Introduction to DevOps & Cloud

Understand Development and Operations(DevOps)

1. Software Development & IT Operations

Key Concepts

1. **Dev (Development):** The team responsible for writing code, designing features, and fixing bugs.

2. **Ops (Operations):** The team responsible for servers, networks, security, scaling, and keeping the application running 24/7.

2. Problems with Traditional IT (Silos & Slow Releases)

Key Concepts

1

- **Silos:** Teams working in isolation, not sharing information or tools.

- **Wall of Confusion:** The barrier where Devs don’t understand Ops problems, and Ops don’t understand the code.

- **Bottlenecks:** Manual processes (like manual testing) that slow everything down

3. What is DevOps

Key Concepts

1

- **- Collaboration:** Dev and Ops work together daily, not just on release day.
- **-Shared Responsibility:** Everyone owns the quality and availability of the product.
- **- Automation:** Replacing manual human work with computer scripts

4. DevOps Culture & Principles

Key Concepts

1

- **- C.A.L.M.S. Framework:**
- - **C**ulture: Communication and trust
- - **A**utomation: Automate manual tasks.
- - **L**ean: Minimize waste and wait times.
- - **M**easurement: Use data to improve.
- - **S**haring: Share knowledge and success.

5. DevOps Lifecycle

Key Concepts

1

- **- Plan:** Decide what to build.
- **- Code:** Write the software.
- **- Build:** Package the code into an executable file.
- **- Test:** Checks for bugs.
- **- Release:** Approve the changes.
- **- Deploy:** Put it on the live servers.
- **- Operate:** Keep it running.
- **- Monitor:** Watch for errors and user feedback.

.

6. CI/ CD Concept (High Level)

Key Concepts

1

- **-Build Automation:** Compiling code automatically.
- **- Automated Testing:** Running tests without humans.
- **- Pipeline:** The set of automated steps code goes through (Build → Test → Deploy).

.

7. Automation & Infrastructure as Code (IaC)

Key Concepts

1

- **- Declarative:** You tell the tool what you want (e.g., “I want 3 servers”), and it figures out how to do it.
- **- Version Control:** You can save your infrastructure scripts in Git, just like software code.

8. What is Cloud Computing

Key Concepts

1

- **- On-Demand:** Get a server whenever you want.
- **- Pay-as-you-go:** Pay only for what you use, like a utility bill.
- **- Broad Network Access**: Access via the internet.

.

9. Why Cloud is Needed

Key Concepts

1

- **- Elasticity:** Automatic scaling up and down.
- **- Cost Efficiency:** No wasted resources.
- **- Global Reach:** Deploy code to Japan, Us, and Europe instantly.

.

10. Cloud Service Models (IaaS, PaaS, SaaS)

Key Concepts

1

- **- IaaS:** AWS EC2. Google Compute Engine. You manage the OS and Apps.
- **- PaaS:** Heroku, Google App Engine. You manage only the code.
- **- SaaS:** Gmail, Salesforce, Dropbox. You manage nothing but your settings.

.

11. Cloud Deployment Models

Key Concepts

1

- **- Public:** AWS, Azure, Google Cloud (GCP).
- **- Private:** Servers dedicated to one company (often on-premise).
- **- Hybrid:** Connecting Public and Private clouds together.
- **- Multi-Cloud:** Using AWS and Azure at the same time.

.

12. DevOps + Cloud Together

Key Concepts

1

- **- Cloud-Native:** Building apps specifically to run on the cloud (Using containers/microservices).
- **- Programmable Infrastructure:** The cloud allows hardware to be controlled by code, which is essential for DevOps automation.

13. Real-World DevOps & Cloud Workflow

Key Concepts

1

- **- Source Control (Git):** Developer saves code.
- **- CI Server (Jenkins/GitHub Actions):** Robot picks up code and tests it.
- **- Artifact Registry:** Robot saves the approved application.
- **- Cloud Deploy:** Robot sends app to AWS//Azure.

14. Career Path Overview

Key Concepts

1

- **- Junior DevOps:** Knows Linux, basic Scripting, and Git, Can manage simple pipelines.
- **- Senior DevOps:** Knows Cloud architecture, Security (DevSecOps), and complex automation.
- **- SRE (Site Reliability Engineer):** A specialized role focused purely on reliability and scaling (Google's version of DevOps).

.