Databricks-Generative-AI-Engineer-Associate試験の準備方法｜完璧なDatabricks-Generative-AI-Engineer-Associate日本語版問題解説試験｜便利なDatabricks Certified Generative AI Engineer Associateトレーリング学習

P.S.JPNTestがGoogle Driveで共有している無料の2026 Databricks Databricks-Generative-AI-Engineer-Associateダンプ：https://drive.google.com/open?id=177oZUwpEWLwt62R9GWHM23mGyNwiRFmh

JPNTestは多くの人に便利を与えるとともに、多くの人の夢が実現させるサイトでございます。もし君はまだIT試験で心配すれば、私達JPNTestのDatabricks-Generative-AI-Engineer-Associate問題集を選んでください。JPNTestは長年の研究をわたって研ITの認証試験に関する品質が高く、範囲は広い教育資料が開発しました。それは確かに君のDatabricks-Generative-AI-Engineer-Associate試験に役に立つとみられます。

Databricks Databricks-Generative-AI-Engineer-Associate 認定試験の出題範囲：

トピック	出題範囲
トピック 1	• 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.
トピック 2	• Application Development: In this topic, Generative AI Engineers learn about tools needed to extract data, Langchain • similar tools, and assessing responses to identify common issues. Moreover, the topic includes questions about adjusting an LLM's response, LLM guardrails, and the best LLM based on the attributes of the application.
トピック 3	• Data Preparation: Generative AI Engineers covers a chunking strategy for a given document structure and model constraints. The topic also focuses on filter extraneous content in source documents. Lastly, Generative AI Engineers also learn about extracting document content from provided source data and format.
トピック 4	• 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.

>> Databricks-Generative-AI-Engineer-Associate日本語版問題解説 <<

最新なDatabricks-Generative-AI-Engineer-Associate試験問題、Databricks-Generative-AI-Engineer-Associateトレーニング資料、有効Databricks-Generative-AI-Engineer-Associate学習資料

弊社の資料はすばらしくて、DatabricksのDatabricks-Generative-AI-Engineer-Associate問題集などを含めています。これらの問題集は詳しい答えと解説があります。それに、我々は一番行き届いたアフターサービスを提供して、あなたの利益を保証します。お客様はDatabricks-Generative-AI-Engineer-Associate問題集を購入するなら、一年の更新サービスと半年の返金サービスが得られています。この期間、我々はDatabricks-Generative-AI-Engineer-Associate問題集に関するサービスを提供します。

Databricks Certified Generative AI Engineer Associate 認定 Databricks-Generative-AI-Engineer-Associate 試験問題 (Q38-Q43):

質問 # 38
A Generative Al Engineer at an automotive company would like to build a question-answering chatbot for customers to inquire about their vehicles. They have a database containing various documents of different vehicle makes, their hardware parts, and common maintenance information.
Which of the following components will NOT be useful in building such a chatbot?

- A. Invite users to submit long, rather than concise, questions
- B. Vector database
- C. Embedding model
- D. Response-generating LLM

正解：A

解説：
The task involves building a question-answering chatbot for an automotive company using a database of vehicle-related documents. The chatbot must efficiently process customer inquiries and provide accurate responses. Let's evaluate each component to determine which isnotuseful, per Databricks Generative AI Engineer principles.
* Option A: Response-generating LLM
* An LLM is essential for generating natural language responses to customer queries based on retrieved information. This is a core component of any chatbot.
* Databricks Reference:"The response-generating LLM processes retrieved context to produce coherent answers"("Building LLM Applications with Databricks," 2023).
* Option B: Invite users to submit long, rather than concise, questions
* Encouraging long questions is a user interaction design choice, not a technical component of the chatbot's architecture. Moreover, long, verbose questions can complicate intent detection and retrieval, reducing efficiency and accuracy-counter to best practices for chatbot design. Concise questions are typically preferred for clarity and performance.
* Databricks Reference: While not explicitly stated, Databricks' "Generative AI Cookbook" emphasizes efficient query processing, implying that simpler, focused inputs improve LLM performance. Inviting long questions doesn't align with this.
* Option C: Vector database
* A vector database stores embeddings of the vehicle documents, enabling fast retrieval of relevant information via semantic search. This is critical for a question-answering system with a large document corpus.
* Databricks Reference:"Vector databases enable scalable retrieval of context from large datasets"("Databricks Generative AI Engineer Guide").
* Option D: Embedding model
* An embedding model converts text (documents and queries) into vector representations for similarity search. It's a foundational component for retrieval-augmented generation (RAG) in chatbots.
* Databricks Reference:"Embedding models transform text into vectors, facilitating efficient matching of queries to documents" ("Building LLM-Powered Applications").
Conclusion: Option B is not a usefulcomponentin building the chatbot. It's a user-facing suggestion rather than a technical building block, and it could even degrade performance by introducing unnecessary complexity. Options A, C, and D are all integral to a Databricks-aligned chatbot architecture.

質問 # 39
A Generative Al Engineer interfaces with an LLM with prompt/response behavior that has been trained on customer calls inquiring about product availability. The LLM is designed to output "In Stock" if the product is available or only the term "Out of Stock" if not. Which prompt will work to allow the engineer to respond to call classification labels correctly?

- A. Respond with "In Stock" if the customer asks for a product.
- B. You will be given a customer call transcript where the customer inquires about product availability. Respond with "In Stock" if the product is available or "Out of Stock" if not.
- C. You will be given a customer call transcript where the customer asks about product availability. The outputs are either "In Stock" or "Out of Stock". Format the output in JSON, for example: {"call_id": "123", "label": "In Stock"}.
- D. Respond with "Out of Stock" if the customer asks for a product.

正解：C

解説：
* Problem Context: The Generative AI Engineer needs a prompt that will enable an LLM trained on customer call transcripts to classify and respond correctly regarding product availability. The desired response should clearly indicate whether a product is "In Stock" or "Out of Stock," and it should be formatted in a way that is structured and easy to parse programmatically, such as JSON.
* Explanation of Options:
* Option A: Respond with "In Stock" if the customer asks for a product. This prompt is too generic and does not specify how to handle the case when a product is not available, nor does it provide a structured output format.
* Option B: This option is correctly formatted and explicit. It instructs the LLM to respond based on the availability mentioned in the customer call transcript and to format the response in JSON.
This structure allows for easy integration into systems that may need to process this information automatically, such as customer service dashboards or databases.
* Option C: Respond with "Out of Stock" if the customer asks for a product. Like option A, this prompt is also insufficient as it only covers the scenario where a product is unavailable and does not provide a structured output.
* Option D: While this prompt correctly specifies how to respond based on product availability, it lacks the structured output format, making it less suitable for systems that require formatted data for further processing.
Given the requirements for clear, programmatically usable outputs,Option Bis the optimal choice because it provides precise instructions on how to respond and includes a JSON format example for structuring the output, which is ideal for automated systems or further data handling.

質問 # 40
A Generative AI Engineer is testing a simple prompt template in LangChain using the code below, but is getting an error:
Python
from langchain.chains import LLMChain
from langchain_community.llms import OpenAI
from langchain_core.prompts import PromptTemplate
prompt_template = "Tell me a {adjective} joke"
prompt = PromptTemplate(input_variables=["adjective"], template=prompt_template)
# ... (Error-prone section)
Assuming the API key was properly defined, what change does the Generative AI Engineer need to make to fix their chain?

- A. (Incorrect structure)
- B. prompt_template = "Tell me a {adjective} joke"
prompt = PromptTemplate(input_variables=["adjective"], template=prompt_template) llm = OpenAI() llm_chain = LLMChain(prompt=prompt, llm=llm) llm_chain.generate([{"adjective": "funny"}])
- C. (Incorrect structure)
- D. (Incorrect structure)

正解：B

解説：
The error in the original snippet usually stems from the improper instantiation of the LLMChain or the incorrect call to the .generate() method. In LangChain, an LLMChain requires two primary components: an LLM (the engine) and a Prompt (the template). Option C provides the correct syntax: first, the PromptTemplate is defined with the correct input_variables. Second, the OpenAI model is instantiated. Third, the LLMChain binds the model and the prompt together. Finally, the .generate() method expects a list of dictionaries, where each dictionary represents a set of inputs for the prompt variables. Options A, B, and D in the original image

contain syntax errors such as passing the variable directly into the chain initialization or missing the dictionary list format required by the standard LangChain API for batch-like generation.

質問 # 41
A Generative AI Engineer is creating an LLM-powered application that will need access to up-to-date news articles and stock prices.
The design requires the use of stock prices which are stored in Delta tables and finding the latest relevant news articles by searching the internet.
How should the Generative AI Engineer architect their LLM system?

- A. Download and store news articles and stock price information in a vector store. Use a RAG architecture to retrieve and generate at runtime.
- B. Query the Delta table for volatile stock prices and use an LLM to generate a search query to investigate potential causes of the stock volatility.
- C. Use an LLM to summarize the latest news articles and lookup stock tickers from the summaries to find stock prices.
- D. Create an agent with tools for SQL querying of Delta tables and web searching, provide retrieved values to an LLM for generation of response.

正解：D

解説：
To build an LLM-powered system that accesses up-to-date news articles and stock prices, the best approach is tocreate an agentthat has access to specific tools (option D).
* Agent with SQL and Web Search Capabilities:By using an agent-based architecture, the LLM can interact with external tools. The agent can query Delta tables (for up-to-date stock prices) via SQL and perform web searches to retrieve the latest news articles. This modular approach ensures the system can access both structured (stock prices) and unstructured (news) data sources dynamically.
* Why This Approach Works:
* SQL Queries for Stock Prices: Delta tables store stock prices, which the agent can query directly for the latest data.
* Web Search for News: For news articles, the agent can generate search queries and retrieve the most relevant and recent articles, then pass them to the LLM for processing.
* Why Other Options Are Less Suitable:
* A (Summarizing News for Stock Prices): This convoluted approach would not ensure accuracy when retrieving stock prices, which are already structured and stored in Delta tables.
* B (Stock Price Volatility Queries): While this could retrieve relevant information, it doesn't address how to obtain the most up-to-date news articles.
* C (Vector Store): Storing news articles and stock prices in a vector store might not capture the real-time nature of stock data and news updates, as it relies on pre-existing data rather than dynamic querying.
Thus, using an agent with access to both SQL for querying stock prices and web search for retrieving news articles is the best approach for ensuring up-to-date and accurate responses.

質問 # 42
A Generative Al Engineer is creating an LLM-based application. The documents for its retriever have been chunked to a maximum of 512 tokens each. The Generative Al Engineer knows that cost and latency are more important than quality for this application. They have several context length levels to choose from.
Which will fulfill their need?

- A. context length 514; smallest model is 0.44GB and embedding dimension 768
- B. context length 512: smallest model is 0.13GB and embedding dimension 384
- C. context length 2048: smallest model is 11GB and embedding dimension 2560
- D. context length 32768: smallest model is 14GB and embedding dimension 4096

正解：B

解説：
When prioritizing cost and latency over quality in a Large Language Model (LLM)-based application, it is crucial to select a configuration that minimizes both computational resources and latency while still providing reasonable performance. Here's why D is the best choice:
Context length: The context length of 512 tokens aligns with the chunk size used for the documents (maximum of 512 tokens per

chunk). This is sufficient for capturing the needed information and generating responses without unnecessary overhead.
Smallest model size: The model with a size of 0.13GB is significantly smaller than the other options. This small footprint ensures faster inference times and lower memory usage, which directly reduces both latency and cost.
Embedding dimension: While the embedding dimension of 384 is smaller than the other options, it is still adequate for tasks where cost and speed are more important than precision and depth of understanding.
This setup achieves the desired balance between cost-efficiency and reasonable performance in a latency-sensitive, cost-conscious application.

質問 # 43
.....

JPNTestは多くの人に便利を与えるとともに、多くの人の夢が実現させるサイトでございます。もし君はまだIT試験で心配すれば、私達JPNTestのDatabricks-Generative-AI-Engineer-Associate問題集を選んでください。JPNTestは長年の研究をわたって研ITの認証試験に関する品質が高く、範囲は広い教育資料が開発しました。それは確かに君のDatabricks-Generative-AI-Engineer-Associate試験に役に立つとみられます。

Databricks-Generative-AI-Engineer-Associateトレーリング学習: https://www.jpntest.com/shiken/Databricks-Generative-AI-Engineer-Associate-mondaishu

- Databricks-Generative-AI-Engineer-Associate試験の準備方法 | 更新するDatabricks-Generative-AI-Engineer-Associate日本語版問題解説試験 | 認定するDatabricks Certified Generative AI Engineer Associateトレーリング学習 □ 最新✔ Databricks-Generative-AI-Engineer-Associate □✔□問題集ファイルは「 www.jpshiken.com 」にて検索Databricks-Generative-AI-Engineer-Associate最新な問題集
- Databricks-Generative-AI-Engineer-Associate模試エンジン □ Databricks-Generative-AI-Engineer-Associate受験料 □ Databricks-Generative-AI-Engineer-Associate最新な問題集 □ ☀ www.goshiken.com □☀□で➥ Databricks-Generative-AI-Engineer-Associate □を検索し、無料でダウンロードしてくださいDatabricks-Generative-AI-Engineer-Associate最新な問題集
- Databricks-Generative-AI-Engineer-Associate試験の準備方法 | 検証するDatabricks-Generative-AI-Engineer-Associate日本語版問題解説試験 | 便利なDatabricks Certified Generative AI Engineer Associateトレーリング学習 □ ➥ www.jpshiken.com □から簡単に（ Databricks-Generative-AI-Engineer-Associate ）を無料でダウンロードできますDatabricks-Generative-AI-Engineer-Associate日本語版参考資料
- 信頼的なDatabricks-Generative-AI-Engineer-Associate日本語版問題解説 - 合格スムーズDatabricks-Generative-AI-Engineer-Associateトレーリング学習 | 効果的なDatabricks-Generative-AI-Engineer-Associate合格体験記 □ ☀ www.goshiken.com □☀□を開いて《 Databricks-Generative-AI-Engineer-Associate 》を検索し、試験資料を無料でダウンロードしてくださいDatabricks-Generative-AI-Engineer-Associate資格認定
- Databricks-Generative-AI-Engineer-Associate試験準備 □ Databricks-Generative-AI-Engineer-Associate学習範囲 □ Databricks-Generative-AI-Engineer-Associate絶対合格 □ Open Webサイト⇒ www.passtest.jp ⇐検索▹ Databricks-Generative-AI-Engineer-Associate ◃無料ダウンロードDatabricks-Generative-AI-Engineer-Associate勉強方法
- Databricks-Generative-AI-Engineer-Associate模試エンジン □ Databricks-Generative-AI-Engineer-Associate無料サンプル □ Databricks-Generative-AI-Engineer-Associate最新な問題集 □ 今すぐ{ www.goshiken.com }で[Databricks-Generative-AI-Engineer-Associate]を検索して、無料でダウンロードしてくださいDatabricks-Generative-AI-Engineer-Associate日本語対策問題集
- Databricks-Generative-AI-Engineer-Associate無料問題 □ Databricks-Generative-AI-Engineer-Associate模試エンジン □ Databricks-Generative-AI-Engineer-Associate復習対策書 □ ✔ www.goshiken.com □✔□を開いて《 Databricks-Generative-AI-Engineer-Associate 》を検索し、試験資料を無料でダウンロードしてください Databricks-Generative-AI-Engineer-Associate試験時間
- Databricks-Generative-AI-Engineer-Associate練習問題 □ Databricks-Generative-AI-Engineer-Associate受験料 □ Databricks-Generative-AI-Engineer-Associate日本語対策問題集 □ “ www.goshiken.com ”にて限定無料の➥ Databricks-Generative-AI-Engineer-Associate □問題集をダウンロードせよDatabricks-Generative-AI-Engineer-Associate模試エンジン
- 一生懸命にDatabricks Databricks-Generative-AI-Engineer-Associate日本語版問題解説 - 合格スムーズDatabricks-Generative-AI-Engineer-Associateトレーリング学習 | 検証するDatabricks-Generative-AI-Engineer-Associate合格体験記 □ 今すぐ➥ www.goshiken.com □で“ Databricks-Generative-AI-Engineer-Associate ”を検索して、無料でダウンロードしてくださいDatabricks-Generative-AI-Engineer-Associate最新な問題集
- Databricks-Generative-AI-Engineer-Associate復習対策書 □ Databricks-Generative-AI-Engineer-Associate練習問題 □ Databricks-Generative-AI-Engineer-Associate無料問題 □ 今すぐ（ www.goshiken.com ）を開き、▹ Databricks-Generative-AI-Engineer-Associate ◃を検索して無料でダウンロードしてくださいDatabricks-Generative-AI-Engineer-Associate試験対策
- Databricks-Generative-AI-Engineer-Associate合格問題 □ Databricks-Generative-AI-Engineer-Associate合格問題 □ Databricks-Generative-AI-Engineer-Associate無料サンプル □ ➥ www.topexam.jp □で使える無料オンライン版

《 Databricks-Generative-AI-Engineer-Associate 》 の試験問題Databricks-Generative-AI-Engineer-Associate無料問題

- 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, myportal.utt.edu.tt, myportal.utt.edu.tt, premiumads.co.zw, test.marpetshelters.ng, 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, 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, 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, bbs.chaken.net.cn, 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

BONUS！！！ JPNTest Databricks-Generative-AI-Engineer-Associateダンプの一部を無料でダウンロード：https://drive.google.com/open?id=177oZUwpEWLwt62R9GWHM23mGyNwiRFmh