

# DOP-C02 Cheat Sheet

High-yield AWS DevOps Engineer Professional review for the last pass before practice tests.

<b>BEST FOR</b>  Last review before mini test or app session	<b>FOCUS</b>  CI/CD, IaC, observability, security, and resilience	<b>USE WITH</b>  Quick Summary + Service Map
--	---	--

<b>1. SDLC automation and CI/CD</b>  <b>CodePipeline</b> - Use to orchestrate source, build, test, approval, and deploy stages. <b>CodeBuild</b> - Ephemeral build environments fit repeatable package, test, and scan jobs. <b>Blue/green</b> - Reduce deployment risk by shifting traffic only after validation passes. <b>GitOps inputs</b> - Keep infra and app changes versioned, reviewed, and rollback-friendly. <b>Artifact discipline</b> - Promote immutable build outputs instead of rebuilding per stage.	<b>2. Configuration management and IaC</b>  <b>CloudFormation</b> - Declarative stacks suit repeatable infra with change sets and drift checks. <b>Systems Manager</b> - Parameter Store, Automation, Patch Manager, and Run Command reduce toil. <b>AMI vs containers</b> - Bake images for stable baselines; use containers for app portability. <b>Secrets Manager</b> - Store and rotate credentials instead of embedding secrets in pipelines. <b>Config rules</b> - Continuously detect noncompliant resources and trigger remediation.
---	---

<b>3. Monitoring and observability</b>  <b>CloudWatch</b> - Metrics, alarms, dashboards, logs, and composite alarms anchor operations. <b>X-Ray and tracing</b> - Use request traces to isolate latency in distributed services. <b>Structured logs</b> - Emit searchable fields so alarms and analysis are reliable. <b>EventBridge</b> - Route operational events to automation, ticketing, or remediation flows. <b>SLO thinking</b> - Alert on user-impact signals, not just raw host or service noise.	<b>4. Resilience, security, and governance</b>  <b>Auto Scaling</b> - Scale on demand and replace unhealthy instances automatically. <b>IAM least privilege</b> - Scope pipeline, runtime, and human roles separately. <b>KMS + encryption</b> - Protect data at rest and sign or encrypt deployment artifacts. <b>Cross-account patterns</b> - Use delegated roles and separate environments for blast-radius control. <b>Rollback paths</b> - Every deployment plan should include fast revert or safe forward-fix steps.
---	---