CNPA최신업데이트시험대비자료, CNPA시험대비최신버전덤프샘플


Cisco 300-620최신 업데이트 시험공부자료 - 300-620최신버전 시험공부자료

최신 CCNP Data Center 300-620 무료샘플문제 (Q191-Q196):

Fast2test의 덤프선택으로Linux Foundation CNPA인증시험에 응시한다는 것 즉 성공과 멀지 않았습니다. 여러분의 성공을 빕니다.

Linux Foundation CNPA 시험요강:

주제	소개
주제 1	• Platform Engineering Core Fundamentals: This section of the exam measures the skills of Supplier Management Consultants and covers essential foundations such as declarative resource management, DevOps practices, application environments, platform architecture, and the core goals of platform engineering. It also includes continuous integration fundamentals, delivery approaches, and GitOps principles.
주제 2	• Continuous Delivery & Platform Engineering: This section measures the skills of Supplier Management Consultants and focuses on continuous integration pipelines, the fundamentals of the CI • CD relationship, and GitOps basics. It also includes knowledge of workflows, incident response in platform engineering, and applying GitOps for application environments.

주제 3	• Platform Observability, Security, and Conformance: This part of the exam evaluates Procurement Specialists on key aspects of observability and security. It includes working with traces, metrics, logs, and events while ensuring secure service communication. Policy engines, Kubernetes security essentials, and protection in CI • CD pipelines are also assessed here.

CNPA시험대비 최신버전 덤프샘플 - CNPA높은 통과율 인기 덤프자료

IT업계에서 자신만의 위치를 찾으려면 자격증을 많이 취득하는것이 큰 도움이 될것입니다. Linux Foundation 인증 CNPA시험은 아주 유용한 시험입니다. Linux Foundation 인증CNPA시험출제경향을 퍼펙트하게 연구하여Fast2test에서는Linux Foundation 인증CNPA시험대비덤프를 출시하였습니다. Fast2test에서 제공해드리는Linux Foundation 인증 CNPA시험덤프는 시장에서 판매하고 있는Linux Foundation 인증CNPA덤프중 가장 최신버전덤프로서 덤프에 있는 문제만 공부하시면 시험통과가 쉬워집니다.

최신 Cloud and Containers CNPA 무료샘플문제 (Q24-Q29):

질문 # 24
In a GitOps workflow, how should application environments be managed when promoting an application from staging to production?

- A. Use a tool to package the application and deploy it directly to production.
- B. Manually update the production environment configuration files.
- C. Create a new environment for production each time an application is updated.
- D. Merge changes and let a tool handle the deployment

정답：D

설명:
In GitOps workflows, the source of truth for environments is stored in Git. Promotion from staging to production is managed by merging changes into the production branch or repository. Option A is correct because once changes are merged, the GitOps operator (e.g., Argo CD, Flux) automatically detects the updated desired state in Git and reconciles it with the production environment.
Option B (creating new environments each time) is inefficient and unnecessary. Option C (manual updates) violates GitOps principles of automation and auditability. Option D (direct deployments) reverts to a push- based CI/CD model rather than GitOps' pull-based reconciliation.
By relying on Git as the single source of truth, GitOps ensures version control, auditability, and rollback capabilities. This allows consistent, reproducible promotion between environments while reducing human error.
References:- CNCF GitOps Principles- CNCF Platforms Whitepaper- Cloud Native Platform Engineering Study Guide

질문 # 25
Which of the following statements describes the fundamental relationship between Continuous Integration (CI) and Continuous Delivery (CD) in modern software development?

- A. CD is a prerequisite for CI; CD automates the deployment of code and CI builds upon this by automating the integration of code changes.
- B. CI is a prerequisite for CD; CI automates the building and testing of code, and CD builds upon this by automating the release process.
- C. CI and CD are interchangeable terms; they both refer to the process of automating software release management.
- D. CI and CD are entirely separate practices; CI focuses on code quality, while CD focuses on infrastructure management.

정답：B

설명:
Continuous Integration (CI) and Continuous Delivery (CD) are complementary practices. Option A is correct:

CI is a prerequisite for CD. CI focuses on automating code integration by building, testing, and validating changes, ensuring code quality and early detection of defects. CD builds upon CI by automating the process of releasing validated builds into staging and production environments, making delivery repeatable and reliable.
Option B incorrectly treats them as entirely separate. Option C reverses the relationship, as CD cannot exist without CI pipelines. Option D is inaccurate because CI and CD are not interchangeable-they represent distinct stages in the software delivery lifecycle. Together, CI/CD accelerates software delivery, reduces risk, and improves quality. In platform engineering, CI /CD pipelines are critical enablers of developer productivity and efficient operations.
References:- CNCF Platforms Whitepaper- Continuous Delivery Foundation Guidance- Cloud Native Platform Engineering Study Guide

질문 # 26
In designing a cloud native platform, which architectural feature is essential for allowing the integration of new capabilities like self-service delivery and observability without specialist intervention?

- A. Monolithic architecture with no APIs.
- B. Extensible architecture with modular components.
- C. Centralized integration through specialist API gateways.
- D. Static architecture with rigid components.

정답： B

설명:
An extensible architecture with modular components is crucial for modern platform engineering. Option C is correct because modularity allows new capabilities (e.g., self-service delivery, observability, or security features) to be added or replaced without disrupting the whole system. This approach promotes agility, scalability, and maintainability.
Option A (monolithic architecture) restricts flexibility and slows innovation. Option B (centralized API gateways) may help integration but still creates bottlenecks if every addition requires specialist intervention.
Option D (static architecture) locks the platform into rigid patterns, preventing adaptation to evolving needs.
Extensible, modular design is a hallmark of cloud native platforms. It enables composability, where services (like service mesh, logging, monitoring, or provisioning APIs) can be plugged in as needed. This architecture supports golden paths and self-service abstractions, reducing developer friction while keeping governance intact.
References:- CNCF Platforms Whitepaper- CNCF Platform Engineering Maturity Model- Cloud Native Platform Engineering Study Guide

질문 # 27
In the context of observability, which telemetry signal is primarily used to record events that occur within a system and are timestamped?

- A. Metrics
- B. Alerts
- C. Logs
- D. Traces

정답： C

설명:
Logs are detailed, timestamped records of discrete events that occur within a system. They provide granular insight into what has happened, making them crucial for debugging, auditing, and incident investigations.
Option A is correct because logs capture both normal and error events, often containing contextual information such as error codes, user IDs, or request payloads.
Option B (alerts) are secondary outputs generated from telemetry signals like logs or metrics and are not raw data themselves.
Option C (traces) represent the flow of requests across distributed systems, showing relationships and latency between services but not arbitrary events. Option D (metrics) are numeric aggregates sampled over intervals (e.g., CPU usage, latency), not discrete, timestamped events.
Observability guidance in cloud native systems emphasizes the "three pillars" of telemetry: logs, metrics, and traces. Logs are indispensable for root cause analysis and compliance because they preserve historical event context.
References:- CNCF Observability Whitepaper- OpenTelemetry Documentation (aligned with CNCF)- Cloud Native Platform Engineering Study Guide

질문 # 28
How can an internal platform team effectively support data scientists in leveraging complex AI/ML tools and infrastructure?

- A. Implement strict resource quotas and isolation for AI/ML workloads for stability.
- B. Offer workflows and easy access to specialized AI/ML tools, data, and compute.
- C. Integrate AI/ML steps into standard developer CI/CD systems for maximum reuse
- D. Focus the portal on UI-driven execution of predefined AI/ML jobs via abstraction.

정답：B

설명:
The best way for platform teams to support data scientists is by enabling easy access to specialized AI/ML workflows, tools, and compute resources. Option C is correct because it empowers data scientists to experiment, train, and deploy models without worrying about the complexities of infrastructure setup. This aligns with platform engineering's principle of self-service with guardrails.
Option A (integrating into standard CI/CD) may help, but AI/ML workflows often require specialized tools like MLflow, Kubeflow, or TensorFlow pipelines. Option B (strict quotas) ensures stability but does not improve usability or productivity. Option D (UI-driven execution only) restricts flexibility and reduces the ability of data scientists to adapt workflows to evolving needs.
By offering AI/ML-specific workflows as golden paths within an Internal Developer Platform (IDP), platform teams improve developer experience for data scientists, accelerate innovation, and ensure compliance and governance.
References:- CNCF Platforms Whitepaper- CNCF Platform Engineering Maturity Model- Cloud Native Platform Engineering Study Guide

질문 # 29
.....