

ANIL KUMAR YALLA (AYALLA)

Conceptual Questions

1. Explain class philosophy of "Understand how it works"

The class philosophy of "Understand how it works" is to build the very thing what we are trying to understand. As time progresses, there are no specific tools which will always work, we will be facing many failures and it is hard to trace these failures. So, the best way is to figure out what it's doing inside the tool, which will help us understand out what it's doing.

2. What is the difference between a terminal and shell?

Shell is a command Line interpreter, which process the commands of the user and interacts with Kernel and outputs the results, where as terminal is a program which runs shell and allows users to enter commands to interact with the kernel.

The user cannot interact with the kernel, so the user enters the commands in the terminal, which in turn invokes the shell and outputs the results after interaction with kernel.

3. Even if you're not a windows user, after seeing information about windows what would be your preferred environment for running commands and why?

In windows, there is no one environment to performing all the tasks. Different environments are required to perform different tasks.

For example, to perform administration tasks PowerShell suits the best. But it doesn't support some common Linux commands like `cd ~`.

MSYS2 is full compatible with linux but it is very slow where as MIngW is faster but limited functionality.

Wsl is efficient but it is complex.

Git Bash is my preferred environment because it is faster and supports most of the linux commands. And easier to access the git hub environment and colours and works well with copy things.

4. What are some reasons why a change to a environment variable may not be immediately visible?

When we make any changes to environment variables, we must refresh the terminal (cmd prompt) to reflect the changes using "refreshEnv" command.

The terminal already loads the environment variables when the terminal is opened, so we have to us the above shell command which interacts with kernel to update the environment variables. If we don't refresh it, the terminal won't get or look for updated environment variables.

5. **What is the difference between stdout and stderr in a running process?**

Stdout is used to deliver text output to the shell whereas stderr is used for error messages. When we just normally execute a program, they are displayed on the terminal as normal text, but the real advantage is, when we use redirecting to segregate output from the errors. This can be achieved in multiple ways like redirecting just output to the file or redirecting error to a file and displaying the output in command line or redirecting both standard output and errors to two separate files, which helps in debugging to analyze the script and do the fixes.

6. **Explain what does it mean by "Commits are NOT diffs"?**

Git commit creates a snapshot of the repository which includes a message, usually contains description about changes made, we commit often based on changes made, they are like save points, whereas diffs are dynamically generated from commits by comparing the commit and its parent. We can also compute diff between any two commits.

7. **What are signs of a *bad* kanban board?**

Kanban board is usage of tools for workflow visualization in the organizations. These boards have cards to represent work item, and columns to represent steps in workflow. Some major signs of a bad Kanban board are.

1. Kanban board is empty
2. Kanban board contains only high-level goals but not low-level goals like weekly and today
3. Kanban board containing too many things which makes complex and difficult to track.
4. Items on Kanban board which are sticking around without any progress.

8. **What is heredoc, and why might it be useful?**

Heredoc refers to special block of code spanned over multiple lines of strings which we will be passing to a shell command. The most used is cat command. If there is no command the interpreter reads the block and ignores it. Heredoc is also useful for variable substitution (both user-defined and environmental variables). Heredoc can also be used to execute multiple commands on a remote system over SSH.

9. **What was a new feature, challenge, or interesting learning experience that you encountered while doing the homework or classroom exercises?**

I have previously enrolled for software engineering course, where it deals with traditional methods and cost for errors and details about how to get code to production after rigorous testing which takes huge amount of time, but in this class I learnt about how to adapt to the new principles and also how can the code become useless if we wait too long and importance of finding a way to put it into production but also having safe and limiting the risk exposure.

