**What is DevOps?**

DevOps is a software engineering culture and practice that aims to unify software development (Dev) and software operation (Ops) and IT operations. The main goal of DevOps is to shorten the systems development life cycle and provide continuous delivery with high software quality. DevOps promotes collaboration, communication, and integration between software developers and IT operations. The philosophy of DevOps aims to bring applications and services faster to the market.

# ****What is Automation?****

Automation refers to the use of technology to perform tasks without the need for human input. Automation can be used in a variety of contexts, such as manufacturing, transportation, and information technology. Automation typically refers to the use of software tools and scripts to perform tasks that would otherwise be done manually by a person. Automation can be used to improve efficiency and speed by eliminating the need for manual processes also it can help reduce the risk of errors.

# ****What is Scaling?****

Scaling is the ability of a system to handle an increasing workload. This can be achieved through horizontal scaling, which involves adding more resources to a system e.g. adding more servers to a network, or through vertical scaling, which involves increasing the capacity of individual resources e.g. upgrading the hardware of a single server)

# ****What is Infrastructure?****

Infrastructure is viewed as code that can be managed and maintained through the use of automation and configuration management tools. This means that rather than manually configuring and maintaining infrastructure, the infrastructure can be defined and managed using code and automation tools. This enables the infrastructure to be treated in the same way as the application code, with the ability to version control, test, and deploy infrastructure changes alongside application code changes.Managing infrastructure in this way can help to improve the reliability, scalability, and security of the infrastructure,

# ****What are the common technologies used in DevOps?****

**1) Version control systems** (e.g. Git) for managing and tracking code changes

**2)Continuous integration tools** (e.g. Jenkins, Travis CI) for automating the build and testing of code

**3) Containerization technologies** (e.g. Docker) for packaging and deploying applications

**4) Configuration management tools** (e.g. Ansible, Puppet) for automating the configuration and management of infrastructure

**5) Monitoring tools** (e.g. Nagios, New Relic) for tracking the performance and availability of systems

**6)Cloud platforms** (e.g. AWS, Azure, GCP) for providing infrastructure and services for deployment

# ****Why DevOps is Important?****

**1)Faster delivery of software updates:** By automating the build, test, and deployment process, DevOps enables organizations to deliver software updates faster and more frequently.

**2)Improved collaboration:** DevOps promotes collaboration between software developers and IT operations, which can help to improve communication and integration between these teams.

**3)Increased reliability and stability:** By automating the deployment process and implementing testing and monitoring at various stages, DevOps can help to improve the reliability and stability of systems. This can reduce the risk of errors and downtime, and can help to improve the overall quality of the software.

**4)Better security:** By incorporating security testing and monitoring into the development and deployment process, DevOps can help to improve the security of systems. This can help to protect against vulnerabilities and threats and can help to ensure that systems are compliant with relevant security standards and regulations.

**What is Linux?**

Linux is **an open-source operating system (OS)**. An operating system is software that directly manages a system’s hardware and resources, like CPU, memory, and storage. The OS sits between applications and hardware and makes the connections between all of your software and the physical resources that do the work.

**What are the Features of Linux Operating Systems?**

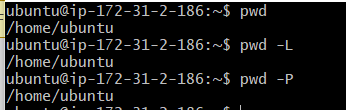
* Free and Open-Source.
* Extremely Flexible.
* Lightweight Infrastructure.
* Graphical User Interface (GUI)
* Linux doesn’t require any antivirus to install.
* End-to-end encryption
* Portable Environment
* Shell/ Command-line Interface
* Customized keyboard

**Basic Linux command:**

**1. Linux command to check your present working directory.**  
**pwd** command is used to find the path of your current working directory.  
Simply entering pwd will return the full current path — a path of all the directories that starts with a forward slash (/).  
For example, /home/username.

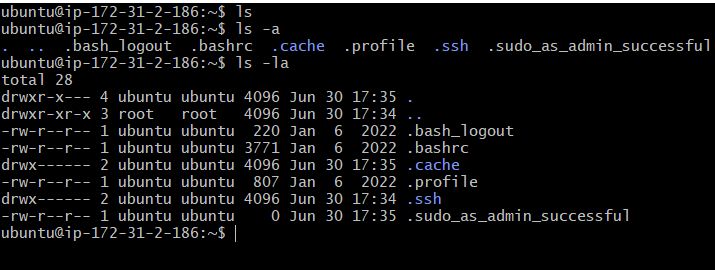
The pwd command uses the following syntax:  
**pwd [option]**  
It has two acceptable options:

**-L or –logical prints**environment variable content, including symbolic links.  
**-P or –physical** prints the actual path of the current directory.

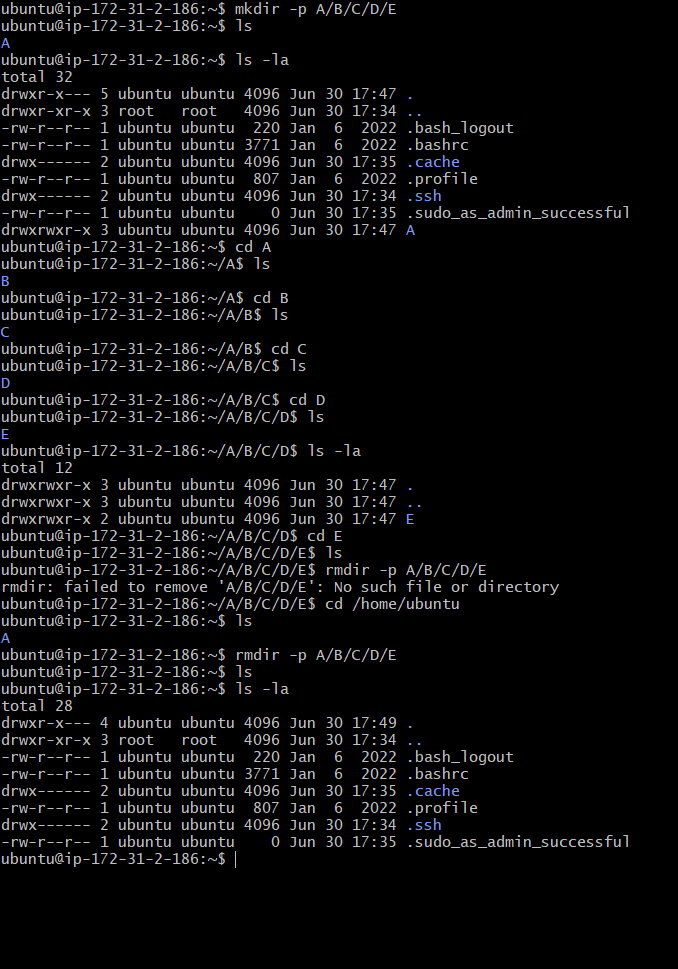


**2. Linux command to list all the files or directories including hidden files.**

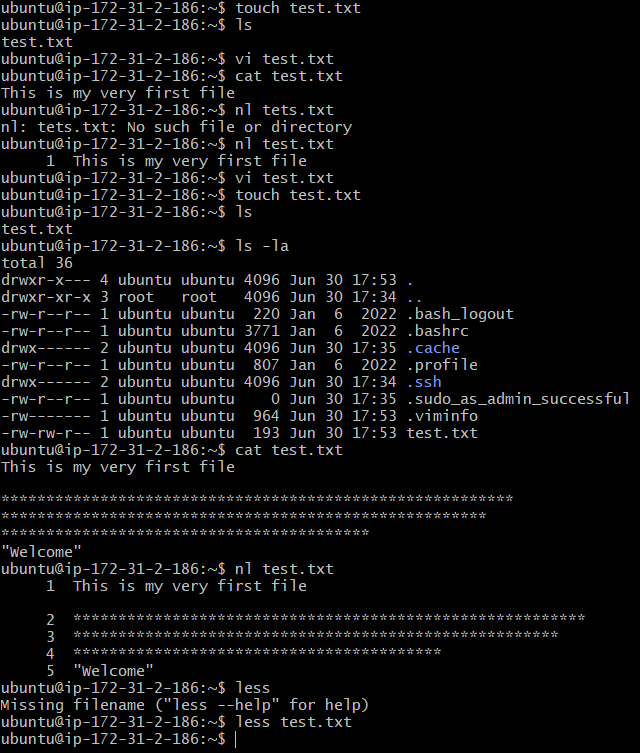
The**ls**command lists files and directories within a system. Running it without a flag or parameter will show the current working directory’s content.  
**“ls -la”** is the command to list all the files or directories including hidden files.  
Using**‘l’** will display a long listing format of content one per line of the current directory.  
Using **“a”** will list all including hidden files and directories in the Linux machine.



**3. Linux command to create a nested directory A/B/C/D/E**  
The command to create a nested directory is mkdir -p A/B/C/D/E. The -p switch creates parents' directories.



**1)Linux command to view what’s written in a file.**  
i)The simplest way to view text files in Linux is the cat command.  
It displays the complete contents in the command line without using inputs to scroll through it.  
ii) **nl**  
The**nl command** is almost like the cat command. The only difference is that it prepends line numbers while displaying the text in the terminal.  
iii) Less  
Less command views the file one page at a time. The best thing is that you exit less (by pressing q), and there are no lines displayed on the screen. Your terminal remains clean and pristine.



**2)Linux command to change the access permissions of files.**  
In Linux, there are three types of owners: users, groups, and others.  
Linux user: A user is the owner of the file.  
Linux group: Group is a collection of users.  
Other: Any user who is not part of our user or group category is called other.

**Linux file permissions:**  
i)Read permission: Read permission allow users to open and read the file only.  
ii)Write permission: It allows the user to modify the file.  
iii)Executable permission: It allows the user to run an executable script.

Note that we can find permissions of files and folders using a long listing **(ls -l)**on a Linux terminal.  
**chmod** is the command which is used to change the access permission of files.  
Syntax :  
**chmod permissions filename**

Permissions can be changed using two modes:  
i)Symbolic mode  
ii)Absolute mode

We will take an example of **Absolute mode**  
The absolute mode uses numbers to represent permissions and mathematical operators to modify them.  
Permissions are provided using an **addition**  
Permission | Provide permission  
**read | add 4  
write | add 2  
execute | add 1**

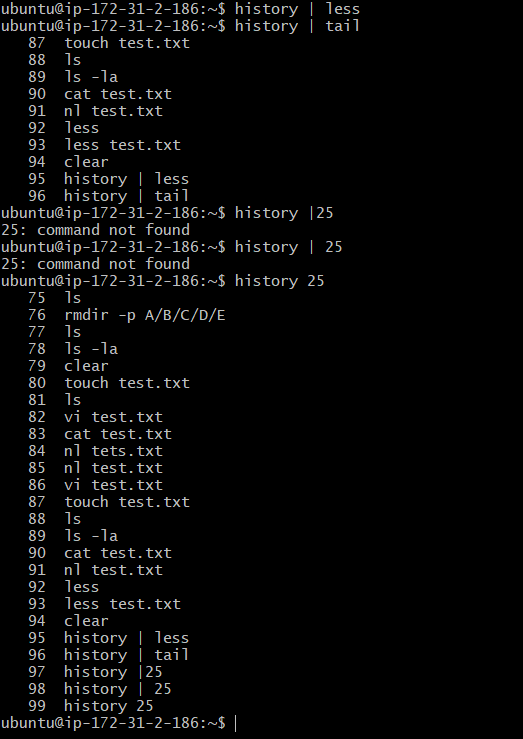
Permissions are revoked using **subtraction**  
Permission | Provide permission  
read | subtract 4  
write | subtract 2  
execute | subtract 1

Example:

Set read (add 4) for the user, read (add 4) and execute (add 1) for the group, and only execute (add 1) for others.  
**chmod 451 file-name**

**3)Linux command to check which commands you have run till now.**  
**history** command is used to check the commands you have run till now

i)To view history one page at a time, you can use the following:  
**history | less**  
ii)To view just the last ten commands, you can use the following:  
**history | tail**  
iii)To view the last 25 commands, just use the following:  
**history 25**

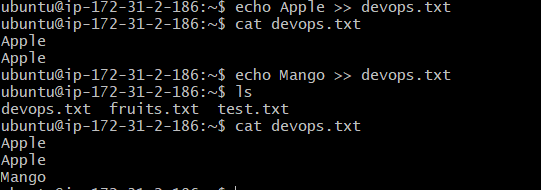


**4)Linux command to remove a directory/ Folder.**  
The **rm command** removes complete directories, including subdirectories and files.  
The **rmdir command** removes empty directories.  
It is important to note that the**rm** and**rmdir** command permanently remove directories without moving them to the Trash directory.  
This means that you cannot restore a directory removed using these commands.

**5)Linux command to create a fruits.txt file and to view the content.**  
So to create a fruits.txt file we are using echo command and then we use **cat** command to display the contents  
**eg:  
echo Happy new year >> fruits.txt  
cat fruits.txt**



**6)Linux command to add content in devops.txt (One in each line) — Apple,** **Mango, Banana, Cherry, Kiwi, Orange, Guava.**  
We are going to use the **echo** command to concatenate each line to the devops.txt file.  
**eg:  
echo Apple >> devops.txt  
echo Mango >> devops.txt**



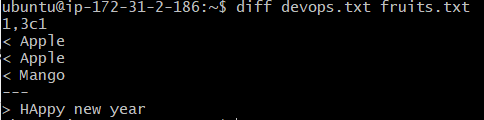
**7)Linux command to show only the top three fruits from the file.**  
To show only the first three fruits we will use **head command**, however by default, it shows 10 lines. We can pass a **-n**, where n is the number of lines from the top that we want as an output.

**i)Show only the bottom three fruits from the file.**  
To show only the bottom three fruits we will use **tail command**, however by default, it shows 10 lines. We can pass a**-n**, where n is the number of lines from the bottom that we want as an output.

**ii)To create another file Colors.txt and to view the content.**  
We are going to use the **touch command** to create the file and **cat** **command** to view the content of the file

**iii)Add content in Colors.txt (One in each line) — Red, Pink, White, Black, Blue, Orange, Purple, Grey.**  
We can add each color in the file using the echo command.  
**eg:  
echo Red >> Colors.txt  
echo Pink >> Pink**

**iv)To find the difference between the fruits.txt and Colors.txt files.**  
To find the difference between the content of two files we can use**diff** command.  
eg  
**diff fruits.txt Colors.txt**



## What is Kernel?

The kernel is a computer program that is the core of a computer’s operating system, with complete control over everything in the system.

## What is Shell?

A shell is a special user program that provides an interface for the user to use operating system services. Shell accepts human-readable commands from users and converts them into something which the kernel can understand. It is a command language interpreter that executes commands read from input devices such as keyboards or from files. The shell gets started when the user logs in or starts the terminal.

## What is Linux Shell Scripting for DevOps?

Shell Scripting is a program to write a series of commands for the shell toexecute. Shell is a command-line interpreter and Shell script is nothing but a list of commands executed by the shell. It can combine lengthy and repetitive sequences of commands into a single and simple script that can be stored and executed anytime which, reduces programming efforts.

## **What is #!/bin/bash?**

**#!/bin/bash**is a shebang line used in script files to set bash, present in the ‘/bin’ directory, as the default shell for executing commands present in the file. “#!” is a **shebang** in the character sequence consisting of the character's number sign and exclamation mark ( #!) at the beginning of a script.

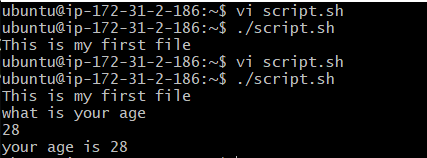
## Can we write **#!/bin/sh** as well?

**#!/bin/sh**: It is used to execute the file using sh, which is a Bourne shell, or a compatible shell.

Just like that, we have **#!/bin/csh**. It is used to execute the file using csh, the C shell, or a compatible shell.

**Write a Shell Script that prints “I will complete #90DaysofDeVops challenge"**

**Write a Shell Script to take user input, input from arguments, and print the variables.**



**Write an Example of If else in Shell Scripting by comparing two numbers**

