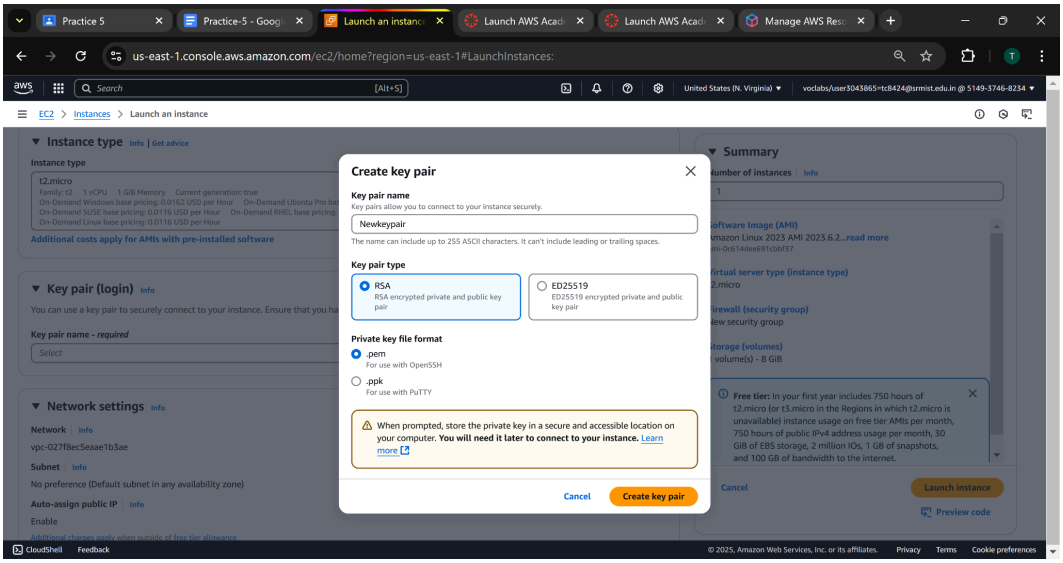
UNIT- 2

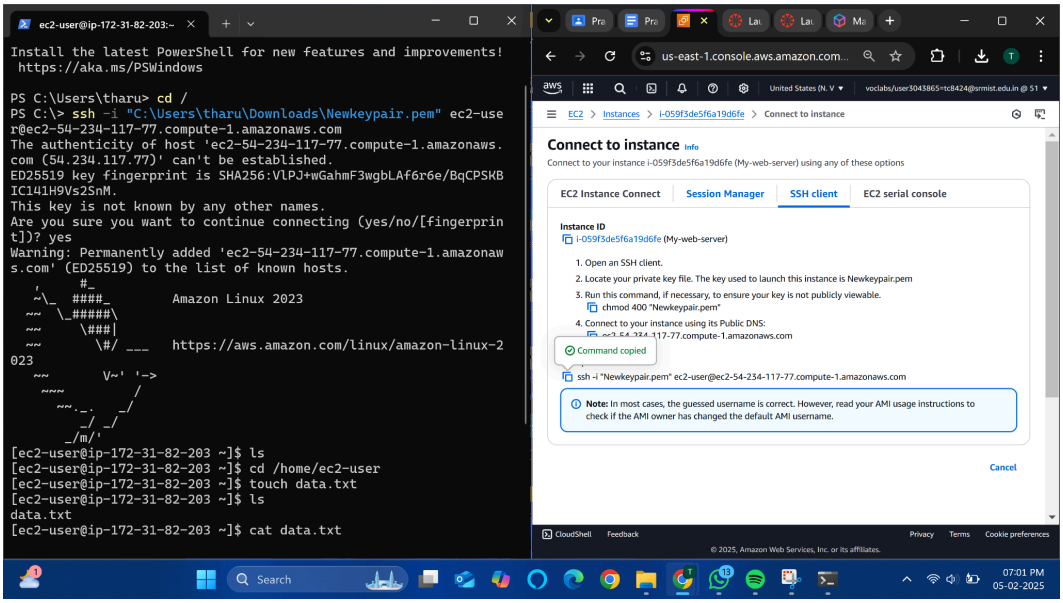
Practice – 5

Tharun Subramanian C
RA2211003011187

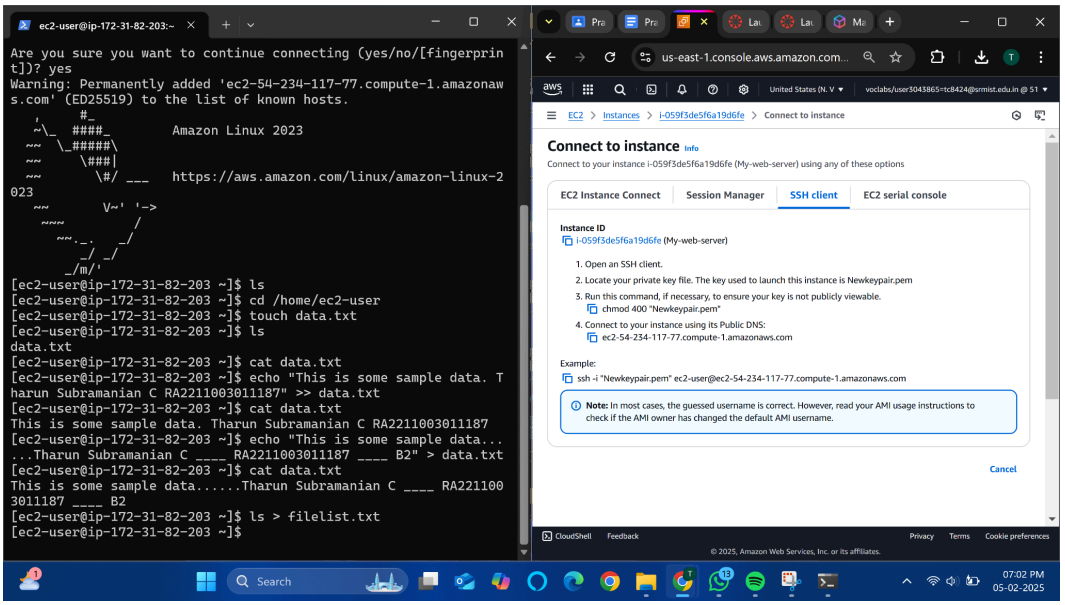
B2

1. **File Manipulation:**
    - List the contents of your current directory using `ls`.
    - Change directory to a different location using `cd`.
    - Create a new empty file named `data.txt` using `touch`.
    - View the contents of `data.txt` using `cat`.





2. **Redirection Operators:**
    - Append the following line to `data.txt` using `echo "This is some sample data" >> data.txt`.
    - Overwrite the contents of `data.txt` with the same line using `echo "This is some sample data" > data.txt`.
    - Redirect the standard output of the `ls` command to a new file named `filelist.txt` using `ls > filelist.txt`.



3. **Standard Streams (STDIN, STDOUT, STDERR):**
    - Explain the difference between Standard Input (STDIN), Standard Output (STDOUT), and Standard Error (STDERR).

Answer:
STDIN (Standard Input): This is where input comes from (usually the keyboard).

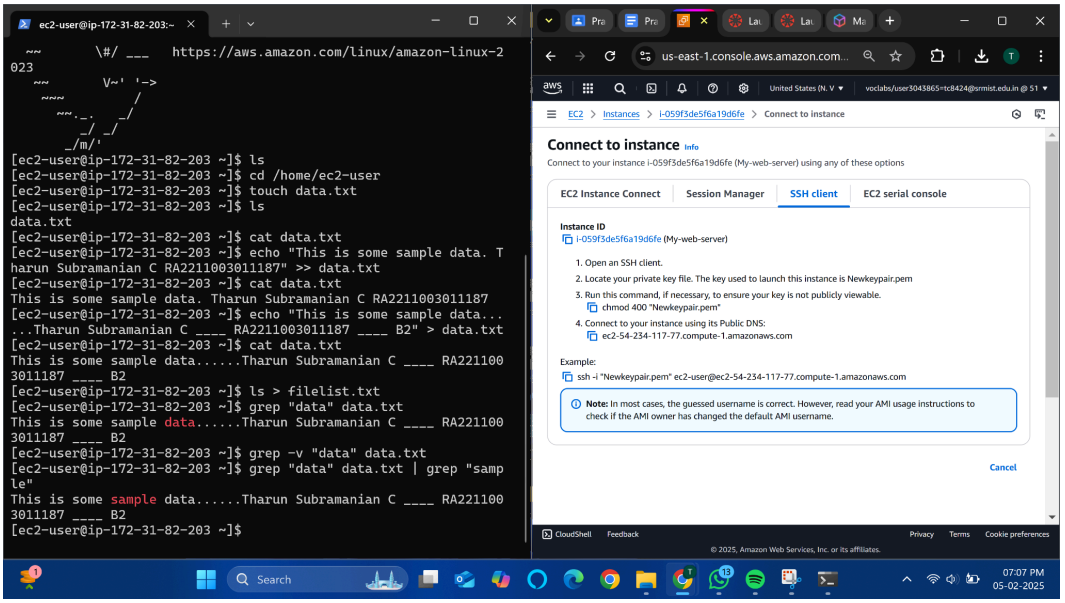
STDOUT (Standard Output): This is where output goes (usually the terminal/screen).

STDERR (Standard Error): This is where error messages go.

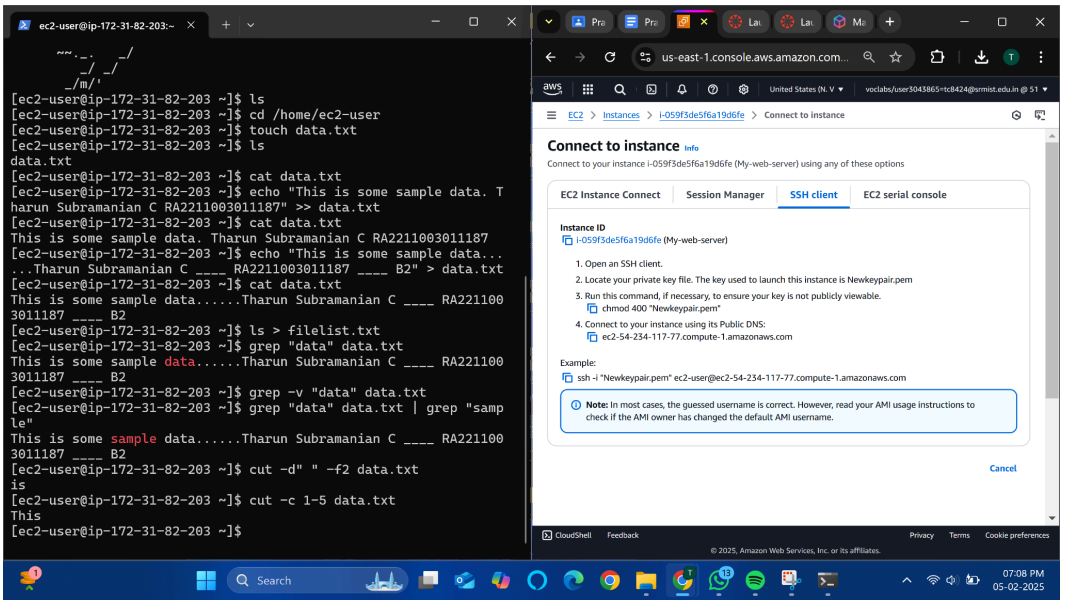
    - Provide an example of how each is used in a shell command.

Answer:
STDIN: Inputting text via “cat” or a script.
STDOUT: Output from commands like “ls” or “echo”.
STDERR: Errors from commands like “cat non_existent_file”.

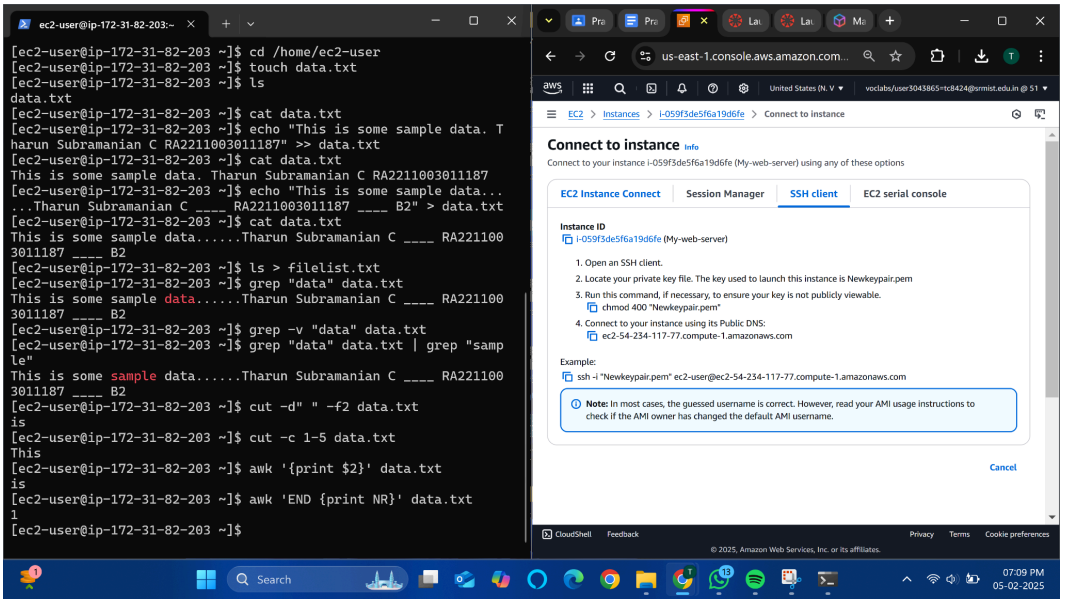
4. **Grep Command:**
    - Use `grep "data"` on `data.txt` to find lines containing the word "data".
    - Search for lines that **don't** contain "data" using `grep -v "data" data.txt`.
    - Combine grep with pipes (`|`) to search for lines with "data" and then filter for those containing "sample" using `grep "data" data.txt | grep "sample"`.



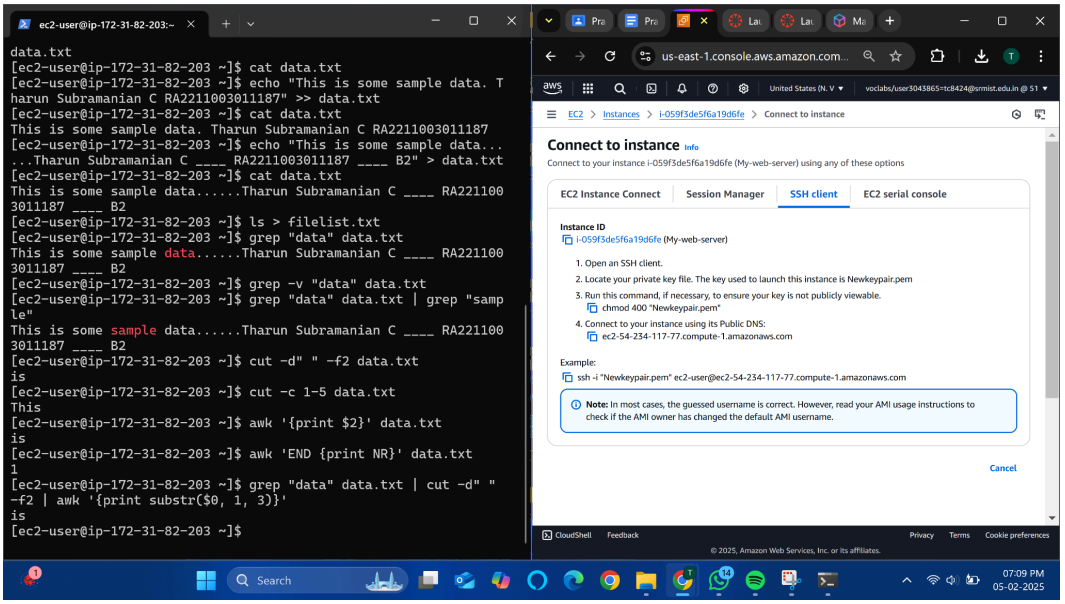
5. **Cut Command:**
    - Use `cut -d" " -f2` on `data.txt` to extract the second word (separated by spaces) from each line.
    - Modify the command to extract a specific range of characters (e.g., first 5 characters) using `cut -c 1-5 data.txt`.



6. **Awk Command:**
    - Print only the second field from `data.txt` using `awk '{print $2}' data.txt`.
    - Count the number of lines in `data.txt` with `awk END {print NR}' data.txt` (NR refers to the number of records).



7. **Command Chaining:**
    - Combine `grep`, `cut`, and `awk` to find lines containing "data", extract the second word, and print only the first three characters using a single command with pipes.



Instance Termination: