DEVOPS and DEVSECOPS

**DevOps :**

DevOps is a combination of "development" and "operations," is a set of practices and tools designed to streamline the software delivery process. It fosters collaboration, communication, integration, and automation between development and operations teams, leading to faster and more reliable software delivery.

**Application types :**

Thick Client/Desktop Applications :

Desktop applications are software programs installed directly on a user's computer (or desktop/laptop), allowing them to function even without an internet connection.

Thin Client/Web Based Applications:

Web-based applications, also known as web apps, are software programs that run in a web browser, like Google Chrome, Firefox, or Safari.

Mobile Applications :

Mobile applications are software programs designed to run on mobile devices like smart phones, tablets, and even smart watches. They are downloaded and installed through app stores like Apple's App Store and Google Play, unlike web applications that run in a browser.

Kubernetes :

Kubernetes (often abbreviated as K8s) is an open-source container orchestration platform that automates the deployment, scaling, and management of containerized applications. It was originally developed by Google and is now maintained by the Cloud Native Computing Foundation (CNCF).

Core components :

A Kubernetes cluster is composed of two main parts: the control plane and the worker nodes.

Control plane (Master) :

* The "brain" of the cluster, responsible for making global decisions and responding to cluster events.
* API server (kube-apiserver): The central hub that exposes the Kubernetes API and handles all communication within the cluster.
* Scheduler (kube-scheduler): Watches for newly created pods and assigns them to the best-suited nodes based on resource availability and other constraints.
* Controller Manager (kube-controller-manager): Runs controller processes that regulate the shared state of the cluster. For example, a replication controller ensures the correct number of pods is running.
* etcd: A consistent and highly available key-value store that holds all cluster data, including configuration and state.

Worker nodes :

* The machines where the actual application workloads run.
* Kubelet: An agent that runs on each node and communicates with the control plane. It ensures that containers are running and healthy as defined in the pod specifications.
* Kube-proxy: A network proxy that maintains network rules on each node, enabling network communication to and from pods.
* Container runtime: The software that runs the containers, such as Docker, containerd, or CRI-O.

kubernetes in AI, ML, and IOT :

Kubernetes provides scalable infrastructure and automation for AI, ML, and IoT workloads, which are often computationally intensive and have diverse deployment requirements.

In AI and ML :

Kubernetes has become a foundational platform for MLOps (Machine Learning Operations), helping data science and operations teams move models from experimentation to production.

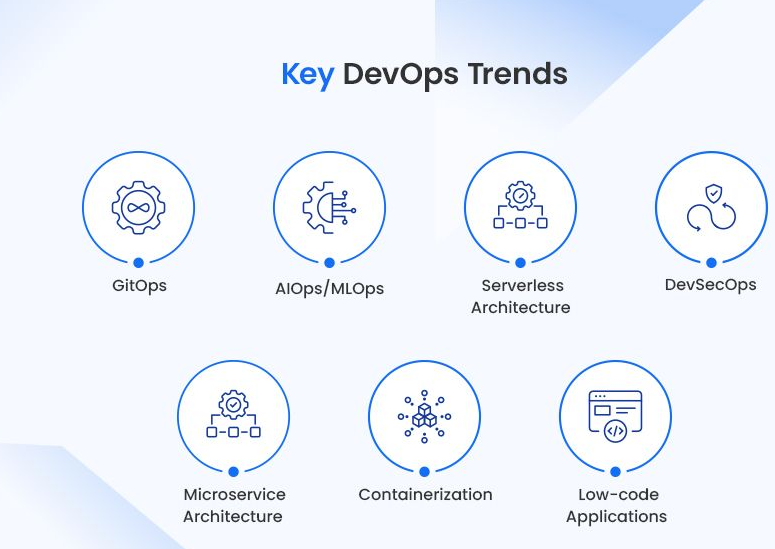
In IoT and edge computing :

Kubernetes is used to manage and orchestrate the workloads that run on edge devices, addressing the unique challenges of a distributed network. Specialized, lightweight distributions of Kubernetes, such as K3s, are designed for resource-constrained edge devices.

Multi Party Computing :

Multi-party computation (MPC) provides a significant security upgrade for DevOps by protecting sensitive data, such as encryption keys and credentials, throughout the CI/CD pipeline.

DevOps Trends :



Low-Code and No-code Platforms :

Low-code platforms :

Low-code platforms are development environments for creating applications with minimal manual coding. They are primarily used by professional developers and "citizen developers" (non-technical users with some coding aptitude) to accelerate complex development tasks.

Ex : Mendix, Appain, GeneXus

No-code platforms :

No-code platforms allow anyone to build applications using visual tools, with no coding knowledge required. They are ideal for business users and entrepreneurs who need to develop simple apps or automate workflows.

Ex : Webflow, Airtable, Bubble

Devops Edge Computing :

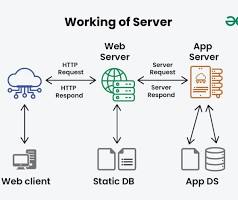
DevOps for edge computing, often referred to as "DevEdgeOps," is the practice of applying DevOps principles and practices to develop, deploy, and manage applications and infrastructure at the edge of the network.

Why DevOps is essential for edge computing ?

Edge computing moves data processing and storage closer to the source, reducing latency and reliance on centralized cloud infrastructure. DevOps methodologies, with their focus on automation, collaboration, and continuous delivery, are crucial for managing this distributed and complex environment.

Server :

what is server?



A server is a software or hardware device that processes requests and provides services to other computers, known as clients, over a network.

Types of Servers :

Web server :

A web server delivers static content, such as HTML pages, images, and videos, to clients via the internet. When you enter a URL into a web browser, the browser sends an HTTP request to a web server, which then sends the requested content back to you.

* Examples: Apache HTTP Server and Nginx.

Application server :

An application server hosts and runs software applications, providing business logic and dynamic content to multiple users simultaneously. It typically works with a web server to handle requests that require more complex processing, like database interactions.

* Examples: Oracle WebLogic, Apache Tomcat, and JBoss.

Database server :

This type of server manages, stores, and retrieves organized data for a network. Applications, clients, and web servers can query a database server to access or manipulate data.

* Examples: MySQL, Oracle, and Microsoft SQL Server.

Software Development Lifecycle :

The software development lifecycle (SDLC) is a structured process used by software teams to design, develop, test, and maintain high-quality software.



* Planning
* Define Requirements
* Design and prototyping
* Software Development
* Testing
* Deployment
* Operations and maintainence

Types of Environments :



Development Sandbox Environment Testing Environment Staging Environment Production

Environment

DEVSECOPS :

DevSecOps (Development, Security, and Operations) is a set of practices that integrates security into every stage of the software development lifecycle (SDLC), or "shifts security left".

Benefits :

* Observability
* Traceability
* Confidence
* Compilance

Use Cases :

* Lack of visibility
* Troubled audits
* Unified Governance
* Risk Mitigation