The Payoff Letter Fed script needs the ability to be run from a session or from Maui DUDE. For now, they both use a different version of RelfectionInterface making it difficult to run from both locations. The QInterface class was created as a wrapper class that will take in a Q.ReflectionInterface object and use it to create a new Uheaa.Common.ReflectionInterface object and call the main class. Here are a few things that are needed for this to work.

1. Both Uheaa.Common.dll and Q.dll need to be added to the project.
2. Your public override void Main() will be in the class that is inheriting the Uheaa.Common.Scripts.FedScriptBase project and will be built like any other c# application.
3. Q.dll will only be used in the QInterface class and it will inherit Q.FedScriptBase.
4. QInterface needs a constructor that receives a Q.ReflectionInterface, Q.MDBorrower and an int for the run number so MauiDUDE can find the constructor.
5. The public override void Main in QInterface will create a new Uheaa.Common.Scripts.ReflectionInterface object using the session being used in the Q.ReflectionInterface object being passed by MD.
6. Call the Main method in your non Q starting class passing the new RI.
7. To make the code easy to find, use the Fody package to merge all the dll’s that your project will use into one dll.
8. Open the NuGet package Manager for solution.
9. Search for package Fody. Install this package to the PYOFFLTRFD project.
10. Search for package Costura.Fosy. Install this package to the PYOFFLTRFD project.
11. Make sure project contains a file named FodyWeavers.xml with the following content:

<?xml version="1.0" encoding="utf-8"?>

<Weavers>

<Costura />

</Weavers>

1. Build the solution, Fody weaves all projects into single assembly.
2. Make sure to not include the Interop.Reflection.dll or the Q.dll as MD will use its own when calling the constructor in Q.interface.