

# CI/CD Best Practices for Data Pipeline Deployment and Monitoring

## Basic Concepts (Beginner – 15 Questions)

1. What is CI/CD?
2. What is the difference between Continuous Integration and Continuous Deployment?
3. What are the benefits of using CI/CD?
4. What problems does CI/CD solve?
5. How does CI/CD relate to DevOps?
6. What are the core components of a CI/CD pipeline?
7. What is the difference between Continuous Deployment and Continuous Delivery?
8. What is a pipeline in CI/CD?
9. Why is automation important in CI/CD?
10. What is a build server?
11. What is a version control system, and why is it essential for CI/CD?
12. How does CI/CD improve software quality?
13. What are some common CI/CD tools?
14. What is artifact management in CI/CD?
15. What is the purpose of staging environments?

## Tools & Platforms (Intermediate – 15 Questions)

16. Compare Jenkins and GitLab CI.
17. What is CircleCI and how does it differ from Travis CI?
18. How do you configure a Jenkins pipeline?
19. What is a Jenkinsfile?
20. What are GitHub Actions, and how do they support CI/CD?
21. What is an agent in Azure DevOps?
22. What is the role of a runner in GitLab CI?
23. What is a self-hosted runner?

24. What are webhooks and how are they used in CI/CD?
25. How do you manage secrets and credentials in a CI/CD pipeline?
26. What is Argo CD?
27. What is Spinnaker?
28. What is Bamboo, and how does it fit into CI/CD?
29. What is the difference between hosted and on-premise CI/CD solutions?
30. How would you trigger a pipeline based on specific conditions?

## **Testing & Quality (Intermediate – 10 Questions)**

31. What types of tests should be included in a CI pipeline?
32. How do you handle flaky tests in CI?
33. What is test coverage, and how is it tracked in CI/CD?
34. How can you run unit tests and integration tests separately?
35. What are some tools for automated testing in CI/CD?
36. What is static code analysis, and how does it integrate with CI/CD?
37. What is SonarQube used for?
38. How can you ensure security testing is part of the CI/CD pipeline?
39. What is a code smell?
40. How do you handle failed builds?

## **Deployment Strategies (Advanced – 10 Questions)**

41. What is blue-green deployment?
42. What is a canary deployment?
43. How do you perform a rolling deployment?
44. What is feature flagging?
45. How would you implement zero-downtime deployment?
46. What is the purpose of a rollback mechanism?
47. How do you manage database migrations in CI/CD?
48. What is immutable infrastructure?
49. What is infrastructure as code (IaC), and how does it fit into CI/CD?
50. How do you handle configuration management in CI/CD?

## Security, Monitoring & Best Practices (Advanced – 10 Questions)

51. What is DevSecOps?
52. How do you integrate vulnerability scanning into a pipeline?
53. What is a software supply chain attack and how can CI/CD be protected against it?
54. How do you implement audit logging in CI/CD?
55. What is the principle of least privilege in CI/CD?
56. What is container scanning and when should it occur in the pipeline?
57. What tools are used for secrets management in CI/CD?
58. How can you monitor the health of a CI/CD pipeline?
59. What is drift detection in infrastructure?
60. How do you secure pipeline code itself (e.g., Jenkinsfile, GitHub Actions YAML)?

## Scenario-Based & Troubleshooting (Expert – 10 Questions)

61. CI builds are taking too long. How would you optimize the pipeline?
62. A deployment failed in production — what steps do you take to investigate?
63. How do you handle dependency versioning issues in CI/CD?
64. Your tests are passing locally but failing in CI. What could be wrong?
65. How would you structure a monorepo for multiple CI/CD pipelines?
66. How do you test environment-specific configurations before deployment?
67. A feature is deployed but not working — what CI/CD logs would you check?
68. How would you roll back a deployment if it introduces a bug?
69. What would you do if two developers' commits are causing merge conflicts in CI?
70. How do you handle hotfixes in a CI/CD process?

## 10 Interview Questions

**Q1. What are the key differences between CI/CD for software applications and CI/CD for data pipelines?**

*Expected: Emphasis on schema management, data validation, and stateful processing.*

**Q2. How do you implement version control for data pipeline code and configurations?**

*Expected: Using Git, storing DAGs/scripts as code, tracking schema changes.*

**Q3. What testing strategies are best suited for validating data pipelines in CI/CD?**

*Expected: Unit tests for transforms, integration tests for data flow, data quality checks.*

**Q4. How can you use CI/CD tools (like Jenkins, GitLab CI, or GitHub Actions) to deploy Apache Airflow or dbt pipelines?**

*Expected: Using Docker containers, YAML config files, triggering DAG uploads or dbt runs.*

**Q5. How do you handle secrets and credentials securely in a CI/CD process for data platforms?**

*Expected: Use of secret managers like AWS Secrets Manager, Vault, or CI tool's secret storage.*

**Q6. What are some effective ways to monitor deployed data pipelines automatically?**

*Expected: Integrating with monitoring tools like Prometheus, Airflow logs, alerting via Slack/Email.*

**Q7. How do you roll back a failed data pipeline deployment in a CI/CD environment?**

*Expected: Versioning DAGs/scripts, immutable releases, infrastructure snapshots.*

**Q8. How do you ensure data quality during and after CI/CD deployments?**

*Expected: Automated checks using tools like Great Expectations or custom validation scripts.*

**Q9. What's the role of containerization (Docker, Kubernetes) in CI/CD for data pipelines?**

*Expected: Consistent environment, scalable deployment, reproducibility.*

**Q10. How do you set up CI/CD to handle schema evolution or data model changes safely?**

*Expected: Schema registry, backward compatibility checks, migration scripts tested pre-deploy.*