For future reference, all these errors occurred because we are working with custom made Python IDE’s/libraries/tools. They are new, and are in the alpha/beta stage, thus are understandably slightly complicated to work with and install.  
  
Problems:

1. Python version conflicts
2. Import issues (unable to locate library)
3. Issues with sample scripts given by Gamry.

Solutions:

1. Python version conflicts:
   1. We followed the requirements of Gamry and used a Python 3.7.9 interpreter to run the NorthIDE. Additionally, we installed each given library wheel one by one using the 3.7.9 interpreter on the NorthIDE.
   2. Moving to 3.7.9 led to the “north” library being unable to be located. We simply copy-pasted 3.7.6 version of the “north” library into the site packages where the interpreter searched for “3.7.9” libraries. We have not run into any problems with that yet.
2. Import issues:
   1. Our thought process: 3.7.6 can’t be much different from 3.7.9
   2. The general rule that miraculously worked for libraries that are already installed but are for some reason inaccessible, simply:
      1. Locate the library (let’s call it L) in the file explorer
      2. Copy the entirety of L
      3. Figure out which folder (let’s call it x) Python is searching for packages
      4. Paste L into x
      5. Run and hope for the best
   3. Alternatively, import the built-in sys library and use sys.path.append() to add whichever folder the library is in to PATH, which basically tells Python to search in that folder for the library.
3. Issues with scripts, coding issues etc
   1. We realized there were many discrepancies between the sample scripts given and the actual library functions. E.g. the script calls “SetCell” but the library function is named “set\_cell”.
   2. There’s no quick workaround to this you’re gonna have to run the script one by one, locate the error and call the function by its correct name