Programatic tracer(PTR)

Commands: (case-sensitive)

1. sout

🡪 print out watchvar information every iteration

Text

Description automatically generated

1. qout

🡪 stop printing out watchvar information at every iteration

Text

Description automatically generated

1. pout

🡪 temporarialy print out watchvar infromation

1. list

🡪 start printing current frame lines with arrow below showing current line execution.

Print every program iteration.

Text

Description automatically generated

1. qlist

🡪 stop printing current frame lines every iteration

Text

Description automatically generated

1. s

🡪 step or stepin

1. n

🡪 step-over current function that you are in. Note if you step-over a function with other function calls the program will skip all executions of the function calls.

Text

Description automatically generated

1. p + space + variablename

🡪 print out temporarily a variable and its value.

Usage of (PTR):

$Python3 calling\_script.py

Sample calling script:

Text

Description automatically generated

Note here the file is in a subdirectory of the Tracer code and the current working directory is the directory where the Tracer code (myscript4.py) is. The code gets the current working directory file path and adds on the file pat that is passed as a parameter to setFilePath method. If in local directory use “/myscript.py” instead of just the script name alone. This part needs the backslash. This needs to be set before exec\_steps() is called. Also the watchVar() methods must be called before the execution of exec\_steps() method.

The watchVar will add the variable that is passed as a parameter to the watchvar list.

The parameter passed to the exec\_steps will be used to create the keywords output.