Databricks.Databricks-Certified-Professional-Data-Engineer.v2024-07-10.q88

試験コード:	Databricks-Certified-Professional-Data-Engineer
試験名称:	Databricks Certified Professional Data Engineer Exam
認定資格:	Databricks
無料問題数:	88
バージョン:	v2024-07-10
アクセス数:	444
ページビュー数:	880
https://www.jpnpdf.com/Databricks.Databricks-Certified-Professional-Data-Engineer.v2024-07-10.q88-mondaishu.html	

最新問題: **1**
AUTO LOADER を使用しているときに、ロードの一部として推論された列のほとんどが、整数であるはずの列を含む文字列データ型であることに気づきました。これを修正するにはどうすればよいでしょうか?
A. cloudfiles.schemalocation にソース テーブルのスキーマを指定します。
B. cloudfiles.schemalocation にターゲット テーブルのスキーマを指定します。
C. スキーマのヒントを提供します
D. チェックポイントの場所を更新します
E. データ型を明示的にキャストして受信データを修正します。
Answer: (解答を表示する**)**
説明
答えは、Dスキーマのヒントを提供する」です。
1.spark.readStream \
2.format("クラウドファイル") \
3.option("cloudFiles.format", "csv") \
4.option("ヘッダー", "true") \
5.option("cloudFiles.schemaLocation", schema_location) \
6.option("cloudFiles.schemaHints", "id int, 説明文字列")
7.load(生データの場所)
8.writeStream \
9.オプション("チェックポイントの場所", チェックポイントの場所) \
10.start(target_delta_table_location)option("cloudFiles.schemaHints", "id int, description string")

ここでは、id 列が int で、説明が文字列であるというヒントを提供しています。cloudfiles.schemalocation を使用してロード プロセス中にスキーマ推論の出力を保存する場合、スキーマ ヒントを使用すると、事前に既知の列のデータ型を強制できます。時間。

最新問題: **2**
データ アーキテクトは、Lakehouse 内のすべてのテーブルを外部 (「アンマネージド」とも呼ばれる) Delta Lake テーブルとして構成することを義務付けています。
この要件を確実に満たせるのはどのアプローチですか?
A. データベースの作成時に、LOCATION キーワードが使用されていることを確認してください。
B. すべてのテーブル ストレージに対して外部データ ウェアハウスを構成する場合は、すべての ELT に対して Databricks を利用します。
C. データをテーブルに保存するときは、デルタ形式とともに完全なファイル パスが指定されていることを確認してください。
D. テーブルを作成するときは、CREATE TABLE ステートメントで EXTERNAL キーワードが使用されていることを確認してください。
E. ワークスペースを構成するときは、外部クラウド オブジェクト ストレージがマウントされていることを確認してください。
Answer: D (メーッセージを残す**)**
外部テーブルまたは管理対象外の Delta Lake テーブルを作成するには、CREATE TABLE ステートメントで EXTERNAL キーワードを使用する必要があります。これは、テーブルがカタログによって管理されておらず、テーブルの削除時にデータ ファイルが削除されないことを示します。データ ファイルが保存されるパスを指定する LOCATION 句も指定する必要があります。例えば :
CREATE EXTERNAL TABLE events (date DATE、eventId STRING、eventType STRING、data STRING) USING DELTA LOCATION '/mnt/delta/events'; これにより、「/mnt/delta/events」パス内のデータ ファイルを参照する events という名前の外部 Delta Lake テーブルが作成されます。このテーブルを削除しても、データ ファイルはそのまま残り、同じステートメントでテーブルを再作成できます。
参考文献:
* https://docs.databricks.com/delta/delta-batch.html#create-a-table
* https://docs.databricks.com/delta/delta-batch.html#drop-a-table

最新問題: **3**
各ビジュアライゼーションには入力するデータが大量に含まれているため、ブラウザーへの読み込みに時間がかかるダッシュボードに取り組んでいます。この問題に対処するには、次のどのアプローチを使用できますか?
A. SQL エンドポイント クラスターのサイズを増やす

B. SQL エンドポイント クラスターの最大範囲のスケールを増加します。
C. Databricks SQL クエリ フィルターを使用して、各ビジュアライゼーションのデータ量を制限します。
D. Delta Lake からデータを削除します
E. デルタ キャッシュを使用して中間結果を保存します
Answer: (解答を表示する**)**
説明
注*: この質問は誤解を招くように聞こえるかもしれませんが、これらは試験で出題されるタイプの質問です。
クエリ フィルターを使用すると、ビジュアライゼーションに表示されるデータの量を対話的に減らすことができます。クエリ パラメーターと似ていますが、いくつかの重要な違いがあります。クエリ フィルターは、ブラウザーに読み込まれた後のデータを制限します。これにより、フィルターは、クエリの実行に時間がかかる、レートが制限される、またはコストがかかる小規模なデータセットや環境に最適になります。
このクエリ フィルターは、データ レベルで適用する必要があるフィルターとは異なり、視覚化レベルにあるため、表示するデータの量を切り替えることができます。
1.SELECT アクション AS `action::filter`、COUNT(0) AS "アクション数"
2.FROMイベント
3.GROUP BYアクション
クエリにフィルターがある場合、ダッシュボード レベルでフィルターを適用することもできます。すべてのクエリにフィルターを適用するには、 ダッシュボード レベル フィルターを使用」チェックボックスを選択します。
ダッシュボードフィルター
クエリフィルター | AWS 上のデータブリック

最新問題**: 4**
ユーザーからのコンテンツ投稿に関するメタデータを表す Delta Lake テーブルには、次のスキーマがあります。
user_id LONG、post_text STRING、post_id STRING、経度 FLOAT、緯度 FLOAT、post_time TIMESTAMP、date DATE このテーブルは日付列によってパーティション化されています。クエリは次のフィルターを使用して実行されます。
経度 < 20 & 経度 > -20
データがどのようにフィルタリングされるかを説明するステートメントはどれですか?
A. デルタ ログの統計は、フィルターされた範囲にファイルが含まれる可能性のあるパーティションを識別するために使用されます。
B. オプティマイザはパーティション列と経度の関係を認識していないため、ファイルのスキップは発生しません。
C. デルタ エンジンは、トランザクション ログの行レベルの統計を使用して、フィルター基準を満たすフライを識別します。

D. デルタ ログの統計は、フィルターされた範囲内のレコードを含む可能性のあるデータ ファイルを識別するために使用されます。
E. デルタ エンジンは寄木細工のファイル フッターをスキャンして、フィルター基準を満たす各行を識別します。
Answer: D (メーッセージを残す**)**
説明
これは、経度 < 20 & 経度 > -20 のフィルターを使用してクエリを実行したときにデータがどのようにフィルターされるかを説明しているため、正解です。クエリは、次のスキーマを持つ Delta Lake テーブルで実行されます。
user_id LONG、post_text STRING、post_id STRING、経度 FLOAT、緯度 FLOAT、post_time TIMESTAMP、日付 DATE。このテーブルは日付列によってパーティション化されています。パーティション化された Delta Lake テーブルでクエリが実行されると、Delta Lake は Delta Log の統計を使用して、フィルターされた範囲内のレコードを含む可能性のあるデータ ファイルを識別します。統計には、各データ ファイルの各列の最小値と最大値などの情報が含まれます。これらの統計を使用することで、Delta Lake はフィルター条件に一致しないデータ ファイルの読み取りをスキップできるため、クエリのパフォーマンスが向上し、I/O コストが削減されます。検証済みの参照: [Databricks Certified Data Engineer Professional]、 Delta Lake」セクション; Databricks ドキュメントの データスキップ」セクション。

最新問題**: 5**
データ サイエンス チームは、ユーザー レビューからの自由形式テキストに対するクエリを高速化するための支援を要求しました。
データは現在、以下のスキーマを使用して Parquet に保存されています。
item_id INT、user_id INT、review_id INT、評価 FLOAT、レビュー STRING
レビュー欄には、ユーザーが残したレビューの全文が表示されます。具体的には、データ サイエンス チームは、この分野に 30 のキーワードのいずれかが存在するかどうかを特定しようとしています。
若手のデータ エンジニアは、このデータを Delta Lake に変換するとクエリのパフォーマンスが向上すると提案しています。
ジュニア データ エンジニアの提案に対する正しい回答はどれですか?
A. Delta Lake の統計は、カーディナリティの高いフリー テキスト フィールドに対しては最適化されていません。
B. テキスト データは Delta Lake に保存できません。
C. パフォーマンスの向上を確認するには、ZORDER ON レビューを実行する必要があります。
D. デルタ ログは、選択的フィルタリングをサポートするためにフリー テキスト フィールドの用語マトリックスを作成します。
E. Delta Lake 統計は、テーブルの最初の 4 列についてのみ収集されます。

Answer: A (メーッセージを残す)
データを Delta Lake に変換しても、レビュー列などのカーディナリティの高いフリー テキスト フィールドではクエリのパフォーマンスが向上しない可能性があります。これは、Delta Lake が各列の最小値と最大値に関する統計を収集するためであり、フリー テキスト フィールドのデータをフィルタリングしたりスキップしたりする場合にはあまり役に立ちません。
さらに、Delta Lake はデフォルトで最初の 32 列の統計を収集しますが、テーブルにさらに多くの列がある場合、レビュー列が含まれない場合があります。したがって、ジュニア データ エンジニアの提案は正しくありません。より良いアプローチは、Elasticsearch などの全文検索エンジンを使用してレビュー列のインデックスを作成し、クエリを実行することです。あるいは、トークン化、ステミング、見出し語化などの自然言語処理手法を使用して、レビュー列を前処理し、データのフィルタリングやスキップに使用できる正規化された用語を含む新しい列を作成することもできます。参考文献:
* 最適化: https://docs.delta.io/latest/optimizations-oss.html
* Elasticsearch による全文検索: https://docs.databricks.com/data/data-sources/elasticsearch.html
* 自然言語処理: https://docs.databricks.com/applications/nlp/index.html

最新問題**: 6**
上流システムは、特定のデータ バッチの日付をパラメータとして Databricks Jobs API に渡すように構成されています。スケジュールされるノートブックは、このパラメーターを使用して、次のコードでデータを読み込みます。
df =spark.read.format("parquet").load(f"/mnt/source/(date)")
上記のコード ブロックで使用されている日付 Python 変数を作成するには、どのコード ブロックを使用する必要がありますか?
A. 日付 =spark.conf.get("日付")
B. input_dict = input()
date= input_dict["日付"]
C. インポート システム
日付 = sys.argv[1]
D. 日付 = dbutils.notebooks.getParam("日付")
E. dbutils.widgets.text("日付", "null") 日付 = dbutils.widgets.get("日付")
Answer: E (メーッセージを残す)
説明
上記のコード ブロックで使用されている日付 Python 変数を作成するために使用するコード ブロックは次のとおりです。
dbutils.widgets.text("date", "null") date = dbutils.widgets.get("date") このコード ブロックは、dbutils.widgets API を使用して、日付を受け入れることができる "date" という名前のテキスト ウィジェットを作成および取得します。パラメータとしての文字列値1。ウィジェットのデフォルト値は 「null」です。これは、パラメーターが渡されない場合、日付変数は 「null」

になることを意味します。ただし、パラメーターが Databricks Jobs API を介して渡される場合、日付変数にはパラメーターの値が割り当てられます。たとえば、パラメータが 「2021-11-01」の場合、日付変数は 「2021-11-01」になります。このようにして、ノートブックは日付変数を使用して、指定されたパスからデータをロードできます。
他のオプションは次の理由により正しくありません。
オプション A は正しくありません。spark.conf.get("date") は、Databricks Jobs API 経由で渡されるパラメーターを取得する有効な方法ではありません。Spark.conf API は、ノートブックのパラメーターではなく、Spark 構成プロパティを取得または設定するために使用されます2。
input() は Databricks Jobs API 経由で渡されるパラメーターを取得する有効な方法ではないため、オプション B は不正解です。input() 関数は、API リクエストからではなく、標準入力ストリームからユーザー入力を取得するために使用されます3。
sys.argv1 は Databricks Jobs API 経由で渡されるパラメーターを取得する有効な方法ではないため、オプション C は不正解です。sys.argv リストは、notebook ではなく Python スクリプトに渡されるコマンドライン引数を取得するために使用されます4。
dbutils.notebooks.getParam("date") は Databricks Jobs API 経由で渡されるパラメーターを取得する有効な方法ではないため、オプション D は不正解です。dbutils.notebooks API は、API5 を介してパラメーターを渡す場合ではなく、ノートブックをジョブまたはサブノートブックとして実行するときにノートブック パラメーターを取得または設定するために使用されます。
参考資料: ウィジェット、Spark 構成、input()、sys.argv、ノートブック

最新問題**: 7**
デルタ湖とレイクハウスについて正しいのは次のうちどれですか?
A. Parquet はデータを行ごとに圧縮するためです。文字列は、文字が複数回繰り返される場合にのみ圧縮されます。
B. Delta Lake は、各テーブルの最初の 32 列に関する統計を自動的に収集します。これらの統計は、クエリ フィルターに基づくデータ スキップに利用されます。
C. Lakehouse のビューは、ソース テーブルの最新バージョンの有効なキャッシュを常に維持します。
D. 主キー制約と外部キー制約を利用して、重複した値がディメンション テーブルに入力されないようにすることができます。
E. Z オーダーは、Delta Lake テーブルに格納されている数値にのみ適用できます。
Answer: (解答を表示する**)**
説明
https://docs.delta.io/2.0.0/table-properties.html
Delta Lake は、各テーブルの最初の 32 列に関する統計を自動的に収集し、クエリ フィルター 1 に基づくデータ スキップに利用します。データ スキップは、ストレージ層からの無関係なデータの読み取りを回避することを目的としたパフォーマンス最適化手法です

1。Delta Lake は、最小/最大値、NULL カウント、ブルーム フィルターなどの統計を収集することにより、クエリ プランから不要なファイルやパーティションを効率的に取り除くことができます1。これにより、クエリのパフォーマンスが大幅に向上し、I/O コストが削減されます。
他のオプションは次の理由から false です。
Parquet はデータを行ごとではなく列ごとに圧縮します2。これにより、特に列内で繰り返される値や類似の値の圧縮率が向上します2。
Lakehouse のビューは、ソース テーブルの最新バージョンの有効なキャッシュを常に維持しているわけではありません3。ビューは、1 つ以上のベース テーブルに対する SQL クエリによって定義される論理構造です3。ビューはデフォルトでは実体化されません。つまり、ビューにはデータは保存されず、クエリ定義のみが保存されます3。したがって、ビューはクエリ時に常にソース テーブルの最新の状態を反映します3。
ただし、CACHE TABLE または CREATE TABLE AS SELECT コマンドを使用してビューを手動でキャッシュできます。
主キー制約と外部キー制約を利用して、重複した値がディメンション テーブルに入力されないようにすることはできません。Delta Lake は、テーブルに対する主キー制約と外部キー制約の強制をサポートしていません。制約は、テーブル内のデータの整合性と有効性を定義する論理ルールです。Delta Lake は、データの品質と一貫性を確保するためにアプリケーション ロジックまたはユーザーに依存します。
Z オーダーは、数値だけでなく、Delta Lake テーブルに格納されているあらゆる値に適用できます。Z オーダーは、1 つ以上の列でデータ ファイルを並べ替えることによって、データ ファイルのレイアウトを最適化する手法です。Z オーダーは、関連する値をクラスター化し、より効率的なデータ スキップを可能にすることで、クエリのパフォーマンスを向上させることができます。Z オーダーは、数値、文字列、日付、ブール値など、順序が定義されている任意の列に適用できます。
参考資料: データ スキップ、パーケット形式、ビュー、[キャッシュ]、[制約]、[Z オーダー]

最新問題**: 8**
マルチホップアーキテクチャにおけるゴールド層の目的は何ですか?
A. ETL スループットと分析クエリのパフォーマンスを最適化します。
B. 重複レコードを削除します
C. 集計を行わずに、元のデータの粒度を保持します。
D. データ品質チェックとスキーマの適用
E. ビジネスクリティカルなデータに対するクエリ パフォーマンスの最適化
Answer: (解答を表示する**)**
説明
メダリオン アーキテクチャ - Databricks
ゴールドレイヤー:
1.ML アプリケーション、レポート、ダッシュボード、アドホック分析を強化します

2. データの洗練されたビュー (通常は集計を使用)
3. 実稼働システムの負担を軽減します
4. ビジネスクリティカルなデータのクエリパフォーマンスを最適化します。
試験の焦点: 下の画像を確認して、メダリオン建築における各層 (ブロンズ、シルバー、ゴールド) の役割を理解してください。各層とその目的を対象としたさまざまな質問が表示されます。
Udemy 内の一部の人が私のコンテンツをコピーしているため、ウォーターマークを追加する必要がありました。

最新問題: **9**
Databricks Re-pos で実装できる CI/CD フローの開発者操作は次のうちどれですか?
A. コードがコミットされたときにマージします
B. プル リクエストとレビュー プロセス
C. Databricks Repos API をトリガーして、最新バージョンのコードを運用フォルダーにプルします
D. マージ競合を解決します。
E. ブランチを削除します
Answer: C (メーッセージを残す**)**
説明
CI/CD ワークフローを構築するときに Databricks Repos と Git プロバイダーが果たす役割を理解するには、以下の図を参照してください。
黄色で強調表示されたすべての手順は Databricks Repo で実行できます。灰色で強調表示されたすべての手順は Github や Azure DevOps などの Git プロバイダーで実行できます。

最新問題: **10**
若手開発者は、ノートブック内のコードが開発環境で正しい結果を生成しないと不満を述べています。共有されたスクリーンショットを見ると、Databricks Repos でバージョン管理されたノートブックを使用している一方で、古いロジックを含む個人ブランチを使用していることがわかります。dev-2.3.9 という名前の目的のブランチは、ブランチ選択ドロップダウンからは利用できません。
この開発者がこのノートブックの現在のロジックをレビューできるのはどのアプローチですか?
A. Repos を使用してプル リクエストを作成します。Databricks REST API を使用して現在のブランチを dev-2.3.9 に更新します。
B. リポジトリを使用してリモート Git リポジトリから変更をプルし、dev-2.3.9 ブランチを選択します。
C. Repos を使用して dev-2.3.9 ブランチをチェックアウトし、現在のブランチとの競合を自動解決します
D. すべての変更をリモート Git リポジトリのメイン ブランチにマージして戻し、リポジトリのクローンを再度作成します。

E. Repos を使用して現在のブランチと dev-2.3.9 ブランチをマージし、プル リクエストを作成してリモート リポジトリと同期します。
Answer: B (メーッセージを残す)
説明
これは、開発者がリモート リポジトリからの最新の変更でローカル リポジトリを更新し、目的のブランチに切り替えることができるため、これが正解です。変更をプルすると、変更がフェッチされるだけでマージされないため、現在のブランチに影響を与えたり、競合が発生したりすることはありません。ドロップダウンから dev-2.3.9 ブランチを選択すると、そのブランチがチェックアウトされ、その内容がノートブックに表示されます。
検証済みの参照: [Databricks Certified Data Engineer Professional]、 Databricks ツール」セクション; Databricks ドキュメントの 『リモート リポジトリからの変更のプル」セクション。

最新問題**: 11**
CREATE DATABASEsample_db ステートメントを使用してデータベース sample_db を作成する場合、DBFS 内のデータベースのデフォルトの場所はどこになりますか?
A. デフォルトの場所、DBFS:/user/
B. デフォルトの場所、/user/db/
C. デフォルトのストレージ アカウント
D. ステートメントは失敗します 場所がないとデータベースを作成できません」
E. デフォルトの場所、dbfs:/user/hive/warehouse
Answer: E (メーッセージを残す)
説明
答えは dbfs:/user/hive/warehouse です。これは、Spark がユーザー データベースを保存するデフォルトの場所です。デフォルトは、spark.sql.warehouse.dir パラメータを使用して変更できます。LOCATION キーワードを使用してカスタムの場所を指定することもできます。
これがどのように機能するかは次のとおりです。
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます
デフォルトの場所
参考までに、これはクラスターのスパーク構成またはセッション構成を使用して変更できます。
デフォルトの場所を変更するには、spark.sql.warehouse.dir の場所を変更します。
グラフィカル ユーザー インターフェイス、テキスト、アプリケーションの説明が自動生成される

最新問題**: 12**
次のエラー トレースバックを確認してください。

発生したエラーを説明しているのはどのステートメントですか?
A. 実行されたコードは PvSoark ですが、Scala ノートブックで実行されました。
B. テーブルに heartrateheartrateheartrate という名前の列がありません。
C. 列オブジェクトを乗算できないため、型エラーが発生しました。
D. DataFrame オブジェクトを乗算できないため、型エラーが発生しました。
E. 心拍数列が列として正しく識別されていないため、構文エラーが発生しています。
Answer: B (メーッセージを残す)
説明
発生するエラーは AnalysisException です。これは、何らかの論理エラーまたはセマンティック エラーが原因で Spark SQL がクエリを分析または実行できない場合に発生する例外の 1 つです1。この場合、エラー メッセージは、入力列を指定すると、クエリが列名 「heartrateheartrateheartrate」を解決できないことを示します。
「心拍数」と 「年齢」。これは、テーブルに 「heartrateheartrateheartrate」という名前の列が存在せず、クエリが無効であることを意味します。このエラーの原因としては、クエリ内のタイプミスまたはコピー＆ペーストの間違いが考えられます。このエラーを修正するには、クエリでテーブル内に存在する有効な列名を使用する必要があります。
'心拍数'。参考: AnalysisException

最新問題**: 13**
Lakehouse は、データと分析ソリューションでデータ レイクとデータ ウェアハウスを使用することへの依存関係をどのように置き換えるのでしょうか?
A. 標準データ形式で保存されたデータへのオープンな直接アクセス。
B. ACID トランザクションをサポートします。
C. BI および機械学習のワークロードをサポートします
D. エンドツーエンドのストリーミングおよびバッチ ワークロードのサポート
E. 上記すべて
Answer: E (メーッセージを残す)
説明
Lakehouse は、データ ウェアハウスとデータ レイクの利点を組み合わせたもので、
レイクハウス = データレイク + データウェアハウス
レイクハウスの主な利点をいくつか紹介します
テキスト、文字 説明が自動生成されます
レイクハウス = データレイク + データウェアハウス
テキストを含む画像、自動生成された黒板説明

最新問題**: 14**
Databricks CLI がインストールされ、正しく構成されていると仮定すると、本番ジョブで使用するために DBFS でマウントされたオブジェクト ストレージにカスタム Python Wheel をアップロードするには、どの Databricks CLI コマンドを使用できますか?

A. 設定します
B. fs
C. ジョブ
D. ライブラリ
E. ワークスペース
Answer: (解答を表示する**)**
説明
ライブラリ コマンド グループを使用すると、Databricks クラスターでライブラリをインストール、アンインストール、および一覧表示できます。library install コマンドを使用して、--whl オプションとホイール ファイルへのパスを指定することで、クラスターにカスタム Python Wheel をインストールできます。たとえば、次のコマンドを使用して、mylib-0.1-py3-none-any.whl という名前のカスタム Python Wheel を ID 1234-567890-abcde123 のクラスターにインストールできます。
databricks ライブラリをインストール --cluster-id 1234-567890-abcde123 --whl dbfs:/mnt/mylib/mylib-0.1-py3-none-any.whl
これにより、カスタム Python Wheel がクラスターにアップロードされ、実稼働ジョブで使用できるようになります。
また、libraries uninstall コマンドを使用してクラスターからライブラリをアンインストールしたり、libraries list コマンドを使用してクラスターにインストールされているライブラリを一覧表示したりすることもできます。
参考文献:
ライブラリ CLI (レガシー): https://docs.databricks.com/en/archive/dev-tools/cli/libraries-cli.html ライブラリ操作: https://docs.databricks.com/en/dev-tools/cli/commands.html#library-operations Databricks CLI をインストールまたは更新します: https://docs.databricks.com/en/dev-tools/cli/install.html

最新問題**: 15**
特定のジョブの実行が時間の経過とともにますます遅くなっているように見えます。チームは、最近の運用変更が実装されたときにこれが起こり始めたと考えています。ジョブ履歴を調べて、傾向と根本原因を特定できるかどうかを確認するように求められました。
ワークスペース UI でこの分析を実行できますか?
A. ジョブ UI で興味のあるジョブを選択します。実行で、現在のアクティブな実行と過去 60 日間の履歴実行が表示されます。
B. ジョブ UI でジョブ クラスターを選択し、Spark UI でアプリケーション ジョブ ログを選択すると、過去 60 日間の履歴実行にアクセスできます。
C. [ワークスペース ログ] でジョブ ログを選択し、監視するジョブを選択して、過去 60 日間の実行履歴を表示します。
D. [コンピューティング UI] で [ジョブ クラスター] を選択し、過去 60 日間の履歴実行を表示するジョブ クラスターを選択します。
E. 履歴ジョブ実行には REST API からのみアクセスできます

Answer: A (メーッセージを残す)
説明
答えは、
ジョブ UI で興味のあるジョブを選択し、実行で現在のアクティブな実行と過去 60 日間の履歴実行を確認できます。

最新問題**: 16**
日曜日に実行に時間がかかりすぎる Databricks ジョブをデバッグするように求められました。実行に時間がかかるステップを特定するためにどのような手順を実行しますか?
A. ジョブ実行のノートブック アクティビティは、汎用クラスターを使用している場合にのみ表示されます。
B. ワークフロー UI とジョブで監視したいジョブを選択し、実行を選択すると、ノートブックのアクティビティが表示されます。
C. ジョブの出力アクティビティを確認するには、ジョブでデバッグ モードを有効にします。出力が表示できるようになります。
D. ジョブが開始されると、ジョブのノートブック アクティビティにアクセスできなくなります。
E. コンピューティングの Spark UI を使用して、ジョブ アクティビティを監視します。
Answer: B (メーッセージを残す)
説明
答えは、「ワークフロー UI とジョブ」で監視したいジョブを選択し、実行を選択すると、ノートブックのアクティビティが表示されるということです。
現在アクティブな実行または完了した実行を表示できます。実行をクリックすると、自動的に生成されたグラフィカル ユーザー インターフェイスの説明を含む画像が表示されます。
実行をクリックしてノートブックの出力を表示します
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます

最新問題: **17**
データの問題を調査しているときに、以下のコマンドを使用してテーブルの昨日のバージョンを確認しようとしましたが、タイム トラベルを使用してテーブルの前のバージョンをクエリしているときに、テーブル内の履歴データを表示できなくなっていることに気付き、昨日テーブル履歴(DESCRIBE HISTORY table_name)コマンドに基づいてテーブルが更新されたのですが、このデータにアクセスできない理由は何でしょうか?
SELECT * FROM table_name TIMESTAMP AS OF date_sub(current_date(), 1)
A. 現在、履歴データを表示するアクセス権がありません。
B. デフォルトでは、履歴データは DELTA で 180 日ごとにクリーンアップされます。
C. コマンド VACUUM table_name RETAIN 0 がテーブルに対して実行されました
D. タイムトラベルは無効です
E. 以前のデータをクエリする前にタイム トラベルを有効にする必要があります
Answer: C (メーッセージを残す**)**
説明
答えは、VACUUM table_name RETAIN 0 が実行されたということです。
VACUUM コマンドは、デルタ テーブルに関連付けられたディレクトリを再帰的にバキューム処理し、テーブルのトランザクション ログの最新の状態ではなくなり、保持しきい値より古いデータ ファイルを削除します。デフォルトは 7 日です。
VACUUM table_name RETAIN 0 を実行すると、データの履歴バージョンがすべて失われ、タイム トラベルで提供できるのは現在の状態のみです。

最新問題: **18**
John Smith はマーケティング チームに新しく加わったチーム メンバーで、現在販売テーブルへの読み取りアクセス権を持っていますが、テーブルから行を削除するアクセス権を持っていません。これを達成するのに役立つコマンドは次のうちどれですか?
A. テーブル table_name の使用を john.smith@marketing.com に許可します
B. テーブル table_name に対する削除を john.smith@marketing.com に許可します
C. john.smith@marketing.com のテーブル table_name に DELETE を許可します
D. john.smith@marketing.com のテーブル table_name に変更を許可します
E. テーブル table_name に対する変更を john.smith@marketing.com に許可します
Answer: (解答を表示する**)**
説明
答えは GRANT MODIFY ON TABLE table_name TO john.smith@marketing.com です。INSERT、UPDATE、DELETE は MODIFY という 1 つのロールに結合されることに注意してください。
以下は、ユーザーまたはグループに付与できる権限のリストです。SELECT: オブジェクトへの読み取りアクセスを与えます。

CREATE: オブジェクト (スキーマ内のテーブルなど) を作成する機能を提供します。
MODIFY: オブジェクトにデータを追加、削除、およびオブジェクトからデータを変更する機能を提供します。
USAGE: いかなる機能も提供しませんが、スキーマ オブジェクトに対してアクションを実行するための追加要件です。
READ_METADATA: オブジェクトとそのメタデータを表示する機能を提供します。
CREATE_NAMED_FUNCTION: 既存のカタログまたはスキーマに名前付き UDF を作成する機能を提供します。
MODIFY_CLASSPATH: Spark クラスパスにファイルを追加する機能を提供します。
ALL PRIVILEGES: すべての権限を与えます (上記のすべての権限に変換されます)

最新問題**: 19**
受信データを処理するために AUTO LOADER をセットアップするように求められます。このデータは JSON 形式で到着し、クラウド オブジェクト ストレージにドロップされます。データはクラウド ストレージに到着したらすぐに処理する必要があります。次のステートメントはどれですか?正しい
A. AUTO LOADER は DELTA Lake にネイティブであり、外部のクラウド オブジェクト ストレージをサポートできません
B. ファイルがクラウド ストレージに到着したときに、外部プロセスから AUTO LOADER をトリガーする必要があります
C. AUTO LOADER は構造化ストリーム プロセスに変換する必要があります
D. AUTO LOADER は、DELTA レイクに保存されている場合のみ連続データを処理できます。
E. AUTO LOADER はファイル通知メソッドをサポートできるため、データが到着したときに処理できます。
Answer: E (メーッセージを残す**)**
説明
Auto Loader は、クラウド オブジェクト ストレージから新しいファイルを取り込むときに 2 つのモードをサポートします。ディレクトリ リスト: Auto Loader は、入力ディレクトリをリストすることで新しいファイルを識別し、ディレクトリ ポーリング アプローチを使用します。
ファイル通知: Auto Loader は、入力ディレクトリからのファイル イベントをサブスクライブする通知サービスとキュー サービスを自動的にセットアップできます。
自動生成される図の説明
ファイル通知はより効率的であり、データがクラウド オブジェクト ストレージに到着するとリアルタイムでデータを処理するために使用できます。
ファイル通知モードとディレクトリー覧モードの選択 | AWS 上のデータブリック

最新問題**: 20**

あなたは、データ サイエンティスト チームが git 統合によるノートブックのバージョン管理機能を使用していることに気づき、Databricks Repos の使用に切り替えるようチームに推奨しました。チームが Databricks Repos に切り替える必要がある理由としては、次のどれが考えられます。
A. Databricks Repos では複数のユーザーが変更を行うことができます
B. Databricks Repos ではマージと競合の解決が可能
C. Databricks Repos にはバージョン管理システムが組み込まれています
D. Databricks Repos は変更を自動的に保存します
E. Databricks リポジトリを使用すると、コメントを追加し、コミットする変更を選択できます。
Answer: E (メーッセージを残す**)**
説明
答えは、Databricks Repos を使用すると、コメントを追加し、コミットする変更を選択できるからです。

最新問題**: 21**
単体テストを PySpark アプリケーションに組み込むには、ジョブの設計に事前に注意するか、既存のコードを大幅にリファクタリングする必要があります。
この追加の取り組みを相殺する主な利点について説明しているのはどれですか?
A. すべてのステップが正しく相互作用して、望ましい最終結果が達成されることを確認します。
B. データの品質を向上させます
C. すべてのステップが分離され、個別にテストされるため、トラブルシューティングが容易になります。
D. アプリケーションの完全なユースケースを検証します。
E. デプロイメントと実行時間が短縮されます。
Answer: C (メーッセージを残す**)**

最新問題**: 22**
データ エンジニアは、ジョブを使用して自動的に処理するようにノートブックを設定しました。データエンジニアのマネージャーが望むこと
複雑なため、スケジュールをバージョン管理する必要があります。
データ エンジニアは次のどのアプローチを使用して、バージョン管理可能な構成を取得できますか?
仕事のスケジュールは？
A. Databricks リポジトリの一部であるノートブックにジョブをリンクできます。
B. ジョブのページからジョブの XML 記述をダウンロードできます。
C. 汎用クラスター上でジョブを 1 回送信できます
D. ジョブ クラスター上でジョブを 1 回送信できます
E. ジョブのページからジョブの JSON 記述をダウンロードできます。

Answer: E (メーッセージを残す)

最新問題: **23**
時間単位のバッチ ジョブは、クラウド オブジェクト ストレージ コンテナーからデータ ファイルを取り込むように構成されており、各バッチは特定の時間にソース システムによって生成されたすべてのレコードを表します。これらのレコードを処理して Lakehouse に取り込むバッチ ジョブは、遅れて到着したデータが見逃されないように十分に遅延されます。user_id フィールドは、次のスキーマを持つデータの一意のキーを表します。
user_id BIGINT、username STRING、user_utc STRING、user_region STRING、last_login BIGINT、auto_pay BOOLEAN、last_updated BIGINT 新しいレコードはすべて、ソースと同じスキーマ内のすべてのデータの完全なレコードを保持する account_history という名前のテーブルに取り込まれます。システム内の次のテーブルは、account_current という名前で、各 uniqueuser_id の最新の値を表すタイプ 1 テーブルとして実装されます。
数百万のユーザー アカウントと数万のレコードが 1 時間ごとに処理されると仮定すると、1 時間ごとのバッチ ジョブの一部として記述された account_currenttable を効率的に更新するには、どの実装を使用できますか?
A. Auto Loader を使用して、アカウント履歴ディレクトリ内の新しいファイルをサブスクライブします。Structured Streaminq トリガー 1 回ジョブを設定して、新しく検出されたファイルをアカウントの現在のテーブルにバッチ更新します。
B. ユーザー ID でグループ化したアカウント履歴テーブルに対するクエリの結果と、最終更新の最大値のフィルタリングを使用して、各バッチでアカウントの現在のテーブルを上書きします。
C. 最終更新フィールドと処理された最新時間、およびユーザー ID ごとの最大最後の iogin を使用して、アカウント履歴のレコードをフィルターします。マージ ステートメントを作成して、各ユーザー ID の最新の値を更新または挿入します。
D. Delta Lake バージョン履歴を使用して、アカウント履歴の最新バージョンと 1 つ前のバージョンの違いを取得し、これらのレコードを現在のアカウントに書き込みます。
E. 最終更新フィールドと処理された最新時間を使用してアカウント履歴のレコードをフィルターし、ユーザー名で重複を排除します。マージ ステートメントを作成して、各ユーザー名の最新の値を更新または挿入します。
Answer: C (メーッセージを残す)
説明
これは、各ユーザー ID の最新の値のみを使用してアカウントの現在のテーブルを効率的に更新するため、正しい答えです。このコードは、最終更新フィールドと処理された最新の時間を使用してアカウント履歴のレコードをフィルター処理します。つまり、最新のデータ バッチのみが処理されます。また、ユーザー ID ごとの最大最終ログイン数によってフィルター処理します。これは、そのバッチ内の各ユーザー ID の最新のレコードのみを保持することを意味します。次に、各ユーザー ID の最新の値を現在のアカウントに更新または挿入するマージ ステートメントを作成します。これは、ユーザー ID 列に基づいて更新/挿入操

作を実行することを意味します。検証済みの参照: [Databricks Certified Data Engineer Professional]、Delta Lake」セクション; Databricks ドキュメントの マージを使用したテーブルへの Upsert」セクション。

最新問題: **24**
Databricks ジョブで 2 つのタスクを設定するように求められます。最初のタスクはノートブックを実行してリモート システムからデータをダウンロードし、2 番目のタスクはこのデータを処理できる DLT パイプラインです。ジョブでこれをどのように構成する予定ですか? UI
A. 1 つのジョブにノートブック タスクと DLT パイプライン タスクを含めることはできません。線形依存関係のある 2 つの異なるジョブを使用してください。
B. ジョブ UI は DTL パイプラインをサポートしていません。ジョブ UI を使用して最初のタスクをセットアップし、連続モードで実行するように DLT をセットアップします。
C. ジョブ UI は DTL パイプラインをサポートしていません。ジョブ UI を使用して最初のタスクをセットアップし、トリガー モードで実行されるように DLT をセットアップします。
D. 単一のジョブを使用してノートブックと DLT パイプラインの両方をセットアップし、線形依存関係を持つ 2 つの異なるタスクを使用できます。
E. DLT パイプラインに最初のステップを追加し、ジョブ UI でトリガー モードとして DLT パイプラインを実行します。
Answer: D (メーッセージを残す**)**
説明
答えは、単一のジョブを使用してノートブックと DLT パイプラインの両方をセットアップでき、線形依存関係を持つ 2 つの異なるタスクを使用できます。これが JOB UI です。
1.ノートブックタスクを作成する
2.DLTタスクの作成
a.ノートブックタスクを依存関係として追加します
3.完成図
ノートブックタスクを作成する
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます
DLT タスク
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます
最終ビュー
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、PowerPoint の説明が自動生成されます。
フォームの下部
フォームの先頭

最新問題: **25**
ジョブ クラスターの開始に 6 ～ 8 分かかっており、これによりジョブが時間どおりに終了するのが遅れています。クラスターの起動時間を短縮するために実行できる手順は何ですか?
A. クラスターを開始する最初のジョブの前に 2 番目のジョブをセットアップします。これにより、ジョブの開始時にクラスターがリソースを準備できるようになります。
B. クラスターの起動時間を短縮するには、代わりに汎用クラスターを使用します。
C. クラスターのサイズを小さくします。クラスターのサイズが小さくなると、クラスターの起動にかかる時間が短くなります。
D. クラスター プールを使用してジョブの起動時間を短縮します。
E. SQL エンドポイントを使用して起動時間を短縮します
Answer: (解答を表示する**)**
説明
答えは、クラスター プールを使用してジョブの起動時間を短縮することです。
クラスター プールを使用すると、新しいジョブ クラスターが作成されるときに VM をプールから取得するときに、事前に VM を予約できます。注: VM がクラスターによる使用を待機している場合、発生するコストは Azure のみです。Databricks の実行時間コストは、VM がクラスターに割り当てられた場合にのみ請求されます。
ここでは、セットアップ方法といくつかのベスト プラクティスに従う方法のデモを示します。
https://www.youtube.com/watch?v=FVtITxOabxg&ab_channel=DatabricksAcademy

最新問題: **26**
user_ltvis という名前のテーブルは、さまざまなチームのデータ アナリストが使用するビューの作成に使用されます。ワークスペース内のユーザーはグループに構成され、ACL を使用したデータ アクセスの設定に使用されます。
user_ltvtable には次のスキーマがあります。
メールアドレス STRING、年齢 INT、Ltv INT
次のビュー定義が実行されます。
マーケティング グループのメンバーではないアナリストが次のクエリを実行します。
SELECT * FROM email_ltv
このクエリによって返される結果を説明するステートメントはどれですか?
A. 3 つの列が返されますが、1 つの列には 編集済み」という名前が付けられ、null 値のみが含まれます。
B. email 列と itv 列のみが返されます。email 列にはすべて null 値が含まれます。
C. email 列と ltv 列は、ユーザー itv の値とともに返されます。
D. 電子メール、年齢。ltv 列はユーザー ltv の値とともに返されます。
E. email 列と ltv 列のみが返されます。電子メール列には文字列が含まれます

各行に 編集済み」と表示されます。
Answer: E (メーッセージを残す)
このコードは、user_ltv というテーブルから email 列と ltv 列を選択する email_ltv というビューを作成します。このテーブルには、email STRING、age INT、ltv INT というスキーマがあります。また、このコードでは、ユーザーがマーケティング グループのメンバーではない場合、CASE WHEN 式を使用して電子メールの値を文字列 REDACTED」に置き換えます。クエリを実行するユーザーはマーケティング グループのメンバーではないため、電子メール列と ltv 列のみが表示され、電子メール列の各行には文字列 REDACTED」が含まれます。
検証済みの参考文献: [Databricks Certified Data Engineer Professional]、 Lakehouse」セクションの下。Databricks ドキュメントの CASE 式」セクション。

最新問題**: 27**
実稼働用に Structured Streaming ジョブをスケジュールする場合、クエリの失敗から自動的に回復し、コストを低く抑える構成はどれですか?
A. クラスター: 新しいジョブ クラスター。
再試行: 無制限。
最大同時実行数: 無制限
B. クラスター: 新しいジョブ クラスター。
再試行: なし。
最大同時実行数: 1
C. クラスター: 既存の汎用クラスター。
再試行: 無制限。
最大同時実行数: 1
D. クラスター: 既存の汎用クラスター。
再試行: 無制限。
最大同時実行数: 1
E. クラスター: 既存の汎用クラスター。
再試行: なし。
最大同時実行数: 1
Answer: (解答を表示する)
クエリの失敗から自動的に回復し、コストを低く抑える構成では、新しいジョブ クラスターを使用し、再試行回数を無制限に設定し、最大同時実行数を 1 に設定します。この構成には次の利点があります。
* 新しいジョブ クラスターは、ジョブの実行ごとに作成および終了されるクラスターです。つまり、クラスター リソースはジョブの実行中にのみ使用され、アイドル コストは発生しません。これにより、クラスターが常にクリーンな状態に保たれ、job1 の最新の構成とライブラリが保持されるようになります。

* 再試行を無制限に設定すると、ネットワークの問題、ノード障害、一時的なエラーなどの障害が発生した場合に、ジョブが自動的にクエリを再開することを意味します。これにより、ストリーミング ジョブの信頼性と可用性が向上し、データの損失や不整合が回避されます2。
* 最大同時実行を 1 に設定すると、一度に実行できるジョブのインスタンスは 1 つだけになります。これにより、複数のクエリが同じリソースをめぐって競合したり、同じ出力場所に書き込んだりすることがなくなり、パフォーマンスの低下やデータ破損が発生する可能性があります3。
したがって、この構成は、ジョブの回復力、効率性、一貫性を保証するため、実稼働用の Structured Streaming ジョブをスケジュールするためのベスト プラクティスです。
参照: ジョブ クラスター、ジョブの再試行、最大同時実行数

最新問題**: 28**
以下の 2 つのコマンドの主な違いは何ですか?
1.INSERT OVERWRITE テーブル名
2.SELECT * FROMテーブル
1.テーブルの作成または置換 table_name
2.AS SELECT * FROM テーブル
A. INSERT OVERWRITE はデフォルトでデータを置き換え、CREATE OR REPLACE はデフォルトでデータとスキーマを置き換えます
B. INSERT OVERWRITE はデフォルトでデータとスキーマを置き換え、CREATE OR REPLACE はデフォルトでデータを置き換えます
C. INSERT OVERWRITE はデフォルトで履歴データのバージョンを維持し、CREATE OR REPLACE はデフォルトで履歴データのバージョンをクリアします
D. INSERT OVERWRITE はデフォルトで履歴データのバージョンをクリアし、CREATE OR REPLACE はデフォルトで履歴データのバージョンを維持します
E. 両方とも同じであり、同じ結果になります。
Answer: A (メーッセージを残す**)**
説明
INSERT OVERWRITE と CREATE OR REPLACE TABLE(CRAS) の主な違いは、CRAS がテーブルのスキーマを変更できること、つまり、新しい列を追加したり、既存の列のデータ型を変更したりできることです。デフォルトでは、INSERT OVERWRITE はデータのみを上書きします。
次の場合に限り、INSERT OVERWRITE を使用してスキーマを上書きすることもできます。
このオプションが有効になっておらず、スキーマの不一致がある場合、コマンドは失敗する場合、spark.databricks.delta.schema.autoMerge.enabled は true に設定されます。

最新問題**: 29**

JDBC を使用して SQLite データベースのローカル インスタンスに接続し、外部 SQL テーブルを定義します。
A. 1.CREATE TABLE users_jdbc
2.SQLITEの使用
3.オプション(
4. URL = "jdbc:/sqmple_db",
5. dbtable = "ユーザー"
6.)
B. 1.CREATE TABLE users_jdbc
2.SQLの使用
3.URL = {サーバー:"jdbc:/sqmple_db",dbtable: "ユーザー"}
C. 1.CREATE TABLE users_jdbc
2.SQLの使用
3.オプション(
4. URL = "jdbc:sqlite:/sqmple_db",
5. dbtable = "ユーザー"
6.)
D. 1.CREATE TABLE users_jdbc
2. org.apache.spark.sql.jdbc.sqlite の使用
3.オプション(
4. URL = "jdbc:/sqmple_db",
5. dbtable = "ユーザー"
6.)
E. 1.CREATE TABLE users_jdbc
2. org.apache.spark.sql.jdbc の使用
3.オプション(
4. URL = "jdbc:sqlite:/sqmple_db",
5. dbtable = "ユーザー"
6.)
Answer: D (メーッセージを残す**)**
説明
答えは、
1.テーブルusers_jdbcの作成
2. org.apache.spark.sql.jdbc の使用
3.オプション(
4. URL = "jdbc:sqlite:/sqmple_db",
5. dbtable = "ユーザー"
6.)

Databricks ランタイムは現在、JDBC を使用した SQL Server、My SQL、SQL Lite、Snowflake などの SQL Database のいくつかのフレーバーへの接続をサポートしています。
1.CREATE TABLE <jdbcTable>
2.org.apache.spark.sql.jdbc または JDBC の使用
3.オプション(
4. URL = "jdbc:<データベースサーバータイプ>://<jdbcホスト名>:<jdbcポート>",
5. dbtable " = <jdbcDatabase>.atable",
6. ユーザー = "<jdbcユーザー名>",
7. パスワード = "<jdbcパスワード>"
8.)
より詳細なドキュメントについては
JDBC を使用した SQL データベース - Azure Databricks | Microsoft ドキュメント

最新問題: **30**
マーケティング チームに新しく参加したチーム メンバーの John Smith は現在、販売テーブルへの読み取りアクセス権を持っていますが、テーブルを更新するアクセス権を持っていません。これを達成するのに役立つコマンドは次のどれですか?
A. テーブル table_name の更新を john.smith@marketing.com に許可します
B. テーブル table_name の使用を john.smith@marketing.com に許可します
C. テーブル table_name に対する変更を john.smith@marketing.com に許可します
D. john.smith@marketing.com のテーブル table_name に更新を許可します
E. john.smith@marketing.com のテーブル table_name に変更を許可します
Answer: (解答を表示する**)**
説明
答えは GRANT MODIFY ON TABLE table_name TO john.smith@marketing.com です。
https://docs.microsoft.com/en-us/azure/databricks/security/access-control/table-acls/object-privileges#privileges

最新問題: **31**
Databricks ワークスペース管理者は、データ エンジニアリング グループごとに対話型クラスターを構成しました。コストを制御するために、クラスターは 30 分間非アクティブ状態が続いた後に終了するように設定されています。各ユーザーは、割り当てられたクラスターに対して 1 日中いつでもワークロードを実行できる必要があります。
ユーザーがワークスペースに追加されているが、権限が付与されていないと仮定すると、ユーザーが既に構成されているクラスターを起動して接続するために必要な最小限の権限を説明するものは次のどれですか。
A. 必要なクラスターに対する 管理可能」権限
B. ワークスペース管理者権限、クラスターの作成が許可されています。必要なクラスターに対する 接続可能」権限

C. クラスターの作成が許可されます。必要なクラスターに対する「接続可能」権限
D. 必要なクラスターに対する「再起動可能」権限
E. クラスターの作成が許可されます。必要なクラスターに対する「再起動可能」権限
Answer: (解答を表示する**)**
https://learn.microsoft.com/en-us/azure/databricks/security/auth-authz/access-control/cluster-acl
https://docs.databricks.com/en/security/auth-authz/access-control/cluster-acl.html

最新問題**: 32**
DBFS の場所「dbfs:/mnt/delta/databases/sales.db/」を使用して販売データベースを作成します。
A. CREATE DATABASE sales FORMAT DELTA LOCATION 'dbfs:/mnt/delta/databases/sales.db/''
B. LOCATION 'dbfs:/mnt/delta/databases/sales.db/' を使用してデータベース sales を作成します
C. CREATE DATABASE sales LOCATION 'dbfs:/mnt/delta/databases/sales.db/'
D. 販売データベースはデルタ湖でのみ作成できます
E. デルタデータベース販売場所「dbfs:/mnt/delta/databases/sales.db/」を作成します。
Answer: D (メーッセージを残す**)**
説明
答えは
データベース販売場所「dbfs:/mnt/delta/databases/sales.db/」を作成します
注: Unity カタログと 3 層の名前空間の導入により、SCHEMA と DATABASE の使用は交換可能になります。

最新問題**: 33**
ユーザーのアクセスに基づいてデルタ テーブルの行と列に対するきめ細かいアクセス制御を実装するために使用できる手法は次のどれですか?

A. Unity カタログを使用して行と列へのアクセスを許可します
B. 行および列のアクセス制御リスト
C. 動的ビュー関数を使用する
D. データ アクセス制御リスト
E. Unity カタログを使用した動的アクセス制御リスト
Answer: C (メーッセージを残す**)**
説明
答えは、ダイナミック ビュー機能を使用することです。
これは、ユーザーがマネージャー グループに属していることに基づいて行へのアクセスを制限する例です。ユーザーがマネージャー グループのメンバーではない場合、以下のビューでは、合計金額が 1000000 以下の行のみが表示されます。動的ビュー機能により、行をフィルタリングする
1.CREATE VIEW sales_redacted AS
2.user_id、国、製品、合計を選択します
3.FROM販売_生
4.WHERE CASE WHEN is_member('managers') THEN TRUE ELSE 合計 <= 1000000 END; ユーザーのアクセスに基づいて列データを非表示にするダイナミックビュー機能、
1.CREATE VIEW sales_redacted AS
2.ユーザーIDを選択し、
3. CASE WHEN is_member('auditors') THEN 電子メール ELSE 'REDACTED' END AS email、
4.国、
5.製品、
6.合計
7.FROM販売_生
詳細については以下をご覧ください
https://docs.microsoft.com/en-us/azure/databricks/security/access-control/table-acls/object-privileges#dynamic-v

最新問題**: 34**
ほぼリアルタイムのワークロードを促進するために、データ エンジニアは、Databricks Auto Loader のスキーマ検出および展開機能を活用するヘルパー関数を作成しています。目的の関数は、ソースのスキーマを直接自動的に検出し、JSON ファイルがソース ディレクトリに到着すると段階的に処理し、新しいフィールドが検出されたときにテーブルのスキーマを自動的に進化させます。
関数は以下に空白で表示されます。
指定された要件を満たすために空白を正しく埋める応答はどれですか?
A. オプション A
B. オプション B
C. オプション C

D. オプション D
E. オプション E
Answer: B (メーッセージを残す)
オプション B は、指定された要件を満たすために空白を正しく埋めます。オプション B では、
「cloudFiles.schemaLocation」オプション。Databricks Auto Loader のスキーマ検出および展開機能に必要です。さらに、オプション B では、Databricks Auto Loader のスキーマ進化機能に必要な「mergeSchema」オプションを使用します。最後に、オプション B では、「writeStream」メソッドを使用します。これは、JSON ファイルがソース ディレクトリに到着したときの増分処理に必要です。他のオプションは、必要なオプションが省略されているか、間違った方法が使用されているか、または間違った形式が使用されているため、正しくありません。参考文献:
* Auto Loader でスキーマの推論と展開を構成します。
https://docs.databricks.com/en/ingestion/auto-loader/schema.html
* ストリーミング データの書き込み:
https://docs.databricks.com/spark/latest/structord-streaming/writing-streaming-data.html

最新問題**: 35**
Spark へのデータの取り込み時に Spark パーティションのサイズに直接影響する構成パラメーターはどれですか?
A. spark.sql.files.maxPartitionBytes
B. spark.sql.autoBroadcastJoinThreshold
C. spark.sql.files.openCostInBytes
D. spark.sql.adaptive.coalescePartitions.minPartitionNum
E. spark.sql.adaptive.advisoryPartitionSizeInBytes
Answer: A (メーッセージを残す)
これは正しい答えです。spark.sql.files.maxPartitionBytes は、Spark へのデータの取り込み時に Spark パーティションのサイズに直接影響する構成パラメーターであるためです。このパラメータは、Parquet、JSON、ORC などのファイルベースのソースからファイルを読み取るときに、単一のパーティションにパックする最大バイト数を構成します。デフォルト値は 128 MB です。これは、小さなファイルが多すぎるか、大きなファイルが 1 つしかない限り、各パーティションのサイズはおよそ 128 MB になることを意味します。検証済みの参照: [Databricks Certified Data Engineer Professional]、「Spark 構成」セクション; Databricks ドキュメントの「利用可能なプロパティ -spark.sql.files.maxPartitionBytes」セクション。

最新問題**: 36**
特定の雑誌の月間購読者数の合計を予測するモデルを作成するように求められます。

1 年分のサブスクリプションおよび支払いデータ、ユーザー人口統計データ、および 10 年分のデータが提供されます。
雑誌のコンテンツ（記事と写真）の価値。どのアルゴリズムが構築に最も適しているか
加入者のための予測モデル?
A. ロジスティック回帰
B. デシジョン ツリー
C. 線形回帰
D. TF-IDF
Answer: C (メーッセージを残す**)**

最新問題**: 37**
CSV、JSON、TEXT、BINARY 形式を使用して外部テーブルを定義する場合、外部テーブルに対するクエリはパフォーマンス上の理由からデータと場所をキャッシュするため、特定の Spark セッション内では、到着した可能性のある新しいファイルは最初のセッション後に使用できなくなります。クエリ。この制限にどう対処すればよいでしょうか?
A. UNCACHE TABLE table_name
B. キャッシュ テーブル テーブル名
C. REFRESH TABLE テーブル名
D. ブロードキャスト テーブル table_name
E. CLEAR CACH テーブル名
Answer: C (メーッセージを残す**)**
説明
答えは REFRESH TABLE table_name です。
REFRESH TABLE table_name は、Spark に外部ファイルの可用性と変更を強制的に更新させます。
Spark が外部テーブルにクエリを実行すると、それに関連付けられたファイルがキャッシュされるため、テーブルが再度クエリされた場合にキャッシュされたファイルを使用できるため、クラウド オブジェクト ストレージから再度取得する必要がなくなります。ただし、ここでの欠点は、新しいファイルが利用可能かどうかは、Refresh コマンドが実行されるまで Spark にはわかりません。

最新問題**: 38**
運用チームは集中型のデータ品質監視システムを使用しており、ユーザーは Webhook を通じてデータ品質メトリクスを公開できます。重複レコードが少なくとも 1 つある場合に Webhook を使用してメッセージを送信するプロセスを開発するように求められました。次のアプローチのうちどれが考えられますか?現在のデータ品質監視システムとアラートを統合するために採用されました
A. ノートブックとジョブを使用して Python を使用して DQ メトリクスを公開します
B. 電子メールを送信するためのアラートを設定し、Python を使用して電子メールを解析し、Webhook メッセージを公開します。

C. カスタム テンプレートを使用してアラートをセットアップする
D. カスタム Webhook 宛先を使用してアラートをセットアップします
E. 動的テンプレートを使用してアラートをセットアップする
Answer: D (メーッセージを残す**)**
説明
アラートは複数の宛先をサポートしており、電子メールがデフォルトの宛先です。
アラートの宛先 | AWS 上のデータブリック
グラフィカル ユーザー インターフェイス、アプリケーションの説明が自動的に生成される

最新問題**: 39**
デルタ ライブ テーブル パイプラインには、STREAMING LIVE TABLE を使用して定義された 2 つのデータセットが含まれています。
LIVE TABLE を使用して、Delta Lake テーブル ソースに対して 3 つのデータセットが定義されます。テーブルは次のように構成されています
トリガーされたパイプライン モードを使用して開発モードで実行します。
以前に処理されていないデータが存在し、すべての定義が有効であると仮定すると、その後の期待される結果は何ですか?
「開始」をクリックしてパイプラインを更新しますか?
A. すべてのデータセットは、パイプラインがシャットダウンされるまで、設定された間隔で更新されます。コンピューティング リソースは、
更新のためにデプロイされ、パイプラインが停止すると終了します
B. すべてのデータセットは継続的に更新され、パイプラインはシャットダウンされません。コンピューティングリソース
パイプラインとともに存続します
C. すべてのデータセットが一度更新され、パイプラインがシャットダウンされます。コンピューティング リソースは次のようになります。
終了しました
D. すべてのデータセットは、パイプラインがシャットダウンされるまで、設定された間隔で更新されます。コンピューティング リソースは、
追加のテストを可能にするために、パイプラインが停止した後も保持されます。
E. すべてのデータセットが一度更新され、パイプラインがシャットダウンされます。コンピューティング リソースは持続します。
追加のテストを可能にする
Answer: E (メーッセージを残す**)**

最新問題**: 40**
各デバイスごとに温度センサーがしきい値温度 (100.00) を超えた回数を特定するように求められました。各行には 5 分ごとに収集された 5 つの測定値が含まれており、空白に適切な関数を入力します。
スキーマ: deviceId INT、deviceTemp ARRAY<double>、dateTimeCollected TIMESTAMP

SELECT deviceId, __ (__ (__(deviceTemp], i -> i > 100.00)))
FROMデバイス
GROUP BY デバイス ID
A. 合計、カウント、サイズ
B. 合計、サイズ、スライス
C. SUM、SIZE、ARRAY_CONTAINS
D. 合計、サイズ、ARRAY_FILTER
E. 合計、サイズ、フィルター
Answer: (解答を表示する**)**
説明
FILER 関数を使用すると、式に基づいて配列をフィルタリングできます
SIZE関数は配列のサイズを取得するために使用できます
SUM はデバイスごとの合計を計算するために使用されます
自動生成される図の説明

最新問題**: 41**
あなたは現在、データ パイプラインの構築に取り組むよう求められています。現在、多くのデータ依存関係を伴う非常に大規模な ETL に取り組んでいることに気づきました。この問題に対処するために使用できるツールは次のうちどれですか?
A. オートローダー
B. ジョブとタスク
C. SQL エンドポイント
D. デルタ ライブ テーブル
E. マルチホップを使用した構造化ストリーミング
Answer: (解答を表示する**)**
説明
答えは、DELTA LIVE TABLES です。
DLT は、ライブ テーブル間に DAG ベースの結合を構築することにより、データの依存関係を簡素化します。以下は、追加のメタデータなしでデータ依存関係を持つ DAG がどのように見えるかを示しています。
1.ライブビュー顧客を作成または置き換える
2.顧客から * を選択します。
3.
4.ライブビュー sales_orders_raw を作成または置き換えます
5. sales_orders から * を選択します。
6.
7.ライブビュー sales_orders_cleaned を作成または置き換えます
8.as
9.販売.*から選択してください
10.live.sales_orders_raw

11. ライブに参加する.顧客 c
12.on c.customer_id = s.customer_id
13.ここで、c.city = 'LA';
14.
15.ライブテーブル sales_orders_in_la を作成または置換する
16. sales_orders_cleaned から選択します。
上のコードは以下の DAG を作成します
DELTA LIVE TABLES に関するドキュメント、
https://databricks.com/product/delta-live-tables
https://databricks.com/blog/2022/04/05/payment-generally-availability-of-databricks-delta-live-tables-dlt.htm DELTA LIVE TABLES は、ETL プロセスを構築する際の以下の課題に対処します
1.大規模ETLの複雑さ
a.依存関係の構築と維持が難しい
b.バッチとストリームの切り替えが難しい
2.データ品質とガバナンス
a.データ品質の監視と強制が難しい
b.データ系統を追跡することが不可能
3.難しいパイプライン運用
a.粒度の高いデータ レベルでの可観測性が低い
b.エラー処理と回復に手間がかかる

最新問題: **42**
あなたは現在、さまざまな顧客調査から受け取ったデータの保存に取り組んでいますが、このデータは非常に構造化されておらず、時間の経過とともに変化します。なぜデータ ウェアハウスと比較して Lakehouse がより良い選択肢なのでしょうか?
A. Lakehouse はデータの整合性を強制します
B. Lakehouse は ACID をサポートします
C. Lakehouse はスキーマの適用と進化をサポートしていますが、従来のデータ ウェアハウスにはスキーマの進化がありません。
D. Lakehouse は SQL をサポートします
E. Lakehouse はデータ ウェアハウスのような主キーと外部キーをサポートします
Answer: C (メッセージを残す**)**

最新問題: **43**
デルタ レイク テーブルに適用できる次のテーブル制約のうち、サポートされているものはどれですか?
A. 主キー、外部キー、Not Null、チェック制約
B. 主キー、NULL ではない、制約をチェック
C. デフォルト、NULL ではない、チェック制約

D. Null ではありません。制約を確認してください
E. 一意、NULL ではない、チェック制約
Answer: D (メーッセージを残す)
説明
答えは「NULL ではありません。制約を確認してください」です。
https://docs.microsoft.com/en-us/azure/databricks/delta/delta-constraints
* CREATE TABLE events(id LONG,
* 日付STRING、
* 場所STRING、
* 説明文字列
*) デルタを使用します。
ALTER TABLE イベント CHANGE COLUMN ID SET NOT NULL;
ALTER TABLE events ADD CONSTRAINT dateWithinRange CHECK (date > '1900-01-01'); 注: DBR 11.1 の Databricks では、Unity カタログが有効な場合の主キーと外部キーのサポートが追加されましたが、これは情報提供のみを目的としており、実際には強制されません。では、強制されていないのになぜこれらを定義するのかと疑問に思われるかもしれません。そのため、特にこれらの情報制約は、テーブル間の関係を知ることで恩恵を受ける BI ツールを使用している場合に非常に役立ち、レポート/ダッシュボードの作成が容易になります。または、データ モデリング ツールを使用する場合のデータ モデルの理解。
主キーと外部キー
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます

最新問題**: 44**
データ エンジニアは次のクエリを作成しました。
1. *を選択します
2. json.`/path/to/json/file.json` から;
データ エンジニアは、このクエリをデルタ ライブ テーブル (DLT) で使用できるように変換するために同僚に協力を求めます。
パイプライン。クエリは DLT パイプラインに最初のテーブルを作成する必要があります。
同僚がクエリに対して行う必要がある変更を説明しているものは次のうちどれですか?
A. ライブを追加する必要があります。json の前にプレフィックスを付けます。FROM行内
B. クエリの先頭に CREATE LIVE TABLE table_name AS 行を追加する必要があります
C. クエリの先頭に COMMENT 行を追加する必要があります。
D. Cloud_files(...) ラッパーを JSON ファイル パスに追加する必要があります
E. クエリの先頭に CREATE DELTA LIVE TABLE table_name AS 行を追加する必要があります
Answer: B (メーッセージを残す)

最新問題: **45**
最初のサイコロが 6 の場合、2 つのサイコロの合計が 8 より大きくなる確率はいくらですか?
A. 1/6
B. 2/3
C. 1/3
D. 2/6
Answer: B (メーッセージを残す**)**

最新問題: **46**
あなたは、同僚が以前のバージョンを保存するために _bkp を使用してノートブックを手動でコピーしていることに気づきました。代わりに次の機能のうちどれをお勧めしますか。
A. Databricks ノートブックは変更の追跡とバージョン管理をサポートしています
B. Databricks ノートブックをローカル マシンにコピーし、ソース管理をローカルでセットアップしてノートブックをバージョン管理する必要があります。
C. Databricks ノートブックは dbc アーカイブ ファイルにエクスポートし、データ レイクに保存できます。
D. Databricks ノートブックは HTML としてエクスポートし、後でインポートできます。
Answer: A (メーッセージを残す**)**
説明
答えは、Databricks ノートブックは自動変更追跡とバージョン管理をサポートしているということです。
右側でノートブックを編集しているときは、バージョン履歴を確認してすべての変更を表示し、加えたすべての変更がキャプチャされて保存されます。

最新問題: **47**

ジュニア データ エンジニアは、Spark がデータとデータの両方を管理する Spark SQL テーブル my_table を作成する必要があります。
メタデータ。メタデータとデータも Databricks ファイルシステム (DBFS) に保存する必要があります。
上級データ エンジニアがジュニア データ エンジニアと共有すべきコマンドは次のどれですか。
このタスクを完了しますか?
A. 1. CREATE TABLE my_table (id STRING, value STRING) USING
2. org.apache.spark.sql.parquet オプション (PATH "storage-path")
B. 1. CREATE TABLE my_table (ID STRING、値 STRING);
C. 1. 管理テーブル my_table (ID STRING、値 STRING) USING を作成します。
2. org.apache.spark.sql.parquet オプション (PATH "storage-path");
D. 1. 管理テーブル my_table を作成します (ID STRING、値 STRING);
E. 1. DBFS を使用してテーブル my_table (ID STRING、値 STRING) を作成します。
Answer: B (メーッセージを残す**)**

最新問題**: 48**
AUTO LOADER を使用して 1 日に何百万ものファイルを処理していましたが、読み込みプロセスの遅さに気づき、Databricks クラスターをスケールアップしましたが、Auto Loader のパフォーマンスがまだ向上していないことに気づきました。これを解決する最善の方法は何ですか。
A. AUTO LOADER は、1 日に何百万ものファイルを処理するのには適していません
B. データを処理するための 2 番目の AUTO LOADER プロセスをセットアップします。
C. maxFilesPerTrigger オプションを十分な数に増やします。
D. アクセスを高速化するために、クラウド ストレージからクラスター上のローカル ディスクにデータをコピーします。
E. ファイルを 1 つの大きなファイルに結合します
Answer: (解答を表示する**)**
説明
maxFilesPerTrigger のデフォルト値は 1000 で、これよりもさらに大きな値に増やすことができますが、処理にはさらに大規模なコンピューティングが必要になります。
グラフィカル ユーザー インターフェイス、テキスト、アプリケーション、電子メールの説明が自動生成されます
https://docs.databricks.com/ingestion/auto-loader/options.html

最新問題**: 49**
COPY INTO コマンドを利用して外部 CSV ファイルを差分テーブルにロードするプロセスに取り組んでいますが、コマンドを 2 回目に実行した後、テーブル名にデータがロードされませんでした。これはなぜですか?
1.COPY INTO テーブル名

2. 「dbfs:/mnt/raw/*.csv」から
3.ファイル形式 = CSV
A. COPY INTO はデータのロードを 1 回のみ実行します
B. COPY INTO を実行する前に REFRESH TABLE 販売を実行します。
C. COPY INTO は最後のロード後に新しいファイルを検出しませんでした
D. 新しいファイルをロードするには、incremental = TRUE オプションを使用します。
E. COPY INTO は増分ロードをサポートしていません。AUTO LOADER を使用してください
Answer: C (メーッセージを残す)
説明
答えは、COPY INTO が最後のロード後に新しいファイルを検出しなかったことです。
COPY INTO はテーブルに正常にロードされたファイルを追跡し、次回 COPY INTO を実行するときにそれらのファイルをスキップします。
参考までに、COPY_OPTIONS 'force'= 'true' を使用してこの動作を変更できます。このオプションが有効な場合、パス/パターン内のすべてのファイルがロードされます。
1.COPY INTO テーブル識別子
2. FROM [ファイルの場所 | (ファイルの場所から識別子リストを選択)]
3. ファイル形式 = データソース
4. [FILES = [ファイル名, ... | PATTERN = '正規表現パターン']
5. [FORMAT_OPTIONS ('data_source_reader_option' = 'value', ...)]
6. [COPY_OPTIONS 'force' = ('false'|'true')]

最新問題**: 50**
実稼働ワークロードは、外部変更データ キャプチャ フィードからの更新を、常時稼働の構造化ストリーム ジョブとして Delta Lake テーブルに段階的に適用します。このテーブルのデータが最初に移行されたとき、OPTIMIZE が実行され、ほとんどのデータ ファイルのサイズが 1 GB に変更されました。ストリーミング制作ジョブでは、自動最適化と自動圧縮の両方がオンになりました。データ ファイルの最近のレビューによると、テーブル内の各パーティションには少なくとも 1 GB のデータが含まれており、テーブルの合計サイズは 10 TB を超えていますが、ほとんどのデータ ファイルは 64 MB 未満です。ファイル サイズが小さくなる理由
として適切なものは次のうちどれですか?
A. Databricks は、MERGE 操作の時間を短縮するために、より小さいターゲット ファイル サイズに自動調整されました。
B. テーブルで計算された Z オーダー インデックスがファイルの圧縮を妨げています。
テーブルで計算された C ブルーム フィラー インデックスがファイルの圧縮を妨げています。
C. Databricks は、テーブル内のデータ全体のサイズに基づいて、より小さいターゲット ファイル サイズに自動調整されました。

D. Databricks は、各パーティション内のデータ量に基づいて、より小さいターゲット ファイル サイズに自動調整されました。
Answer: (解答を表示する**)**
これは正解です。Databricks には自動最適化と呼ばれる機能があり、小さなファイルを大きなファイルに結合し、各ファイル内のデータを指定された列で並べ替えることで、Delta Lake テーブルのレイアウトを自動的に最適化します。ただし、自動最適化では、ファイル サイズとマージ パフォーマンスの間のトレードオフも考慮され、特に既存のレコードを頻繁に更新するストリーミング ワークロードの場合、マージ操作の時間を短縮するために、より小さいターゲット ファイル サイズが選択される場合があります。したがって、ストリーミング制作ジョブの特性に基づいて、自動最適化がより小さな目標ファイル サイズに自動調整した可能性があります。検証済みの参照: [Databricks Certified Data Engineer Professional]、「Delta Lake」セクション; Databricks ドキュメントの「自動最適化」セクション。
https://docs.databricks.com/en/delta/tune-file-size.html#autotune-table 「ワークロードに基づいてファイル サイズを自動調整する」

最新問題**: 51**
データ エンジニアは、それぞれ夜間に実行される 2 つのジョブを設定しました。最初のジョブは午前 12:00 に開始され、通常は
約20分で完了します。2 番目のジョブは最初のジョブに依存しており、午前 12 時 30 分に開始されます。時々、
最初のジョブが午前 12:30 までに完了しない場合、2 番目のジョブは失敗します。
データ エンジニアがこの問題を回避するために使用できるアプローチは次のうちどれですか?
A. 線形依存関係を持つ 1 つのジョブで複数のタスクを利用できます。
B. 最初のジョブに再試行ポリシーを設定して、ジョブをより迅速に実行できるようにします。
C. クラスター プールを使用して、ジョブをより効率的に実行できます。
D. 最初のジョブから 2 番目のジョブにストリーミングするデータを設定できます。
E. 2 番目のジョブの出力サイズを制限して、簡単に失敗しないようにすることができます。
Answer: A (メッセージを残す**)**

最新問題**: 52**
以下の各構成は、各クラスターに合計 400 GB の RAM、合計 160 個のコア、および VM ごとに 1 つのエグゼキューターしかない点で同一です。
少なくとも 1 つの幅広い変換を含むジョブの場合、次のクラスター構成のうちどれが最大のパフォーマンスをもたらしますか?
A. * VM の合計数。1

* エグゼキューターあたり 400 GB
* 160 コア/エグゼキュータ
B. * 合計 VM: 8
* エグゼキュータあたり 50 GB
* 20 コア/エグゼキュータ
C. * 合計 VM 数: 4
* エグゼキューターあたり 100 GB
* 40 コア/エグゼキュータ
D. * 合計 VM 数:2
* エグゼキューターあたり 200 GB
* 80 コア/エグゼキュータ
Answer: (解答を表示する**)**
説明
これは正しい答えです。少なくとも 1 つのワイド変換を含むジョブのパフォーマンスが最大になるのはクラスター構成だからです。ワイド変換は、join、groupBy、orderBy など、パーティション間でデータをシャッフルする必要がある変換のタイプです。特にパーティションが多すぎたり少なすぎたりする場合、シャッフルにはコストと時間がかかる可能性があります。したがって、並列処理とネットワーク オーバーヘッドの間のトレードオフのバランスがとれるクラスター構成を選択することが重要です。この場合、エグゼキュータあたり 50 GB、エグゼキュータあたり 20 コアを持つ 8 つの VM があると、シャッフルを効率的に処理するのに十分なメモリと CPU リソースを備えた 8 つのパーティションが作成されます。VM の数が減り、エグゼキュータあたりのメモリとコアが増えると、作成されるパーティションの数が減り、並列処理が低下し、各シャッフル ブロックのサイズが増加します。エグゼキュータあたりのメモリとコアが少なく、より多くの VM を使用すると、より多くのパーティションが作成され、並列処理が向上しますが、ネットワーク オーバーヘッドとシャッフル ファイルの数も増加します。検証済みの参照: [Databricks Certified Data Engineer Professional]、『パフォーマンス チューニング」セクション; Databricks ドキュメントの 『クラスター構成」セクション。

最新問題**: 53**
セキュリティ チームは、Databricks シークレット モジュールを外部データベースへの接続に利用できるかどうかを検討しています。
すべての Python 変数が文字列で定義されているコードをテストした後、パスワードを Secrets モジュールにアップロードし、現在アクティブなユーザーに正しい権限を構成します。次に、コードを次のように変更します (他のすべての変数は変更しないままにします)。
上記のコードが実行されると何が起こるかを説明するステートメントはどれですか?
A. 外部テーブルへの接続は失敗します。文字列 『編集済み」が出力されます。

B. インタラクティブな入力ボックスがノートブックに表示されます。正しいパスワードが指定された場合、接続は成功し、エンコードされたパスワードが DBFS に保存されます。
C. インタラクティブな入力ボックスがノートブックに表示されます。正しいパスワードが指定された場合、接続は成功し、パスワードはプレーンテキストで出力されます。
D. 外部テーブルへの接続は成功します。パスワードの文字列値はプレーンテキストで出力されます。
E. 外部テーブルへの接続は成功します。 編集済み」という文字列が出力されます。
Answer: (解答を表示する**)**
コードでは dbutils.secrets.get メソッドを使用して Secrets モジュールからパスワードを取得し、それを変数に格納しているため、これは正しい答えです。Secret モジュールを使用すると、ユーザーはパスワード、トークン、API キーなどの機密情報を安全に保存し、アクセスできます。パスワード変数には実際のパスワード値が含まれるため、外部テーブルへの接続は成功します。ただし、パスワード変数を印刷するときは、ノートブック内の機密情報の漏洩を防ぐためのセキュリティ対策として、プレーン テキストのパスワードの代わりに文字列 「redacted」が表示されます。検証済みの参照: [Databricks Certified Data Engineer Professional]、 セキュリティとガバナンス」セクション; Databricks ドキュメントの以下 「秘密」セクション。

最新問題**: 54**
セッションスコープの一時ビューに関する正しいステートメントは次のうちどれですか?
A. ノートブックを切り離して再接続すると、一時ビューは失われます。
B. メモリに保存された一時ビュー
C. ノートブックがデタッチされてアタッチされていても、一時ビューには引き続きアクセスできます。
D. クラスターが再起動されても、一時ビューには引き続きアクセスできます
E. 一時ビューは local_temp データベースに作成されます
Answer: A (メーッセージを残す**)**
説明
答えは、ノートブックをデタッチしてアタッチすると、一時ビューは失われます。作成できる一時ビューには、セッション スコープとグローバルの 2 種類があります。
*ローカル/セッション スコープの一時ビューは Spark セッションでのみ使用できるため、同じクラスター内の別のノートブックはアクセスできません。ノートブックが切り離され、再接続された場合、ローカルの一時ビューが失われます。
*クラスターが再起動してグローバル一時ビューが失われた場合、グローバル一時ビューはクラスター内のすべてのノートブックで使用できます。

最新問題**: 55**

データ ガバナンス チームは、GDPR に準拠するためにレコードの削除に使用されるコードをレビューしています。彼らは、users という名前の Delta Lake テーブルからレコードを削除するために次のロジックが使用されていることを指摘しています。
user_id が一意の識別キーであり、削除を要求したすべてのユーザーが含まれていると仮定すると、上記のロジックを正常に実行すると、削除されるレコードにアクセスできなくなることが保証されるかどうか、またその理由を説明するステートメントはどれですか?
A. はい。Delta Lake ACID 保証は、削除コマンドが成功してこれらのレコードを完全かつ永久にパージしたことを保証します。
B. いいえ。デルタ キャッシュは、クラスターが再起動されるまで、テーブルの以前のバージョンからレコードを返す場合があります。
C. はい。デルタ キャッシュはすぐに更新され、ディスクに記録された最新のデータ ファイルが反映されます。
D. いいえ。Delta Lake delete コマンドは、mergeinto コマンドと組み合わせた場合にのみ ACID 保証を提供します。
E. いいえ。削除されたレコードを含むファイルは、バキューム コマンドを使用して無効化されたデータ ファイルを削除するまで、タイム トラベルで引き続きアクセスできる可能性があります。
Answer: E (メーッセージを残す**)**
説明
このコードは、DELETE FROM コマンドを使用して、削除を要求したすべてのユーザーを含む、delete_requests という別のテーブルとの結合に基づく条件に一致するレコードを users テーブルから削除します。
DELETE FROM コマンドは、削除されたレコードを含まない新しいバージョンのテーブルを作成することにより、Delta Lake テーブルからレコードを削除します。ただし、Delta Lake はタイム トラベルをサポートしており、タイムスタンプまたはバージョン番号を使用してテーブルの以前のバージョンをクエリできるため、これは削除されるレコードにアクセスできなくなることを保証するものではありません。したがって、バキューム コマンドを使用して無効化されたデータ ファイルを物理ストレージから削除するまでは、削除されたレコードを含むファイルにタイム トラベルで引き続きアクセスできる可能性があります。
検証済みの参照: [Databricks Certified Data Engineer Professional]、 Delta Lake」セクション; Databricks ドキュメントの テーブルからの削除」セクション。Databricks ドキュメントの デルタ テーブルによって参照されなくなったファイルの削除」セクション。

最新問題: **56**
エンジニアリング マネージャーは、Databricks SQL クエリを使用して、次の点に関連する修正に関するチームの進捗状況を監視します。
顧客から報告されたバグ。マネージャーは毎日クエリの結果をチェックしますが、それらは手動で行われます。

毎日クエリを再実行し、結果を待ちます。
マネージャーはクエリの結果が毎回確実に更新されるようにするには、次のどのアプローチを使用できますか?
日？
A. Databricks SQL のクエリのページから 1 日ごとにクエリを更新するようにスケジュールできます。
B. Databricks SQL の SQL エンドポイントのページから 12 時間ごとにクエリを更新するようにスケジュールできます。
C. ジョブ UI からクエリを 1 日ごとに実行するようにスケジュールできます。
D. ジョブ UI から 12 時間ごとにクエリを実行するようにスケジュールできます。
E. Databricks SQL の SQL エンドポイントのページからクエリを 1 日ごとに更新するようにスケジュールできます。
Answer: (解答を表示する**)**

最新問題**: 57**
AUTO LOADER にスキーマの場所が必要なのはなぜですか?
A. スキーマの場所は、ユーザーが指定したスキーマを保存するために使用されます。
B. スキーマの場所は、ターゲット テーブルのスキーマを識別するために使用されます。
C. AUTO LOADER はスキーマの進化をサポートしているため、スキーマの場所を必要としません。
D. スキーマの場所は、AUTO LOADER によって推論されたスキーマを保存するために使用されます。
E. スキーマの場所は、ターゲット テーブルとソース テーブルのスキーマを識別するために使用されます。
Answer: (解答を表示する**)**
説明
答えは、スキーマの場所は AUTO LOADER によって推論されたスキーマを保存するために使用されるため、次回 AUTO LOADER がより高速に実行されるため、最後に知られているスキーマを使用しようとするたびにスキーマを推論する必要がなくなります。
Auto Loader は、最初に検出した 50 GB または 1000 ファイルのいずれか最初に制限を超えた方をサンプリングします。ストリームの起動ごとにこの推論コストが発生するのを回避し、ストリームの再起動後も安定したスキーマを提供できるようにするには、オプションcloudFiles.schemaLocationを設定する必要があります。Auto Loader は、入力データに対するスキーマの変更を経時的に追跡するために、この場所に隠しディレクトリ _schemas を作成します。
以下のリンクには、さまざまなオプションに関する詳細なドキュメントが含まれています
オートローダーのオプション | AWS 上のデータブリック

最新問題**: 58**

次のコマンドのうち、重複する既存のデルタ テーブル my_table からレコードを返すものはどれですか。
削除されましたか？
A. 1. my_table a にマージします。
2. new_records bの使用;
B. 1. 選択 *
2. my_table から
3. WHERE 重複 = False;
C. 1. 重複を削除します
2. my_table から;
D. 1. my_table a にマージします。
2. new_records b ON a.id = b.id の使用
3. 一致しない場合
4. 次に * を挿入します。
E. 1. SELECT DISTINCT *
2. my_table から;
Answer: (解答を表示する**)**

最新問題**: 59**
データ レイクハウスの次の機能のうち、両方のワークロードのニーズを満たすのに役立つものはどれですか?
A. データ レイクハウスでは、データ モデリングはほとんど必要ありません。
B. データ レイクハウスは、コンピューティングとストレージを組み合わせてシンプルなガバナンスを実現します。
C. データ レイクハウスは、コンピューティング クラスターの自動スケーリングを提供します。
D. データ レイクハウスは非構造化データを保存し、ACID トランザクションをサポートできます。
E. データ レイクハウスは完全にクラウドに存在します。
Answer: D (メーッセージを残す**)**
説明
答えは、データ レイクハウスには非構造化データが保存され、ACID に準拠しています。

最新問題**: 60**
次の構造化ストリーミング クエリのうち、ブロンズ テーブルからシルバー テーブルへのホップを実行しているのはどれですか?
A. 1.(spark.table("sales").groupBy("store")
2..agg(sum("売上")).writeStream
3..option("チェックポイントの場所",チェックポイントのパス)
4..outputMode("完全")

5..table("集計売上"))
B. 1.(spark.table("売上").agg(sum("売上"),sum("単位"))
2..writeStream
3..option("チェックポイントの場所",チェックポイントのパス)
4..outputMode("完全")
5..table("集計売上"))
C. 1.(spark.table("売上")
2..withColumn("平均価格",col("売上高") /col("単位"))
3..writeStream
4..option("チェックポイントの場所", チェックポイントパス)
5..outputMode("追加")
6..table("cleanedSales"))
D. 1.(spark.readStream.load(rawSalesLocation)
2..writeStream
3..option("チェックポイントの場所", チェックポイントパス)
4..outputMode("追加")
5..table("uncleanedSales"))
E. 1.(spark.read.load(rawSalesLocation)
2..writeStream
3..option("チェックポイントの場所", チェックポイントパス)
4..outputMode("追加")
5..table("uncleanedSales"))
Answer: C (メーッセージを残す**)**
説明
低信頼度で自動的に生成された家の説明図

最新問題**: 61**
データ エンジニアのユーザー A は、REST API を使用してプログラムでいくつかのジョブを作成し、新しいパイプラインを運用環境にプロモートしました。DevOps エンジニアのユーザー B は、REST API を介してジョブの実行をトリガーするように外部オーケストレーション ツールを構成しました。どちらのユーザーも、個人のアクセス トークンを使用して REST API 呼び出しを承認しました。
これらのイベントに関するワークスペース監査ログの内容を説明しているのはどれですか?
A. REST API はジョブの作成と実行のトリガーに使用されたため、これらのイベントを識別するためにサービス プリンシパルが自動的に使用されます。
B. ユーザー B が最後にジョブを構成したため、ユーザー B の ID はジョブ作成イベントとジョブ実行イベントの両方に関連付けられます。
C. これらのイベントは個別に管理されるため、ユーザー A の ID はジョブ作成イベントに関連付けられ、ユーザー B の ID はジョブ実行イベントに関連付けられます。

D. REST API はジョブの作成と実行のトリガーに使用されたため、ユーザー ID は監査ログにキャプチャされません。
E. ユーザー A がジョブを作成したため、ユーザー A の ID はジョブ作成イベントとジョブ実行イベントの両方に関連付けられます。
Answer: C (メーッセージを残す**)**
イベントは、データ エンジニアのユーザー A が、REST API を使用してプログラムでいくつかのジョブを作成することにより、新しいパイプラインを運用環境にプロモートし、DevOps エンジニアのユーザー B が、REST を介してジョブの実行をトリガーする外部オーケストレーション ツールを構成したことです。API。どちらのユーザーも、個人のアクセス トークンを使用して REST API 呼び出しを承認しました。ワークスペース監査ログは、クラスター、ジョブ、ノートブック、テーブルなどのオブジェクトの作成、更新、削除など、Databricks ワークスペース内のユーザー アクティビティを記録するログです。ワークスペース監査ログには、各アクティビティを実行したユーザーの ID、アクティビティの時間と詳細もキャプチャされます。これらのイベントは個別に管理されるため、ワークスペース監査ログでは、ユーザー A の ID はジョブ作成イベントに関連付けられ、ユーザー B の ID はジョブ実行イベントに関連付けられます。検証済みの参照: [Databricks Certified Data Engineer Professional]、以下
Databricks ワークスペース」セクション。Databricks ドキュメントの「ワークスペース監査ログ」セクション。

最新問題**: 62**
Databricks Repos がどのように CI/CD ワークフローの促進に役立つかを説明しているのは次のどれですか。
Databricks Lakehouse プラットフォーム?
A. Databricks Repos を使用して、Git 自動化パイプラインを設計、開発、トリガーできます。
B. Databricks Repos はコード変更をコミットまたはプッシュして CI/CD プロセスをトリガーできます

C. Databricks Repos は、ブランチをマージする前のプル リクエスト、レビュー、承認プロセスを容易にします。
D. Databricks Repos は単一の真実の情報源である Git リポジトリを保存できます
E. Databricks Repos は、セカンダリ Git ブランチからメイン Git ブランチに変更をマージできます。
Answer: (解答を表示する**)**

最新問題**: 63**
次のデータ ワークロードのうち、宛先としてブロンズ テーブルを利用するものはどれですか?
A. クリーンアップされたデータを集約して標準の要約統計を作成するジョブ
B. 集約されたデータをクエリして重要な分析情報をダッシュボードに公開するジョブ
C. ストリーミング ソースから生データを Lakehouse に取り込むジョブ
D. 機械学習アプリケーションの機能セットを開発するジョブ
E. タイムスタンプを人間が判読できる形式に解析してデータを強化するジョブ
Answer: C (メーッセージを残す**)**
説明
答えは、生データをストリーミング ソースから Lakehouse に取り込むジョブです。
Kafka などの生のストリーミング データ ソースから取り込まれたデータは、さらに最適化されて Silver に保存される前に、まず最初の宛先として Bronze レイヤーに保存されます。
メダリオン アーキテクチャ - Databricks
ブロンズ層:
1. 取り込まれたデータの生のコピー
2. 従来のデータレイクを置き換えます
3. 完全な未処理のデータ履歴の効率的なストレージとクエリを提供します。
4. この層ではスキーマは適用されません
試験の焦点: 下の画像を確認して、メダリオン建築における各層 (ブロンズ、シルバー、ゴールド) の役割を理解してください。各層とその目的を対象としたさまざまな質問が表示されます。
Udemy 内の一部の人が私のコンテンツをコピーしているため、ウォーターマークを追加する必要がありました。
メダリオンアーキテクチャの各層の目的

最新問題**: 64**
Databricks は、カスタム Python コード パッケージのインストールについてどのディストリビューションをサポートしていますか?
A. sbt
B. 瓶
C. ホイール
D. CRAM

E. CRAN
F. 名目
Answer: F (メーッセージを残す**)**

最新問題**: 65**
次の SQL コマンドのうち、行が存在するかどうかを確認する条件に基づいて行を挿入、更新、または削除するために使用できるものはどれですか?
A. table_name にマージします
B. テーブル名にコピーします
C. UPDATE テーブル名
D. INSERT INTO OVERWRITE テーブル名
E. EXISTS テーブル名を挿入
Answer: A (メーッセージを残す**)**
説明
ここにレビュー用の追加ドキュメントがあります。
https://docs.databricks.com/spark/latest/spark-sql/ language-manual/delta-merge-into.html
1.MERGE INTO target_table_name [target_alias]
2.source_table_reference [source_alias] の使用
3. マージ条件をオンにする
4. [WHEN MATCHED [AND 条件] THEN matched_action] [...]
5. [WHEN NOT MATCHED [AND 条件] THEN not_matched_action] [...]
6.
7.一致したアクション
8. { 削除 |
9. セットの更新 * |
10. UPDATE SET { 列 1 = 値 1 } [, ...] }
11.
12.一致しないアクション
13. { * を挿入 |
14. INSERT (column1 [, ...]) VALUES (value1 [, ...])

最新問題**: 66**
2 つのテーブルを結合してオブジェクトを作成しようとしていますが、そのテーブルはデータ サイエンティストのチームがアクセスできるため、クラスターが再起動したり、ノートブックが切り離されてもオブジェクトは削除されません。どのような種類のオブジェクトを作成しようとしていますか?
A. 一時的なビュー
B. グローバル一時ビュー
C. キャッシュ オプションを使用したグローバル一時ビュー
D. 外観図

E. ビュー
Answer: (解答を表示する**)**
説明
答えはビューです。ビューは複数のテーブルを結合するために使用できますが、他のユーザーがアクセスできるようにメタ ストアに永続化することもできます。

最新問題: **67**
データ アーキテクトは、データが外部ソースから Databricks Lakehouse に取り込まれたら、テーブル アクセス制御を利用してすべての運用テーブルとビューのアクセス許可を管理することを決定しました。
次のロジックは、実稼働データベースに対する対話型クエリの権限をコア エンジニアリング グループに付与するために実行されました。
データベース prod の使用を eng に許可します。
データベース prod への選択を eng に許可します。
これらが eng グループに付与されている唯一の権限であり、これらのユーザーがワークスペース管理者ではないと仮定すると、その権限を説明するステートメントはどれですか?
A. グループ メンバーは prod データベースに対する完全な権限を持ち、他のユーザーまたはグループに権限を割り当てることもできます。
B. グループ メンバーは、prod データベース内のすべてのテーブルを一覧表示できますが、それらのテーブルに対するクエリの結果を確認することはできません。
C. グループ メンバーは、prod データベース内のすべてのテーブルとビューをクエリおよび変更できますが、新しいテーブルやビューを作成することはできません。
D. グループ メンバーは、prod データベース内のすべてのテーブルとビューにクエリを実行できますが、データベース内で何も作成または編集することはできません。
E. グループ メンバーは、prod データベース内のすべてのテーブルとビューを作成、クエリ、および変更できますが、カスタム関数を定義することはできません。
Answer: (解答を表示する**)**
GRANT USAGE ON DATABASE prod TO eng コマンドは、eng グループに prod データベースを使用する権限を付与します。これは、eng グループがデータベース内のテーブルとビューを一覧表示し、アクセスできることを意味します。GRANT SELECT ON DATABASE prod TO eng コマンドは、eng グループに prod データベース内のテーブルおよびビューからデータを選択する権限を付与します。つまり、SQL または DataFrame API を使用してデータをクエリできることになります。
ただし、これらのコマンドは、テーブルやビューの作成、変更、削除、カスタム関数の定義など、その他の権限を eng グループに付与しません。したがって、eng グループのメンバーは、prod データベース内のすべてのテーブルとビューに対してクエリを実行できますが、データベース内で何も作成または編集することはできません。参考文献:
* データベースに対する権限を付与します。

https://docs.databricks.com/en/security/auth-authz/table-acls/grant-privileges-database.html
* Hive メタストア オブジェクトに付与できる権限:
https://docs.databricks.com/en/security/auth-authz/table-acls/privileges.html

最新問題**: 68**
データ エンジニアリング チームのメンバーが、より大規模なデータ パイプラインの一部としてスケジュールを設定したい短いノートブックを提出しました。以下に示すコマンドを、示されているとおりに実行すると、論理的に正しい結果が生成されると仮定します。ジョブとしてスケジュールする前にノートブックから削除する必要があるコマンドはどれですか?
A. コマンド 2
B. コマンド 3
C. コマンド 4
D. コマンド 5
E. コマンド 6
Answer: E (メーッセージを残す**)**
Cmd 6 は、ジョブとしてスケジュールする前にノートブックから削除する必要があるコマンドです。このコマンドは、finalDF データフレームからすべての列を選択し、ノートブックに表示します。FinalDF データフレームはすでに Cmd 7 のテーブルに書き込まれているため、これはジョブには必要ありません。ノートブックでのデータフレームの表示はリソースと時間を消費するだけで、ジョブの出力には影響しません。
したがって、Cmd 6 は冗長なので削除する必要があります。
他のコマンドは次のタスクを実行するため、ジョブに不可欠です。
* コマンド 1: raw_data テーブルを rawDF という Spark データフレームに読み取ります。
* Cmd 2: rawDF データフレームのスキーマを出力します。これは、デバッグやデータ構造の理解に役立ちます。
* Cmd 3: rawDF データフレームからすべての列と、値構造列からネストされた列を選択し、flattenedDF という新しいデータフレームを作成します。
* Cmd 4: flattenedDF データフレームから値列を削除します (平坦化後は不要になるため)。そして、finalDF という新しいデータフレームを作成します。
* Cmd 5: FinalDF データフレームの物理計画について説明します。これは、ジョブのパフォーマンスの最適化と調整に役立ちます。
* Cmd 7: 既存のテーブルに新しいデータを追加する追加モードを使用して、finalDF データフレームを flat_data というテーブルに書き込みます。

最新問題**: 69**
Delta Lake テーブルの下流の利用者は、アプリケーションのパフォーマンスに影響を与えるデータ品質の問題について不満を抱いています。具体的には、activity_detailstable の無効

な緯度と経度の値により、他の地理位置情報プロセスを使用する機能が損なわれていると苦情が寄せられています。
若手エンジニアは、Delta Lake テーブルに CHECKconstraints を追加する次のコードを作成しました。
上級エンジニアは、上記のロジックが正しく、緯度と経度の有効な範囲が提供されていることを確認しましたが、コードを実行すると失敗します。
この失敗の原因を説明しているのはどれですか?
A. 別のチームがこのテーブルを使用して頻繁に実行されるアプリケーションをサポートしているため、2 フェーズ ロックにより操作のコミットが妨げられています。
B. アクティビティ詳細テーブルはすでに存在します。CHECK 制約は、最初のテーブル作成時にのみ追加できます。
C. アクティビティの詳細テーブルには、制約に違反するレコードがすでに含まれています。既存のデータを既存のテーブルに追加するには、すべての既存のデータが CHECK 制約に合格する必要があります。
D. アクティビティ詳細テーブルにはすでにレコードが含まれています。CHECK 制約は、テーブルに値を挿入する前にのみ追加できます。
E. 現在のテーブル スキーマにはフィールドの有効な座標が含まれていません。テーブルを変更して制約を追加する前に、スキーマ進化を有効にする必要があります。
Answer: (解答を表示する**)**
説明
問題は、Delta Lake テーブルに CHECK 制約を追加するコードが実行時に失敗することです。このコードでは、ALTER TABLE ADD CONSTRAINT コマンドを使用して、activity_details という名前のテーブルに 2 つの CHECK 制約を追加します。最初の制約は緯度の値が -90 ～ 90 の範囲にあるかどうかをチェックし、2 番目の制約は経度の値が -180 ～ 180 の範囲にあるかどうかをチェックします。この失敗の原因は、activity_details テーブルにこれらの制約に違反するレコードがすでに含まれていることです。これらの範囲外にある無効な緯度または経度の値が含まれていることを確認します。既存のテーブルに CHECK 制約を追加する場合、Delta Lake は、既存のすべてのデータがテーブルに追加する前に制約を満たしていることを検証します。いずれかのレコードが制約に違反すると、Delta Lake は例外をスローし、操作を中止します。検証済みの参考文献:
[Databricks Certified Data Engineer Professional]、 Delta Lake」セクションの下。Databricks ドキュメントの 既存のテーブルに CHECK 制約を追加する」セクション。

最新問題**: 70**
実稼働クラスターには 3 つのエグゼキューター ノードがあり、ドライバーとエグゼキューターに同じ仮想マシン タイプを使用します。
このクラスターの Ganglia メトリックを評価する場合、ドライバー上で実行されるコードによって引き起こされるボトルネックを示すインジケーターはどれですか?
A. 5 分間の負荷平均は一貫性/フラットを維持します

B. 受信バイト数が 1 秒あたり 8,000 万バイトを超えることはありません
C. 合計ディスク容量は一定のまま
D. ネットワーク I/O が急増することはありません
E. クラスター全体の CPU 使用率は約 25%
Answer: (解答を表示する**)**
説明
これはドライバー上で実行されるコードによって引き起こされるボトルネックを示しているため、これは正しい答えです。ボトルネックとは、システムのパフォーマンスまたは容量が単一のコンポーネントまたはリソースによって制限される状況です。ボトルネックにより、実行速度の低下、遅延の増加、またはスループットの低下が発生する可能性があります。実稼働クラスターには 3 つのエグゼキューター ノードがあり、ドライバーとエグゼキューターに同じ仮想マシン タイプを使用します。このクラスターの Ganglia メトリックを評価するときは、CPU、メモリ、ディスク、ネットワークなどのクラスター リソースがどのように利用されているかを示す指標を探すことができます。クラスター全体の CPU 使用率が約 25% である場合、4 つのノード (ドライバー + 3 つのエグゼキューター) のうち 1 つだけが CPU 能力をフルに使用しており、他の 3 つのノードはアイドル状態か十分に活用されていないことを意味します。これは、ドライバー上で実行されるコードに時間がかかりすぎるか、CPU リソースを過剰に消費し、実行プログラムが処理するタスクやデータを受信できないことを示唆しています。これは、ドライバーへの大量のデータの収集、ドライバーでの複雑な計算の実行、ドライバーでの非 Spark ライブラリの使用など、並列化または分散されていないドライバー側の操作がコードに含まれている場合に発生する可能性があります。検証済みの参照: [Databricks Certified Data Engineer Professional]、「Spark Core」セクション; Databricks ドキュメントの「クラスターのステータスとイベント ログの表示 - Ganglia メトリクス」セクション。Databricks ドキュメントの「大規模な RDD の収集を避ける」セクション。
Spark クラスターでは、ドライバー ノードは、タスクのスケジュール設定、実行計画の管理、クラスター マネージャーとの対話など、Spark アプリケーションの実行の管理を担当します。全体的なクラスターの CPU 使用率が低い場合 (たとえば、約 25%)、ドライバー ノードが利用可能なリソースを効果的に利用しておらず、ボトルネックになっている可能性があることを示している可能性があります。

最新問題**: 71**
VACUUM のデフォルトのしきい値は 7 日であり、内部監査チームは特定のテーブルに少なくとも 7 日を維持するよう依頼しました。
コンプライアンス要件の一部として 365 日。実装するには次の設定のどれが必要ですか。
A. ALTER TABLE table_name set TBLPROPERTIES (del-ta.deletedFileRetentionDuration= '間隔 365 日')
B. MODIFY TABLE table_name set TBLPROPERTY (delta.maxRetentionDays = 'inter-val 365 days')

C. ALTER TABLE table_name set EXENDED TBLPROPERTIES (del-
ta.deletedFileRetentionDuration
「間隔 365 日」)
D. ALTER TABLE table_name set EXENDED TBLPROPERTIES (delta.vaccum.duration=
'間隔 365 日')
Answer: (解答を表示する**)**
説明
1.ALTER TABLE table_name SET TBLPROPERTIES (property_key [=] property_val
[, ...]) TBLPROPERTIES を使用すると、キーと値のペアを設定できます テーブル プロパティとテーブル オプション (Databricks SQL) | AWS 上のデータブリック

最新問題**: 72**
若手のデータ エンジニアが、Databricks ジョブ UI を使用して一連のジョブを手動で構成しました。エンジニアは自分の作業を確認すると、自分が各ジョブの 「所有者」としてリストされていることに気付きます。彼らは転送しようとします
「DevOps」グループに対する 「所有者」権限がありますが、このタスクを正常に実行できません。
この権限の譲渡を妨げている原因を説明しているのはどれですか?
A. Databricks ジョブには所有者が 1 人だけ必要です。 「所有者」権限をグループに割り当てることはできません。
B. Databricks ジョブの作成者は常に 「所有者」権限を持ちます。この設定は変更できません。
C. デフォルトの 「admins」グループ以外、ジョブに対する権限を付与できるのは個々のユーザーのみです。
D. ユーザーは、そのグループのメンバーでもある場合にのみ、ジョブの所有権をそのグループに譲渡できます。
E. ワークスペース管理者のみがグループに 「所有者」権限を付与できます。
Answer: A (メッセージを残す**)**
ジュニア データ エンジニアが 「所有者」権限を 「DevOps」グループに譲渡できない理由は、Databricks ジョブには 1 人の所有者が必要であり、所有者はグループではなく個人ユーザーである必要があるためです。ジョブに複数の所有者を設定することはできません。また、ジョブにグループを所有者として設定することもできません。ジョブの所有者は、ジョブを作成したユーザー、または別のユーザーによって所有権を割り当てられたユーザーです。ジョブの所有者は、ジョブに対する最高レベルの権限を持ち、他のユーザーまたはグループに権限を付与または取り消すことができます。ただし、所有者は所有権をグループに譲渡することはできず、別のユーザーにのみ譲渡します。したがって、ジュニアデータ エンジニアが 「所有者」権限を 「DevOps」グループに移管しようとすることはできません。参考文献:

* ジョブのアクセス制御: https://docs.databricks.com/security/access-control/table-acls/index.html
* ジョブ権限: https://docs.databricks.com/security/access-control/table-acls/privileges.html#job-permissions

最新問題: **73**
顧客データベースおよび関連するテーブルとデータを削除すると、データベース内のすべてのテーブルが管理対象テーブルになります。これを達成するのに役立つ SQL コマンドは次のうちどれですか?
A. データベースの顧客を強制的に削除します
B. データベース顧客カスケードの削除
C. DROP DATABASE の顧客に含まれるもの
D. データベースを削除する前に、最初にすべてのテーブルを削除する必要があります
E. DROP DELTA DATABSE の顧客
Answer: (解答を表示する**)**
説明
答えは、DROP DATABASE の顧客 CASCADE です。
カスケード オプションを使用してデータベースを削除すると、すべてのテーブルが削除されます。データベース内のすべてのテーブルは管理されたテーブルであるため、ストレージ内のデータをクリーンアップするために追加の手順を実行する必要はありません。

最新問題: **74**
サプライ チェーン チームが在庫と製品の注文を監視するために SQL ダッシュボードが構築されましたが、ダッシュボードに表示されるタイムスタンプはすべて UTC 形式で表示されているため、タイム ゾーンをニューヨークの場所に変更するように要求されました。この問題の解決にどのようにアプローチしますか?
A. ワークスペースを米国中部ゾーンから米国東部ゾーンに移動します
B. デルタ テーブルのタイムスタンプを America/New_York 形式に変更します
C. SQL エンドポイントの Spark 構成を変更して、タイムスタンプを America-ca/New_York にフォーマットします。
D. SQL Admin Console で、SQL 構成パラメータのタイムゾーンを America/New_York に設定します。
E. ダッシュボードのすべての SQL クエリに SET Timezone = America/New_York を追加します。
Answer: D (メーッセージを残す**)**
説明
答えは、SQL 管理コンソールで、SQL 構成パラメーターのタイム ゾーンを America/New_York に設定します。これを構成する手順は次のとおりです。これにより、

個々のクエリを変更せずにダッシュボード全体が変更されます。SQL パラメーターの構成
SQL パラメーターを使用してすべてのウェアハウスを構成するには:
1.サイドバーの下部にある 設定」をクリックし、 SQL 管理コンソール」を選択します。
2. SQL ウェアハウス設定」タブをクリックします。
3. [SQL 構成パラメータ] テキストボックスで、1 行に 1 つのキーと値のペアを指定します。パラメータの名前と値をスペースで区切ります。たとえば、ANSI_MODE を有効にするには:
グラフィカル ユーザー インターフェイス、テキスト、アプリケーションの説明が自動生成される
同様に、SQL 構成パラメータに行を追加できます。
タイムゾーン アメリカ/ニューヨーク
SQL 構成パラメータ | AWS 上のデータブリック

最新問題: **75**
マーケティング チームは、最初の 2 週間の新しいキャンペーンのパフォーマンスを監視するために新しいキャンペーンを立ち上げています。5 分ごとに実行される更新スケジュールを使用してダッシュボードを設定したいと考えています。削減するには、次のいずれかの手順を実行できます。この更新にかかる費用は時間の経過とともにどれくらいかかるでしょうか?
A. SQL クラスターのサイズを削減します。
B. 自動スケーリングの最大サイズを 10 から 5 に削減します。
C. ダッシュボードの更新スケジュールを 2 週間後に終了するように設定します
D. スポット インスタンス ポリシーを信頼性最適化からコスト最適化に変更します。
E. 常に X-small クラスターを使用します
Answer: C (メーッセージを残す**)**
説明
答えは、ダッシュボードの更新スケジュールを 2 週間後に終了するように設定するです。

最新問題: **76**
マルチホップ アーキテクチャにおけるシルバー層の目的は何ですか?
A. 従来のデータ レイクを置き換えます
B. データの完全な未処理履歴の効率的なストレージとクエリ
C. データ品質チェックとともにスキーマが適用されます。
D. 集約されたデータを使用した洗練されたビュー
E. ビジネスクリティカルなデータに対するクエリ パフォーマンスの最適化
Answer: C (メーッセージを残す**)**
説明
答えは、データ品質チェックとともにスキーマが適用されるということです。
メダリオン アーキテクチャ - Databricks
シルバー層:

1.データストレージの複雑さ、遅延、冗長性を軽減します。
2.ETL スループットと分析クエリのパフォーマンスを最適化します。
3.元のデータの粒度を維持します（集計なし）
4.重複レコードの削除
5.本番スキーマの適用
6.データ品質チェック、破損したデータの隔離
試験の焦点: 下の画像を確認して、メダリオン建築における各層 (ブロンズ、シルバー、ゴールド) の役割を理解してください。各層とその目的を対象としたさまざまな質問が表示されます。
Udemy 内の一部の人が私のコンテンツをコピーしているため、ウォーターマークを追加する必要がありました。

最新問題**: 77**
次のデータ ワークロードのうち、Bronze テーブルをソースとして利用するものはどれですか?
A. 集約されたデータをクエリして重要な分析情報をダッシュボードに公開するジョブ
B. 機械学習アプリケーションの機能セットを開発するジョブ
C. ストリーミング ソースから生データを Lakehouse に取り込むジョブ
D. クリーンアップされたデータを集約して標準の要約統計を作成するジョブ
E. タイムスタンプを人間が判読できる形式に解析してデータを強化するジョブ
Answer: E (メーッセージを残す**)**

最新問題**: 78**
user_ltvis という名前のテーブルは、さまざまなチームのデータ アナリストが使用するビューの作成に使用されます。ワークスペース内のユーザーはグループに構成され、ACL を使用したデータ アクセスの設定に使用されます。
user_ltvtable には次のスキーマがあります。
メールアドレス STRING、年齢 INT、Ltv INT

次のビュー定義が実行されます。
マーケティング グループのメンバーではないアナリストが次のクエリを実行します。
SELECT * FROM email_ltv
このクエリによって返される結果を説明するステートメントはどれですか?
A. 3 つの列が返されますが、1 つの列には 「編集済み」という名前が付けられ、null 値のみが含まれます。
B. email 列と itv 列のみが返されます。email 列にはすべて null 値が含まれます。
C. email 列と ltv 列は、ユーザー itv の値とともに返されます。
D. 電子メール、年齢。ltv 列はユーザー ltv の値とともに返されます。
E. email 列と ltv 列のみが返されます。電子メール列には文字列が含まれます
各行に 「編集済み」と表示されます。
Answer: E (メーッセージを残す)
説明
このコードは、user_ltv というテーブルから email 列と ltv 列を選択する email_ltv というビューを作成します。このテーブルには、email STRING、age INT、ltv INT というスキーマがあります。また、このコードでは、ユーザーがマーケティング グループのメンバーではない場合、CASE WHEN 式を使用して電子メールの値を文字列 「REDACTED」に置き換えます。クエリを実行するユーザーはマーケティング グループのメンバーではないため、電子メール列と ltv 列のみが表示され、電子メール列の各行には文字列 「REDACTED」が含まれます。
検証済みの参考文献: [Databricks Certified Data Engineer Professional]、「Lakehouse」セクションの下。Databricks ドキュメントの 「CASE 式」セクション。

最新問題**: 79**
実稼働クラスターには 3 つのエグゼキューター ノードがあり、ドライバーとエグゼキューターに同じ仮想マシン タイプを使用します。
このクラスターの Ganglia メトリックを評価する場合、ドライバー上で実行されるコードによって引き起こされるボトルネックを示すインジケーターはどれですか?
A. 5 分間の負荷平均は一貫性/フラットを維持します
B. 受信バイト数が 1 秒あたり 8,000 万バイトを超えることはありません
C. 合計ディスク容量は一定のまま
D. ネットワーク I/O が急増することはありません
E. クラスター全体の CPU 使用率は約 25%
Answer: E (メーッセージを残す)
これはドライバー上で実行されるコードによって引き起こされるボトルネックを示しているため、これは正しい答えです。ボトルネックとは、システムのパフォーマンスまたは容量が単一のコンポーネントまたはリソースによって制限される状況です。ボトルネックにより、実行速度の低下、遅延の増加、またはスループットの低下が発生する可能性があります。実稼働クラスターには 3 つのエグゼキューター ノードがあり、ドライバーとエグゼ

キューターに同じ仮想マシン タイプを使用します。このクラスターの Ganglia メトリックを評価するときは、CPU、メモリ、ディスク、ネットワークなどのクラスター リソースがどのように利用されているかを示す指標を探すことができます。クラスター全体の CPU 使用率が約 25% である場合、4 つのノード (ドライバー + 3 つのエグゼキューター) のうち 1 つだけが CPU 能力をフルに使用しており、他の 3 つのノードはアイドル状態か十分に活用されていないことを意味します。これは、ドライバー上で実行されるコードに時間がかかりすぎるか、CPU リソースを過剰に消費し、実行プログラムが処理するタスクやデータを受信できないことを示唆しています。これは、ドライバーへの大量のデータの収集、ドライバーでの複雑な計算の実行、ドライバーでの非 Spark ライブラリの使用など、並列化または分散されていないドライバー側の操作がコードに含まれている場合に発生する可能性があります。検証済みの参照: [Databricks Certified Data Engineer Professional]、「Spark Core」セクション; Databricks ドキュメントの「クラスターのステータスとイベント ログの表示 - Ganglia メトリクス」セクション。Databricks ドキュメントの「大規模な RDD の収集を避ける」セクション。
Spark クラスターでは、ドライバー ノードは、タスクのスケジュール設定、実行計画の管理、クラスター マネージャーとの対話など、Spark アプリケーションの実行の管理を担当します。全体的なクラスターの CPU 使用率が低い場合 (たとえば、約 25%)、ドライバー ノードが利用可能なリソースを効果的に利用しておらず、ボトルネックになっている可能性があることを示している可能性があります。

最新問題**: 80**
DELTA LIVE TABLE パイプラインは 2 つの異なるモードで実行するようにスケジュールできます。これら 2 つの異なるモードとは何ですか?
A. トリガー、増分
B. 1 回、継続
C. トリガー、継続
D. 1 回、増分
E. 連続、増分
Answer: C (メーッセージを残す**)**
説明
答えは「トリガー、継続」です。
https://docs.microsoft.com/en-us/azure/databricks/data-engineering/delta-live-tables/delta-live-tables-concepts#-
*トリガーされたパイプラインは、現在利用可能なデータで各テーブルを更新し、パイプラインを実行しているクラスターを停止します。Delta Live Tables はテーブル間の依存関係を自動的に分析し、外部ソースから読み取った依存関係を計算することから始めます。パイプライン内のテーブルは、依存するデータ ソースが更新された後に更新されます。
*継続的パイプラインは、入力データの変更に応じてテーブルを継続的に更新します。更新が開始されると、手動で停止するまで実行され続けます。継続的なパイプラインには、常に

実行されているクラスターが必要ですが、下流のコンシューマーが最新のデータを確実に入手できるようにします。

最新問題: **81**
次の Python ステートメントのうち、クエリ内のスキーマ名とテーブル名を置換するために使用できるものはどれですか?
A. 1.table_name = "売上"
2.schema_name = "ブロンズ"
3.query = f"select * from schema_name.table_name"
B. 1.table_name = "売上"
2.query = "select * from {schema_name}.{table_name}"
C. 1.table_name = "売上"
2.query = f"select * from {schema_name}.{table_name}"
D. 1.table_name = "売上"
2.query = f"select * from + スキーマ名 +″. ″+テヲル名"
Answer: C (メーッセージを残す**)**
説明
答えは
1.table_name = "売上"
2.query = f"select * from {schema_name}.{table_name}"
Python 変数を置き換えるには、文字列連結を使用するのではなく、f 文字列を使用することが常に最善です。

最新問題: **82**
次の自動ローダー構造化ストリーミング コマンドのうち、ランディング エリアからブロンズへのホップを正常に実行できるものはどれですか?
A. 1.spark\
2..readStream\
3..format("csv")\
4..option("cloudFiles.schemaLocation",checkpoint_directory)\
5..load("着陸")\
6..writeStream.option("チェックポイントの場所", チェックポイントディレクトリ)\
7..テーブル(生)
B. 1.spark\
2..readStream\
3..format("cloudFiles")\
4..option("cloudFiles.format","csv")\
5..option("cloudFiles.schemaLocation",checkpoint_directory)\
6..load("着陸")\
7..writeStream.option("チェックポイントの場所", チェックポイントディレクトリ)\

8..テーブル(生)
（丌し い）
C. 1.spark\
2..読む\
3..format("cloudFiles")\
4..option("cloudFiles.format","csv")\
5..option("cloudFiles.schemaLocation",checkpoint_directory)\
6..load("着陸")\
7..writeStream.option("チェックポイントの場所", チェックポイントディレクトリ)\
8..テーブル(生)
D. 1.spark\
2..readStream\
3..load(rawSalesLocation)\
4..writeStream \
5..option("チェックポイントの場所", チェックポイントパス).outputMode("追加")\
6..table("uncleanedSales")
E. 1.spark\
2..読む\
3..load(rawSalesLocation) \
4..writeStream\
5..option("チェックポイントの場所", チェックポイントパス) \
6..outputMode("追加")\
7..table("uncleanedSales")
Answer: B (メーッセージを残す**)**
説明
答えは
1.スパーク\
2..readStream\
3..format("cloudFiles") \# オートローダーを使用します
4..option("cloudFiles.format","csv") \ # csv 形式のファイル
5..option("cloudFiles.schemaLocation",checkpoint_directory)\
6..load('着陸')\
7..writeStream.option("チェックポイントの場所", チェックポイントディレクトリ)\
8..テーブル(生)
注: 以下のオプションを選択した場合、これは readStream がないため正しくありません。
1.spark.read.format("cloudFiles")
2..option("cloudFiles.format","csv")
3....
4...
5...

試験の焦点: 下の画像を確認して、メダリオン建築における各層 (ブロンズ、シルバー、ゴールド) の役割を理解してください。各層とその目的を対象としたさまざまな質問が表示されます。
Udemy 内の一部の人が私のコンテンツをコピーしているため、ウォーターマークを追加する必要がありました。
低信頼度で自動的に生成された家の説明図

最新問題**: 83**
夜間ジョブは、次のコードを使用してデータを Delta Lake テーブルに取り込みます。
パイプラインの次のステップでは、パイプラインの次のテーブルにまだ処理されていない新しいレコードを操作するために使用できるオブジェクトを返す関数が必要です。
この関数定義を完成させるコード スニペットはどれですか?
def new_records():
A. spark.readStream.table("bronze") を返します
B. spark.readStream.load("bronze") を返す
C.
D. spark.read.option("readChangeFeed", "true").table ("bronze") を返します
E.
Answer: E (メーッセージを残す**)**
説明
https://docs.databricks.com/en/delta/delta-change-data-feed.html

最新問題**: 84**
以下のトランザクション データを保存するために、既存の差分テーブルを作成するか上書きするように求められました。
A. 1.CREATE OR REPLACE DELTA TABLE トランザクション (
2.transactionId int、
3.transactionDate タイムスタンプ、
4.販売単位)
B. 1.トランザクションが存在する場合、テーブルを作成または置換します (
2.transactionId int、
3.transactionDate タイムスタンプ、
4.販売単位)
5.フォーマットデルタ
C. 1.CREATE IF EXSITS REPLACE TABLE トランザクション (
2.transactionId int、
3.transactionDate タイムスタンプ、
4.販売単位)
D. 1.CREATE OR REPLACE TABLE トランザクション (
2.transactionId int、

3.transactionDate タイムスタンプ、
4.販売単位)
Answer: (解答を表示する**)**
説明
答えは
1.CREATE OR REPLACE TABLE トランザクション (
2.transactionId int、
3.transactionDate タイムスタンプ、
4.販売単位)
Databricks でテーブルを作成する場合、デフォルトではテーブルは DELTA 形式で保存されます。

最新問題**: 85**
実行中のすべてのストリームを停止する Python コードを記述するように求められました。次のコマンドのどれを使用すると、現在実行中のすべてのアクティブなストリームのリストを取得して、空白を埋めることができます。
1._____ の場合:
2. s.stop()
A. spark.streams.getActive
B. getActiveStreams()
C. activeStreams()
D. spark.streams.active
E. Spark.getActiveStreams()
Answer: D (メッセージを残す**)**

最新問題**: 86**
データセットはデルタ ライブ テーブルを使用して定義されており、expectation 句が含まれています: CON-STRAINT valid_timestamp EXPECT (timestamp > '2020-01-01') これらの制約に違反するデータを含むデータのバッチが処理される場合、予期される動作は何ですか?
A. 期待に違反するレコードはターゲット データセットに追加され、イベント ログに無効として記録されます。
B. 期待に違反するレコードはターゲット データセットから削除され、イベント ログに無効として記録されます。
C. 期待に反するレコードにより、ジョブは失敗します。
D. 期待に違反するレコードはターゲット データセットに追加され、ターゲット データセットに追加されたフィールドに無効としてフラグが立てられます。
E. 期待に違反するレコードはターゲット データセットから削除され、隔離テーブルにロードされます。
Answer: A (メッセージを残す**)**

説明
答えは、期待に反するレコードがターゲット データセットに追加され、イベント ログに無効として記録されるということです。
デルタ ライブ テーブルは、DLT パイプライン内の不良データを修正するための 3 種類の期待値をサポートしています。これらの期待値を調べるには、以下のコード例を参照してください。図の説明は中程度の信頼度で自動的に生成されます。

最新問題: **87**
米国に本拠を置くある中小企業は最近、人工知能アプリケーションを強化するためのいくつかの新しいデータ エンジニアリング パイプラインを実装するためにインドのコンサルティング会社と契約しました。会社のすべてのデータは、米国の地域クラウド ストレージに保存されています。
会社のワークスペース管理者は、請負業者が使用する Databricks ワークスペースをどこにデプロイする必要があるかがわかりません。
データ ガバナンスに関するすべての考慮事項が考慮されていると仮定すると、この決定を正確に伝える記述はどれですか?
A. Databricks はクラウド ボリューム ストレージ上で HDFS を実行します。そのため、クラウド仮想マシンはデータが保存されているリージョンにデプロイする必要があります。
B. Databricks ワークスペースはリージョン インフラストラクチャに依存しません。したがって、ワークスペース管理者にとって最も都合のよいことに基づいて決定を下す必要があります。
C. クロスリージョンの読み取りと書き込みでは、多大なコストと遅延が発生する可能性があります。可能な限り、データが保存されているのと同じリージョンにコンピューティングをデプロイする必要があります。
D. Databricks は、対話型開発中にユーザー ワークステーションをドライバーとして利用します。したがって、ユーザーは常に、物理的に近いリージョンにデプロイされたワークスペースを使用する必要があります。
E. Databricks ノートブックは、すべての実行可能コードをユーザーのブラウザーからオープン インターネット経由で仮想マシンに送信します。可能な限り、エンド ユーザーに近いワークスペース リージョンを選択するのが最も安全です。
Answer: C (メーッセージを残す**)**
説明
これは、この決定を正確に伝えるため、正しい答えです。決定は、請負業者が使用する Databricks ワークスペースをどこにデプロイするかによって決まります。請負業者はインドに拠点を置いていますが、会社のすべてのデータは米国の地域クラウド ストレージに保存されています。Databricks ワークスペースをデプロイするリージョンを選択する場合、考慮すべき重要な要素の 1 つは、データ ソースとシンクへの近さです。クロスリージョンの読み取りと書き込みでは、ネットワーク帯域幅とデータ転送料金により、多大なコストと遅延が発生する可能性があります。したがって、パフォーマンスを最適化しコストを削

減するには、可能な限り、データが保存されているのと同じリージョンにコンピューティングをデプロイする必要があります。検証済みの参照: [Databricks Certified Data Engineer Professional]、「Databricks ワークスペース」セクション; Databricks ドキュメントの「リージョンの選択」セクション。

最新問題: **88**
データ アーキテクトは、データが外部ソースから Databricks Lakehouse に取り込まれたら、テーブル アクセス制御を利用してすべての運用テーブルとビューのアクセス許可を管理することを決定しました。
次のロジックは、実稼働データベースに対する対話型クエリの権限をコア エンジニアリング グループに付与するために実行されました。
データベース prod の使用を eng に許可します。
データベース prod への選択を eng に許可します。
これらが eng グループに付与されている唯一の権限であり、これらのユーザーがワークスペース管理者ではないと仮定すると、その権限を説明するステートメントはどれですか?
A. グループ メンバーは prod データベースに対する完全な権限を持ち、他のユーザーまたはグループに権限を割り当てることもできます。
B. グループ メンバーは、prod データベース内のすべてのテーブルを一覧表示できますが、それらのテーブルに対するクエリの結果を確認することはできません。
C. グループ メンバーは、prod データベース内のすべてのテーブルとビューをクエリおよび変更できますが、新しいテーブルやビューを作成することはできません。
D. グループ メンバーは、prod データベース内のすべてのテーブルとビューにクエリを実行できますが、データベース内で何も作成または編集することはできません。
E. グループ メンバーは、prod データベース内のすべてのテーブルとビューを作成、クエリ、および変更できますが、カスタム関数を定義することはできません。
Answer: D (メーッセージを残す**)**
説明
GRANT USAGE ON DATABASE prod TO eng コマンドは、eng グループに prod データベースを使用する権限を付与します。これは、eng グループがデータベース内のテーブルとビューを一覧表示し、アクセスできることを意味します。GRANT SELECT ON DATABASE prod TO eng コマンドは、eng グループに prod データベース内のテーブルおよびビューからデータを選択する権限を付与します。つまり、SQL または DataFrame API を使用してデータをクエリできることになります。
ただし、これらのコマンドは、テーブルやビューの作成、変更、削除、カスタム関数の定義など、その他の権限を eng グループに付与しません。したがって、eng グループのメンバーは、prod データベース内のすべてのテーブルとビューに対してクエリを実行できますが、データベース内で何も作成または編集することはできません。参考文献:
データベースに対する権限を付与します。

https://docs.databricks.com/en/security/auth-authz/table-acls/grant-privileges-database.html Hive メタストア オブジェクトに付与できる権限: https://docs.databricks.com/en/security/auth-authz/table-acls/privileges.html