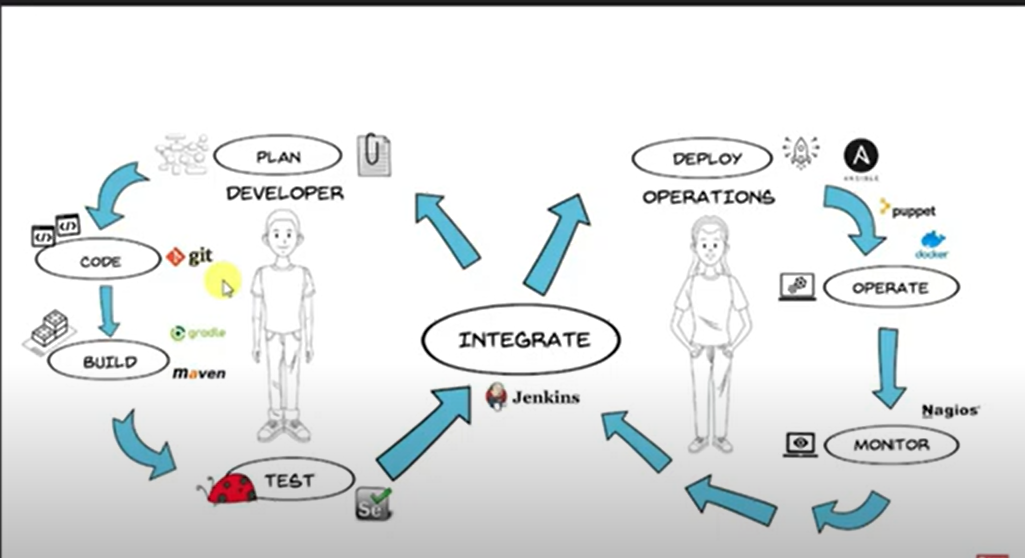
**DevOps:-**

**1.What is DevOps?**

DevOps is a combination of **Development (Dev)** and **Operations (Ops)** that focuses on automating and integrating the processes between software development and IT operations. It aims to improve collaboration, speed, and reliability in software delivery.



A diagram of a software development process

AI-generated content may be incorrect.

**2.Real-World Example: Online Shopping App**

Imagine you are working on an **online shopping app** like Amazon or Flipkart. The development team writes the code for new features (like a new discount system), but they need to make sure:

1. **Code is error-free**
2. **Works smoothly in production**
3. **Runs efficiently on the server**

Before DevOps:

* Developers write code and pass it to another team for testing.
* After testing, another team deploys it to production.
* If errors occur, the feedback loop is slow.

With DevOps:

* **Automation & Continuous Integration:** Whenever a developer writes new code, tools like **Jenkins/GitHub Actions** automatically test it.
* **Continuous Delivery:** If the code passes tests, it is deployed automatically to production using **Docker & Kubernetes** without waiting for manual intervention.
* **Monitoring & Feedback:** If anything goes wrong, monitoring tools like **Prometheus/Grafana** help detect issues quickly.

**3.What is Operations in DevOps?**

In traditional software development, **operations** refers to the **IT team** responsible for managing servers, deploying applications, monitoring performance, and ensuring security. They handle:  
✅ Server setup and maintenance  
✅ Deploying applications to production  
✅ Monitoring system health and fixing issues  
✅ Managing security, backups, and disaster recovery

In **DevOps**, operations is integrated with development to automate these tasks and make software delivery faster and more reliable.

**4.How Does DevOps Work? (Step-by-Step with Real-World Example)**

Let’s take an **online shopping app (like Amazon or Flipkart)** as an example to understand how DevOps works.

**Step 1: Planning & Development**

**Scenario:** A new feature (e.g., "Buy 1 Get 1 Free") is requested.

🔹 **Before DevOps:** Developers write code without collaborating with operations, leading to deployment issues later.  
🔹 **With DevOps:** Developers and operations work together from the start.

**Tools Used:** Jira, Trello (for planning), Git (for version control)

**Step 2: Continuous Integration (CI)**

**Scenario:** A developer writes code for the new feature and pushes it to a repository (GitHub/GitLab).

🔹 **Before DevOps:** Developers submit code manually, and testing is done later. Bugs may appear in production.  
🔹 **With DevOps:** As soon as code is pushed, **CI tools (Jenkins, GitHub Actions)** automatically test the code.

✅ If tests pass → Code moves to the next stage.  
❌ If tests fail → Developer fixes the issue immediately.

**Tools Used:** Jenkins, GitHub Actions, Azure DevOps

**Step 3: Continuous Delivery (CD)**

**Scenario:** The tested code needs to be deployed to a test server.

🔹 **Before DevOps:** Manual deployment takes time and may introduce errors.  
🔹 **With DevOps:** **CD tools** automatically deploy the tested code to a staging environment.

**Tools Used:** Docker, Kubernetes, Azure DevOps, AWS CodeDeploy

**Step 4: Continuous Deployment (CD) to Production**

**Scenario:** The new "Buy 1 Get 1 Free" feature is ready and should go live for customers.

🔹 **Before DevOps:** The operations team manually deploys the feature, which can lead to delays.  
🔹 **With DevOps:** Automated deployment tools push changes to production instantly.

**Tools Used:** Kubernetes, Terraform, Ansible

**Step 5: Monitoring & Feedback**

**Scenario:** The feature is live, but customers complain about slow performance.

🔹 **Before DevOps:** The operations team checks logs manually to find issues.  
🔹 **With DevOps:** Monitoring tools detect issues in real-time and alert the team.

✅ If CPU usage is high → Auto-scaling increases server capacity.  
✅ If errors occur → DevOps tools provide instant alerts.

**Tools Used:** Prometheus, Grafana, ELK Stack, Datadog

**Summary: DevOps Workflow for an Online Shopping App**

1️⃣ **Plan & Develop** – Write code (Git, Jira)  
2️⃣ **Continuous Integration** – Test automatically (Jenkins, GitHub Actions)  
3️⃣ **Continuous Delivery** – Deploy to staging (Docker, Kubernetes)  
4️⃣ **Continuous Deployment** – Release to production (Terraform, Ansible)  
5️⃣ **Monitor & Feedback** – Detect & fix issues (Prometheus, Grafana)