NVIDIA NCP-AAI Practice Questions

Shared by Schwartz on 21-08-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

What benefits does a Kubernetes deployment offer over Slurm?

Options:

A- Kubernetes provides autoscaling, auto-restarts, dynamic task scheduling, error isolation with containers, and integrated monitoring.
B- Kubernetes is the best option for both training and inference, offering advantages for resource management and workload visibility over traditional HPC schedulers like Slurm.
C- Kubernetes is more optimized for batch jobs to achieve high throughput, and also provides for monitoring and failover in large-scale workloads.

Answer:

A

Explanation:

The selected design maps to Kubernetes provides autoscaling auto-restarts dynamic task scheduling error isolation with containers and integrated monitoring, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The deployment logic aligns with NVIDIA NIM for containerized inference, TensorRT-LLM for optimized engines, and Triton for batching, scheduling, and Prometheus-visible inference metrics. Performance comes from matching workload shape to serving topology: small requests, large reasoning calls, embeddings, rerankers, and multimodal models should scale on separate resource signals. GPU utilization, queue depth, dynamic batching, model precision, and container lifecycle are therefore first-class design variables, not after-the-fact tuning knobs. The distractors are weaker because they lean on B: Kubernetes is the best option for both training and inference offering advantages...; C: Kubernetes is more optimized for batch jobs to achieve high throughput and..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 2

Question Type: MultipleChoice

A company is deploying a multi-agent AI system to handle large-scale customer interactions. They want to ensure the system is highly available, cost-effective, and scalable across multiple NVIDIA GPUs using container orchestration tools.

Which practice is most crucial for successfully deploying and scaling an agentic AI system in production?

Options:

A- Use a static assignment of requests across agents to maintain consistent agent operation and simplify coordination while scaling infrastructure resources as needed.
B- Optimize GPU utilization frameworks with workload optimization separate from cost analysis, prioritizing resource performance for peak load scenarios in deployment.
C- Deploy agents on a single machine to obtain a dimensioning baseline and thereby reduce setup complexity before expanding system scope.
D- Implementing automated workload management and resource scheduling frameworks to optimize GPU utilization and maintain service availability.

Answer:

D

Explanation:

The selected design maps to Implementing automated workload management and resource scheduling frameworks to optimize GPU utilization and maintain service availability, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The deployment logic aligns with NVIDIA NIM for containerized inference, TensorRT-LLM for optimized engines, and Triton for batching, scheduling, and Prometheus-visible inference metrics. Performance comes from matching workload shape to serving topology: small requests, large reasoning calls, embeddings, rerankers, and multimodal models should scale on separate resource signals. GPU utilization, queue depth, dynamic batching, model precision, and container lifecycle are therefore first-class design variables, not after-the-fact tuning knobs. The distractors are weaker because they lean on A: Use a static assignment of requests across agents to maintain consistent agent...; B: Optimize GPU utilization frameworks with workload optimization separate from cost analysis prioritizing...; C: Deploy agents on a single machine to obtain a dimensioning baseline and..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 3

Question Type: MultipleChoice

You are developing an agent that needs to perform a complex set of tasks repeatedly.

Why is periodic fine-tuning an important aspect of long-term knowledge retention for this type of agent?

Options:

A- It prevents the agent from becoming overly specialized to a single task.
B- It eliminates the need for external storage like RAG.
C- It prevents the agent from forgetting past successes and failures.
D- It guarantees the agent will produce the same output for the same input.

Answer:

C

Explanation:

The selected design maps to It prevents the agent from forgetting past successes and failures, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. For knowledge-grounded agents, the clean architecture is a RAG path with retrievers and vector indexes externalized from the LLM, then evaluated for retrieval quality and answer faithfulness. Agentic systems need explicit decomposition: a planner or coordinator defines the work, specialized agents or tools execute bounded actions, and memory/state is preserved only where it improves the next decision. That structure increases maintainability because each agent role, message contract, and state transition can be tested independently under load. The distractors are weaker because they lean on A: It prevents the agent from becoming overly specialized to a single task; B: It eliminates the need for external storage like RAG; D: It guarantees the agent will produce the same output for the same..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 4

Question Type: MultipleChoice

In designing an AI workflow which of the following best describes a comprehensive approach to improving the performance of AI agents?

Options:

A- Implementing benchmarking pipelines, deploying physical agents and monitoring user engagement metrics
B- Implementing benchmarking pipelines, collecting user feedback, and tuning model parameters iteratively
C- Implementing benchmarking pipelines and incorporating a dynamic dataset for a real-time fall-back
D- Monitoring agents' throughput and time-to-first-token from the scoring engine

Answer:

B

Explanation:

The selected design maps to Implementing benchmarking pipelines collecting user feedback and tuning model parameters iteratively, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. For optimization, NeMo Agent Toolkit profiling and evaluation expose workflow timing, token flow, tool latency, and quality metrics that single-output grading cannot capture. The evaluation target is the full agent workflow: planning quality, tool selection, intermediate state, latency, retries, user feedback, and final task completion. Instrumentation must expose where degradation starts so remediation can focus on prompts, tool schemas, retrieval, model parameters, or infrastructure rather than random retuning. The distractors are weaker because they lean on A: Implementing benchmarking pipelines deploying physical agents and monitoring user engagement metrics; C: Implementing benchmarking pipelines and incorporating a dynamic dataset for a real-time fall-back; D: Monitoring agents throughput and time-to-first-token from the scoring engine, which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 5

Question Type: MultipleChoice

You are deploying an AI-driven applicant-screening agent that analyzes candidate resumes and social-media data to recommend top applicants. Due to anti-discrimination laws and corporate policy, the system must mitigate bias against protected groups, maintain an audit trail of decisions, and comply with GDPR (including data minimization and explicit consent).

Which of the following strategies is most effective for ensuring your screening agent both mitigates bias in its recommendations and complies with data-privacy regulations?

Options:

A- Perform a post-deployment GDPR and bias audit and process raw personal data as received.
B- Pseudonymize protected attributes, implement fairness-aware debiasing, maintain an audit trail, and enforce GDPR data-minimization and consent.
C- Encrypt all candidate data at rest and in transit, remove protected attributes from analysis, and conduct manual bias checks on recommendations.
D- Exclude gender and ethnicity fields during training, use a generic privacy policy for consent, and do not maintain audit logs or apply targeted debiasing.

Answer:

B

Explanation:

The selected design maps to Pseudonymize protected attributes implement fairness-aware debiasing maintain an audit trail and enforce GDPR data-minimization and consent, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The NVIDIA stack component that anchors this design is NeMo Guardrails, because rails can be placed before retrieval, during dialog, around tool execution, and after generation. The system must constrain behavior at runtime, preserve reviewability, and make human accountability explicit when outputs affect regulated, safety-critical, or rights-sensitive decisions. Guardrails, audit trails, provenance, and intervention controls are stronger than relying on vague ethical prompts or undisclosed autonomous decisions. The distractors are weaker because they lean on A: Perform a post-deployment GDPR and bias audit and process raw personal data...; C: Encrypt all candidate data at rest and in transit remove protected attributes...; D: Exclude gender and ethnicity fields during training use a generic privacy policy..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 6

Question Type: MultipleChoice

When evaluating an agent's integration with external tools and APIs for data retrieval and action execution, which analysis approaches effectively identify reliability and performance issues? (Choose two.)

Options:

A- Implement comprehensive API call tracing with latency measurement, success rates per endpoint, and correlation analysis between tool failures and task completion.
B- Use static API endpoints and parameters configured during development, allowing consistent and effective agent integration across predictable workflows.
C- Connect to external APIs with standard procedures and monitor request and response exchanges to isolate the analysis of integration reliability and effectiveness.
D- Design integration tests simulating API version changes, schema modifications, and backward compatibility scenarios to ensure reliable tool connections across updates.

Answer:

A, D

Explanation:

The selected design maps to Implement comprehensive API call tracing with latency measurement success rates per endpoint and correlation analysis between tool failures... and Design integration tests simulating API version changes schema modifications and backward compatibility scenarios to ensure reliable tool connections..., which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The deployment logic aligns with NVIDIA NIM for containerized inference, TensorRT-LLM for optimized engines, and Triton for batching, scheduling, and Prometheus-visible inference metrics. The evaluation target is the full agent workflow: planning quality, tool selection, intermediate state, latency, retries, user feedback, and final task completion. Instrumentation must expose where degradation starts so remediation can focus on prompts, tool schemas, retrieval, model parameters, or infrastructure rather than random retuning. The distractors are weaker because they lean on B: Use static API endpoints and parameters configured during development allowing consistent and...; C: Connect to external APIs with standard procedures and monitor request and response..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 7

Question Type: MultipleChoice

Integrate NeMo Guardrails, configure NIM microservices for optimized inference, use TensorRT-LLM for deployment, and profile the system using Triton Inference Server with multi-modal support.

Which of the following strategies aligns with best practices for operationalizing and scaling such

Agentic systems?

Options:

A- Use Docker containers orchestrated by Kubernetes, implement MLOps pipelines for CI/CD, monitor agent health with Prometheus/Grafana.
B- Deploy agents on bare-metal servers to maximize performance and avoid container overhead, using manual scripts for orchestration and monitoring.
C- Deploy all agents on a single high-performance GPU node to reduce latency, and use cron jobs for periodic health checks and updates.
D- Run agents as independent serverless functions to minimize infrastructure management, relying primarily on cloud provider auto-scaling and logging tools.

Answer:

A

Explanation:

The selected design maps to Use Docker containers orchestrated by Kubernetes implement MLOps pipelines for CI/CD monitor agent health with Prometheus/Grafana, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The NVIDIA stack component that anchors this design is NeMo Guardrails, because rails can be placed before retrieval, during dialog, around tool execution, and after generation. Performance comes from matching workload shape to serving topology: small requests, large reasoning calls, embeddings, rerankers, and multimodal models should scale on separate resource signals. GPU utilization, queue depth, dynamic batching, model precision, and container lifecycle are therefore first-class design variables, not after-the-fact tuning knobs. The distractors are weaker because they lean on B: Deploy agents on bare-metal servers to maximize performance and avoid container overhead...; C: Deploy all agents on a single high-performance GPU node to reduce latency...; D: Run agents as independent serverless functions to minimize infrastructure management relying primarily..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 8

Question Type: MultipleChoice

A company operates agent-based workloads in multiple data centers. They want to minimize latency for users in different regions, maintain continuous service during infrastructure upgrades, and keep operational costs predictable.

Which deployment practice best supports low-latency, resilient, and cost-efficient agent operations at scale?

Options:

A- Schedule regular agent downtime for system updates and operational recalibration.
B- Implement geo-distributed deployments with rolling updates and resource usage monitoring.
C- Prioritize high-performance GPUs for all agents in geo-distributed deployments.
D- Apply static infrastructure allocation with centralized resource usage monitoring at a single data center.

Answer:

B

Explanation:

The selected design maps to Implement geo-distributed deployments with rolling updates and resource usage monitoring, which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The deployment logic aligns with NVIDIA NIM for containerized inference, TensorRT-LLM for optimized engines, and Triton for batching, scheduling, and Prometheus-visible inference metrics. Performance comes from matching workload shape to serving topology: small requests, large reasoning calls, embeddings, rerankers, and multimodal models should scale on separate resource signals. GPU utilization, queue depth, dynamic batching, model precision, and container lifecycle are therefore first-class design variables, not after-the-fact tuning knobs. The distractors are weaker because they lean on A: Schedule regular agent downtime for system updates and operational recalibration; C: Prioritize high-performance GPUs for all agents in geo-distributed deployments; D: Apply static infrastructure allocation with centralized resource usage monitoring at a single..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.

Question 9

Question Type: MultipleChoice

After a series of adjustments in a supply chain agentic system, the agent has dramatically reduced shipping times and minimized costs, but the team is receiving a high volume of complaints from customers regarding delayed deliveries.

Which metric is MOST important to prioritize when investigating this situation?

Options:

A- The agent's ability to predict future demand fluctuations, as accurate forecasting is crucial for effective logistics.
B- The total cost savings achieved through the agent's optimization, which represents a significant financial benefit.
C- The percentage of delivery times that fall within the acceptable delay window, considering customer satisfaction as a key factor.
D- The agent's adherence to the prescribed delivery schedules, as it's demonstrably improving efficiency.

Answer:

C

Explanation:

The selected design maps to The percentage of delivery times that fall within the acceptable delay window considering customer satisfaction as a key..., which is the highest-control path for this scenario rather than a prompt-only or single-service shortcut. The deployment logic aligns with NVIDIA NIM for containerized inference, TensorRT-LLM for optimized engines, and Triton for batching, scheduling, and Prometheus-visible inference metrics. The evaluation target is the full agent workflow: planning quality, tool selection, intermediate state, latency, retries, user feedback, and final task completion. Instrumentation must expose where degradation starts so remediation can focus on prompts, tool schemas, retrieval, model parameters, or infrastructure rather than random retuning. The distractors are weaker because they lean on A: The agent s ability to predict future demand fluctuations as accurate forecasting...; B: The total cost savings achieved through the agent s optimization which represents...; D: The agent s adherence to the prescribed delivery schedules as it s..., which compromises traceability, resilience, scalability, or policy enforcement in production. The answer therefore fits NVIDIA's production-agent pattern: modular workflow design, measurable runtime behavior, GPU-aware serving where applicable, and controlled integration with enterprise systems.