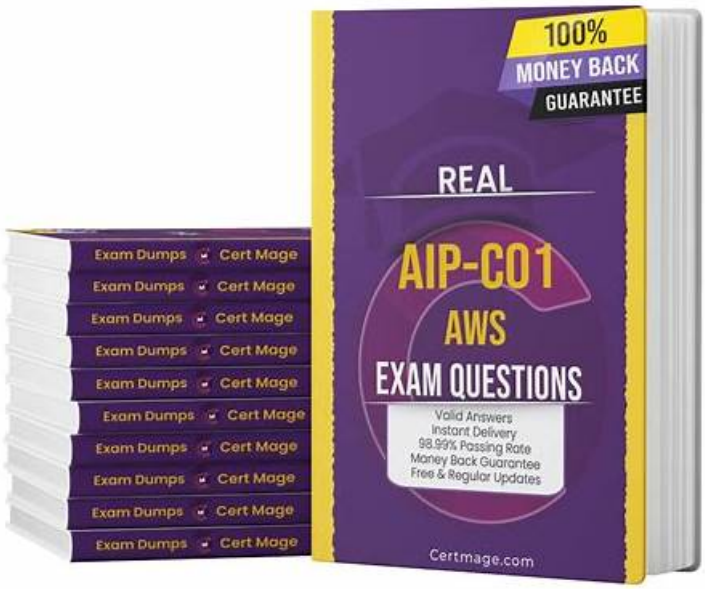
Exam AIP-C01 Quick Prep, AIP-C01 Valid Real Exam



To be successful in your social life and own a high social status you must own good abilities in some area and plenty of knowledge. Passing the test AIP-C01 exam can make you achieve those goals and prove that you are competent. Buying our AIP-C01 practice test can help you pass the exam fluently and the learning costs you little time and energy. The questions and answers of our AIP-C01 Test Question are chosen elaborately and to simplify the important information to make your learning relaxing and efficient.

Amazon AIP-C01 Exam Syllabus Topics:

Topic	Details
Topic 1	• Operational Efficiency and Optimization for GenAI Applications: This domain encompasses cost optimization strategies, performance tuning for latency and throughput, and implementing comprehensive monitoring systems for GenAI applications.
Topic 2	• Implementation and Integration: This domain focuses on building agentic AI systems, deploying foundation models, integrating GenAI with enterprise systems, implementing FM APIs, and developing applications using AWS tools.
Topic 3	• AI Safety, Security, and Governance: This domain addresses input • output safety controls, data security and privacy protections, compliance mechanisms, and responsible AI principles including transparency and fairness.
Topic 4	• Foundation Model Integration, Data Management, and Compliance: This domain covers designing GenAI architectures, selecting and configuring foundation models, building data pipelines and vector stores, implementing retrieval mechanisms, and establishing prompt engineering governance.
Topic 5	• Testing, Validation, and Troubleshooting: This domain covers evaluating foundation model outputs, implementing quality assurance processes, and troubleshooting GenAI-specific issues including prompts, integrations, and retrieval systems.

>> Exam AIP-C01 Quick Prep <<

Trusting Authorized Exam AIP-C01 Quick Prep Is The Eastest Way to Pass AWS Certified Generative AI Developer - Professional

If you have interests with our AIP-C01 practice materials, we prefer to tell that we have contacted with many former buyers of our AIP-C01 exam questions and they all talked about the importance of effective AIP-C01 practice material playing a crucial role in your preparation process. Our AIP-C01 practice materials keep exam candidates motivated and efficient with useful content based wholly on the real AIP-C01 guide materials. There are totally three versions of AIP-C01 practice materials which are the most suitable versions for you: pdf, software and app versions.

Amazon AWS Certified Generative AI Developer - Professional Sample Questions (Q82-Q87):

NEW QUESTION # 82
A company is building a legal research AI assistant that uses Amazon Bedrock with an Anthropic Claude foundation model (FM). The AI assistant must retrieve highly relevant case law documents to augment the FM's responses. The AI assistant must identify semantic relationships between legal concepts, specific legal terminology, and citations. The AI assistant must perform quickly and return precise results.
Which solution will meet these requirements?

- A. Use Amazon OpenSearch Service to deploy a hybrid search architecture that combines vector search with keyword search. Apply an Amazon Bedrock reranker model to optimize result relevance.
- B. Configure an Amazon Bedrock knowledge base to use a default vector search configuration. Use Amazon Bedrock to expand queries to improve retrieval for legal documents based on specific terminology and citations.
- C. Use Amazon OpenSearch Service with vector search and Amazon Bedrock Titan Embeddings to index and search legal documents. Use custom AWS Lambda functions to merge results with keyword-based filters that are stored in an Amazon RDS database.
- D. Enable the Amazon Kendra query suggestion feature for end users. Use Amazon Bedrock to perform post-processing of search results to identify semantic similarity in the documents and to produce precise results.

Answer: A

Explanation:
Option B is the correct solution because legal research workloads require both semantic understanding and exact lexical precision, especially for statutes, citations, and domain-specific terminology. A hybrid search architecture directly addresses this need by combining vector similarity search with traditional keyword-based retrieval.
Vector search alone is often insufficient for legal research because exact phrases, citation formats, and jurisdiction-specific terms must be matched precisely. Keyword search ensures high recall and precision for citations and legal terms, while vector search captures deeper semantic relationships between legal concepts, precedents, and arguments. Amazon OpenSearch Service natively supports hybrid search, enabling efficient scoring and ranking without external orchestration.
Applying an Amazon Bedrock reranker model further improves relevance by reordering retrieved documents based on deeper contextual understanding. Reranking is especially valuable in legal research because multiple documents may appear relevant, but only a subset truly addresses the user's legal question. The reranker optimizes final results before they are passed to the Anthropic Claude FM, improving answer accuracy and reducing hallucinations.
Option A relies on default vector search, which does not reliably handle citations and exact terminology.
Option C focuses on query suggestions and post-processing rather than retrieval quality. Option D introduces unnecessary operational complexity by merging results across multiple systems.
Therefore, Option B best meets the requirements for precision, performance, and semantic understanding in a legal research AI assistant.

NEW QUESTION # 83
A healthcare company is using Amazon Bedrock to build a Retrieval Augmented Generation (RAG) application that helps practitioners make clinical decisions. The application must achieve high accuracy for patient information retrievals, identify hallucinations in generated content, and reduce human review costs.
Which solution will meet these requirements?

- A. Implement automated large language model (LLM)-based evaluations that use a specialized model that is fine-tuned for

medical content to assess all responses. Deploy AWS Lambda functions to parallelize evaluations. Publish results to Amazon CloudWatch metrics that track relevance and factual accuracy.

- B. Deploy a hybrid evaluation system that uses an automated LLM-as-a-judge evaluation to initially screen responses and targeted human reviews for edge cases. Use a built-in Amazon Bedrock evaluation to track retrieval precision and hallucination rates.
- C. Use Amazon Comprehend to analyze and classify RAG responses and to extract medical entities and relationships. Use AWS Step Functions to orchestrate automated evaluations. Configure Amazon CloudWatch metrics to track entity recognition confidence scores. Configure CloudWatch to send an alert when accuracy falls below specified thresholds.
- D. Configure Amazon CloudWatch Synthetics to generate test queries that have known answers on a regular schedule, and track model success rates. Set up dashboards that compare synthetic test results against expected outcomes.

Answer: B

Explanation:
Option D is the correct solution because it directly addresses all three requirements: high retrieval accuracy, hallucination detection, and reduced human review costs. AWS recommends a layered evaluation strategy for high-stakes domains such as healthcare, where generative outputs must be both accurate and safe.
Using an automated LLM-as-a-judge evaluation enables scalable, consistent assessment of generated responses for factual grounding, relevance, and hallucination risk. This automated screening significantly reduces the number of responses that require manual inspection. Only responses that fall below defined quality thresholds or exhibit ambiguous behavior are escalated to targeted human reviews, which optimizes review effort and cost.
The use of Amazon Bedrock built-in evaluations provides standardized metrics specifically designed for RAG systems, including retrieval precision, faithfulness to source documents, and hallucination rates. These evaluations integrate directly with Amazon Bedrock knowledge bases and models, eliminating the need to build and maintain custom evaluation pipelines.
Option A focuses on entity extraction confidence, which does not reliably detect hallucinations in generative text. Option B requires maintaining and scaling a separate fine-tuned evaluation model, increasing complexity and cost. Option C is useful for regression testing but cannot detect hallucinations in real-world, open-ended clinical queries.
Therefore, Option D provides the most effective and operationally efficient approach to maintaining clinical- grade accuracy while minimizing human review effort.

NEW QUESTION # 84
A company is using AWS Lambda and REST APIs to build a reasoning agent to automate support workflows.
The system must preserve memory across interactions, share relevant agent state, and support event-driven invocation and synchronous invocation. The system must also enforce access control and session-based permissions.
Which combination of steps provides the MOST scalable solution? (Select TWO.)

- A. Deploy the reasoning logic as a container on Amazon ECS behind API Gateway. Use Amazon Aurora to store memory and identity data.
- B. Use Amazon Bedrock Agents for reasoning and conversation management. Use AWS Step Functions and Amazon SQS for orchestration. Store agent state in Amazon DynamoDB.
- C. Register the Lambda functions and REST APIs as actions by using Amazon API Gateway and Amazon EventBridge. Enable Amazon Bedrock AgentCore to invoke the Lambda functions and REST APIs without custom orchestration code.
- D. Use Amazon Bedrock AgentCore to manage memory and session-aware reasoning. Deploy the agent with built-in identity support, event handling, and observability.
- E. Build a custom RAG pipeline by using Amazon Kendra and Amazon Bedrock. Use AWS Lambda to orchestrate tool invocations. Store agent state in Amazon S3.

Answer: C,D

Explanation:
The combination of Options A and B provides the most scalable and AWS-native architecture for building reasoning agents with persistent memory, session awareness, secure access control, and flexible invocation models.
Amazon Bedrock AgentCore is purpose-built to manage agent memory, session context, and identity-aware reasoning across interactions. It eliminates the need for developers to manually store and retrieve agent state, manage session lifecycles, or implement custom memory layers. AgentCore natively supports both synchronous requests and event-driven execution, making it ideal for support workflow automation.
Option B complements AgentCore by enabling seamless tool invocation. By registering AWS Lambda functions and REST APIs as agent actions through API Gateway and EventBridge, the agent can invoke tools reactively or synchronously without custom orchestration code. EventBridge enables event-driven execution, while API Gateway supports synchronous request-response patterns.

This combination provides built-in security, observability, and scaling, while avoiding the operational burden of managing queues, databases, or custom workflow engines.
Option C introduces unnecessary orchestration complexity. Option D increases infrastructure management and cost. Option E stores agent state in S3, which is not suitable for low-latency, session-based reasoning.
Therefore, A and B together deliver the most scalable, secure, and low-overhead solution for production- grade reasoning agents on AWS.

NEW QUESTION # 85
A company is creating a workflow to review customer-facing communications before the company sends the communications. The company uses a pre-defined message template to generate the communications and stores the communications in an Amazon S3 bucket. The workflow needs to capture a specific portion from the template and send it to an Amazon Bedrock model. The workflow must store model responses back to the original S3 bucket.
Which solution will meet these requirements?

- A. Create an Amazon Bedrock agent that has an action group. Configure instructions to define how the agent should parse the communications. Configure the action group to retrieve the communications from the S3 bucket, invoke the Amazon Bedrock model, and store the model responses back to the S3 bucket.
- B. Create an AWS Step Functions Express workflow state machine. Use an Amazon S3 integration GetObject step to retrieve the original communications. Use an intrinsic function Pass step to parse the communications and to pass the results to an Amazon Bedrock InvokeModel step. Configure an Amazon S3 integration PutObject step to store the model responses back to the S3 bucket.
- C. Create an Amazon Bedrock agent that has a single action group. Configure three AWS Lambda functions in the action group. Configure the functions to retrieve the communications from the S3 bucket, parse the communications and invoke the Amazon Bedrock model, and store the model responses back to the S3 bucket.
- D. Create a flow in Amazon Bedrock Flows. Configure S3 action nodes at the beginning and end of the flow to retrieve and store the communications and the model responses. In the middle of the flow, configure an expression to parse each communication. Configure an agent step to send the parsed input to the model for review.

Answer: D

Explanation:
Option A is the correct answer because Amazon Bedrock Flows is purpose-built to orchestrate generative AI workflows that combine data access, deterministic transformations, and model invocation with minimal operational overhead. The requirements explicitly state that the workflow must retrieve content from Amazon S3, extract a specific portion of a predefined template, send that portion to an Amazon Bedrock model, and store the model's response back into the same S3 bucket. Amazon Bedrock Flows natively supports all of these steps.
By configuring S3 action nodes at the beginning and end of the flow, the workflow can retrieve the original communications and persist the reviewed output without custom code. The expression step allows deterministic parsing of a specific portion of the template, which is essential when only part of the message should be reviewed. This avoids relying on generative logic for parsing, which would be less predictable and harder to audit. The agent step is then used specifically for the review task, where the foundation model evaluates or modifies the extracted content.
Option B uses AWS Step Functions, which can achieve similar outcomes but requires more explicit orchestration logic and does not provide GenAI-native constructs such as expressions and agent steps in a single managed experience. Options C and D rely on Amazon Bedrock agents and AWS Lambda functions to handle parsing and data movement, which increases complexity, operational overhead, and maintenance burden.
Because Amazon Bedrock Flows directly integrates S3 actions, parsing expressions, and model review steps in a single managed workflow, Option A best meets the requirements with the least development and operational effort.

NEW QUESTION # 86
A financial services company wants to use Amazon Bedrock foundation models (FMs) to analyze call center recordings. When calls end, the call center stores recordings as MP3 files in an Amazon S3 bucket. The company needs to generate summaries and sentiment analysis for the recordings in a structured format as soon as new files are created. The recordings average 20 MB in size.
Which combination of solutions will meet these requirements? (Select TWO.)

- A. Configure the source S3 bucket to send notifications to the Step Functions workflow when an object is created in the bucket.
- B. Configure the source S3 bucket to send events to Amazon EventBridge. Create an EventBridge rule to invoke the Step Functions workflow when an object is created in the bucket.

- C. Use AWS Step Functions to orchestrate a workflow to process the recordings. Configure steps to invoke Amazon Transcribe to convert audio to text, validate job completion, and to invoke an AWS Lambda function to process the text by using Amazon Bedrock FMs to generate structured analysis output.
- D. Use AWS Step Functions to orchestrate a workflow to process the recordings. Configure steps to invoke Amazon Transcribe to convert audio to text, validate job completion, and to directly invoke Amazon Bedrock FMs to generate summaries and sentiment analysis in JSON format.
- E. Use AWS Step Functions to orchestrate a workflow to process the recordings. Configure steps to invoke Amazon Transcribe to convert audio to text, validate job completion, and to invoke an AWS Lambda function to create a prompt to invoke Amazon Bedrock FMs to generate structured analysis output.

Answer: B,D

Explanation:
AWS Step Functions provides native service integrations that minimize code and operational overhead. For this workflow, Step Functions can directly invoke Amazon Transcribe and, upon completion, directly invoke Amazon Bedrock foundation models. Modern foundation models can be prompted to return outputs in JSON format , fulfilling the requirement for structured analysis without needing an intermediate Lambda function to format the request or response. To trigger the process " as soon as new files are created, " configuring Amazon S3 to send events to Amazon EventBridge is the recommended event-driven pattern. This allows for fine-grained routing and decouples the storage layer from the processing logic, ensuring the workflow scales reliably as call volume grows.

NEW QUESTION # 87
.....