

Introduction to Cloud Computing & DevOps Fundamentals

1. What is Cloud Computing?

Cloud computing is the delivery of computing services (servers, storage, databases, networking, software, analytics, etc.) over the internet instead of using local physical hardware.

Instead of buying and maintaining physical servers, organizations rent resources from cloud providers like:

- Amazon Web Services (AWS)
- Microsoft Azure
- Google Cloud Platform

Why Organizations Use Cloud Computing?

- **Cost Efficient** – Pay-as-you-go pricing model
 - **Scalability** – Easily scale resources up or down
 - **High Availability** – Data centers across multiple regions
 - **Security** – Built-in security and compliance tools
 - **Faster Deployment** – Launch servers in minutes
-

2. Cloud Service Models

1. IaaS (Infrastructure as a Service)

Provides virtual machines, storage, and networking.

Example:

- AWS EC2
- Azure Virtual Machines

You manage: OS, runtime, applications
Provider manages: Hardware & infrastructure

2. PaaS (Platform as a Service)

Provides a platform to develop and deploy applications without managing infrastructure.

Example:

- Azure App Service
- Google App Engine

You manage: Application & data

Provider manages: OS, runtime, infrastructure

3. SaaS (Software as a Service)

Ready-to-use software delivered via internet.

Examples:

- Google Docs
- Microsoft Office 365
- Salesforce

Users just use the software — no infrastructure management required.

3. Basics of DevOps

DevOps is a combination of **Development (Dev)** and **Operations (Ops)**.

It is a culture and set of practices that aims to:

- Automate software delivery
 - Improve collaboration between teams
 - Deliver software faster and more reliably
-

DevOps Principles

- **Collaboration** – Dev & Ops work together
 - **Automation** – Automate builds, tests, deployments
 - **Continuous Integration (CI)** – Frequently merge code
 - **Continuous Delivery (CD)** – Automatic deployment
 - **Monitoring & Feedback** – Continuous improvement
-

4. DevOps Lifecycle

- Plan

- Code
- Build
- Test
- Release
- Deploy
- Operate
- Monitor

Popular tools used:

- GitHub – Version control
 - Jenkins – CI/CD
 - Docker – Containers
 - Kubernetes – Container orchestration
-

5. Traditional Software Development vs DevOps

Traditional Model	DevOps Model
Separate Dev & Ops teams	Collaborative culture
Manual deployments	Automated deployments
Slow release cycles	Faster & frequent releases
Late testing	Continuous testing
Higher failure risk	Continuous monitoring & feedback

Traditional Workflow

Development → Testing → Deployment → Maintenance
(Sequential and slow)

DevOps Workflow

Continuous integration, testing, deployment, monitoring
(Automated and fast)

6. How Cloud & DevOps Work Together

Cloud provides:

- Scalable infrastructure
- On-demand resources
- Global availability

DevOps provides:

- Automation
- CI/CD pipelines
- Faster delivery

Together they enable:

- Microservices architecture
- Containerization
- Infrastructure as Code (IaC)
- Rapid application deployment