**Executing saved files:**

I had a general play with the console then with a customtkinter textbox and button.

The subprocess library may work but doesn't output anything, so you can't tell. The call, run and Popen all seem to have this problem. Using the check\_output function and then printing out what it returns results in "b'hello world\r\n'" on the console. After investigation, it seems to be more for running command line stuff etc rather than python files.

The os.system executes the command and produces output to the console if the file is plain, but didn't want to work once tkinter was introduced.

Reading in the file and using exec also worked well in the same way as os.system, but without stopping working when tkinter was introduced.  In tkinter (before an exception was added in), exceptions threw warnings, but it kept running.

Exec also works when just passing the contents of a textbox to it, meaning that saving to a file first isn't necessary for functioning, so it just depends on if we want to do it anyway.

**Syntax highlighting:**

There are a few libraries out there. The common ones seem to be using idlelib or pygments. Pygments seems a little bulky and involved more code to use. Idlelib seems a little more streamlined (at least on this end) and seems suitable for the project.

A screenshot of a computer

AI-generated content may be incorrect.

HTML

HTML pages for instructions. Sadly no nav in tkhtmlview, links go outside of window. Can't find a way to link within window aka for contents (works as HTML on own)

Tried webview (pip install pywebview) instead and contents work, presumably because it treats it as a web page so more interactive. Causes app to crash. Must be run on main thread

Scratch the contents thing. It would appear that once linked up to the main app, the choices are a non-working contents part or an application that crashes if you try to click on it when the html window is open. I'm going to prioritise the not crashing