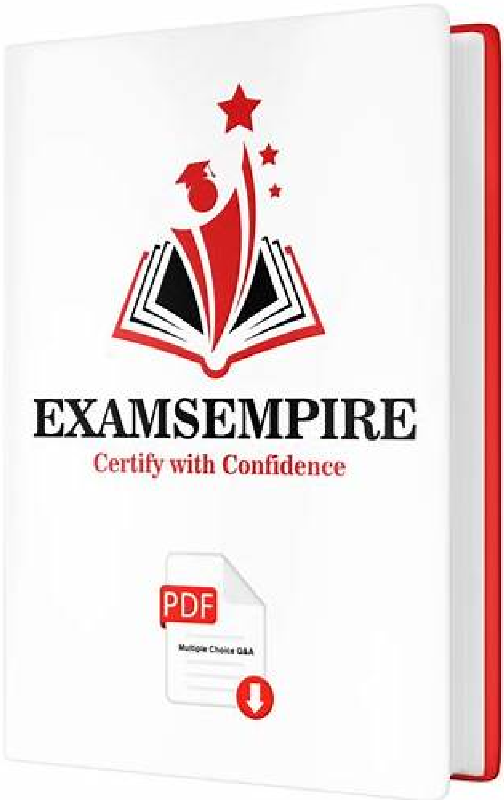
Reliable GES-C01 Source, GES-C01 Exam Torrent



BONUS!!! Download part of TestkingPass GES-C01 dumps for free: https://drive.google.com/open?id=1HJivn5lGCL3SnueO0XsewsjSl_2ZYD7e

Thus, it leads to making your practice quite convenient. Snowflake GES-C01 desktop software functions on Windows-based computers and works without a functional internet connection. Snowflake GES-C01 Exam Questions always provide ease to their consumers. therefore, the committed team is present around the clock to fix any problem.

When you are struggling with those troublesome reference books; when you feel helpless to be productive during the process of preparing GES-C01 exams; when you have difficulty in making full use of your sporadic time and avoiding procrastination. It is time for you to realize the importance of our GES-C01 Test Prep, which can help you solve these annoyance and obtain a GES-C01 certificate in a more efficient and productive way. Not only will you be able to pass any GES-C01 test, but will gets higher score, if you choose our GES-C01 study materials.

>> Reliable GES-C01 Source <<

Snowflake GES-C01 Exam Torrent, Dump GES-C01 Check

The countless candidates have already passed their SnowPro® Specialty: Gen AI Certification Exam (GES-C01) certification exam and they all used the real, valid, and updated GES-C01 exam questions. So, why not, take a decision right now and ace your SnowPro® Specialty: Gen AI Certification Exam (GES-C01) exam preparation with top-notch Snowflake GES-C01 exam questions?

Snowflake SnowPro® Specialty: Gen AI Certification Exam Sample Questions (Q269-Q274):

NEW QUESTION # 269
A Gen AI developer has a Document AI pipeline that uses a query with 'GET PRESIGNED URL' to process multi-page PDF

documents. Despite the internal stage being correctly set up with 'SNOWFLAKE SSE' encryption and the model build being published, they observe inconsistent results. Some documents result in a Received HTTP 403 response for presigned URL. URL may be expired.
error, while other documents (containing complex diagrams and dense text in an unsupported language like Korean) are processed, but the extracted information is often incomplete or inaccurate.
Which two factors are most likely contributing to these observed issues?

- A. The default expiration time for the 'GET PRESIGNED URL' function is causing some URLs to expire before the Document AI model can process them.
- B. The documents are in an unsupported language or exceed the maximum page length of 125 pages, causing extraction failures or inaccuracies.
- C. The '!PREDICT method is being called with an outdated model build version instead of the latest one, leading to performance degradation.
- D. The Document AI model is returning answers longer than its limit of 512 tokens for entity extraction or 2048 tokens for table extraction.
- E. The role lacks the 'EXECUTE TASK' privilege, preventing the scheduled pipeline tasks from running.

Answer: A,B

Explanation:
The error 'Received HTTP 403 response for presigned URL. URL may be expired.' directly indicates that the function's default expiration time is causing some documents to be inaccessible by the Document AI model. This is a common issue when processing pipelines encounter delays. Additionally, the observation of incomplete or inaccurate extraction for documents with 'dense text in an unsupported language like Korean' directly points to language limitations. Document AI explicitly lists supported languages (English, Spanish, French, German, Portuguese, Italian, and Polish) and states that results for other languages might not be satisfactory. While the question mentions 'multi-page PDF documents' without explicitly stating they exceed page limits, the mention of 'complex diagrams and dense text' can also imply potential issues if page length (max 125 pages) is exceeded or other document requirements are not met. Thus, option D comprehensively covers these content- related issues. Option A (outdated model version) is unlikely to cause these specific errors, as the latest model is used by default if not specified. Option C (missing 'EXECUTE TASK privilege) would prevent task execution entirely, not cause intermittent URL issues or content- specific extraction problems. Option E (answers exceeding token limits) would be reflected in truncated output, not necessarily 'incomplete or inaccurate' extraction in the sense of failing to identify information in the first place.

NEW QUESTION # 270
A software development team is building a conversational AI application within Snowflake, aiming to provide a dynamic and stateful chat experience for users. The application needs to handle follow-up questions while maintaining context, provide responses with a degree of creative variation, and actively filter out any potentially harmful content. The team utilizes the SNOWFLAKE. CORTEX. COMPLETE (or AI_COMPLETE) function.

- A.
```
SELECT SNOWFLAKE.CORTEX.COMPLETE(
    model => 'llama3.1-70b',
    prompt => [{ 'role': 'user', 'content': 'First question.' }],
    options => { 'temperature': 0.2, 'guardrails': TRUE, 'max_tokens': 50 }
);
```
```
SELECT SNOWFLAKE.CORTEX.TRY_COMPLETE(
    model => 'mistral-large2',
    prompt => [
        { 'role': 'user', 'content': 'What are the sales trends?' },
        { 'role': 'assistant', 'content': 'Sales are up 10% this quarter.' }
        { 'role': 'user', 'content': 'What about next quarter?' }
    ],
    options => { 'temperature': 0.8, 'guardrails': TRUE }
```
- B.);

```
SELECT SNOWFLAKE.CORTEX.COMPLETE(
    model => 'claude-3-5-sonnet',
    prompt => [
        { 'role': 'system', 'content': 'Respond as a financial advisor.' },
        { 'role': 'user', 'content': 'What is my portfolio performance?' }
    ],
    options => { 'temperature': 0.0, 'guardrails': TRUE }
```

- C. ;

```
SELECT SNOWFLAKE.CORTEX.COMPLETE(
    model => 'snowflake-arctic',
    prompt => 'Analyze the latest market report.',
    options => { 'temperature': 0.5, 'guardrails': FALSE }
);
```

- D.

Answer: B

Explanation:
The scenario requires statefulness (multi-turn conversation), creative variation in responses, and safety (filtering harmful content). Option B correctly demonstrates a multi-turn conversation by passing a history of user and assistant messages in the prompt array. It uses a of which allows for creative variation in the output, as higher temperatures result in more diverse output. It also sets guardrails temperature 0.8, to enable content filtering. Option A's prompt is single-turn, not multi-turn. Option C uses a single-turn prompt and disables guardrails. to TRUE Option D uses a of which produces deterministic results and lacks creative variation. Option E has an incorrect conversation temperature 0.0, history structure with two consecutive 'user' roles without an 'assistant' response in between, which is not supported for stateful conversations.

NEW QUESTION # 271
A data team is refining their Cortex Analyst semantic model to improve the accuracy of responses for specific, frequently asked questions and to enable better literal value searches. Consider a semantic model being developed to address these requirements. Which two configurations or features are directly relevant and correctly applied in the semantic model YAML for these purposes?

- ☐ Integrating a Cortex Search Service for a dimension like `product_name` by defining a `cortex_search_service` block within the dimension's configuration, such as `cortex_search_service: { service: 'my_product_search', literal_column: 'product_id' }`, to enable fuzzy searching for product names.
- ☐ Adding a `verified_queries` section at the model level with `question` and `sql` fields, and explicitly setting `use_as_onboarding_question: true` for certain entries to ensure Cortex Analyst uses pre-defined correct queries and presents them as suggested questions.
- ☐ Defining `metrics` with `expr` that directly references physical column names from multiple underlying tables without defining `relationships`, relying on Cortex Analyst's implicit join capabilities.
- ☐ Specifying `custom_instructions` within the dimension definition to embed business logic directly into the LLM's understanding of that specific dimension.
- ☐ Using the `sample_values` field for high-cardinality dimensions (e.g., millions of unique `customer_id`s) to guide literal search, as it leverages semantic similarity search without exceeding context window limits.

- A. Option D
- B. Option B
- C. Option C
- D. Option A
- E. Option E

Answer: B,D

Explanation:
Option A is correct. Cortex Search Services can be integrated into a dimension's definition (using the field with 'service' and fields) to improve literal matching by performing semantic search over the underlying column, which enhances Cortex Analyst's ability to find literal values for filtering. Option B is correct. The 'verified_querieS section allows pre-defining accurate SQL queries for specific natural language questions. Setting 'use_as_onboarding_question true' ensures these queries are used when relevant and presented as suggested questions to users. Option C is incorrect; while metrics can reference logical columns, 'relationships' between logical tables are necessary for defining joins, especially across different underlying base tables. Option D is incorrect; 'custom_instructions' are provided at the model level to give general context to the LLM for SQL query generation, not embedded within individual dimension definitions. Option E is incorrect; the 'sample_valueS field is recommended for dimensions with relatively

low-cardinality (approximately 1-10 distinct values) to aid in semantic search for literals, not for high-cardinality dimensions.

NEW QUESTION # 272

A data science team is developing a Retrieval Augmented Generation (RAG) application within Snowflake Cortex. They want to use AI Observability to assess how effectively the application retrieves context and how truthful the generated responses are based on that context. The application has two key functions: `retrieve_context(query: str) -> str` to fetch relevant information, and `generate_answer(query: str, context: str) -> str` to formulate the final response using an LLM. Which of the following configurations for instrumenting the application and computing evaluation metrics would best address their requirements?

- Instrument `retrieve_context` with `@instrument(span_type=SpanAttributes.SpanType.RETRIEVAL)` and then compute 'accuracy' and 'latency' metrics.
- Instrument `generate_answer` with `@instrument(span_type=SpanAttributes.SpanType.GENERATION)` and compute 'answer_relevance' and 'groundedness' metrics
- Instrument both `retrieve_context` and `generate_answer` with `@instrument()` and compute 'coherence' and 'cost' metrics globally for the entire application.
- Instrument the overall RAG workflow's entry point with `@instrument(span_type=SpanAttributes.SpanType.RECORD_ROOT)` and then calculate 'context_relevance' and 'correctness' metrics on the run.
- Only a generic `@instrument()` on the main application function is sufficient, as AI Observability automatically infers relevant metrics for RAG applications.

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E

Answer: B,D

Explanation:
Option B is correct because the `generate_answer` function is responsible for the final LLM response, and 'answer_relevance' and 'groundedness' metrics are specifically designed to detect the truthfulness and relevance of this response based on the retrieved context. Instrumenting it with `@instrument(span_type=SpanAttributes.SpanType.GENERATION)` makes this part of the trace clear. Option D is also correct because instrumenting the entry point with `@instrument(span_type=SpanAttributes.SpanType.RECORD_ROOT)` allows for comprehensive tracing of the entire workflow. 'Context relevance' helps detect the quality of search results retrieval, which is crucial for the `retrieve_context` part of a RAG application, and 'correctness' evaluates the overall factual accuracy of the output. Option A is less precise as 'accuracy' and 'latency' are general performance metrics, and while important, 'context_relevance' is more directly tied to retrieval quality. Option C is too general; while 'coherence' and 'cost' are valid metrics, explicitly defining span types provides more granular insights for debugging and optimization, especially in complex RAG workflows. Option E is incorrect; while AI Observability provides a suite of metrics, explicit instrumentation and selection of appropriate metrics are necessary for targeted evaluation.

NEW QUESTION # 273

A Data Scientist has a pre-trained PyCaret model and wants to log it into the Snowflake Model Registry for inference. The model requires specific versions of and 'scipy' , and a configuration file 'my_config.json' needs to be packaged with the model for use during inference. Assuming 'sp_session' is an active Snowpark Session, is an instance of 'PyCaretModel' , and 'train_features' is a Pandas DataFrame for which of the following code snippets correctly logs this custom PyCaret model into the Snowflake Model Registry?

- A.

```
from snowflake.ml.registry import Registry from snowflake.ml.model import custom_model # assuming PyCaretModel is custom_model.CustomModel or similar reg = Registry(session=sp_session, database_name="ML", schema_name="REGISTRY") mv = reg.log_model(my_pycaret_model, model_name="my_custom_pycaret_model", version_name="v1", pip_requirements=["pycaret==3.0.2", "scipy==1.11.4"], user_files={"config": ["/path/to/my_config.json"]}, sample_input_data=train_features)
```

- B.

```
from snowflake.ml.registry import Registry reg = Registry(session=sp_session, database_name="ML", schema_name="REGISTRY") mv = reg.log_model(my_pycaret_model, model_name="my_custom_pycaret_model", version_name="v1", pip_requirements=["pycaret==3.0.2", "scipy==1.11.4"], sample_input_data=train_features)
```

- C.

```
from snowflake.ml.registry import Registry reg = Registry(session=sp_session, database_name="ML", schema_name="REGISTRY") mv = reg.log_model(model_object=my_pycaret_model, model_name="my_custom_pycaret_model", version_name="v1", pip_packages=["pycaret==3.0.2", "scipy==1.11.4"], config_files={"my_config.json": "/path/to/my_config.json"}, input_data=train_features)
```

- D.

```
from snowflake.ml.registry import Registry reg = Registry(session=sp_session, database_name="ML", schema_name="REGISTRY") mv = reg.log_model(my_pycaret_model, model_name="my_custom_pycaret_model", version_name="v1", conda_dependencies=["pycaret==3.0.2", "scipy==1.11.4"], user_files={"config": ["/path/to/my_config.json"]}, sample_input_data=train_features, options={"target_platforms": "WAREHOUSE"})
```

- E.

```
from snowflake.ml.registry import Registry from snowflake.ml.model import custom_model reg = Registry(session=sp_session, database_name="ML", schema_name="REGISTRY") mv = reg.log_model(my_pycaret_model, model_name="my_custom_pycaret_model", version_name="v1", conda_dependencies=["pycaret", "scipy"], model_context=custom_model.ModelContext(config_file='my_config.json'), sample_input_data=train_features)
```

Answer: A

Explanation:

Option A correctly uses `pip_requirements` to specify the required Python packages with their versions. It also correctly uses the `user_files` argument to package the configuration file with the model, mapping a subdirectory name ('config') to the local path of the file. `sample_input_data` is also correctly provided to infer the input signature. Option B is incorrect because `model_context` is typically instantiated within the custom model's definition to handle internal files, not passed directly to `log_model` as an argument alongside `conda_dependencies`. Additionally, `conda_dependencies` is used, but the scenario specified PyPI packages, and Snowflake recommends using either `conda_dependencies` or `pip_requirements`, not both. Option C is incomplete as it does not include the configuration file `my_config.json`. Option D uses `conda_dependencies` when `pip_requirements` would be more direct for PyPI packages, and it also uses a non-standard `options` key to specify `target_platforms` (which should be a top-level argument for `log_model` or `options={'target_platforms': [...]}`). While `conda_dependencies` can specify channel, mixing is discouraged, and the explicit `options` usage for `target_platforms` is less conventional than passing it as a direct argument to `log_model` if needed. Option E uses non-standard parameter names such as `model_object`, `pip_packages`, `config_files`, and `input_data`, which are not part of the `Registry.log_model` API.

NEW QUESTION # 274

......

As the saying goes, opportunities for those who are prepared. If you have made up your mind to get respect and power, the first step you need to do is to get the GES-C01 certification, because the certification is a reflection of your ability. If you have the GES-C01 Certification, it will be easier for you to get respect and power. Our company happened to be designing the GES-C01 exam question.

Snowflake Reliable GES-C01 Source As for PPT online version, as long as you download the app into your computer, With our GES-C01 practice test software, you can simply assess yourself by going through the GES-C01 practice tests, The practice material of TestkingPass GES-C01 Exam Torrent can be instantly accessed just after purchasing it, You can open and use the GES-C01 Questions from any location at any time on your smartphones, tablets, and laptops.

The basic blend modes are easy to find, but the Advanced Layer GES-C01 Blending controls tucked away in the Layer Style dialog box are a little more obscure, Stewards of True Value.

As for PPT online version, as long as you download the app into your computer, With our GES-C01 Practice Test software, you can simply assess yourself by going through the GES-C01 practice tests.

Marvelous Snowflake Reliable GES-C01 Source Are Leading Materials & Verified GES-C01: SnowPro® Specialty: Gen AI Certification Exam

The practice material of TestkingPass can be instantly accessed just after purchasing it, You can open and use the GES-C01 Questions from any location at any time on your smartphones, tablets, and laptops.

We constantly update our SnowPro® Specialty: Gen AI Certification Exam test products with the inclusion of new GES-C01 brain dump questions based on expert's research.

➤ www.pdfvce.com enter 【 GES-C01 】 and obtain a free download Dumps GES-C01 Reviews

- GES-C01 Free Pdf Guide GES-C01 Latest Test Bootcamp Valid GES-C01 Test Registration Download " GES-C01 " for free by simply entering { www.testkingpass.com } website GES-C01 Free Pdf Guide
- 100% Pass Quiz Snowflake - GES-C01 - SnowPro® Specialty: Gen AI Certification Exam –High-quality Reliable Source   Download ➡ GES-C01 for free by simply searching on www.pdfvce.com GES-C01 Free Pdf Guide
- Valid GES-C01 Exam Tutorial GES-C01 Reliable Exam Online Test GES-C01 Practice Search for ➠ GES-C01 and easily obtain a free download on " www.practicevce.com " ↪Valid GES-C01 Exam Tutorial
- www.stes.tyc.edu.tw, hackingworlds.com, exxpertscm.com, 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, www.stes.tyc.edu.tw, study.stcs.edu.np, www.stes.tyc.edu.tw, Disposable vapes

BONUS!!! Download part of TestkingPass GES-C01 dumps for free: https://drive.google.com/open?id=1HJivn5lGCL3SnueO0XsewsjSl_2ZYD7e