Hello friends,

I hope you're all doing well.
The following exercises are not exam questions, but rather open-ended prompts whose answers can often be found online, through chatbots, or even in technical interviews. These are intended to encourage deeper exploration and self-directed learning based on your current knowledge.
We've selected these questions based on real-world experience and past exam patterns to help you better understand the types of concepts that appear in the LPIC exam. Many of the questions don't have a single correct answer — instead, they are designed to broaden your perspective and understanding.

Please be kind and patient with yourself as you write and research your responses.

Please complete the following exercises using the same format as the directory structure below. Combine your answers into a single PDF file named `answer.pdf` and submit a pull request.

📁 Reference format:

https://github.com/devopsdoctors/Academy/tree/main/L1-JavanPahlevanan/Exercises/T(x)/name-family(emailAddress)/answer.pdf

📄 Sample answer file:

https://github.com/devopsdoctors/Academy/tree/main/L1-JavanPahlevanan/Exercises/T1/ali-farhadian(alifrd49@gmail.com)/answer.pdf

1. Please explain how you check **total, used, and free memory** on Linux using `free -m`. What does the **buffer/cache** field mean?
2. When using `htop` or `top`, what is the difference between **us**, **sy**, **id**, **wa**, and **st** in CPU usage breakdown?
3. How can you detect a **CPU bottleneck** using `top` or `htop`? What values or columns are important to monitor?
4. Please explain the difference between **load average** and **CPU utilization**. What does a load average of `4.0` mean on a 2-core vs 8-core system?
5. What is the output of `vmstat 1`? Please explain the meaning of the columns **r**, **b**, **si**, **so**, **wa**, and **us**.
6. How do you identify **I/O bottlenecks** using `iostat`? Please explain the meaning of **await**, **svctm**, and **%util**.
7. What does **iowait** mean in Linux CPU statistics, and how can high iowait impact system performance?
8. Please explain how you would measure **disk throughput** and **latency** with `iostat -x`.
9. How can you monitor **per-process memory usage** with `htop`? What is the difference between **RES**, **VIRT**, and **SHR** memory?
10. How can you measure **network bandwidth usage** with tools like `iftop`, `nload`, or `sar -n DEV`?
11. Please explain how to use `iperf3` to test **network throughput** between two nodes. Which modes does it support (client/server, TCP/UDP)?
12. What is the difference between **packet loss**, **latency**, and **jitter** in network performance testing? How can you measure each with `ping`, `iperf`, or `mtr`?
13. Please explain how you would use `sar` from `sysstat` to monitor CPU, memory, and I/O over time. Why is historical monitoring important?
14. How can you detect a **memory leak** in a process using Linux monitoring tools?
15. Please compare **htop** and **top**: what extra features does htop provide that make it useful for DevOps engineers?
16. When troubleshooting high load, how do you decide if the problem is **CPU bound**, **I/O bound**, or **network bound**? Which commands would you use in sequence?
17. What is an **inode** in Linux? What kind of information does it store?
18. What is stored in the **inode table** vs what is stored in the **directory entry**?
19. How do you check the number of **free inodes** on a filesystem?
20. What is the difference between **inode number** and **filename**?
21. What happens when a filesystem runs out of **inodes** but still has free space?
22. How can you find the **inode number** of a file?
23. Please explain **hard link** vs **soft link** in relation to inodes.

24. What does `ls -i` show and why might it be useful?
25. How does Linux handle **deleting a file that is still open by a process**?
26. What happens to the inode when a file is deleted?
27. Compare **ext4, xfs, btrfs, zfs** — what are their strengths and weaknesses?
28. What is **journaling** in filesystems? Why is it useful?
29. What is the difference between **ordered journaling** and **writeback journaling** in ext4?
30. What is **copy-on-write (CoW)** in filesystems like Btrfs and ZFS?
31. Compare **Btrfs** and **XFS**: which one supports snapshots?
32. Which filesystems are commonly used for **boot partitions**? Why?
33. What is the maximum file size supported by **ext4** vs **xfs** vs **btrfs**?
34. What is the main limitation of **FAT32**? Why is **exFAT** introduced?
35. Compare **NTFS** (Windows) and **ext4** (Linux). What are key differences?
36. What is an **overlay filesystem** and how is it used in Docker?
37. What is **LVM** and why do we use it?
38. What is the difference between **physical volume (PV)**, **volume group (VG)**, and **logical volume (LV)**?
39. How do you create a new LVM logical volume? Which commands are involved?
40. How do you extend an existing LVM logical volume?
41. How do you **reduce** the size of an LVM volume safely?
42. What is a **snapshot** in LVM? How does it work internally?
43. How do you check the **free space** in a volume group?
44. What is the difference between **thin provisioning** and **normal LVM volumes**?
45. What is the purpose of `lvextend` vs `resize2fs`? Why do you need both?
46. What happens if you extend an LVM volume but forget to extend the filesystem?
47. What is **RAID**? Please explain RAID levels 0, 1, 5, 6, and 10.
48. What are the differences between **hardware RAID** and **software RAID**?
49. What tool do we use in Linux to manage software RAID?
50. How do you check the status of a RAID array?
51. What happens when one disk fails in a RAID 5 array?
52. Which RAID level provides **best performance but no redundancy**?
53. Which RAID level is most suitable for **databases**? Why?
54. What is the difference between **RAID 1+0 (RAID 10)** and **RAID 0+1**?
55. How do you rebuild a degraded RAID array in Linux?
56. What is **hot spare** in RAID?
57. What is the difference between **MBR** and **GPT** partition tables?
58. What is the maximum disk size supported by MBR?
59. How do you create a new partition with `fdisk`?
60. What is the difference between `fdisk` and `parted`?
61. How do you list all block devices in Linux?

62. What is the purpose of `lsblk`? Please explain its output columns.
63. What is the difference between `df -h` and `lsblk`?
64. How do you create a filesystem on a partition using `mkfs`?
65. What are common `mkfs` options for ext4?
66. How do you label a filesystem at creation time?
67. How do you manually mount a filesystem?
68. What is the difference between mounting with `mount` vs adding entry in `/etc/fstab`?
69. What happens if you incorrectly configure `/etc/fstab`? How can you recover?
70. How do you mount an ISO file in Linux?
71. What does the `mount` command show without arguments?
72. How do you check which filesystem a mount point is using?
73. How do you resize an **ext4 filesystem** on an LVM logical volume?
74. How do you resize an **xfs filesystem**?
75. What is the difference between `resize2fs` and `xfs_growfs`?
76. How do you shrink an ext4 filesystem? Why can't you shrink xfs easily?
77. What is **NFS**? How does it differ from **SMB**?
78. How do you mount an NFS share in Linux?
79. What is the purpose of `/etc/exports` in NFS?
80. How can you check NFS client connections on a server?
81. What is the main limitation of NFS compared to object storage (e.g., S3)?
82. How does **NFSv4** improve over NFSv3?
83. How can you test NFS performance?