Assignment GUI pipelining

Hi class

This week we will go over methods by which you can control other scripts using system calls, piping, and the subprocess module in python. We will build a PyQt5 graphical user interface for controlling the scripts you have written for former assignments. Also see runMyRfromPYTHON.py and runMyRfromPERL.pl for some examples of system calling R from within another language.

For this week's coding studio

1. Set up your original Python and R scripts from earlier in the course to run as a single pipeline (i.e. make sure the output of your python gene parser file creates a text file that is read by your R script to create plots in the Rplots.pdf file.
2. Install PyQt5 (e.g. pip3 install PyQt5 qt5-tools)
3. Open 'designer' (type designer on command line) and create a user interface template (i.e. a .ui file). Save you .ui file before exiting
4. Convert your .ui file to a .py file (Linux: pyuic5 –x myFile.ui –o myFile.py) or (WIN: python3 –m PyQt5.uic.pyuic –x myFile.ui –o myFile.py) or (MacOS: python3 myfile.ui > myfile.py)
5. Open the .py file in your code editor and add lines and functions to instruct the actions for the GUI (e.g. system calling your other scripts, closing the GUI, opening your plots etc)
6. Pretend you are creating a script for a research group who hates using a command line interface. Design your pipeline GUI to function seamlessly without the user doing anything on the command line except opening you GUI (e.g. python myFile.py)

Some tutorials

https://www.tutorialspoint.com/pyqt5/index.htm

https://build-system.fman.io/pyqt5-tutorial