**Socket Program Documentation**

|  |
| --- |
|  |

**Execution**

To execute our socket program, ensure Python is installed (preferably Python 3.6 and above).

During testing and execution, we decided to create a Python virtual environment to ensure proper dependency management and project isolation. Instructions on how to create and use a virtual environment can be found [here](https://docs.python.org/3/library/venv.html) (optional).

Ensure source code file **SocketProgram.py** is downloaded and easily accessible.

Our program takes in the **hostname**as a command line argument.

In a command line terminal, enter one of the following commands based on optional usage of virtual environment:

* python “{*path to source code file*}” {*hostname*}
* “{*virtual environment directory*}\scripts\python” “{*path to source code*}” {*hostname*}

As an example, the command we used during testing is as follows:

* “C:/Users/Luke/Documents/venv/Scripts/python” "f:/PennWest/Fall 3/NET/SocketProgram/SocketProgram.py" DRACO1

The terminal will now execute the Socket Program.

**Details**

Our program uses the standard protocol of {*code*}#{*message*}.

Code **1** will enforce echo and code **2** will enforce no echo.

Input **0** will exit any input prompt and safely exit the program.

Program will check for and enforce correct protocol format and code.

A screenshot of a computer program

Description automatically generated**Demonstration**

**Contributions**

Luke contributed to virtual environment setup, socket utilization, and code/typing documentation.

Adir contributed to handling clean input, control loop, and error handling.

We both contributed to code cleanup and refactoring, testing, and bug fixes.