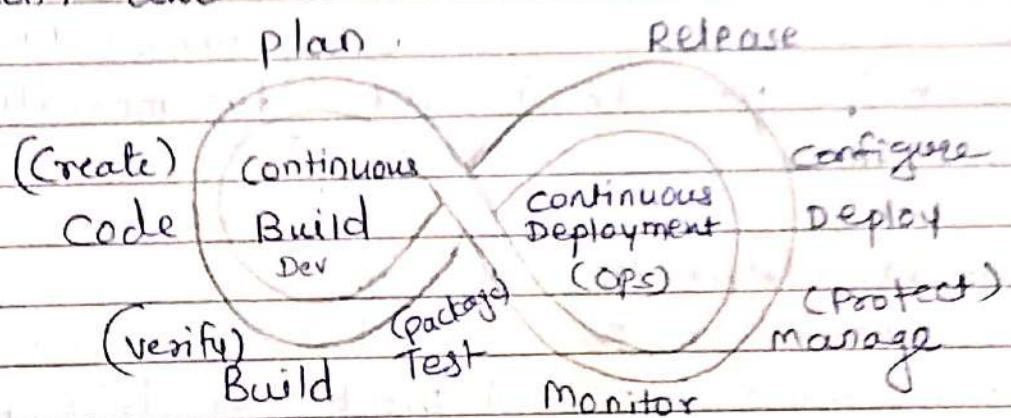


Chapter 1st -

Introduction of Devops :-

Devops is the acronym given to the combination of Development and Operations. It refers to a collaborative approach to make the Application Development team and the IT Operations team of an organization to seamlessly work with better communication. It is a philosophy that encourages adopting iterative software development, automation, and programmable infrastructure deployment and maintenance.

Devops is visualized as an infinite loop comprising the steps plan, code, build, test, release, deploy, operate, monitor, then back to plan, and so on.



While the visualization helps us understand how Devops works, let us try to dig a little deeper to figure out which are the different Devops tools, and how they help an organization.

i) Build Automation:- An automated code is prepared to be deployed in a live environment where the tool used to is tied to the programming language chosen.

2) CI/CD :- Continuous integration and continuous deployment deals with frequent merging of codes and unit testing. Further the deployment of small code changes in a routine and frequent process is also a part of this method.

3) Infrastructure As code :- This is usually to manage and provision IT infrastructure through code and automation. This is reusable, and helps in consistent resource creation and management.

4) Configuration Management:-

This is the process where you can manage and change the state of infrastructure in constant and maintainable ways. Configuration Management also helps provide insight documentation on infrastructure.

5) ORCHESTRATION:-

Orchestration is nothing but an automation that supports processes and workflows, like Resource Planning.

6) Monitoring:- By this method you collect and present data about the performance and stability of services and stability of services and infrastructure, while also detecting problems.

7) Microservices: It is basically an architecture that breaks an application into many small

and loosely coupled services.

Define Devops:-

Devops is the combination of cultural philosophies, practices, and tools that increases an organization's ability to deliver applications and services at high velocity. Evolving and improving products at a faster pace than organizations using traditional software development or infrastructure management processes. This speed enables organizations to better serve their customers and compete more effectively in the market.

What is devops:-

Devops is a combination of software developers (dev) and operations (ops). It is defined as a software engineering methodology which aims to integrate the work of software development and software operations teams by facilitating a culture of collaboration and shared responsibility.

The Devops methodology comprises four key principles that guide the effectiveness and efficiency of application development and deployment. These principles, listed below, center on the best aspects of modern development.

- ① Automation of the software development lifecycle.
- ② Collaboration and communication.
- ③ Continuous improvement and minimization of waste.
- ④ Hyperfocus on user needs with short feedback loops.

* SDLC Models, lean, JTIL, Agile.

-* Agile :-

After many years of using the waterfall software development methodology (and other less than successful approaches), software development met to discuss alternative software development methods in 2001. The manifesto for agile software development was the result of this meeting. The manifesto is a set of four values that are supported by 12 principles. In a 2016 study of Agile practices, the most widely valued Agile principle was related to business people and developers working together. The 2016 study found 84% of respondents rated this of high importance. The (tied for) second most valued Agile principle related to achieving customer satisfaction through early and continuous delivery of valuable software. Some 60% of respondents rated this of high importance.

Essentially, the Agile approach is designed to drive shorter feedback loops that ultimately improve customer value by developing software in a collaborative, iterative, and incremental manner. The main elements of the Manifesto are:

Work is done by self-organizing teams, networks and ecosystems that mobilize the full talents of those doing the work.

Work is focused directly on meeting customers'

needs and interaction with the customer is paramount.

A "lens" focuses attention on the customers' needs and interaction with the customer is paramount

A "lens" focuses attention on the customers' needs (when the lens is a person, as in Scrum, the person is known as a 'product owner')

Work proceeds in an iterative fashion and progress toward fulfilling the needs of customers is assessed at every stage

There are a wide range of Agile Software development methods including extreme programming, Adaptive Software Development, Scrum, Agile Project Management, Crystal Methods, Feature Driven Development, lean Development, and Rational Unified process. As a result, agility could be considered more of a mindset rather than a specific set of techniques.



Lean

Manufacturing took many concepts from leadership practices and learned how to remove constraints along the flow of work using a variety of short feedback loops. The core idea behind lean is maximizing customer value while minimizing waste," states The lean Enterprise Institute. The Institute goes on to state that

"Simply put, lean means creating more value for customers with fewer resources."

There has been a long-time connection between agile and lean in the IT field. In 2003 Poppoendieck published their book, lean software Development. An Agile Toolkit. This book was part of Addison-Wesley's The Agile Software Development series. In their book, Poppoendieck and Poppoendieck illustrate how 22 different lean tools, such as seeing waste and value stream mapping, could be applied to the (Agile) software development process.

In 2010 Bell and Orzan published their book Lean IT; Enabling and sustaining your lean transformation. They applied lean to the entire IT organization. In their lean IT pyramid, the top of their pyramid is culture. The starting building blocks are: consistency of purpose, respect for people and pursuit of perfection. Intermediate layers include: Voice of the customer (originally a marketing term widely adopted in business to describe the in-depth process of capturing customers' expectations, preferences and aversions); quality at the source (a lean manufacturing principle defines that quality output is not only measured at the end of the production line but at every step of the production process at each step the responsibility for quality are the individuals working on the step rather than quality inspectors); systems thinking (a holistic approach to analysis that focuses

on the entire system and the relationships between each of the system's constituent parts over time); and flow/pull just in time (an approach in which materials, goods, and labor are scheduled to arrive or be replenished only when needed in the process - that is, just in time).

* Information Technology Infrastructure Library (ITIL)

Information Technology Service Management (ITSM) is a quality management approach for managing IT services that meet the needs of the business by focusing on the effective and efficient operation of the IT service provider's internal processes. ITSM is defined as the implementation and management of quality IT services that meet the needs of the business and is performed by IT service providers through an appropriate mix of people, processes, and information technology. There are several ITSM frameworks, but the most widely known framework is the Information Technology Infrastructure Library (ITIL). In our data analysis, we use ITIL knowledge to represent ITSM knowledge.

There are five life cycle stages in ITIL viz. Service strategy, Service Design, Service Transition, Service Operation, and Continual Service Improvement (CSI). CSI has many similarities to the lean concept of kaizen. CSI uses methods from quality management such as the doming PDCA

(Plan-Do-Check-Act) Cycle"

Like The agile Manifesto, ITIL explicitly states some essential principles and values. In ITIL v3, each of the five life cycle books begin with a chapter on services and value. In ITIL's definition of a service, it provides an essential core principle - a service, it provides an essential core principle - a service is a means of delivering value to customers by facilitating outcomes. Customers want to achieve without the ownership of specific costs and risks. ITIL further expounds on this principle as IT Service value is composed of two parts: utility and warrenty. Utility is a service's fitness for purpose while warrenty is a service's fitness for use. Utility is simply a service's functional requirements. Warrenty includes availability, capacity, continuity and security.

In terms of devops, service Design (Development) and service Operation (Operations) have considerable relevance because of the overlapping activities in Devops performed in the processes that make up these two ITIL life cycle stages. Service Design includes processes for service-level Management Availability Management, Capacity Management, Service Continuity Management, and Information Security Management. Service Operation consists of five processes: Event Management, Incident Management, Request fulfillment, Problem Management and Access Management. ITIL also suggests four generic functions (employee groups): Service Desk

Technical Management, Application Management, and IT Operations Management.

The updated ITIL framework ITIL 4 expands on the previous versions by providing a practical and flexible basis to support organizations on their journey to the new world of digital transformation. It provides an operating model for the delivery and operation of the IT components that fosters team integration. ITIL 4 also provides a holistic end-to-end picture that integrates frameworks such as lean, Agile, and Devops.

Waterfall

Why Devops:

Devops is important because it's a software development and operations approach that enables faster development of new products and easier maintenance of existing deployments. Devops is the amalgamation of development (Dev) and operations (Ops) teams is an organizational approach that enables faster development of applications and easier maintenance of existing deployments. By enabling organizations to create stronger bonds between Dev, Ops and other stakeholders in the company, Devops is not a technology per se but it covers everything from the organization to culture, processes and tooling. Initial steps usually include continuous integration and continuous delivery (CI/CD), real-time monitoring, incident response systems and collaboration platforms.

Devops adoption is growing rapidly. IDC fore-
casts the wo

Source code management: In this phase, the business owners and software development team discuss project goals and create a plan. Programmers then design and code the application, using tools like Git to store the application code.

Continuous Build and Test:- This phase deals with building tools, like maven and Gradle, then taking code from different repositories and combining them to build the complete application. The application is then tested using automation testing tools, like selenium and JUnit, to ensure software quality.

Continuous Integration :- When the testing is complete, new features are integrated automatically to the existing codebase.

Continuous Deployment: Here, the application is packaged after being released and deployed from the development server to the production server. Once the software is deployed, operation teams perform tasks, such as configuring servers and provisioning them with the required resources.

Continuous Monitoring: Monitoring allows IT organizations to identify issues of specific

releases and understand the impact on end-users.

Software released :- After all the phases are completed and the software meets the user's requirement, it is released into the market.

History of Devops

The Devops movement started to coalesce some time between 2007 & 2008, when IT operations and software development communities raised concerns what they felt was a fatal level of dysfunction in the industry.

They railed against the traditional software development model, which called for those who write code to be organizationally and functionally apart from those who deploy and support that code.

Developers and IT ops professionals had separate (and often competing) objectives, separate department leadership, separate key performance indications by which they were judged and often worked on separate floors or even separate buildings. The result was siloed teams concerned only with what with their own fiefdoms, long hours batched releases, and unhappy customers, surely there's a better way, they said so, the two communities came together and started talking - with people like Patrick Dubois, Gene Kim, and John Willis driving the conversation.

* Devops Stakeholders :-
Stakeholders are users with free but limited access to Azure DevOps features and functions. With stakeholder access, you can add and modify work items, manage build and release pipelines and view dashboards. You can check project status and provide direction, feedback, feature ideas and business alignment to a team.

Dev includes all people involved in developing software products and services including but not exclusive to

Architects, business representatives, customers, product owners, project managers, quality assurance (QA) testers and analysts, support suppliers.

Ops includes all people involved in delivering and managing software products and services including but not exclusive to

Information security professionals, system engineers, system administrators & IT operations engineers, release engineers, database administrators (DBAs), network engineers, support professionals, third party vendors and suppliers

The challenge for any DevOps initiative is how to include all systems necessary to incorporate the activities, requirements and roles that these stakeholders represent. We must

radically shift our thinking. Service providers need to understand that it's not what we are doing that has to change but need to consider how we do it. We still need to deliver services that deliver value to our customers. The old way of doing things does not work in today's world.

Devops goal :-

The primary Devops goal is to optimize the flow of value from idea to end user. Obviously, there's a cultural change that must happen for a company to be successful with Devops, so culture is a big focus, but the Devops goal is to make the delivery of value more efficient and effective. To do this we must first be able to assign value to any changes that we're trying to make in an organization. This is something that we have been very focused on at Digital.ai, formerly VersionOne.

As we are rolling out new strategies and programs, we assign a value to those ideas and break them down along with the value into work items so that our developers or our team members can work on them. When I'm visiting customer companies, my experience has been that most organizations are very good at measuring the output of their work. There are a lot of Devops

tools for our applications. We have support teams and on call lists and tools that can measure how our applications are performing. This doesn't really get to the bottom of the question we should be focused on, which is how do I optimize the flow of value from idea to end user?

Devops is very much a black box; you can track what goes in and kind of track what comes out, but you aren't really sure of what happens in between. To address the Devops goal, we have to focus on how we're performing as an organization with everything in between user stories being completed and end users receiving the product. In most cases, there are a lot of manual processes in between that rarely get measured. You have to be able to bring visibility to the manual interactions throughout your pipeline so that your organization can understand how those manual interactions affect their value delivery process.

As we measure our end-to-end pipeline, and begin to improve it, we are addressing the fundamental Devops goal and that is to 1) break down silos
2) Create cross-functional teams, and
3) Improve the flow of value. At Digital Bazaar, we are truly passionate about making sure that we can also help organizations

assess Devops goals and the three fundamental things to not only report on them, but also to understand them and to understand where they can make better investments in tools, people and processes.

* Devops perspective :-

Devops is an IT mindset that encourages communication, collaboration, integration and automation among software developers and IT operations in order to improve the speed and quality of delivering software.

"Devops is a set of practices and cultural changes - supported by the right tools that creates an automated software delivery pipeline enabling organizations to win, serve and retain customers."

The emerging professional movement that advocates a collaborative working relationship between Development and IT Operations, resulting in the fast flow of planned work (i.e. high deploy rates), while simultaneously increasing the reliability, stability, resilience and security of the production environment."

While Agile helps in bridging the gap between business and development teams, Devops helps in doing this for development and operations teams.

While Agile refers to an iterative approach which focuses on collaboration, customer feedback, and small, rapid releases,

Devops is the practice of bringing development and operations teams together

Devops central concept is to manage end-to-end engineering processes.

Devops and agile

Both Devops and agile are modern software development practices aimed at providing a framework to produce a part of a product a launch, or a release.

Although both Devops and agile result in the development of software, they have different approaches, involve different groups and departments, and structure production differently.

The most important thing to know about Devops versus agile is that they are not mutually exclusive. Devops is a culture, fostering collaboration amongst all participants involved in the development and maintenance of software. Agile can be described as a development methodology designed to maintain productivity and give releases with the common reality of changing needs. Although Devops and agile are different, when used together these two methodologies lead to greater efficiency and more reliable results.

Devops is software development practice that brings people, process and technology together to deliver continuous value. The

approach is divided into planning and tracking development, build and test, delivery and monitoring and operations. DevOps is unique in that development, IT operations, quality engineering and security teams work together to create efficiency across all tasks involved, in launching a new product release or update.

Agile

Agile development is a delivery approach that relates to lean manufacturing. The fundamentals of agile center around creating a working prototype or build amidst the realities of changing needs and requirements. Bridging the gap between the development team and the end user, adaptability is a core attribute of agile, giving precedence to the needs of users and stakeholders over rigid plans.

The fail-fast mindset of agile centers around adaptability and keeping pace with customer needs and expectations. Features are described as user stories, placing the focus on the individual user, what he or she needs, and why.

Agile development is specific to the development team, its productivity, and its progress toward completing the project at hand. Development is completed in incremental Sprints and software delivery, deployment, and

Date _____
Page 50

Ongoing maintenance of each release is managed by different teams

* Devops Tools :-

Devops Tools is an application that helps automate the software development process. It mainly focuses on communication and collaboration between product management, software development and operations professionals. DevOps tool also enables teams to automate most of the software development processes like build, conflict management, dependency management, deployment, etc. and helps reduce manual efforts.

1) QuerySurge :-

QuerySurge is the smart data testing solution that is the first-of-its-kind full DevOps solution for continuous data testing.

2) Basis Technologies :-

Basis Technologies offers the only DevOps for SAP tool; ActiveControl allows business to move their SAP applications from fixed release cycles to an on-demand delivery model based on CI/CD and DevOps.

3) Keysight Eggplant :-

Keysight Eggplant offers continuous intelligent AI-assisted test automation that seamlessly

integrates with your CI/CD pipeline, automates the entire testing lifecycle, and delivers fast, quality software releases.

4) Datadog:-

Datadog provides a complete monitoring solution so that you have full visibility as you transform or adopt your DevOps culture.

5) Solarwinds DevOps

SolarWinds DevOps helps you to reduce server load with an agentless architecture. It enables you to check the performance of your network servers, and VMware and simplify the daily tasks at your organization.

6) Jenkins:-

Jenkins a DevOps tool for monitoring execution of pre-repeated tasks. It is one of the best software deploy tools which helps to integrate project changes more easily by quickly finding issues.

7) Vagrant:-

Vagrant is a DevOps tool. It is one of the best DevOps tools that allows building & managing virtual machine environments in a single workflow. It offers easy-to-use workflow and focuses on automation. Vagrant lowers development environment setup time and increases production parity.

iii) PagerDuty:-

PagerDuty is a DevOps tools that helps businesses to enhance their brand reputation. It is an incident management solution supporting continuous delivery strategy. It also allows DevOps teams to deliver high-performing apps.

iv) Prometheus:-

Prometheus is 100% open source free to use service monitoring systems. It is one of the best DevOps deployment tools that offers support for more than ten languages.

v) Ganglia :-

Ganglia DevOps tooling offers remain to teams with cluster and grid monitoring capabilities. This tool is designed for high-performance computing systems like clusters and grids.

vi) Snort :- Snort :-

Snort is a very powerful open-source DevOps tool that helps in the detection of intruders. It is one at the top DevOps tools, which highlights malicious attacks against the system. It is one of the best software deploy tools that allows real-time traffic analysis and packet logging.

12) Splunk :

Splunk is a tool to make machine data accessible, usable, and valuable to everyone. It is one of the best DevOps technologies that delivers operational intelligence to DevOps teams. It is one of the best software deployment tools that helps companies to be more productive, competitive, and secure.

13) Nagios:

Nagios is another useful tool for DevOps. It helps DevOps teams to find, and correct problems with network & infrastructure.

* Configuration Management

In software development circles, configuration management refers to the process by which all environments hosting software are configured and maintained.

Every development pipeline requires multiple environments for multiple purposes - unit testing, integration testing, acceptance testing, load testing, system testing, end-user testing, etc. These environments become increasingly complex as the testing moves towards pre-prod and prod environments. Configuration management is an automated process that ensures the configuration of these environments is optimal.

The configuration of testing environments is critical for the success of testing teams. Accurate

12) Splunk :

Splunk is a tool to make machine data accessible, usable, and valuable to everyone. It is one of the best DevOps technologies that delivers operational intelligence to DevOps teams. It is one of the best software deployment tools that helps companies to be more productive, competitive, and secure.

13) Nagios

Nagios is another useful tool for DevOps. It helps DevOps teams to find, and correct problems with network & infrastructure.

* Configuration & Management

In software development circles, configuration management refers to the process by which all environments hosting software are configured and maintained.

Every development pipeline requires multiple environments for multiple purposes - unit testing, integration testing, acceptance testing, load testing, system testing, end-user testing, etc. These environments become increasingly complex as the testing moves towards pre-prod and prod environments. Configuration management is an automated process that ensures the configuration of these environments is optimal.

The configuration of testing environments is critical for the success of testing teams. Naive

Configuration makes every source - Servers, networks, data centers, operating systems, IT assets, configuration files, function as they must to facilitate success. These environments must be meticulously managed, and all configuration changes must be tracked to ensure that they are traceable.

Inadequate configuration management can lead to system outages, data breaches, and leaks, not to mention the fact that bad environments make for improper, incomplete, and shallow tests.

Using configuration Management is imperative in Devops infrastructures. Remember, DevOps is about facilitating speed, accuracy and efficiency. Configuration Management helps to automate mundane maintenance tasks, which frees up dev time for actual programming. This increases agility, both on the part of individual devs and the organization as a whole. At this point, it would be correct to state also that configuration Management is fundamentally necessary for setting up a DevOps-driven framework.

Continuous Integration and Deployment

CI/CD is a practice in DevOps that allows developers to quickly and automatically test, package, and deploy their applications. Most commonly it is often achieved by leveraging a server called Jenkins, which serves as the

most common CI/CD orchestrator. Jenkins listens to specific inputs given by the developer and, when triggered, kicks off a pipeline flow.

* Continuous configuration & integration :-

It is a software development method in which the developer's code gets integrated several times a day. Whenever a developer checks-in the code, each time the change is verified by an automated pipeline (server) and gives early feedback, in case if there are any bugs.

Build → → Merge

* Continuous Deployment :-

It comes after CI and it is the ability to introduce changes with every commit making the code ready for production so that it can be deployed to production on demand and as a routine activity. Code changes can be like new features, bug fixes, updates, configuration changes, etc. One important thing to note here is to ensure safety checks, so the bugs get detected before they get promoted to the production.

Building → Testing → Delivery

* Linux Os introduction :

Linux is one of popular version of UNIX operating system. It is open source as its source code is freely available. It is free to use.

Linux was designed considering UNIX compatibility. Its functionality list is quite similar to that of UNIX.

Linux is a community of open-source Unix like operating systems that are based on the Linux Kernel. It was initially released by Linus Torvalds on September 17, 1991. It is a free and open source operating system and the source code can be modified and distributed to anyone commercially or noncommercially under the GNU General Public license & Initially, Linux was created for personal computers and gradually it was used in other machine like servers, mainframe computers, supercomputers. Nowdays, Linux is also used in embedded systems like routers, automation controls, televisions, digital video recorders, video game consoles, smartwatches, etc. The biggest success of Linux is Android (Operating System) It is based on the linux kernel that is running on Smartphones and tablets. Due to android Linux has the largest installed base of all general-purpose operating systems. Linux is generally packaged in a Linux distribution.

Linux distribution :-

Linux distribution is an operating system that is made up of a collection of software based on linux kernel or you can say distribution contains the linux kernel and supporting libraries and software. And you can get Linux based operating system by downloading one of the Linux distributions.

and these distributions are available for different types of devices like embedded devices, personal computers, etc. Around 600+ Linux Distributions are available and some of the popular linux distributions are.

Mx Linux

Manjaro

Linux Mint

elementary

Ubuntu

Debian

Solus

Fedora

OpenSUSE

Deepin

* Importance of linux in devops:-

The customization aspect of linux is of particular importance for DevOps. It allows for design and security applications specific to a particular development environment or development goals to be created. There is much more freedom over how the operating system functions compared to windows.

Linux offers the DevOps team the flexibility and scalability needed to create a dynamic development process. You can set it up any way that suits your needs. Rather than letting the Operating System dictate how you work, you can configure it to work for you.

Linux makes automation easy, hence, it has become an integral part of DevOps professionals. The best advice anybody can get while starting their journey in DevOps is to learn and understand the basics of Linux thoroughly. This makes the DevOps career path easy in the future. Linux is going to be there no matter what, you have to face it and work with it, to become a great DevOps engineer.

Today, we will see why Linux is famous and reasons that make it so popular among developers and DevOps engineers.

Devops is booming, and in DevOps, Linux knowledge is essential because it helps in automation. Firms look for such a combination of skills that fit well in the DevOps team.

3) Linux Basic Command Utilities :-

1) w

Display who is logged into the system and what process executed by them.

\$w

\$w -h

\$w <username>

2) nmon

Nmon or nigel's monitor is a tool which displays performance information of the system.

\$ sudo apt-get install nmon

1) nmon

nmon can dump information related to networks, cpu, memory or disk usage.

2) ncdt

A command utility is a cursor based version of 'du'. This command is used to analyze disk space occupied by various directories.

```
$ apt-get install ncdt
```

```
$ ncdt /
```

4) Slurm

A command line utility used for command based network interface bandwidth monitoring, it will display ascii based graphic

```
$ apt-get install slurm
```

```
$ slurm -i <interface>
```

```
$ slurm -i eth1
```

5) findmnt

Findmnt command is used to find mount file systems. It is used to list mount devices and can also mount or unmount devices as and when required, it comes as a part of util-linux

```
$ findmnt
```

```
$ findmnt -i
```

6) dstat

A combined and flexible tool which can be used to monitor memory process, network or disk space performance, it is a good replacement of ifstat, iostat, dmstat etc.

\$ apt-get install dstat

7) Sudoar

Another cli based system statistics monitoring tool, provide information about disk user, network, memory, swap etc.

\$ sudo apt-get install sudoar

8) Ss

Ss or socket statistics is a good alternative to netstat, netstat it directly gather information from kernel space and play fast in comparison to the netstat utility.

\$ ss | less

9) CCze

A tool that decorate your logs

\$ apt-get install ccze

10) ranwhen.py

A python based terminal utility that can be used to display system activities graphically.

\$ sudo apt-add-repository
ppa:fkrull/deadsnakes

* Linux administration:

Linux system administration is a process of setting up, configuring and managing a computer system in a linux environment. System administration involves creating a user account, taking reports, performing backup, updating configuration files, documentation, and performing recovery actions. The user who manages the server, fixes configuration issues, recommends new software updates, and updates document is the system administrator.

Linux is a major strength in computing technology. Most of the webserver, mobile phones, personal computers, supercomputers, and cloud-servers are powered by Linux. The job of a Linux systems administrator is to manage the operations of a computer system like maintain, enhance, create user account / report, taking backups using account / report, taking backups using Linux tools and command-line interface tools. Most computing devices are powered by Linux because of its high stability, high security, and open-source environment. There are some of the things that a Linux system administrator should know and understand?

1) Linux File Systems

2) File System Hierarchy

3) Managing Root / Super User

4) Basic Bash command

5) Handling file, Directories and Users

System administration has become a solid criterion for an organization and institute that requires a solid IT foundation. Hence, the need for efficient linux administrators is the requirement of the time. The job profile might change from each organization as there may be added responsibilities and duties to the role.

Linux Systems Administrator is to manage the operations like install, observe the software and hardware systems and taking backup. And also have a good ability to describe an in-depth understanding of technical knowledge. Even freshman level professionals has great possibilities for the position of System Administrator with the yearly median salary.

Environment Variables :-

A DevOps environment aims to mitigate such scenarios by encouraging automation and documentation, thereby increasing communication.

In this environment, every release is committed in an automated fashion, enabling the rapid building, testing and deployment.

of every project.

Firstly a DevOps environment leads to an increase in visibility and a decrease in the risk of latent uncertainty because development & operations teams are more tightly aligned and have insight into what each team is working on, upcoming projects and more. This means that the next time a member on the development team asks the operations team to fix a crashed server, members on at the operations team already have visibility into what is in the pipeline and can strategically incorporate requests and other tasks into their workflow. Similarly, because the development team has visibility into the work that the operations team is conducting, they are able to prioritize accordingly. As a result, the increase in visibility not only improves communication between teams, but also increases time to market.

In addition to improving visibility, a DevOps environment also reduces the risk of uncertainty. But this is not only related to timelines and project delivery, as in the example. DevOps environment team members already have visibility and can work closely together to solve such an error and pinpoint its source.

+ DevOps Networking:-

Network DevOps refers to the acceptance of DevOps culture and principles in network engineering and operations. It embodies a preference for network automation including "network as API" concepts along with continuous development, integration and deployment of new networking technologies.

DevOps is a company software development idiom adopted to determine a type of flexible relationship among development and IT Operations. The aim of DevOps is to develop and improve the connection by supporting better communication and collaboration within these couple of business units.

Network DevOps refers to the approach of DevOps culture and policies in network engineering and processes. It represents a decision for network automation including "network as API" thoughts, simultaneously with constant development combination and deployment of novel networking technologies. This methodology follows including an "instrument everything" comprehension that endeavours to programmatically implement network execution metrics behind typical networking organisations to a wide audience including network engineers and also network operations organisation.

DevOps for networking has evolved in the circumstances of IT and production.

Organisations adopting DevOps and rapid development using different programming languages while software development companies seem to work more collaboratively and in a constant process of the development stage onwards, network organisations that are secured in a more standard and siloed way of functioning become more difficult.

f) Linux Server installation :-

- 1) Select your language and location
- 2) In "Network Configuration", choose your network card (eth0) for the first network card.
- 3) Enter your chosen "hostname"
- 4) Enter the "username" and "password" for the system administrator
- 5) Set your time zone.