Hi, we are building a risk system for a stock portfolio. We have a sample-data.py that gets our sample data and saves it to a csv file and we have a exploratory-analysis.py that does a exploratory analysis on the sample data. We also created a my-functions.py file which will store our functions for the risk system project. I'm now creating a GUI with pyqt, I have a script login.py that the user does the login, a script called data\_input that the user inputs data and a script called main.py that calls the other two. I now want that the user gets first the login window and then if the login is successfull he is directed to the data\_input window. Can you create a simple script for this? Show me only the code

Sure, here is a simple example of how this could be done using PyQt5:

Copy code

import sys from PyQt5.QtWidgets import QApplication, QDialog, QWidget class LoginWindow(QWidget): def \_\_init\_\_(self): super().\_\_init\_\_() self.initUI() def initUI(self): # add your code here to create the login window pass def onLoginSuccess(self): # open the data input window when login is successful data\_input\_window = DataInputWindow() data\_input\_window.show() class DataInputWindow(QDialog): def \_\_init\_\_(self): super().\_\_init\_\_() self.initUI() def initUI(self): # add your code here to create the data input window pass if \_\_name\_\_ == '\_\_main\_\_': app = QApplication(sys.argv) login\_window = LoginWindow() login\_window.show() sys.exit(app.exec\_())

In this example, the **LoginWindow** class represents the login window, and the **DataInputWindow** class represents the data input window. When the login is successful, the **onLoginSuccess** method is called and it opens the data input window.

This is just a basic example to give you an idea of how this could be implemented. You may need to modify the code to fit your specific needs.

Please give me only the code part to create the login window method

Here is the code for the **initUI** method of the **LoginWindow** class, which creates the login window:

Copy code

def initUI(self): # create the username and password inputs self.username\_input = QLineEdit() self.password\_input = QLineEdit() self.password\_input.setEchoMode(QLineEdit.Password) # create the login button and connect it to the onLoginSuccess method self.login\_button = QPushButton("Login") self.login\_button.clicked.connect(self.onLoginSuccess) # create a layout to hold the username and password inputs and the login button layout = QFormLayout() layout.addRow("Username:", self.username\_input) layout.addRow("Password:", self.password\_input) layout.addRow("", self.login\_button) # set the layout of the login window self.setLayout(layout)

This creates a **QLineEdit** widget for the username and password inputs, a **QPushButton** widget for the login button, and a **QFormLayout** to hold these widgets. When the login button is clicked, it calls the **onLoginSuccess** method.

You may need to modify this code to fit your specific needs, such as adding validation for the username and password.

How can I add validation for the username and password?