Jupyter – SAS Kernel – auto-completion errors

Kernel.py : <https://raw.githubusercontent.com/sassoftware/sas_kernel/master/sas_kernel/kernel.py>

Kernel.py in my installation : C:\tools\anaconda3\Lib\site-packages\sas\_kernel



* Jupyter notebook is a framework that allows users to submit code to a computational engine (Kernel) and display results inline
* The SAS Kernel is a few hundred lines of python code that facilitates communication between the Jupyter notebook and SAS
* SAS communicates with the SAS Kernel through stdin, stdout, and stderr using the ODS HTML5 engine
* The SAS Kernel has no use without access to SAS

Table des matières

[1) Case 1 – Test of PROC auto-completion 2](#_Toc495675803)

[Using SAS standard kernel.py file 2](#_Toc495675804)

[2) Case 2 – Test of DATA STEP auto-completion 3](#_Toc495675805)

[Using the standard kernel.py file 3](#_Toc495675806)

[Using a modified SAS standard kernel.py file 3](#_Toc495675807)

# Case 1 – Test of PROC auto-completion

## Using SAS standard kernel.py file

#### **Test**

When you ask to complete a PROC name in jupyter, it works, but not but the parameter of the procedure.

1. I type the first letter of PROC and type touch “<tab>”



No suggestion appears and an error appears in SAS Kernel log (SAS kernel of Anaconda/Python server).

* **Error “TypeError: must be str, not bytes” appears.**

|  |
| --- |
| [IPKernelApp] ERROR | Exception in message handler:Traceback (most recent call last): File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 235, in dispatch\_shell handler(stream, idents, msg) File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 434, in complete\_request matches = self.do\_complete(code, cursor\_pos) File "C:\tools\anaconda3\lib\site-packages\metakernel\\_metakernel.py", line 508, in do\_complete matches.extend(self.get\_completions(info)) File "C:\tools\anaconda3\lib\site-packages\sas\_kernel\kernel.py", line 221, in get\_completions method = str(procer.group(0).split(' ')[-1].upper()) + mykey.encode()TypeError: must be str, not bytes |

#### **Python code extract :**

|  |
| --- |
|  if proc: # we are not in data section should see if proc option or statement lastsemi = info['code'].rfind(';') mykey = 's' if lastproc > lastsemi: mykey = 'p'  procer = re.search('(?i)proc\s\w+', info['code'][lastproc:]) method = procer.group(0).split(' ')[-1].upper() + mykey.encode() mylist = self.compglo[method][0] potentials = re.findall('(?i)' + info['obj'] + '.+', '\n'.join(str(x) for x in mylist), re.MULTILINE) return potentials |

# Case 2 – Test of DATA STEP auto-completion

## Using the standard kernel.py file

When you ask to complete a DATA STEP statement in jupyter, it doesn’t work.



Jupyter interface didn’t suggest statement.

* **Error “IndexError: list index out of range” appears.**

|  |
| --- |
|  [IPKernelApp] ERROR | Exception in message handler:Traceback (most recent call last): File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 235, in dispatch\_shell handler(stream, idents, msg) File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 434, in complete\_request matches = self.do\_complete(code, cursor\_pos) File "C:\tools\anaconda3\lib\site-packages\metakernel\\_metakernel.py", line 508, in do\_complete matches.extend(self.get\_completions(info)) File "C:\tools\anaconda3\lib\site-packages\sas\_kernel\kernel.py", line 217, in get\_completions method = procer.group(0).split(' ')[-1].upper() + mykey.encode()TypeError: must be str, not bytes |

|  |
| --- |
|  if info['line\_num'] > 1: relstart = info['column'] - (info['help\_pos'] - info['start']) else: relstart = info['start'] seg = info['line'][:relstart] if relstart > 0 and re.match('(?i)proc', seg.rsplit(None, 1)[-1]): potentials = re.findall('(?i)^' + info['obj'] + '.\*', self.strproclist, re.MULTILINE) return potentials else: |

## Using a modified SAS standard kernel.py file

#### **Test**

1. I changed just this row of Python script :

From:

 if relstart > 0 and re.match('(?i)proc', seg.rsplit(None, 1)[-1]):

To:

 if relstart > 0 and re.match('(?i)proc', 'fixed\_string'):

1. I type the first letter of PROC and type touch “<tab>”

When you ask to complete a PROC name in jupyter, it doesn’t work with this dirty modification. My modification breaks the completion for PROC <name>.



I receive this error :

|  |
| --- |
| [IPKernelApp] ERROR | Exception in message handler:Traceback (most recent call last): File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 235, in dispatch\_shell handler(stream, idents, msg) File "C:\tools\anaconda3\lib\site-packages\ipykernel\kernelbase.py", line 434, in complete\_request matches = self.do\_complete(code, cursor\_pos) File "C:\tools\anaconda3\lib\site-packages\metakernel\\_metakernel.py", line 508, in do\_complete matches.extend(self.get\_completions(info)) File "C:\tools\anaconda3\lib\site-packages\sas\_kernel\kernel.py", line 217, in get\_completions method = procer.group(0).split(' ')[-1].upper() + mykey.encode()TypeError: must be str, not bytes |

1. I type the “DATA X;” completely and ask completion for DATA STEP statements.



It works for DATA STEP now.