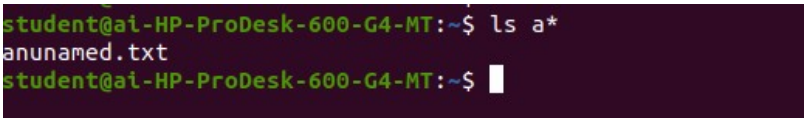
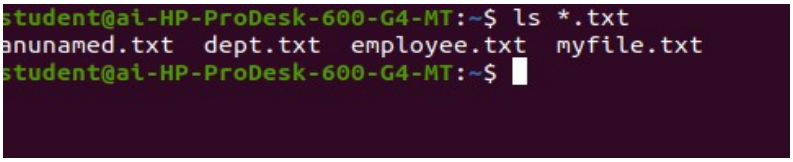
LAB TASK

423132

Question 1:
List all files in the current directory that start with the letter a.

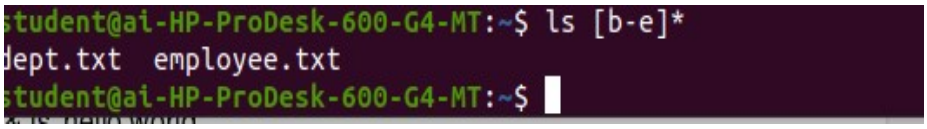


Question 2:
List all files in the current directory with a .txt extension.



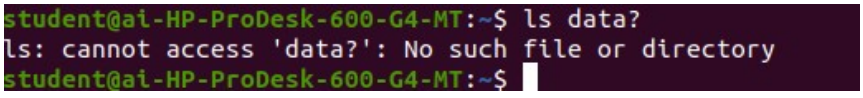
Question 3:
Find all files in the current directory whose names have exactly five characters.

Question 4:
List files in the current directory that start with any letter between b and e.

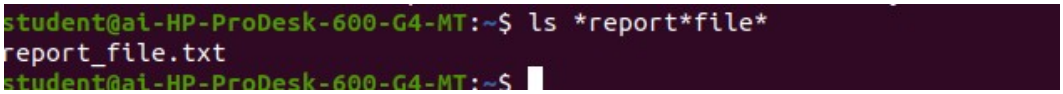


Question 5:
List files in the current directory ending with .log, where the second character is a number (e.g., a1.log).

Question 6:
List all files in the current directory whose names start with data followed by a single character (e.g., data1, dataA).



Question 7:
Find all files in the current directory whose names contain both file and report in any order.



LAB TASK

423132

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls *file*report*
file_report.txt
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 8:
Create a file with the name hello world and use quoting to list it.

```
student@ai-HP-ProDesk-600-G4-MT:~$ touch 'hello world' && ls 'hello world'
'hello world'
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 9:
Use quoting to echo the string I love $HOME, first interpreting $HOME and then treating it as literal text.

```
student@ai-HP-ProDesk-600-G4-MT:~$ echo "I love $HOME"
I love /home/student
student@ai-HP-ProDesk-600-G4-MT:~$ echo 'I love $HOME'
I love $HOME
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 10:
Create a file named file\with\slashes and list it using backslashes.

```
student@ai-HP-ProDesk-600-G4-MT:~$ touch file\with\slashes && ls file\with\slashes
filewithslashes
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 11:
Use echo to print * as a literal character without it being expanded by the shell.

```
student@ai-HP-ProDesk-600-G4-MT:~$ echo '*'
*
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 12:
Write a command to echo the string Today is "Monday" using proper quoting.

```
student@ai-HP-ProDesk-600-G4-MT:~$ echo 'Today is "MOnday"'
Today is "MOnday"
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 13:
Create a file named data'sheet using quoting.

Question 14:r
Create a text file named output.txt and redirect the output of ls into it.

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls >output.txt
```

Question 15:
Append the output of date to the file created in the previous question.

```
student@ai-HP-ProDesk-600-G4-MT:~$ date >>output.txt
student@ai-HP-ProDesk-600-G4-MT:~$
```

LAB TASK

423132

Question 16:
List all files in the current directory, sort them alphabetically, and display only the first 5 files.

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls |sort|head -n 5
anunamed.txt
dept.txt
Desktop
Documents
Downloads
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 17:
Count the number of .txt files in the current directory using pipes.

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls *.txt | wc -l
5
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 18:
Use command substitution to assign the current date to a variable and print it.

```
student@ai-HP-ProDesk-600-G4-MT:~$ curr=$(date) && echo $curr
Tuesday 28 January 2025 03:33:44 PM IST
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 19:
Create a file named backup_<date> where <date> is the current date.

```
student@ai-HP-ProDesk-600-G4-MT:~$ touch backup_$(date  +%Y-%m-%d)
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 20:
Run a command (e.g., sleep 10) in the background and check its status using jobs.

```
student@ai-HP-ProDesk-600-G4-MT:~$ sleep 10 & jobs
[1] 10536
[1]+  Running                 sleep 10 &
```

Question 21:
List all running background jobs and bring one to the foreground.

```
student@ai-HP-ProDesk-600-G4-MT:~$ jobs && fg%<job_number>
bash: syntax error near unexpected token `newline'
[1]+  Done                    sleep 10
```

Question 22:
List all files in a directory where the name contains at least one digit and ends with .sh.

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls *[0-9]*.sh
ls: cannot access '*[0-9]*.sh': No such file or directory
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 23:
Find files where the first character is a, the second is not b, and the name ends in .log.

LAB TASK

423132

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls a[!b]*.log
ls: cannot access 'a[!b]*.log': No such file or directory
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 24:
Use && and || operators to list files only if a directory exists; otherwise, print an error message.

```
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$ [ -d directory ] && ls || echo 'D
irectory does not exist'
Directory does not exist
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$
```

Question 25:
Chain commands to create a file, write to it, and display its contents in one line.

```
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$ touch file && echo 'content' >file && cat file
content
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$
```

Question 26:
Use a pipe to filter files ending in .txt and redirect the output to filtered_files.txt.

```
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$ ls *.txt | tee filtered_files.txt
ls: cannot access '*.txt': No such file or directory
student@ai-HP-ProDesk-600-G4-MT:~/Downloads$
```

Question 27:
Append the number of lines in filtered_files.txt to the end of the file.

```
student@ai-HP-ProDesk-600-G4-MT:~$ wc -l filtered_files.txt >>filtered_files.txt
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 28:
Create a file named *important? and list it without expanding the special characters.

```
student@ai-HP-ProDesk-600-G4-MT:~$ touch '*important? && ls \*important\?
> ^C
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 29:
Use quoting to delete a file named Hello[World].

```
student@ai-HP-ProDesk-600-G4-MT:~$ rm 'Hello[World]'
rm: cannot remove 'Hello[World]': No such file or directory
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 30:
List all files that start with test, have exactly two characters after it, and end with .txt (e.g., test12.txt, testAA.txt).

```
student@ai-HP-ProDesk-600-G4-MT:~$ ls test??.txt
ls: cannot access 'test??.txt': No such file or directory
student@ai-HP-ProDesk-600-G4-MT:~$
```

Question 31:
Find all files in the current directory that do not have the extension .log using wildcards.

LAB TASK

423132

```
student@66:~$ ls *[^.log]
 1.sh                boxdemo.class   'data\sheet'          game.css      'hello world'   stack.java
 alias               boxdemo.java     datasheet            game.js       '*important?'   student.c
 backup_2025-01-28   box.java         filewithslashes      hello.class    output.txt     student.java
 Box.class          "data'sheet"      filtered_files.txt   hello.java     reportfile     test12.txt

423116:
c  file.c  hello.c

anaconda3:
 bin              doc       info          licensing     pkgs        sbin    translations
 compiler_compat  envs      lib           man           plugins     share   var
 condabin         etc       libexec       mkspecs       qml         shell  'wekafiles (copy)'
 conda-meta       include   LICENSE.txt   phrasebooks   resources   ssl     x86_64-conda_cos6-linux-gnu

COP_290_Lab3:
get-pip.py  main.py    requirements.txt  test_seller_views.py  website
htmlcov     README.md  test_auth.py      test_user_views.py

Desktop:
422129  423134  423182  a1.log  ran3  unix

Documents:
 bdlab   eclipse-inst-jre-linux64   hadoop-2.10.2   sort  'week1 TDPL.odt'   xv6-public-master

Downloads:
 0.CNN
 1-16.hs
 1-17
 1-17.hi
 1-17.hs
 1-17.o
'1-18 (1).hs'
 1-18.hs
 aipl1.png
 App.java
 blender-3.1.2-linux-x64
```

```
Public:
snap  wekafiles

__pycache__:
queue.cpython-37.pyc  queue.cpython-38.pyc

SAMPLE:

snap:
dog  jupyter  snap-store  umbrello

Templates:

TRAIL:

Videos:

wekafiles:
native  packages  props  repCache  systemDialogs  weka.log  wekaMetaStore
student@66:~$
```

Question 32:
Create a file with the name hello$world using quoting, and write a command to list it without expanding $.

```
student@66:~$ touch 'hello$world' && ls hello\$world
'hello$world'
```

LAB TASK

423132

Question 33:
Write a command to rename all files in the current directory with .txt extension to .bak extension in a single line.

```
student@66:~$ for file in *.txt; do mv "$file" "${file%.txt}.bak"; done
```

Question 34:
Combine quoting and command substitution to create a file whose name is the current date and time (e.g., 2025-01-28_10-30.txt).

```
student@66:~$ touch $(date +%Y-%m-%d_%H-%M).txt
```

Question 35:
Write a command that uses redirection to write the output of ls to a file, appends the current date and time to the same file, and then displays the contents of the file.

```
student@66:~$ ls > file && date >> file && cat file
1.sh
2025-01-28_16-19.txt
423116
ac.log
alias
anaconda3
b1.log
backup_2025-01-28
Box.class
boxdemo.class
boxdemo.java
box.java
COP_290_Lab3
data'sheet
data\sheet
datasheet
Desktop
Documents
Downloads
file
filewithslashes
filtered_files.bak
game.css
game.html
game.js
hadoop-2.10.2
hello$world
hello.class
hello.java
hello world
*important?
input
Music
new
OOPS
output.bak
Pictures
```

Question 36:
List files in the current directory that do not contain the letter e anywhere in their names.

Question 37:
Write a command to find and delete all files in a directory that have more than 10 characters in their name.

Question 38:
Using metacharacters and pipes, list all files in the current directory, display only the ones with .txt extension, and count the total number of these files.

LAB TASK

423132

Question 39:
Create a file named important|file and use quoting or escaping to delete it without triggering errors.

Question 40:
Write a command to find all files with .txt or .log extensions and copy them to a directory named backup, ensuring the directory is created only if it does not exist.

Question 41:
Using wildcards, find all files that start and end with a number (e.g., 1data1, 4file7) in a directory.

Question 42:
Write a single command to replace all spaces in filenames in the current directory with underscores (_).

Question 43:
Create a chain of commands using && to check if a file exists, delete it if it does, and create a new file with the same name.

Question 44:
Write a command that redirects both stdout and stderr of a program (e.g., ls non_existent_file) into a single file.

Question 45:
Find all files with names starting with a lowercase letter and ending with .sh, and move them to a directory named scripts.

Question 46:
Use pipes and grep to list only the hidden files in the current directory (files starting with a .).

Question 47:
Create a file with the name *wild?card.txt and write a command to delete it without expanding the special characters.

Question 48:
Write a command to search for a string error in all .log files in a directory, count the number of matches, and save the count to a file named error_count.txt.

Question 49:
Find all files in a directory that contain the letter a in the third position of their name (e.g., __a__).

Question 50:
Using quoting and metacharacters, write a command that appends the output of ls to a file named summary_$(date +%Y%m%d).txt in a single line.

Question 51:
Write a command to find all files in the current directory and its subdirectories that:

- Start with the letter a,
- Have exactly 8 characters before the file extension,
- End with .txt or .log.
  The output should display the relative paths of these files.

LAB TASK

423132

Question 52:
Create a directory structure with multiple nested folders and files using a single command (e.g., mkdir -p). Use wildcards in a find command to locate and list all files that:

- Start with a vowel,
- Contain at least two numbers,
- End with .sh.

Question 53:
Using metacharacters and wildcards, write a single command to:

- Find all files that do **not** have a .txt or .log extension,
- Exclude any files with a name containing temp or backup,
- Copy the remaining files into a new directory named filtered_files.

Question 54:
Write a single command to:

- Identify all files whose names have at least one uppercase letter, one lowercase letter, and one number.
- Append the full path of each matching file

Question 55:
You are given a text file data.txt with more than 100 lines. Write a single command to:

- Extract lines 20 to 30 from the file and save them to output.txt.
- Ensure the original file remains unchanged.

Question 56:
Given two files file1.txt and file2.txt, write a command to:

- Display the first 5 lines of file1.txt concatenated with the last 5 lines of file2.txt.
- Use head, tail, and cat in a pipeline.

Question 57:
You have two files, list1.txt and list2.txt, containing sorted lists of names. Write a command to:

- Find the common names in both files, the names unique to list1.txt, and the names unique to list2.txt.
- Display the result in three separate columns.

LAB TASK

423132

Question 58:
Given two files, config_old.txt and config_new.txt, use diff to:

- Display only the lines that differ between the two files with their line numbers.
- Write a command to suppress all unchanged lines in the output.

Question 59:
You have a large log file named system.log. Write a single command to:

- Extract the last 50 lines of the file, sort them alphabetically, and display only the first 10 sorted lines.
-