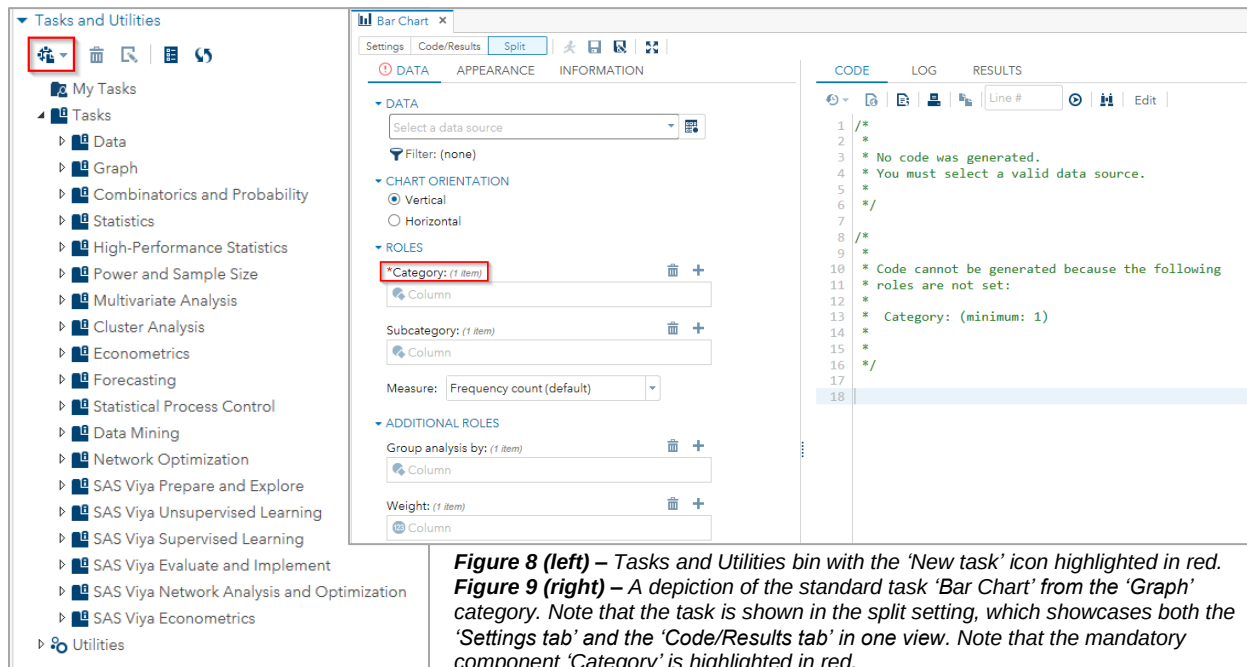
Each task is based on such a CTM file. These files define the template for a task. In the CTM file, the user interface (Settings tab) is defined alongside the code that is needed to run the desired task. The CTM file is composed of many different elements.

A CTM file that generates a 'Task' functionality has a 'TASK' element, which has the following child elements:

1. Registration element → Defines the type of task; i.e. task name, description, and other desired properties.
2. Metadata element → Specifies whether or not an input data source is required.
 - a. Roles element → Specify the type of variables required.
 - b. Options element → How to display options in the Settings tab.
3. User Interface (UI) element → Presentation of user interface (top-down layout is supported).
4. Dependencies element → Possible dependencies in the user interface.
5. Requirements element → Required conditions for code to be generated.
6. CodeTemplate element → Final output (illustration or SAS code etc.).

STANDARD TASKS

As detailed in Figure 8 there are many different categories for which SAS Studio offers standard tasks, ranging from simple graphical tasks (such as a 'Bar Chart' task, see Figure 9 and 10) to complex statistical tasks such as a 'Mixed Models' task.



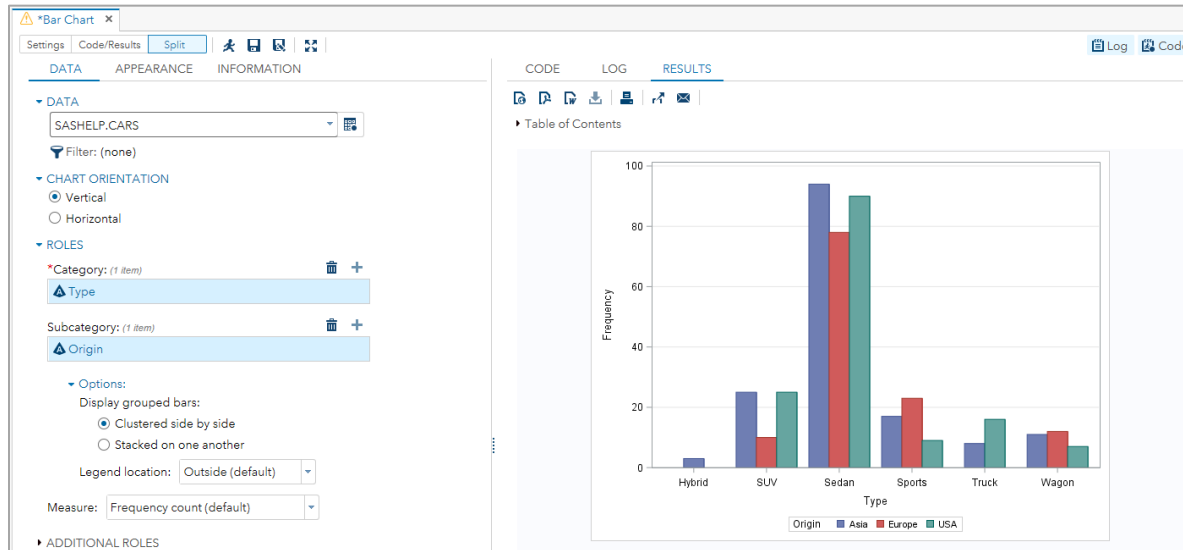


Figure 10 – A ‘Bar Chart’ standard task applied to the SASHELP.CARS dataset categorising the ‘Type’ of cars clustered by their region of ‘Origin’ in a vertical orientation. Note that on the right the results of the task are shown.

As standard tasks are already predefined, the user merely needs to select their dataset and fill in the ‘ROLES’ (and the required items therein) to be able to run the task. These standard tasks allow for quick analyses to be done in just a few clicks. Hence, the standard tasks offer any user, novice or expert, a manageable clear environment from which to comfortably run either basic or multifaceted SAS procedures.

CUSTOM TASKS

If a user requires a multifaceted SAS program but cannot find a suitable standard task to accommodate their requirements, there is always the possibility to create one’s own task. There are several ways to create a custom task.

A quick and minimally challenging way is to edit an existing predefined task. This is done by right-clicking a standard task and selecting the ‘Add to My Tasks’ option, consequently a copy of the standard task is saved under the ‘My Tasks’ folder. Select the standard task and then click the ‘Edit’ button to edit the CTM file and customise the XML and Velocity code. Save the customised CTM file and run it to generate the CTK (task interface) file, from which you can now run your own customised task.

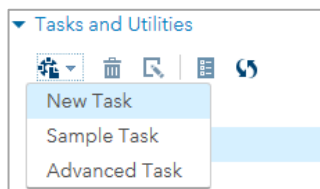


Figure 11 – Clicking the ‘New task’ icon leads to a drop-down list. There are three options to choose from: ‘New Task’, ‘Sample Task’ and ‘Advanced Task’.

Another option is to create a custom task from blank templates. This starts by clicking the ‘New task’ icon, see Figure 11. There is a selection of custom tasks to choose from.

Selecting ‘New Task’ will give the user a blank template from which to start the creation of a custom task. The user needs to complete all elements of the Task element using XML/VTL, see Figure 12 for the blank template.

There are several guides available online to help novices and experts alike to create their own custom tasks from scratch. These guides provide valuable step by step explanations for each element. A handy task to help a user get started, is to create a custom task that converts character variables in a dataset to numeric or vice-versa (see ‘SAS Studio3.4: Writing Your First Custom Task.pdf’ provided by SAS® documentation).^[1]

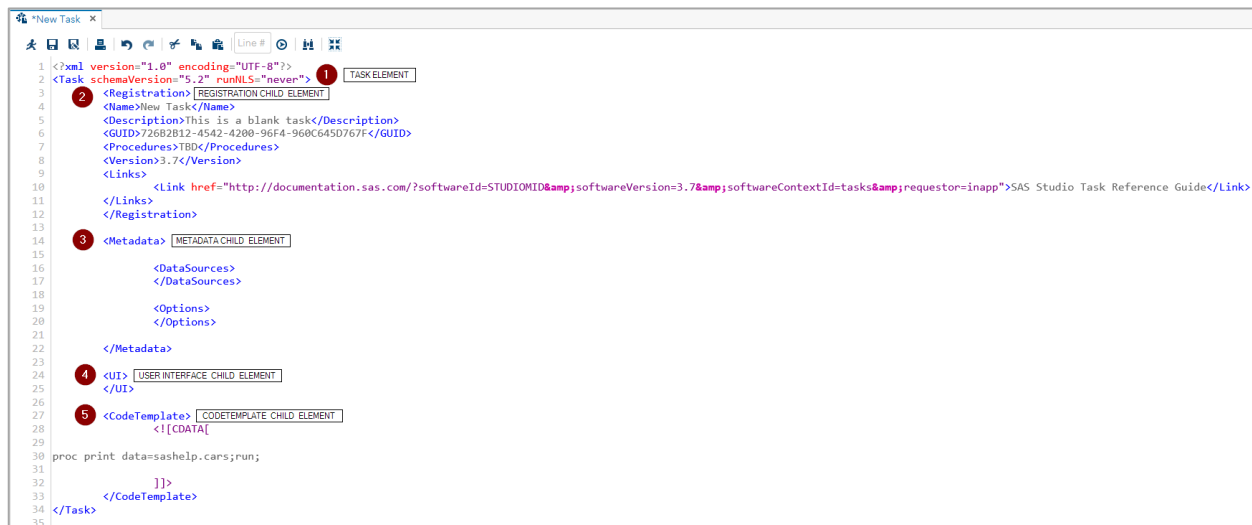


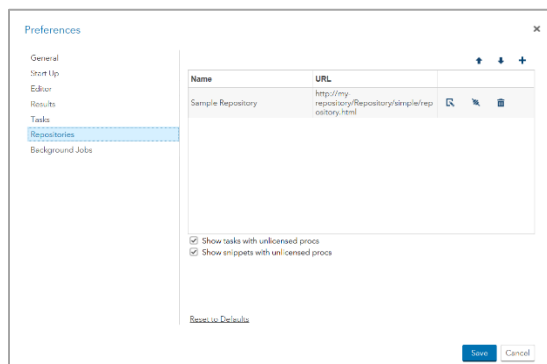
Figure 12 – A ‘New Task’ blank template denoting all the child elements of the ‘Task’ element (1), except for the ‘Dependencies’ and the ‘Requirements’ child elements.

A ‘Sample Task’ provides a user with a simple task outline to which only small adaptations need to be made for customisation. The sample task shows the controls that are available to you when writing a task. The ‘Advanced Task’, as indicated by the term, provides a template that already captures some of the more complex functionalities in the CTM file such as dependencies, the mixed effects builder, data linking, and return values.

Each selection provides a template from which a custom task can be built with XML and VTL. Once editing a standard task becomes rudimentary and fundamental knowledge exists regarding the different elements and how they compose a CTM file, creating one’s own custom task becomes relatively straight-forward. And there are many papers available online to help guide in the creation of a custom task.

Though tasks provide independent SAS processes outside a program, one can also copy the code behind a task and add that to a program. Tasks can likewise be individually linked to programs in a process flow using the ‘Visual Programmer’ view in SAS Studio. In short, tasks provide a very user-friendly environment from which to run simple to very complex SAS processes with simple clicks of a mouse.

REPOSITORY



Repositories are centralised locations where data and code can be stored, and they are shared via URLs (Uniform Resource Locator, i.e. an address) that can be accessed by users. An administrator can configure and create a general repository that is available to any user that has been granted permission. A global repository is immediately available in the ‘Tasks & Utilities’ and the ‘Snippets’ bins in the navigation pane. Additionally, any user can add their own personal repositories. This is done by going to the ‘Preferences’ pop-up window and selecting ‘Repositories’ and then clicking the plus icon. To share personal repositories, share the URL and the receiving user can add it to their repositories so as to have access to them in their sessions.

Figure 13 – ‘Preferences’ pop-up window displaying the ‘Repositories’ section with the ‘Add repositories’ icon highlighted in red.

The repository will appear below the utilities folder in the Tasks & Utilities bin and below the standard snippets folder in the Snippets bin. From there, one can use the stored Tasks and Snippets and upload new ones.

CONCLUSION

In conclusion, SAS Studio offers a host of useful standard Tasks and Snippets that can easily and quickly be implemented. This approach can help ease program writing for novices as well as for experienced programmers. Additionally, the added bonus of customising these features and sharing them via a repository allows for company-wide use which can help in uniformity and increase efficiency through the sharing of useful tasks or snippets amongst colleagues.

This paper provided a short overview on how to create, use and implement standard and custom Tasks and Snippets. The topics discussed allow the reader to successfully implement these useful features when using SAS Studio.

REFERENCES AND RECOMMENDED READING

1. SAS Studio 3.4: Writing Your First Custom Task, SAS® documentation (2015-12) at <https://support.sas.com/documentation/cdl/en/webeditortaskgs/68948/PDF/default/webeditortaskgs.pdf>
2. SAS® Studio 3.5: Developers Guide to Writing Custom Tasks, SAS® Documentation (2016-02) at <https://support.sas.com/documentation/cdl/en/webeditorug/68829/PDF/default/webeditorug.pdf>
3. SAS® Studio 3.7: Writing Your First Custom Task, SAS® Documentation (2017-04-18) at <https://documentation.sas.com/api/docsets/webeditortaskgs/3.7/content/webeditortaskgs.pdf?locale=en>
4. SAS® Studio 3.7: User's Guide, SAS® Documentation (2017-09-15) at <https://documentation.sas.com/api/docsets/webeditorug/3.7/content/webeditorug.pdf?locale=en>
5. Appendix 5 of SAS® Studio 3.7: User's Guide - Keyboard Shortcuts in SAS Studio Workspace

For further information on SAS Studio, please visit the official SAS website (SAS Studio related information is available at: https://www.sas.com/en_us/software/studio.html). Brand and product names mentioned are trademarks of their respective companies.

ACKNOWLEDGMENTS

I would like to acknowledge my wonderful colleagues who helped me finalise this paper by providing constructive feedback and encouragement along the way. In particular, I would like to mention Mylène van Doorn and Sofia Vale for proof-reading and providing critical talking points.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Fatima Kassim
OCS Life Sciences – OCS Consulting B.V.
Ruwkampweg 2G
's-Hertogenbosch / 5222 AT
+31 (0)73 523 6000
Fatima.kassim@ocs-consulting.com
<https://ocs-consulting.nl/ocs-life-sciences/>