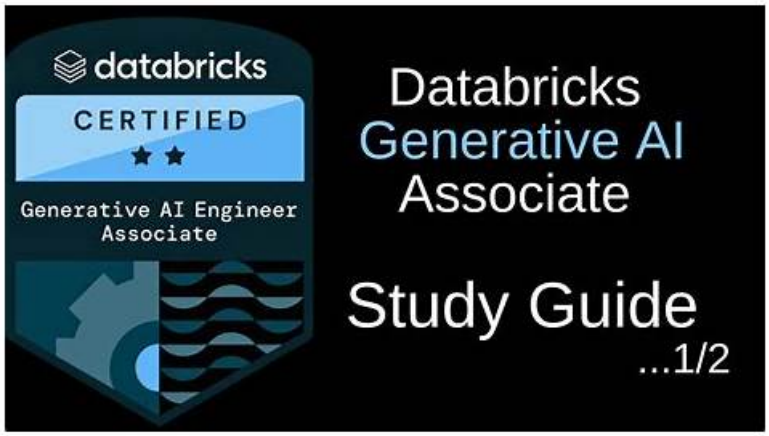
Get High-quality Databricks-Generative-AI-Engineer-Associate Valid Dumps Sheet and High Pass-Rate Prep Databricks-Generative-AI-Engineer-Associate Guide



Our Databricks-Generative-AI-Engineer-Associate preparation exam have assembled a team of professional experts incorporating domestic and overseas experts and scholars to research and design related exam bank, committing great efforts to work for our candidates. Most of the experts have been studying in the professional field for many years and have accumulated much experience in our Databricks-Generative-AI-Engineer-Associate Practice Questions. So we can say that our Databricks-Generative-AI-Engineer-Associate exam questions are the first-class in the market. With our Databricks-Generative-AI-Engineer-Associate learning guide, you will get your certification by your first attempt.

Databricks Databricks-Generative-AI-Engineer-Associate Exam Syllabus Topics:

Topic	Details
Topic 1	• Design Applications: The topic focuses on designing a prompt that elicits a specifically formatted response. It also focuses on selecting model tasks to accomplish a given business requirement. Lastly, the topic covers chain components for a desired model input and output.
Topic 2	• Assembling and Deploying Applications: In this topic, Generative AI Engineers get knowledge about coding a chain using a pyfunc mode, coding a simple chain using langchain, and coding a simple chain according to requirements. Additionally, the topic focuses on basic elements needed to create a RAG application. Lastly, the topic addresses sub-topics about registering the model to Unity Catalog using MLflow.
Topic 3	• Evaluation and Monitoring: This topic is all about selecting an LLM choice and key metrics. Moreover, Generative AI Engineers learn about evaluating model performance. Lastly, the topic includes sub-topics about inference logging and usage of Databricks features.

>> Databricks-Generative-AI-Engineer-Associate Valid Dumps Sheet <<

Prep Databricks-Generative-AI-Engineer-Associate Guide - Databricks-Generative-AI-Engineer-Associate Online Tests

Now you can trust ActualCollection Databricks-Generative-AI-Engineer-Associate exam questions as these Databricks Certified Generative AI Engineer Associate (Databricks-Generative-AI-Engineer-Associate) exam questions have already helped countless candidates in their Databricks-Generative-AI-Engineer-Associate exam preparation. They easily got success in their challenging and dream Databricks Databricks-Generative-AI-Engineer-Associate Certification Exam. Now they have become certified Databricks

professionals and offer their services to top world brands.

Databricks Certified Generative AI Engineer Associate Sample Questions (Q37-Q42):

NEW QUESTION # 37
A Generative AI Engineer is testing a simple prompt template in LangChain using the code below, but is getting an error.
Assuming the API key was properly defined, what change does the Generative AI Engineer need to make to fix their chain?

- A.
- B.
- C.
- D.

Answer: A

Explanation:
To fix the error in the LangChain code provided for using a simple prompt template, the correct approach is Option C. Here's a detailed breakdown of why Option C is the right choice and how it addresses the issue:
* Proper Initialization: In Option C, the LLMChain is correctly initialized with the LLM instance specified as OpenAI(), which likely represents a language model (like GPT) from OpenAI. This is crucial as it specifies which model to use for generating responses.
* Correct Use of Classes and Methods:
* The PromptTemplate is defined with the correct format, specifying that adjective is a variable within the template. This allows dynamic insertion of values into the template when generating text.
* The prompt variable is properly linked with the PromptTemplate, and the final template string is passed correctly.
* The LLMChain correctly references the prompt and the initialized OpenAI() instance, ensuring that the template and the model are properly linked for generating output.
Why Other Options Are Incorrect:
* Option A: Misuses the parameter passing in generate method by incorrectly structuring the dictionary.
* Option B: Incorrectly uses prompt.format method which does not exist in the context of LLMChain and PromptTemplate configuration, resulting in potential errors.
* Option D: Incorrect order and setup in the initialization parameters for LLMChain, which would likely lead to a failure in recognizing the correct configuration for prompt and LLM usage.
Thus, Option C is correct because it ensures that the LangChain components are correctly set up and integrated, adhering to proper syntax and logical flow required by LangChain's architecture. This setup avoids common pitfalls such as type errors or method misuses, which are evident in other options.

NEW QUESTION # 38
A Generative Al Engineer would like an LLM to generate formatted JSON from emails. This will require parsing and extracting the following information: order ID, date, and sender email. Here's a sample email:
They will need to write a prompt that will extract the relevant information in JSON format with the highest level of output accuracy.
Which prompt will do that?

- A. You will receive customer emails and need to extract date, sender email, and order ID. Return the extracted information in JSON format.
  Here's an example: {"date": "April 16, 2024", "sender_email": "sarah.lee925@gmail.com", "order_id": "RE987D"}
- B. You will receive customer emails and need to extract date, sender email, and order ID. Return the extracted information in JSON format.
- C. You will receive customer emails and need to extract date, sender email, and order ID. Return the extracted information in a human-readable format.
- D. You will receive customer emails and need to extract date, sender email, and order ID. You should return the date, sender email, and order ID information in JSON format.

Answer: A

Explanation:
Problem Context: The goal is to parse emails to extract certain pieces of information and output this in a structured JSON format. Clarity and specificity in the prompt design will ensure higher accuracy in the LLM' s responses.

Explanation of Options:
* Option A: Provides a general guideline but lacks an example, which helps an LLM understand the exact format expected.
* Option B: Includes a clear instruction and a specific example of the output format. Providing an example is crucial as it helps set the pattern and format in which the information should be structured, leading to more accurate results.
* Option C: Does not specify that the output should be in JSON format, thus not meeting the requirement.
* Option D: While it correctly asks for JSON format, it lacks an example that would guide the LLM on how to structure the JSON correctly.
Therefore,Option Bis optimal as it not only specifies the required format but also illustrates it with an example, enhancing the likelihood of accurate extraction and formatting by the LLM.

NEW QUESTION # 39
All of the following are Python APIs used to query Databricks foundation models. When running in an interactive notebook, which of the following libraries does not automatically use the current session credentials?

- A. Databricks Python SDK
- B. MLflow Deployments SDK
- C. OpenAI client
- D. REST API via requests library

Answer: D

Explanation:
When working within a Databricks notebook, several high-level SDKs are "Databricks-aware." The MLflow Deployments SDK (C) and the Databricks Python SDK (D) are designed to automatically look for the DATABRICKS_HOST and DATABRICKS_TOKEN environment variables provided by the notebook context. The OpenAI client (A), when configured for Databricks via Mosaic AI Gateway, also typically handles authentication via workspace integration in recent versions. However, the REST API via the requests library (B) is a generic Python HTTP client. It has no intrinsic knowledge of the Databricks environment. To use it, an engineer must manually extract the token (e.g., via dbutils.notebook.entry_point...) and explicitly pass it in the Authorization: Bearer <token> header of the request. Without this manual step, the requests library will fail with a 401 Unauthorized error.

NEW QUESTION # 40
A Generative Al Engineer has created a RAG application to look up answers to questions about a series of fantasy novels that are being asked on the author's web forum. The fantasy novel texts are chunked and embedded into a vector store with metadata (page number, chapter number, book title), retrieved with the user's query, and provided to an LLM for response generation. The Generative AI Engineer used their intuition to pick the chunking strategy and associated configurations but now wants to more methodically choose the best values.
Which TWO strategies should the Generative AI Engineer take to optimize their chunking strategy and parameters? (Choose two.)

- A. Choose an appropriate evaluation metric (such as recall or NDCG) and experiment with changes in the chunking strategy, such as splitting chunks by paragraphs or chapters.
  Choose the strategy that gives the best performance metric.
- B. Pass known questions and best answers to an LLM and instruct the LLM to provide the best token count. Use a summary statistic (mean, median, etc.) of the best token counts to choose chunk size.
- C. Add a classifier for user queries that predicts which book will best contain the answer. Use this to filter retrieval.
- D. Change embedding models and compare performance.
- E. Create an LLM-as-a-judge metric to evaluate how well previous questions are answered by the most appropriate chunk. Optimize the chunking parameters based upon the values of the metric.

Answer: A,E

NEW QUESTION # 41
A Generative AI Engineer at a legal firm is designing a RAG system to analyze historical legal cases. The system needs to process millions of court opinions and legal documents, already organized by time and topic, to track how interpretations of specific laws have evolved over time. All of these documents are in plain-text. The engineer needs to choose a chunking method that would most effectively preserve continuity and the temporal nature of the cases. Which method do they choose?

- A. Implement windowed summarization with overlapping chunks.
- B. Implement sentence level embeddings with each chunk tagged with the time to enable metadata filtering.
- C. Implement a hierarchical tree structure, like RAPTOR, to group similar legal concepts.
- D. Implement paragraph level embeddings with each chunk.

Answer: A

Explanation:
In the context of legal document analysis where the "evolution of interpretation" is the primary goal, preserving narrative continuity is paramount. Windowed summarization with overlapping chunks is the most effective method for this use case. Overlapping (e.g., 10-15% of the chunk size) ensures that sentences or concepts split at the boundary of one chunk are preserved in the next, preventing the loss of critical context that often occurs in legal jargon. Furthermore, windowed summarization allows the system to condense long-form court opinions into manageable parts while maintaining the chronological "thread" of the argument. While sentence-level embeddings with metadata (D) are useful for filtering, they often lack the sufficient context required to understand the nuances of a legal ruling. A windowed approach provides the LLM with enough surrounding text to understand the "why" behind a legal evolution, rather than just the "when."

NEW QUESTION # 42
.....

- Free PDF Quiz 2026 Databricks-Generative-AI-Engineer-Associate: Fantastic Databricks Certified Generative AI Engineer Associate Valid Dumps Sheet □ Search for ➤ Databricks-Generative-AI-Engineer-Associate □ and obtain a free download on □ www.examcollectionpass.com □ □New Databricks-Generative-AI-Engineer-Associate Exam Price
- Databricks-Generative-AI-Engineer-Associate Valid Exam Preparation □ Databricks-Generative-AI-Engineer-Associate Training Tools □ Databricks-Generative-AI-Engineer-Associate Learning Mode □ Open website 【 www.pdfvce.com 】 and search for □ Databricks-Generative-AI-Engineer-Associate □ for free download □Databricks-Generative-AI-Engineer-Associate Reliable Exam Question
- HOT Databricks-Generative-AI-Engineer-Associate Valid Dumps Sheet 100% Pass | High Pass-Rate Databricks Prep Databricks Certified Generative AI Engineer Associate Guide Pass for sure □ Download （ Databricks-Generative-AI-Engineer-Associate ） for free by simply searching on “ www.practicevce.com ” □Databricks-Generative-AI-Engineer-Associate Reliable Exam Question
- myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.xiaokedou20.com, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, shop.shouxishe.ltd, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, Disposable vapes