DevOps Road Map

**Programming language:-**

**Python**

1. Used in automation of Devops life cycle management
2. Used in CI/CD Pipeline automation
3. Python script can be used to automate the small day to day observing task and checking
4. Organizing and using with the help of python tools

**Operating System:-**

**Linux:**

1. Linux's open-source nature allows for extensive customization to meet specific needs.
2. Its stability and performance make it ideal for high-load production environments.
3. Powerful scripting tools in Linux facilitate automation and streamline DevOps processes.

Package managers in Linux simplify software installation and updates.

**Version Control System:-**

**Git:**

1. Git provides version control to track and manage code changes effectively.
2. Its branching and merging features support parallel development workflows.
3. Git integrates with CI/CD tools to automate testing and deployments.
4. Collaboration is streamlined through pull requests and code reviews.
5. Git's rollback capabilities allow for easy recovery from errors.

**Container:-**

**Docker:**

1. Docker containers package applications and dependencies together for consistent environments.
2. It simplifies deployment across different environments by ensuring consistency.
3. Docker's lightweight containers improve resource efficiency and scalability.
4. It supports rapid scaling and orchestration with tools like Kubernetes.

**Networking & Protocols:-**

**Http:-**

1. HTTP creates communication between DevOps tools and web services.
2. It enables RESTful APIs for integrating various applications and services.
3. HTTP is used for retrieving and sending data during CI/CD pipelines.
4. It supports monitoring and logging by accessing web-based dashboards and metrics.

**Cloud providers:-**

**Azure:-**

1. Azure DevOps provides comprehensive tools for planning, developing, and managing software projects.
2. Azure Pipelines automates build and release processes with continuous integration and deployment.
3. Azure Repos offers Git-based version control for efficient source code management.
4. Azure Monitor and Application Insights enable detailed monitoring and performance tracking of applications.

**AWS**:-

1. AWS Code Pipeline automates build, test, and deployment workflows for continuous integration and delivery.
2. AWS Code Build provides scalable and managed build services for compiling source code.
3. AWS Code Deploy automates application deployment across various compute services.
4. AWS Cloud Watch offers comprehensive monitoring and logging for performance and operational insights.

**Serverless:-**

**Cloudflare:-**

1. Cloudflare provides global content delivery network services for faster and more reliable website performance.
2. It offers DDoS protection to safeguard applications from distributed denial-of-service attacks.
3. Cloudflare’s DNS services ensure low-latency and high-availability domain resolution.
4. It supports secure communication with SSL/TLS certificates and web application firewall features.

**Configuration Management:-**

**Ansible:-**

1. Ansible automates configuration management and deployment tasks using simple, readable yaml playbooks.
2. It ensures consistency across environments by defining and managing infrastructure as code.
3. Ansible integrates with various systems and tools, enhancing DevOps workflows.

**CI/CD TOOLS:-**

**Jenkins:-**

1. Jenkins streamlines the continuous integration and continuous delivery processes by automating builds, tests, and deployments.
2. Jenkins can integrate with many tools and technologies, allowing customization to fit various workflows
3. Jenkins can distribute build tasks across multiple machines, enhancing scalability and efficiency

**Circle CI:-**

1. Circle CI automates build, test, and deployment pipelines for faster development.
2. It integrates with version control systems like Git Hub and Bit bucket for workflows.
3. Circle CI provides scalable infrastructure with parallel execution to optimize performance.

**Container orchestration:-**

**Kubernetes:**

* 1. Kubernetes automates container orchestration, scaling, and management.
  2. It ensures high availability and fault tolerance by distributing workloads across clusters.
  3. Kubernetes supports rolling updates and rollbacks for application deployments.
  4. It provides advanced networking and service discovery for efficient micro services communication.