

HercJobs Pre-Release Alpha

2024, S Johnson, westdalefarmer@gmail.com.
Free for whatever use, just be nice.

Pre-release notice: This product isn't finished. It might not even work properly, so please don't do anything rash like try to use it in a production environment or for something that will cause you distress.

This application has been written for my enjoyment. I make no claims about it's quality, code or even thought processes. It made me happy to create it, and if you can find it useful all the better.

NOTE: If your system does not already contain the .NET 8.0 runtime or SDK, you will need it. You can find it at this URL, <https://dotnet.microsoft.com/en-us/download/dotnet/8.0>

At this time, .NET 9 is out and I haven't experimented with it yet. This application has been targeted for the 8.0 releases.

Installation

Since you're reading this, I'm assuming you've already downloaded the .ZIP file containing the application and it's supporting files. Unzip it into the directory of your choosing, and if desired create a shortcut to HercJobs.exe for your desktop. That's it! This is Windows, so a lot of the weirdness is already taken care of.

Now for the more interesting parts. In order to use this application for submitting jobs, you'll need to make sure your copy of Hercules has the card reader set up as a TCP/IP sockdev. Most Turnkey installations already have this set up, however it's pretty simple.

For the turnkey systems (TK4-, TK5 and possibly TK3) you'll find it in your distributions conf directory. For example, my tk5, it's conf/tk5.cnf **BACK UP YOUR CONFIGURATION FILE BEFORE YOU MODIFY IT!!! PRACTICE SAFE COMPUTING!**

Look for the 3505 device line which will look something similar to this. Again, this is from TK5

```
000C 3505 ${RDRPORT:=3505} sockdev ascii trunc eof
```

In some cases, it might not have the RDRPORT established and look like this:

```
000C 3505 3505 sockdev ascii trunc eof
```

This establishes device 000C as a 3505 (card reader) listening on port 3505 as a sockdev using ascii and some parameters to make things play nice. If your configuration is missing this line, or the 3505 is defined a different way, you can add or alter the line to match that above. If something on your system is already using port 3505, you can make it anything that works for you. Again, the port in the above line is the SECOND 3505 on the line.

As I mentioned, most of the turnkeys already have this set up.

Next up, you'll have to decide if you want this program to act as one or more of your system printers. If you do, **all** printer activity for the device you link will be directed to this program. Please be aware of that. It does NOT distinguish between your submitted job's output and any other output generated by the system for that printer. In your Hercules configuration, follow the same process for the card reader, but without all the extra parameters. It's enough to just change it to <port> sockdev.

For example, mine are:

```
000E 1403 1403 sockdev
000F 1403 1404 sockdev
010E 1403 1405 sockdev
```

You probably won't have the line for device 010E. That's an additional printer I've added to my TK5's JES2 configuration for other uses. In any event, that sets up 3 printers, the usual 2 that the turnkey's establish and my third. The first printer, 000E is a 1403 line printer, listening on port 1403. The second 000F is also a 1403 listening on 1404. I'm sure you can see the pattern here. Again if something else is using those ports, you can change the ports in your configuration.

IMPORTANT! In most turnkey installations, device 0002 and device 030E are also defined as printers. Please don't mess with those two devices. They are there to help you figure out what's going on in case of trouble.

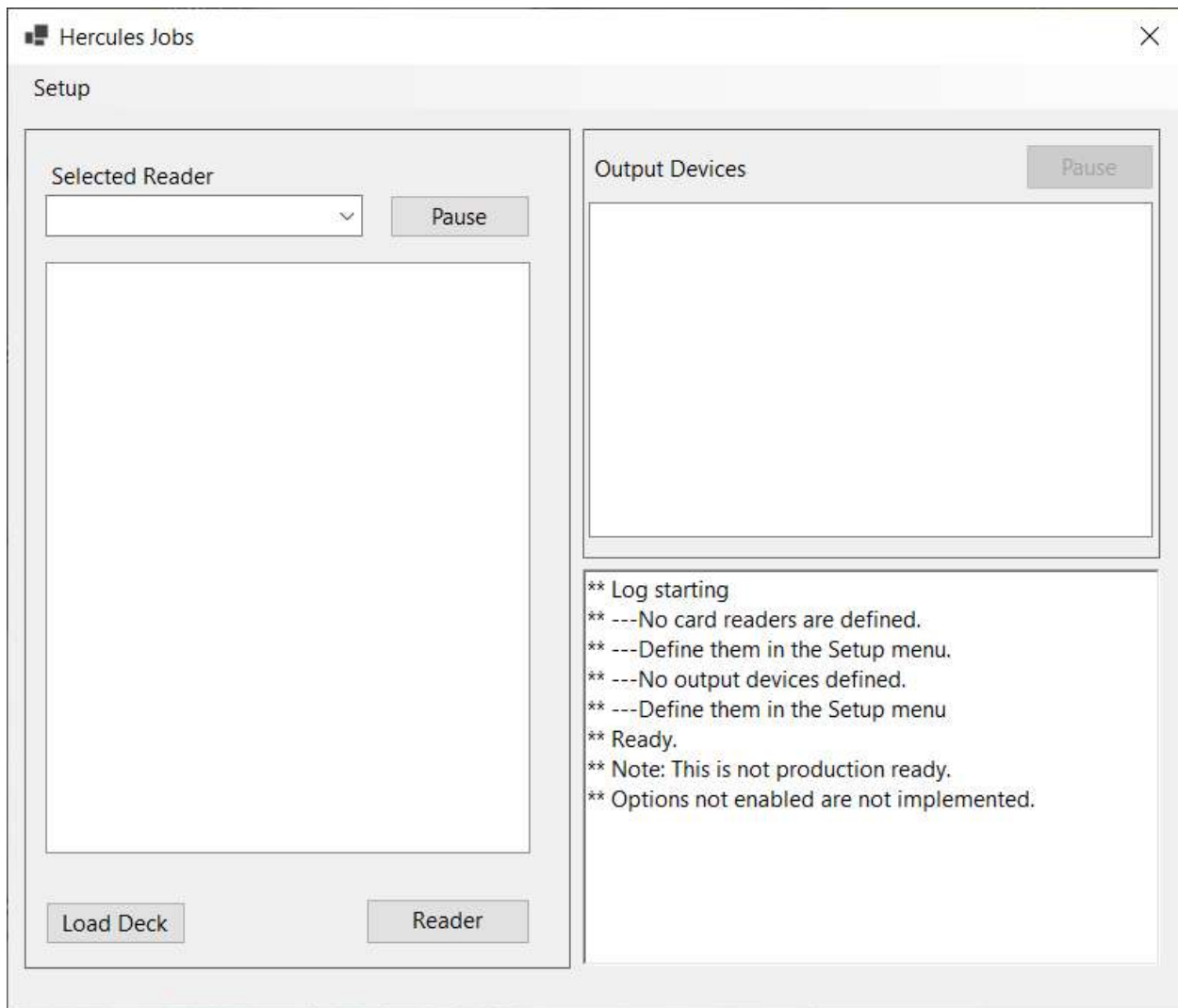
While it's not yet implemented, you can also set up your card punch as a sockdev device in the same manner. Mine is

```
000D 3525 3525 sockdev ascii
```

Save your configuration file, and restart Hercules. **BE SURE TO NICELY SHUT DOWN YOUR MAINFRAME IF IT'S RUNNING!**

APPLICATION CONFIGURATION

When you start the application for the first time, there are no readers or printers defined.



These are defined in the Setup Menu, Configuration and the Input/Output Device menu options. Each has a window with prompts to configure your devices.

3505 Card readers

<add new reader>

Remove

Save

Close

Friendly name

Host or IP address

Port

Output Devices

<add new output device>

☐ IBM 1403 Printer ☐ Auto print

☐ IBM 3525 Punch

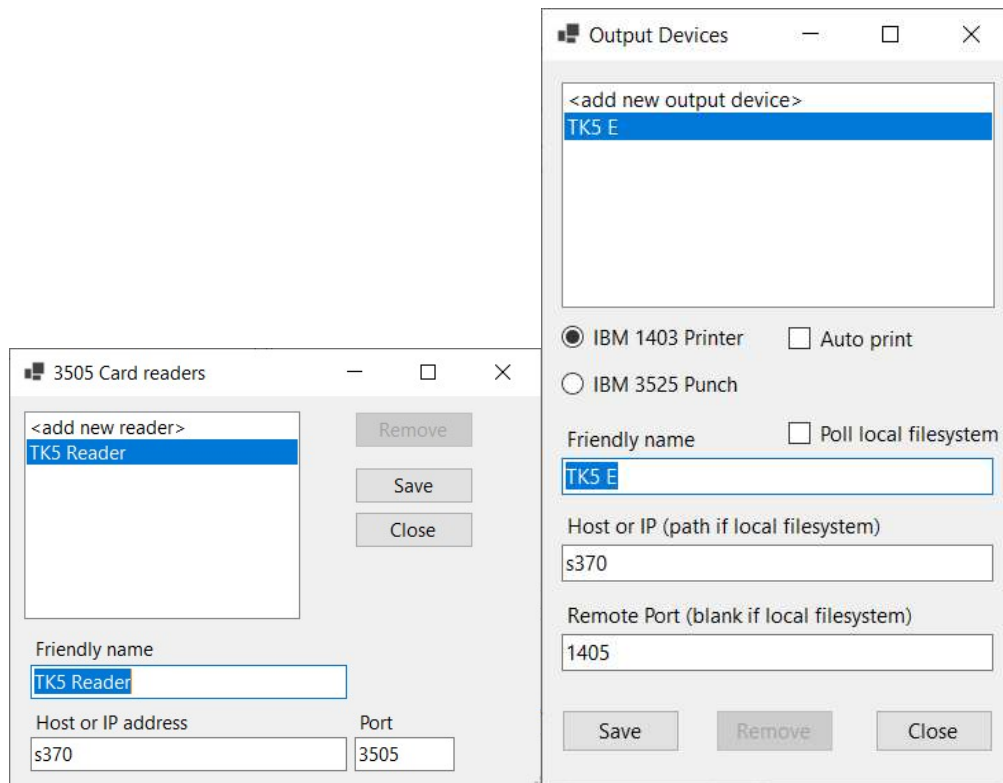
Friendly name ☐ Poll local filesystem

Host or IP (path if local filesystem)

Remote Port (blank if local filesystem)

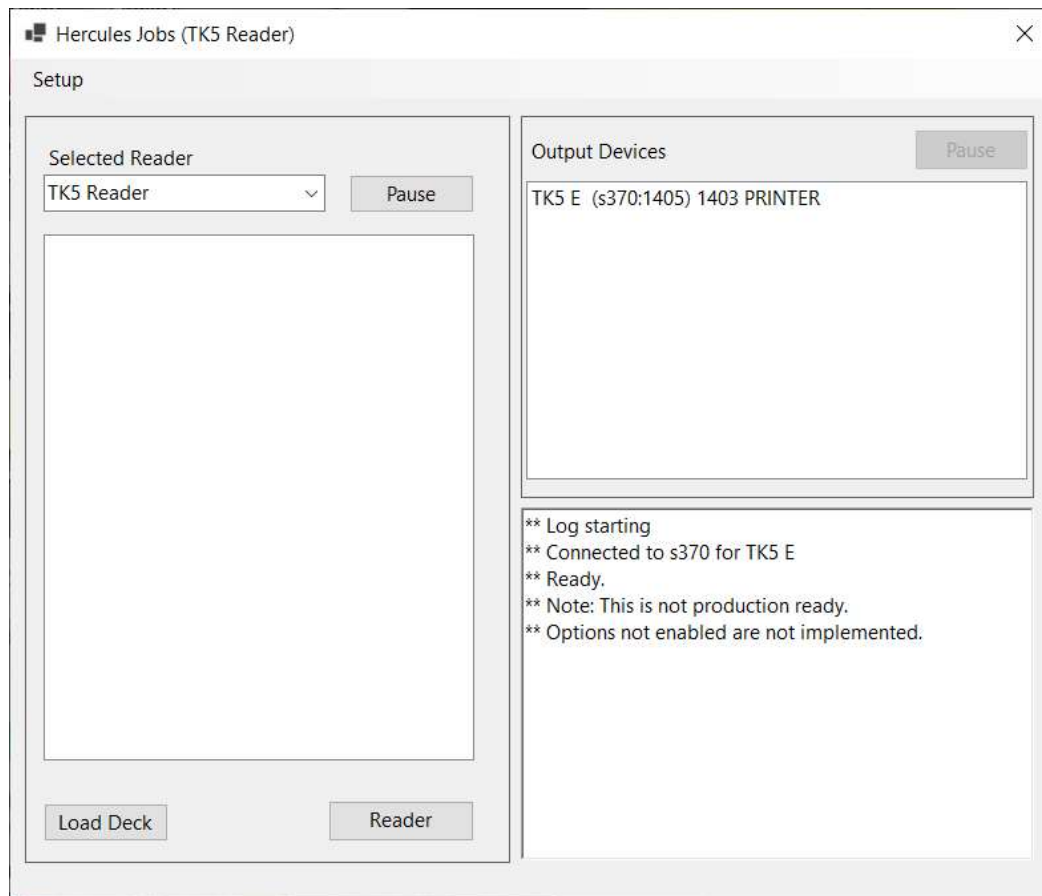
Save Remove Close

Both windows work roughly the same. To add a new device, click on the <add new...> option in the listbox, and fill out the relevant information below. Then press the Save Button. Once you do this, you can edit an existing device by clicking on it's entry in the listbox and changing the information that is presented in the area below, then again Save the changes. Once you save this information for both readers and outputs, they will be saved in the same directory as the program as an xml class file to reload each time you start the program. If you're brave and want to try to alter the information in them, you're welcome to. Be aware however, that if you mess it up, .NET won't be able to deserialize these objects. You'll either have to fix them, or delete them and start over again.



These are what mine look like. Note, my printer is set up to use that special third printer I set up. Yours would be pointing to port 1403 or 1404.

Now, close the application and restart it. You should see your devices connect and become ready.

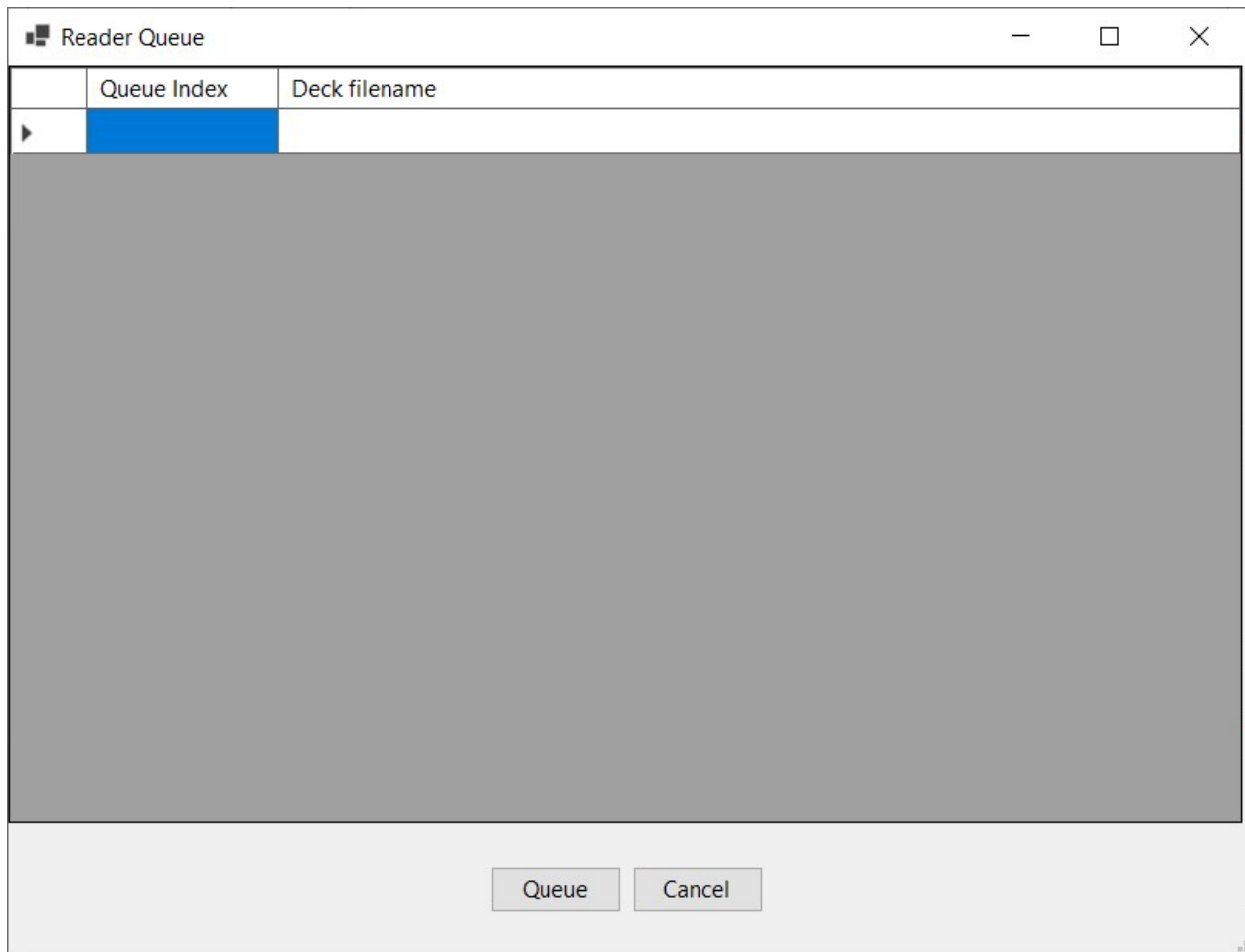


From here, you can use the “Load Deck” button to browse your system to load a “deck” into your reader. Unless you’ve pressed the Pause button beside the reader name, the job will be immediately submitted to your Hercules instance. If you have the correct printer defined, you’ll see the output pop up on your screen in a separate window, and in the log window you’ll see the name of a PDF file of that generated output. If you’ve pressed the Pause button, the listbox window to the left will be yellow, indicating that jobs will not be sent to the reader until you press the “Start” button.

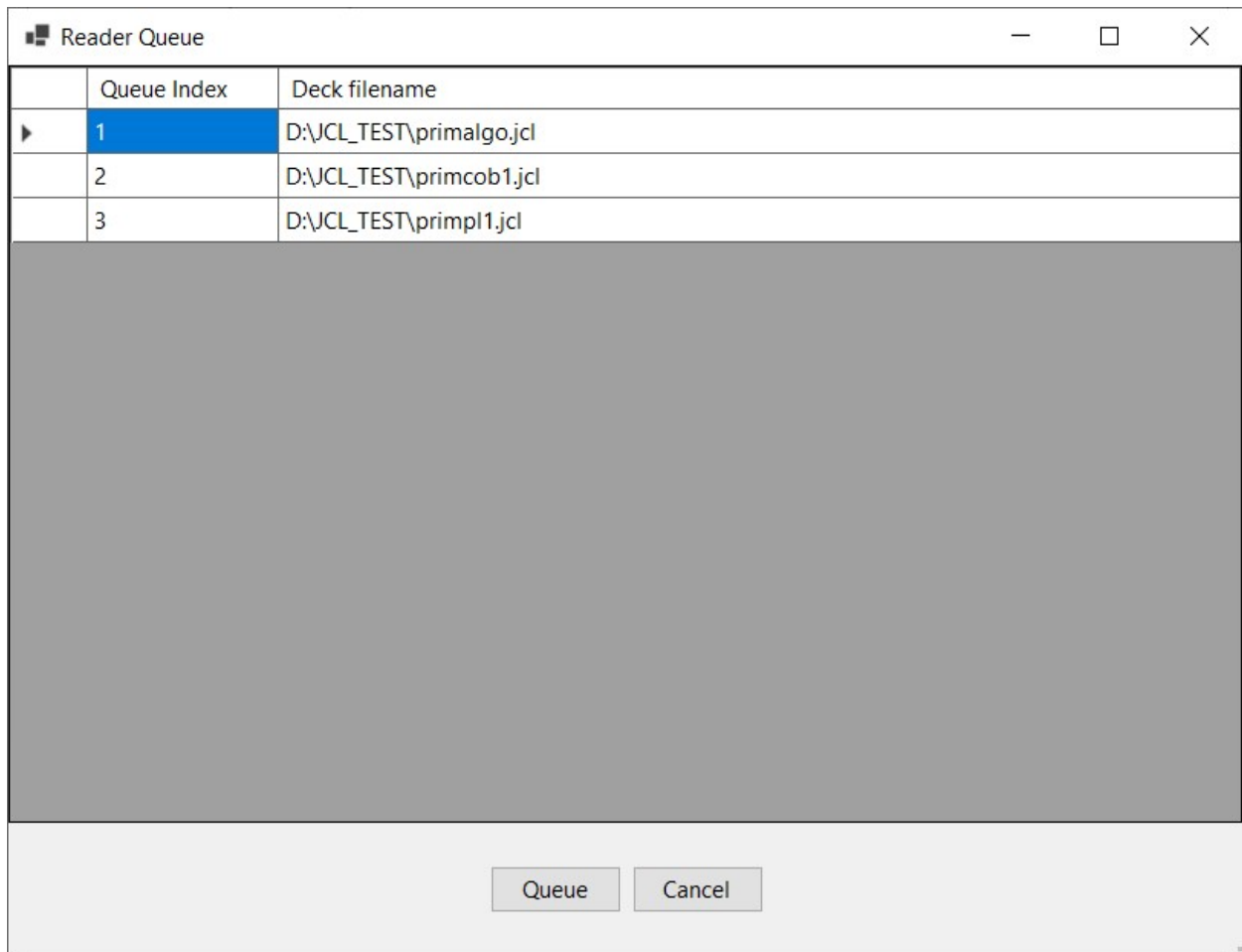
Additional methods to submit jobs.

You can also drag/drop file from your explorer window directly into the job queue listbox. This is done the usual way. I’m assuming that if you’ve gotten this far in your computer life, drag and drop is a second nature, so I won’t discuss it any further.

The “Reader” button is for submitting more than one job at a time, with the ability to specify the order the jobs will be submitted.



On this screen, select a cell as shown above, and right click. This will show you a drop down menu with the option to Select Files, or clear the list. Select files, will allow you to browse your system for the card decks you want to submit. This dialog allows multiple files to be selected. Once selected, the application adds them to the grid in the order that the browser dialog has returned them.



Reader Queue

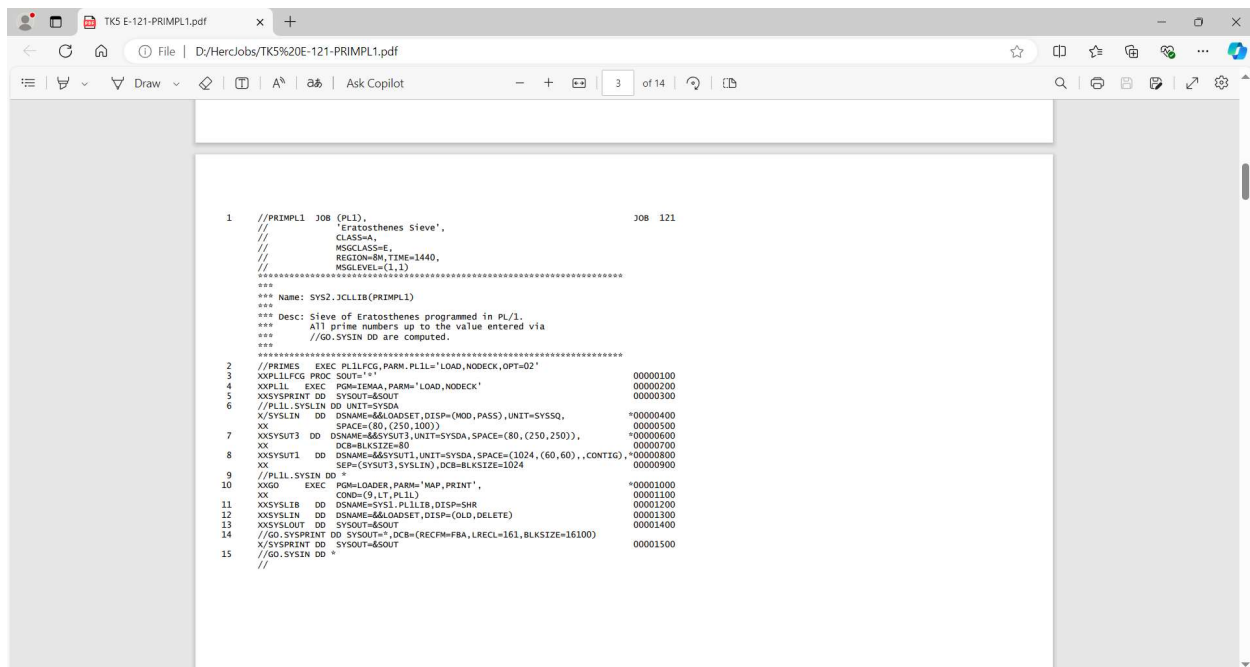
	Queue Index	Deck filename
▶	1	D:\JCL_TEST\primalgo.jcl
	2	D:\JCL_TEST\primcob1.jcl
	3	D:\JCL_TEST\primpl1.jcl

Queue Cancel

At this point, selecting any cell and right clicking to the menu will allow you to move that cell up or down the list to change the order they're submitted. Once you have the order to your liking, press the Queue button to add them to the prior screen's list. Note, when you start this window a dialog asks you about Pausing the reader. It's a good idea if you want to have a bit more control over when the jobs are submitted. If you did not elect to pause the reader, the decks will begin to send as soon as you click "Queue".

[illegible]

For a PDF output, you'll see the output file in the log window. It should be (for this job) TK5 E-121-PRIMPL1.pdf which is the Printer friendly name, the job number, and the jobname.



I hope this helps you get started with the program. Enjoy!