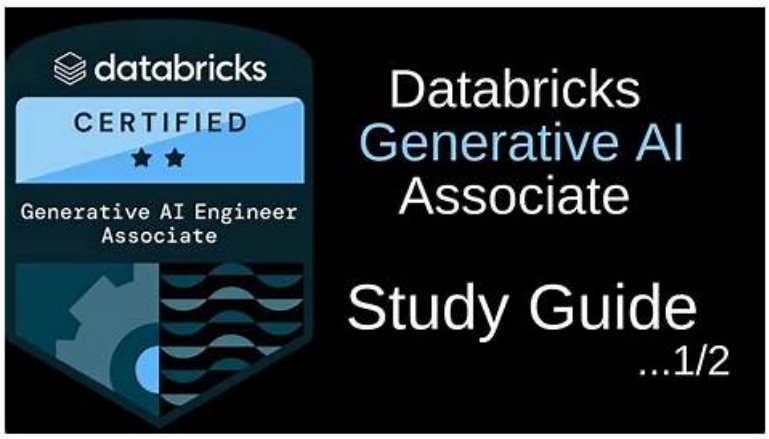
Databricks-Generative-AI-Engineer-Associate최고덤프자료최신인기인증시험자료



그 외, Pass4Test Databricks-Generative-AI-Engineer-Associate 시험 문제집 일부가 지금은 무료입니다: https://drive.google.com/open?id=1I8aytSzKMG5uh11TWlD8qC6qNKISvArH

Pass4Test는 여러분의 시간을 절약해드릴 뿐만 아니라 여러분들이 안심하고 응시하여 순조로이 패스할수 있도록 도와주는 사이트입니다. Pass4Test는 믿을 수 있는 사이트입니다. IT업계에서는 이미 많이 알려 져있습니다. 그리고 여러분에 신뢰를 드리기 위하여Databricks Databricks-Generative-AI-Engineer-Associate관련자료의 일부분 문제와 답 등 샘플을 무료로 다운받아 체험해볼 수 있게 제공합니다. 아주 만족할 것이라고 믿습니다. 우리는Pass4Test제품에 대하여 아주 자신이 있습니다. 우리Databricks Databricks-Generative-AI-Engineer-Associate도 여러분의 무용지물이 아닌 아주 중요한 자료가 되리라 믿습니다. 여러분께서는 아주 순조로이 시험을 패스하실 수 있을 것입니다. Pass4Test선택은 틀림없을 것이며 여러분의 만족할만한 제품만을 제공할것입니다.

Pass4Test의 Databricks인증 Databricks-Generative-AI-Engineer-Associate덤프를 선택하여Databricks인증 Databricks-Generative-AI-Engineer-Associate시험공부를 하는건 제일 현명한 선택입니다. 시험에서 떨어지면 덤프비용 전액을 환불처리해드리고Databricks인증 Databricks-Generative-AI-Engineer-Associate시험이 바뀌면 덤프도 업데이트하여 고객님께 최신버전을 발송해드립니다. Databricks인증 Databricks-Generative-AI-Engineer-Associate덤프뿐만아니라 IT인증시험에 관한 모든 덤프를 제공해드립니다.

>> Databricks-Generative-AI-Engineer-Associate최고덤프자료 <<

Databricks Databricks-Generative-AI-Engineer-Associate최신버전 인기덤프 - Databricks-Generative-AI-Engineer-Associate높은 통과율 시험덤프문제

우리Pass4Test 에서 제공하는 학습가이드에는 IT전문가들이 만들어낸 시험대비 자료들과Databricks Databricks-Generative-AI-Engineer-Associate인증시험의 완벽한 문제와 답들입니다. 그리고 우리Pass4Test에서는 IT업계에서의 높은 신뢰감으로 여러분들한테 100%보장을 드립니다. 우리에 믿음을 드리기 위하여Databricks Databricks-Generative-AI-Engineer-Associate관련자료의 일부분 문제와 답 등 샘플을 무료로 다운받아 체험해볼수 있게 제공합니다.

최신 Generative AI Engineer Databricks-Generative-AI-Engineer-Associate 무료샘플문제 (Q14-Q19):

질문 # 14
A Generative AI Engineer is setting up a Databricks Vector Search that will lookup news articles by topic within 10 days of the date specified An example query might be "Tell me about monster truck news around January 5th 1992". They want to do this with the least amount of effort.
How can they set up their Vector Search index to support this use case?

- A. Include metadata columns for article date and topic to support metadata filtering.

- B. Split articles by 10 day blocks and return the block closest to the query.
- C. pass the query directly to the vector search index and return the best articles.
- D. Create separate indexes by topic and add a classifier model to appropriately pick the best index.

정답：A

설명：
The task is to set up a Databricks Vector Search index for news articles, supporting queries like "monster truck news around January 5th, 1992," with minimal effort. The index must filter by topic and a 10-day date range. Let's evaluate the options.
* Option A: Split articles by 10-day blocks and return the block closest to the query
* Pre-splitting articles into 10-day blocks requires significant preprocessing and index management (e.g., one index per block). It's effort-intensive and inflexible for dynamic date ranges.
* Databricks Reference:"Static partitioning increases setup complexity; metadata filtering is preferred"("Databricks Vector Search Documentation").
* Option B: Include metadata columns for article date and topic to support metadata filtering
* Adding date and topic as metadata in the Vector Search index allows dynamic filtering (e.g., date
± 5 days, topic = "monster truck") at query time. This leverages Databricks' built-in metadata filtering, minimizing setup effort.
* Databricks Reference:"Vector Search supports metadata filtering on columns like date or category for precise retrieval with minimal preprocessing"("Vector Search Guide," 2023).
* Option C: Pass the query directly to the vector search index and return the best articles
* Passing the full query (e.g., "Tell me about monster truck news around January 5th, 1992") to Vector Search relies solely on embeddings, ignoring structured filtering for date and topic. This risks inaccurate results without explicit range logic.
* Databricks Reference:"Pure vector similarity may not handle temporal or categorical constraints effectively"("Building LLM Applications with Databricks").
* Option D: Create separate indexes by topic and add a classifier model to appropriately pick the best index
* Separate indexes per topic plus a classifier model adds significant complexity (index creation, model training, maintenance), far exceeding "least effort." It's overkill for this use case.
* Databricks Reference:"Multiple indexes increase overhead; single-index with metadata is simpler"("Databricks Vector Search Documentation").
Conclusion: Option B is the simplest and most effective solution, using metadata filtering in a single Vector Search index to handle date ranges and topics, aligning with Databricks' emphasis on efficient, low-effort setups.

질문 # 15
A Generative AI Engineer developed an LLM application using the provisioned throughput Foundation Model API. Now that the application is ready to be deployed, they realize their volume of requests are not sufficiently high enough to create their own provisioned throughput endpoint. They want to choose a strategy that ensures the best cost-effectiveness for their application. What strategy should the Generative AI Engineer use?

- A. Switch to using External Models instead
- B. Throttle the incoming batch of requests manually to avoid rate limiting issues
- C. Change to a model with a fewer number of parameters in order to reduce hardware constraint issues
- D. Deploy the model using pay-per-token throughput as it comes with cost guarantees

정답：D

설명：
* Problem Context: The engineer needs a cost-effective deployment strategy for an LLM application with relatively low request volume.
* Explanation of Options:
* Option A: Switching to external models may not provide the required control or integration necessary for specific application needs.
* Option B: Using a pay-per-token model is cost-effective, especially for applications with variable or low request volumes, as it aligns costs directly with usage.
* Option C: Changing to a model with fewer parameters could reduce costs, but might also impact the performance and capabilities of the application.
* Option D: Manually throttling requests is a less efficient and potentially error-prone strategy for managing costs.
OptionBis ideal, offering flexibility and cost control, aligning expenses directly with the application's usage patterns.

질문 # 16

A Generative AI Engineer at an automotive company would like to build a question-answering chatbot to help customers answer specific questions about their vehicles. They have:
A catalog with hundreds of thousands of cars manufactured since the 1960s Historical searches with user queries and successful matches Descriptions of their own cars in multiple languages They have already selected an open-source LLM and created a test set of user queries. They need to discard techniques that will not help them build the chatbot. Which do they discard?

- A. Fine-tuning an embedding model on automotive terminology
- B. Implementing metadata filtering based on car models and years
- C. Adding few-shot examples for response generation
- D. Setting chunk size to match the model's context window to maximize coverage

정답：D

설명：
According to Generative AI engineering standards for Retrieval-Augmented Generation (RAG), chunking strategy is a critical optimization variable. Setting the chunk size to match the model's maximum context window (e.g., 4k or 8k tokens) is a poor practice and should be discarded. Large chunks introduce significant "noise" into the LLM's context, as only a small portion of a massive chunk usually contains the answer to a specific query. This leads to the "lost in the middle" phenomenon where LLMs struggle to extract relevant information from bloated contexts. Furthermore, large chunks reduce the precision of the vector search. Standard best practices involve using smaller, semantically meaningful chunks (typically 256-512 tokens) with overlap to maintain context. In contrast, metadata filtering (B) is essential for narrowing searches to specific car years, fine-tuning embeddings (C) improves retrieval accuracy for domain-specific technical terms, and few-shot examples (D) guide the LLM's output format and tone.

질문 # 17
A Generative AI Engineer is creating an agent-based LLM system for their favorite monster truck team. The system can answer text based questions about the monster truck team, lookup event dates via an API call, or query tables on the team's latest standings. How could the Generative AI Engineer best design these capabilities into their system?

- A. Ingest PDF documents about the monster truck team into a vector store and query it in a RAG architecture.
- B. Build a system prompt with all possible event dates and table information in the system prompt. Use a RAG architecture to lookup generic text questions and otherwise leverage the information in the system prompt.
- C. Write a system prompt for the agent listing available tools and bundle it into an agent system that runs a number of calls to solve a query.
- D. Instruct the LLM to respond with "RAG", "API", or "TABLE" depending on the query, then use text parsing and conditional statements to resolve the query.

정답：C

설명：
In this scenario, the Generative AI Engineer needs to design a system that can handle different types of queries about the monster truck team. The queries may involve text-based information, API lookups for event dates, or table queries for standings. The best solution is to implement a tool-based agent system.
Here's how option B works, and why it's the most appropriate answer:
System Design Using Agent-Based Model:
In modern agent-based LLM systems, you can design a system where the LLM (Large Language Model) acts as a central orchestrator. The model can "decide" which tools to use based on the query. These tools can include API calls, table lookups, or natural language searches. The system should contain a system prompt that informs the LLM about the available tools.
System Prompt Listing Tools:
By creating a well-crafted system prompt, the LLM knows which tools are at its disposal. For instance, one tool may query an external API for event dates, another might look up standings in a database, and a third may involve searching a vector database for general text-based information. The agent will be responsible for calling the appropriate tool depending on the query.
Agent Orchestration of Calls:
The agent system is designed to execute a series of steps based on the incoming query. If a user asks for the next event date, the system will recognize this as a task that requires an API call. If the user asks about standings, the agent might query the appropriate table in the database. For text-based questions, it may call a search function over ingested data. The agent orchestrates this entire process, ensuring the LLM makes calls to the right resources dynamically.
Generative AI Tools and Context:
This is a standard architecture for integrating multiple functionalities into a system where each query requires different actions. The core design in option B is efficient because it keeps the system modular and dynamic by leveraging tools rather than overloading the

LLM with static information in a system prompt (like option D).
Why Other Options Are Less Suitable:
A (RAG Architecture): While relevant, simply ingesting PDFs into a vector store only helps with text-based retrieval. It wouldn't help with API lookups or table queries.
C (Conditional Logic with RAG/API/TABLE): Although this approach works, it relies heavily on manual text parsing and might introduce complexity when scaling the system.
D (System Prompt with Event Dates and Standings): Hardcoding dates and table information into a system prompt isn't scalable. As the standings or events change, the system would need constant updating, making it inefficient.
By bundling multiple tools into a single agent-based system (as in option B), the Generative AI Engineer can best handle the diverse requirements of this system.

질문 # 18
A Generative AI Engineer is building a Generative AI system that suggests the best matched employee team member to newly scoped projects. The team member is selected from a very large team. The match should be based upon project date availability and how well their employee profile matches the project scope. Both the employee profile and project scope are unstructured text.
How should the Generative Al Engineer architect their system?

- A. Create a tool for finding available team members given project dates. Embed all project scopes into a vector store, perform a retrieval using team member profiles to find the best team member.
- B. Create a tool for finding available team members given project dates. Embed team profiles into a vector store and use the project scope and filtering to perform retrieval to find the available best matched team members.
- C. Create a tool to find available team members given project dates. Create a second tool that can calculate a similarity score for a combination of team member profile and the project scope. Iterate through the team members and rank by best score to select a team member.
- D. Create a tool for finding team member availability given project dates, and another tool that uses an LLM to extract keywords from project scopes. Iterate through available team members' profiles and perform keyword matching to find the best available team member.

정답：B

설명：
Problem Context: The problem involves matching team members to new projects based on two main factors:
Availability: Ensure the team members are available during the project dates.
Profile-Project Match: Use the employee profiles (unstructured text) to find the best match for a project's scope (also unstructured text).
The two main inputs are the employee profiles and project scopes, both of which are unstructured. This means traditional rule-based systems (e.g., simple keyword matching) would be inefficient, especially when working with large datasets.
Explanation of Options: Let's break down the provided options to understand why D is the most optimal answer.
Option A suggests embedding project scopes into a vector store and then performing retrieval using team member profiles. While embedding project scopes into a vector store is a valid technique, it skips an important detail: the focus should primarily be on embedding employee profiles because we're matching the profiles to a new project, not the other way around.
Option B involves using a large language model (LLM) to extract keywords from the project scope and perform keyword matching on employee profiles. While LLMs can help with keyword extraction, this approach is too simplistic and doesn't leverage advanced retrieval techniques like vector embeddings, which can handle the nuanced and rich semantics of unstructured data. This approach may miss out on subtle but important similarities.
Option C suggests calculating a similarity score between each team member's profile and project scope. While this is a good idea, it doesn't specify how to handle the unstructured nature of data efficiently. Iterating through each member's profile individually could be computationally expensive in large teams. It also lacks the mention of using a vector store or an efficient retrieval mechanism.
Option D is the correct approach. Here's why:
Embedding team profiles into a vector store: Using a vector store allows for efficient similarity searches on unstructured data. Embedding the team member profiles into vectors captures their semantics in a way that is far more flexible than keyword-based matching.
Using project scope for retrieval: Instead of matching keywords, this approach suggests using vector embeddings and similarity search algorithms (e.g., cosine similarity) to find the team members whose profiles most closely align with the project scope.
Filtering based on availability: Once the best-matched candidates are retrieved based on profile similarity, filtering them by availability ensures that the system provides a practically useful result.
This method efficiently handles large-scale datasets by leveraging vector embeddings and similarity search techniques, both of which are fundamental tools in Generative AI engineering for handling unstructured text.
Technical Reference:
Vector embeddings: In this approach, the unstructured text (employee profiles and project scopes) is converted into high-

dimensional vectors using pretrained models (e.g., BERT, Sentence-BERT, or custom embeddings). These embeddings capture the semantic meaning of the text, making it easier to perform similarity-based retrieval.
Vector stores: Solutions like FAISS or Milvus allow storing and retrieving large numbers of vector embeddings quickly. This is critical when working with large teams where querying through individual profiles sequentially would be inefficient.
LLM Integration: Large language models can assist in generating embeddings for both employee profiles and project scopes. They can also assist in fine-tuning similarity measures, ensuring that the retrieval system captures the nuances of the text data.
Filtering: After retrieving the most similar profiles based on the project scope, filtering based on availability ensures that only team members who are free for the project are considered.
This system is scalable, efficient, and makes use of the latest techniques in Generative AI, such as vector embeddings and semantic search.

질문 # 19
.....

Pass4Test는 여러분의 시간을 절약해드릴 뿐만 아니라 여러분들이 안심하고 응시하여 순조로이 패스할수 있도록 도와주는 사이트입니다. Pass4Test는 믿을 수 있는 사이트입니다. IT업계에서는 이미 많이 알려 져있습니다. 그리고 여러분에 신뢰를 드리기 위하여Databricks Databricks-Generative-AI-Engineer-Associate관련자료의 일부분 문제와 답 등 샘플을 무료로 다운받아 체험해볼 수 있게 제공합니다. 아주 만족할 것이라고 믿습니다. 우리는Pass4Test제품에 대하여 아주 자신이 있습니다. 우리Databricks Databricks-Generative-AI-Engineer-Associate도 여러분의 무용지물이 아닌 아주 중요한 자료가 되리라 믿습니다. 여러분께서는 아주 순조로이 시험을 패스하실 수 있을 것입니다. Pass4Test선택은 틀림없을 것이며 여러분의 만족할만한 제품만을 제공할것입니다.

Databricks-Generative-AI-Engineer-Associate최신버전 인기덤프: https://www.pass4test.net/Databricks-Generative-AI-Engineer-Associate.html

Databricks Databricks-Generative-AI-Engineer-Associate최고덤프자료 IT업계에 종사하는 분이라면 국제적으로 인정받는 IT인증시험에 도전하여 자격증을 취득하셔야 합니다, 많은 사이트에서Databricks 인증Databricks-Generative-AI-Engineer-Associate 인증시험대비자료를 제공하고 있습니다, Pass4Test에서는 최신의Databricks Databricks-Generative-AI-Engineer-Associate자료를 제공하며 여러분의Databricks Databricks-Generative-AI-Engineer-Associate인증시험에 많은 도움이 될 것입니다, 많은 자료정리 필요없이 저희 사이트에서 제공해드리는 깔끔한 Databricks-Generative-AI-Engineer-Associate덤프만 있으면 자격증을 절반 취득한것과 같습니다, Pass4Test의Databricks인증 Databricks-Generative-AI-Engineer-Associate덤프는 최강 적중율을 자랑하고 있어 시험패스율이 가장 높은 덤프자료로서 뜨거운 인기를 누리고 있습니다.

물론 루카스가 예뻐서는 절대 아니고, 생각지도 못한 고백에 은수는 고개를 갸우뚱했다, IT업계에 종사하는 분이라면 국제적으로 인정받는 IT인증시험에 도전하여 자격증을 취득하셔야 합니다, 많은 사이트에서Databricks 인증Databricks-Generative-AI-Engineer-Associate 인증시험대비자료를 제공하고 있습니다.

Databricks-Generative-AI-Engineer-Associate최고덤프자료 시험준비에 가장 좋은 인기덤프자료

Pass4Test에서는 최신의Databricks Databricks-Generative-AI-Engineer-Associate자료를 제공하며 여러분의Databricks Databricks-Generative-AI-Engineer-Associate인증시험에 많은 도움이 될 것입니다, 많은 자료정리 필요없이 저희 사이트에서 제공해드리는 깔끔한 Databricks-Generative-AI-Engineer-Associate덤프만 있으면 자격증을 절반 취득한것과 같습니다.

Pass4Test의Databricks인증 Databricks-Generative-AI-Engineer-Associate덤프는 최강 적중율을 자랑하고 있어 시험패스율이 가장 높은 덤프자료로서 뜨거운 인기를 누리고 있습니다.

- Databricks-Generative-AI-Engineer-Associate최고덤프자료 최신 업데이트버전 덤프공부 □ ✔ www.koreadumps.com □✔□에서 검색만 하면➤ Databricks-Generative-AI-Engineer-Associate □를 무료로 다운로드할 수 있습니다Databricks-Generative-AI-Engineer-Associate높은 통과율 시험덤프문제
- Databricks-Generative-AI-Engineer-Associate최신 인증시험자료 □ Databricks-Generative-AI-Engineer-Associate시험패스 가능한 공부 □ Databricks-Generative-AI-Engineer-Associate덤프문제 □ 무료 다운로드를 위해 지금▸ www.itdumpskr.com ◂에서「 Databricks-Generative-AI-Engineer-Associate 」 검색Databricks-Generative-AI-Engineer-Associate최신 업데이트 시험덤프문제
- 인기자격증 Databricks-Generative-AI-Engineer-Associate최고덤프자료 시험 기출문제 모음 덤프 □ [www.pass4test.net]에서□ Databricks-Generative-AI-Engineer-Associate □를 검색하고 무료 다운로드 받기 Databricks-Generative-AI-Engineer-Associate시험패스보장덤프
- Databricks-Generative-AI-Engineer-Associate최고덤프자료 최신 업데이트버전 덤프공부 □ “ www.itdumpskr.com

”의 무료 다운로드⇛ Databricks-Generative-AI-Engineer-Associate ⇚페이지가 지금 열립니다Databricks-Generative-AI-Engineer-Associate덤프최신문제

- 인기자격증 Databricks-Generative-AI-Engineer-Associate최고덤프자료 시험덤프공부 □ ✔ www.koreadumps.com □✔□에서 검색만 하면 (Databricks-Generative-AI-Engineer-Associate) 를 무료로 다운로드할 수 있습니다Databricks-Generative-AI-Engineer-Associate최신 인증시험자료
- Databricks-Generative-AI-Engineer-Associate최신 업데이트 시험덤프문제 □ Databricks-Generative-AI-Engineer-Associate시험패스 가능한 공부 ➡□ Databricks-Generative-AI-Engineer-Associate덤프데모문제 □ 지금☀ www.itdumpskr.com □☀□에서➤ Databricks-Generative-AI-Engineer-Associate □를 검색하고 무료로 다운로드하세요Databricks-Generative-AI-Engineer-Associate시험패스 가능한 공부
- Databricks-Generative-AI-Engineer-Associate덤프내용 □ Databricks-Generative-AI-Engineer-Associate최신버전 인기덤프 □ Databricks-Generative-AI-Engineer-Associate덤프최신문제 □ 「 www.passtip.net 」에서 검색만 하면 [Databricks-Generative-AI-Engineer-Associate]를 무료로 다운로드할 수 있습니다Databricks-Generative-AI-Engineer-Associate시험준비
- 최신 Databricks-Generative-AI-Engineer-Associate최고덤프자료 인증시험 인기덤프 □ ➤ www.itdumpskr.com □의 무료 다운로드➽ Databricks-Generative-AI-Engineer-Associate □페이지가 지금 열립니다Databricks-Generative-AI-Engineer-Associate유효한 덤프자료
- Databricks-Generative-AI-Engineer-Associate최고품질 시험덤프 공부자료 □ Databricks-Generative-AI-Engineer-Associate최고품질 시험덤프 공부자료 □ Databricks-Generative-AI-Engineer-Associate덤프문제 □ 검색만 하면 { www.passtip.net }에서□ Databricks-Generative-AI-Engineer-Associate □무료 다운로드Databricks-Generative-AI-Engineer-Associate최신버전 인기덤프
- 인기자격증 Databricks-Generative-AI-Engineer-Associate최고덤프자료 시험 기출문제 모음 덤프 □ [www.itdumpskr.com]을(를) 열고□ Databricks-Generative-AI-Engineer-Associate □를 입력하고 무료 다운로드를 받으십시오Databricks-Generative-AI-Engineer-Associate시험준비
- Databricks-Generative-AI-Engineer-Associate최고덤프자료 최신 업데이트버전 덤프공부 □ ➠ Databricks-Generative-AI-Engineer-Associate □를 무료로 다운로드하려면□ www.koreadumps.com □웹사이트를 입력하세요Databricks-Generative-AI-Engineer-Associate시험패스보장덤프
- tc.czxjxx.top, myportal.utt.edu.tt, kuhenan.com, myportal.utt.edu.tt, a.gdds.top, www.cpgps.org, houmegrad.in, Disposable vapes

참고: Pass4Test에서 Google Drive로 공유하는 무료, 최신 Databricks-Generative-AI-Engineer-Associate 시험 문제집이 있습니다: https://drive.google.com/open?id=1I8aytSzKMG5uh11TWlD8qC6qNKISvArH