**Create a Development Environment Exercise**

Patrick Southcott

Artificial Intelligence for Human-Computer Interaction (MSAI-631-B01)

Prof. Dennis

University of the Cumberlands

26 October 2025

**Exercise: Create a Development Environment Exercise**

A development environment can be created using virtual environments. Figure 1 shows checking the python version, creating a virtual environment, activating the virtual environment, creating a “hello world” python app, and running the app.

Figure 1.

A screen shot of a computer

AI-generated content may be incorrect.

The project can also be viewed in Pycharm. See Figure 2 for a screenshot of the project running in PyCharm.

Figure 2.

A screenshot of a computer

AI-generated content may be incorrect.

PyCharm makes it easy to create Jupiter Notebooks. See figure 3 for a screenshot of creating a new Jupyter Notebook in PyCharm.

Figure 3.

A screenshot of a computer

AI-generated content may be incorrect.

After the notebook is created, it can run “hello world” as seen in Figure 4.

Figure 4.

A screenshot of a computer

AI-generated content may be incorrect.

To demo required libraries, we can create a requirements.txt file and install it with pip, as shown in Figure 5.

Figure 5.

A screenshot of a computer program

AI-generated content may be incorrect.

Figure 5 shows installing the “termcolor” library. The hello.py script was updated to use the termcolor library, and the output in show in cyan which demonstrates it working.

My github username is p-s-dev ( <https://github.com/p-s-dev> ).

The repo I will use for this class is: <https://github.com/p-s-dev/MSAI-631-B01_Artificial-Intelligence-for-Human-Computer-Interaction>

# Conclusion

Virtual Environment and PyCharm are effective tools for developing Python apps. As Sakib (2025) notes, “Modern software development heavily relies on third-party packages to accelerate progress, yet two critical challenges persist: managing dependency conflicts during package installation and addressing the frequent absence or incompleteness of configuration files in Python projects.” (p. 1).

# References

Sakib, S. J. (2025). Toward better dependency management in Python projects (Master’s thesis, University of Windsor). ProQuest Dissertations & Theses Global. <https://www.proquest.com/openview/27800550fa4f12f51d380297902def0e/1?pq-origsite=gscholar&cbl=18750&diss=y>