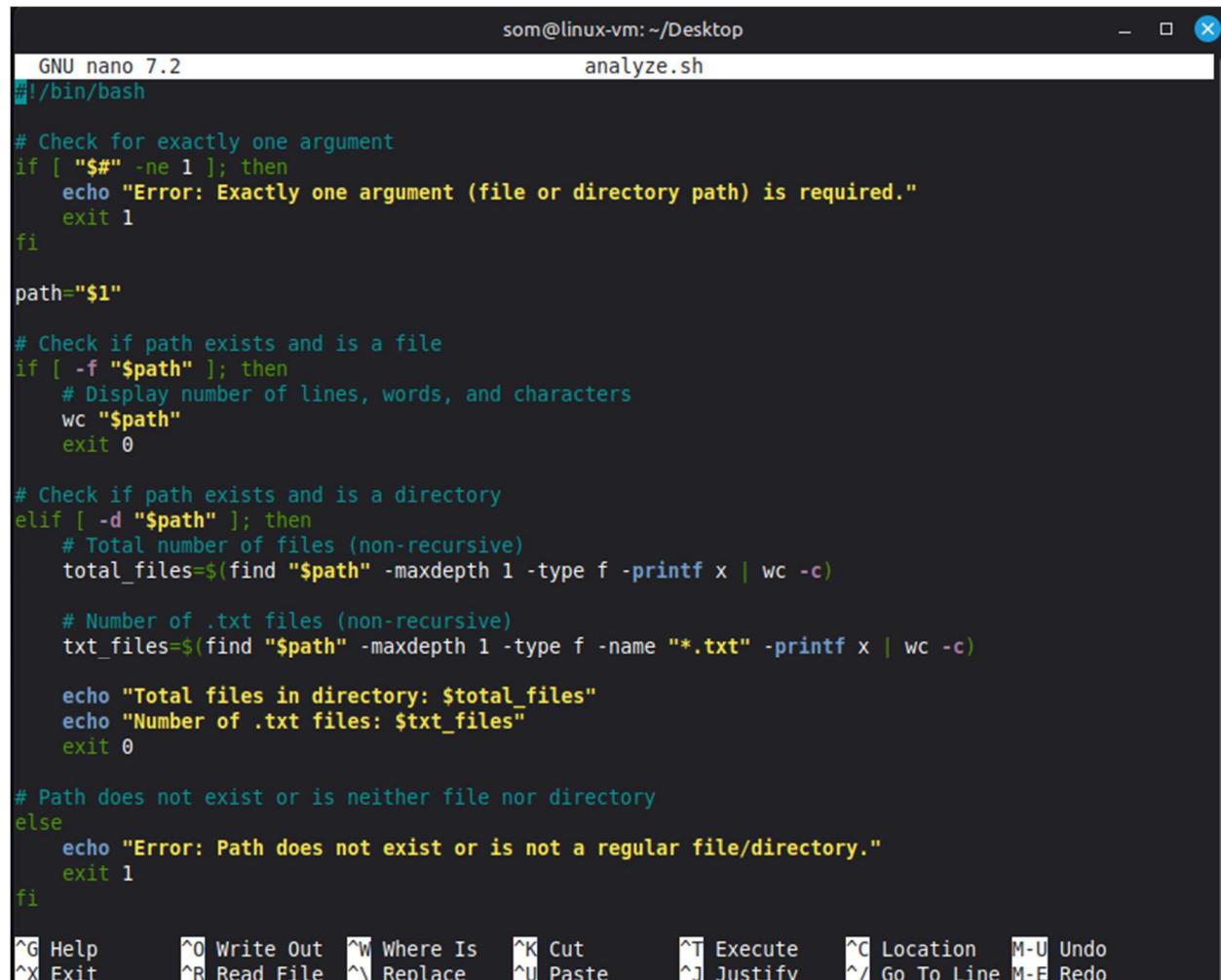


Question 1 (2024eb03003):

Please find screenshot of shell script below and attaching analyze.sh script to GitHub repository:



The screenshot shows a terminal window titled "analyze.sh" running on a Linux VM. The window title bar says "som@linux-vm: ~/Desktop". The terminal content displays a shell script named "analyze.sh" written in nano editor version 7.2. The script performs several checks on a provided argument (file or directory path) and outputs the total number of files and .txt files in the directory.

```
GNU nano 7.2                               analyze.sh
#!/bin/bash

# Check for exactly one argument
if [ "$#" -ne 1 ]; then
    echo "Error: Exactly one argument (file or directory path) is required."
    exit 1
fi

path="$1"

# Check if path exists and is a file
if [ -f "$path" ]; then
    # Display number of lines, words, and characters
    wc "$path"
    exit 0
else
    # Check if path exists and is a directory
    elif [ -d "$path" ]; then
        # Total number of files (non-recursive)
        total_files=$(find "$path" -maxdepth 1 -type f -printf x | wc -c)

        # Number of .txt files (non-recursive)
        txt_files=$(find "$path" -maxdepth 1 -type f -name "*.txt" -printf x | wc -c)

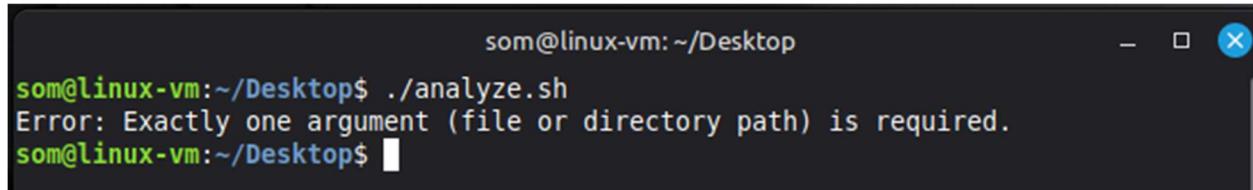
        echo "Total files in directory: $total_files"
        echo "Number of .txt files: $txt_files"
        exit 0
    else
        echo "Error: Path does not exist or is not a regular file/directory."
        exit 1
    fi
fi

^G Help      ^O Write Out  ^W Where Is   ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Paste    ^J Justify   ^/ Go To Line M-E Redo
```

Testing the analyze.sh Script

Test 1: Invalid number of arguments

```
./analyze.sh
```



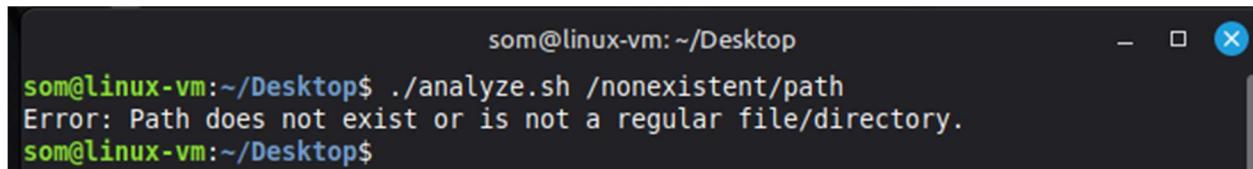
```
som@linux-vm:~/Desktop$ ./analyze.sh
Error: Exactly one argument (file or directory path) is required.
som@linux-vm:~/Desktop$
```

A screenshot of a terminal window titled "som@linux-vm: ~/Desktop". It shows the command "./analyze.sh" being run. The output is an error message: "Error: Exactly one argument (file or directory path) is required." followed by a new line and the prompt "som@linux-vm:~/Desktop\$".

Script displays "Error: Exactly one argument (file or directory path) is required." and exits with code 1. This validates the argument count check works correctly.

Test 2: Non-existent path

1. `./analyze.sh /nonexistent/path`



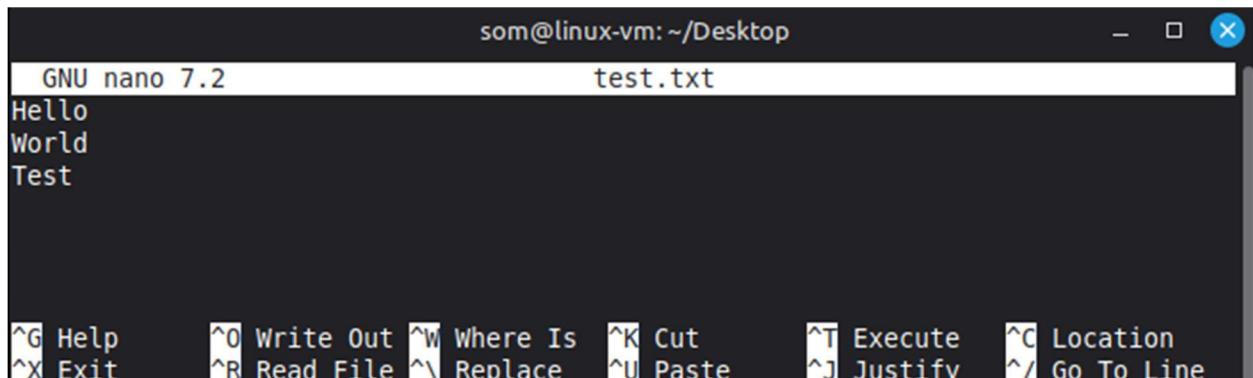
```
som@linux-vm:~/Desktop$ ./analyze.sh /nonexistent/path
Error: Path does not exist or is not a regular file/directory.
som@linux-vm:~/Desktop$
```

A screenshot of a terminal window titled "som@linux-vm: ~/Desktop". It shows the command "./analyze.sh /nonexistent/path" being run. The output is an error message: "Error: Path does not exist or is not a regular file/directory." followed by a new line and the prompt "som@linux-vm:~/Desktop\$".

Script outputs "Error: Path does not exist or is not a regular file/directory." and exits with code 1. Proper error handling for invalid paths confirmed.

Test 3: Regular text file

```
echo -e "Hello\nWorld\nTest" > test.txt
```



```
GNU nano 7.2
Hello
World
Test

^G Help      ^O Write Out  ^W Where Is   ^K Cut      ^T Execute  ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste    ^J Justify  ^/ Go To Line
```

A screenshot of a terminal window titled "som@linux-vm: ~/Desktop". It shows the command "echo -e \"Hello\nWorld\nTest\" > test.txt" being run. The output shows the contents of the file "test.txt": "Hello", "World", and "Test". Below the file content is a menu bar for the nano editor with various keyboard shortcuts. At the bottom of the screen, there is a status bar with the same set of keyboard shortcuts.

```
./analyze.sh test.txt
```

```
som@linux-vm:~/Desktop
som@linux-vm:~/Desktop$ echo -e "Hello\nWorld\nTest" > test.txt
som@linux-vm:~/Desktop$ ./analyze.sh test.txt
3 3 17 test.txt
som@linux-vm:~/Desktop$
```

Breakdown:

- 3 lines: "Hello", "World", "Test" (each terminated by \n)
- 3 words: One word per line (no multi-word lines)
- 17 characters: H-e-l-l-o + \n + W-o-r-l-d + \n + T-e-s-t + \n = 17 bytes total

The wc output 3 3 17 test.txt is perfectly correct for this file content. The script analyze.sh works exactly as designed - it passes the file directly to wc, which produces standard output format matching the requirements

Test 4: Directory with mixed files

```
mkdir testdir
touch testdir/file1.txt testdir/file2.txt testdir/image.jpg testdir/doc.pdf
./analyze.sh testdir
```

```
som@linux-vm:~/Desktop
som@linux-vm:~/Desktop$ mkdir testdir
som@linux-vm:~/Desktop$ touch testdir/file1.txt testdir/file2.txt testdir/image.jpg testdir/doc.pdf
som@linux-vm:~/Desktop$ ./analyze.sh testdir
Total files in directory: 4
Number of .txt files: 2
som@linux-vm:~/Desktop$
```

Shows "Total files in directory: 4" and "Number of .txt files: 2" accurately. Directory counting logic performs correctly (non-recursive).

Test 5: Empty directory

```
mkdir emptydir
./analyze.sh emptydir
```

```
som@linux-vm:~/Desktop$ som@linux-vm:~/Desktop$ mkdir emptydir
som@linux-vm:~/Desktop$ ./analyze.sh emptydir
Total files in directory: 0
Number of .txt files: 0
som@linux-vm:~/Desktop$
```

Reports "Total files in directory: 0" and "Number of .txt files: 0". Handles edge case properly.