This document provides instructions on how to install and enter the environment to run the Genelytics pipeline.

Requirements:

- Must run on x86 architecture (i.e. Windows machines)
- Windows Subsystem for Linux, WSL (install instructions below)

Steps:

1. Download the project repository:
    - Download project repository from Dr. Almeyda or from the project github: https://github.com/brooks-hw/Bioinformatics-Capstone-CS-490
2. Install WSL
    - Press the windows key (or enter search bar)
    - Type “powershell,” right click and “run as administrator
    - Type the command: “wsl –install”
    - Restart PC to properly ensure installation
3. Install Ubuntu:
    - A command shell labeled “Ubuntu” may pop up, if so skip next step
    - If not → Enter search bar again and type “Ubuntu”
    - It will then install and likely take a few minutes
    - Create a username and password when prompted (Simple password helps, doesn’t need to be super secure, especially if using personal computer)
    - Once complete, you are now inside a Linux virtual environment while on windows.
4. Update Ubuntu Packages:
    - Type “sudo apt update && sudo apt upgrade -y” to pull the most recent versions of necessary packages. (this may require you to type in password)
    - This shell will be your work environment, all tools will be hosted and run here
5. Download Miniforge:
    - Run: “cd ~” to navigate to the root directory
    - Run the following command (inside of the Ubuntu window): ‘wget https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-Linux-x86_64.sh -O miniforge.sh’

Followed by: ‘bash miniforge.sh’

- Type ‘yes’ when asked to accept the license
- Press ‘Enter’ when prompted to install to /home/username/miniforge3
  (this makes Conda visible to your PC upon restarting the terminal)
- Type ‘yes’ when asked to ‘initialize conda?’
- Close your Ubuntu terminal and reopen it to activate conda

6. Install Mamba (faster version of Conda for some parts):
   - Run: ‘conda install -n base mamba -c conda-forge -y’
   - If successfully installed, the project environment is ready to be built
7. Create the Genelytics Conda environment:
   - Navigate to the project directory (wherever it’s located on your computer), to do this type: **“cd /mnt/c/Users/<WINDOWS_USERNAME>/”**
   - After the last / type “Desktop” or “Downloads” or wherever you have the project folder. At each step of the way, you can type “ls” to see what files/folders are currently in that directory. From there you can “cd” into them as necessary
   - An example entire command may be:
     “cd /mnt/c/Users/John/Desktop/Genelytics-Project”

     **Note that your project folder may be called something different (feel free to rename it to something easy to type)*
   - Run: “mamba env create -f environment.yml” to create the mamba environment containing the necessary dependencies

     **This will likely take a few minutes to install everything*
   - Activate the environment by running: “conda activate genelytics”
   - You should now see something like:

     (genelytics) student$DESKTOP:~$
8. After entering the environment, you are ready to run the pipeline.
   - Ensure you are currently inside the same directory (folder) that contains “ProcFast.py”
   - While still inside the genelytics environment, run:
     “python ProcFast.py –port 500X” to initialize the program
   - Each user will need to use a different port, so change the command accordingly (example –port 5000, –port 5001, –port 5002)
   - It should now initialize and say something such as:
     Running on http://127.0.0… along with other info.

    - Open a browser and navigate to: http://localhost:500X (use the same port number as initialized with above) and the dashboard will appear.
    - Once here, use the tools as necessary via the upload boxes, the outputs will be located in the same folder as your main project

9. After you're finished:
    - Run CTRL+C to stop the program from running
    - Close Ubuntu

Additional Info/Comments:

- The command line can be very specific, if you're getting errors at any step, copy them (with CTRL+C) and give them to ChatGPT or any other LLM and it will most likely help you find the root issue.
- Most steps listed here should be general knowledge to CS students so I recommend reaching out to someone in CS if you encounter any issues along the way.
- If a tool fails, the pipeline website should output which command it tried to run. You may run that same command directly in the Ubuntu terminal to see what went wrong and try to debug it.