databricks

Guia de exame do Databricks

Databricks Certified Data Engineer Associate

Dar feedback sobre o guia de exame

Finalidade desse guia de exame

O objetivo deste guia de exame é fornecer uma visão geral do exame e sua abordagem para ajudar a determinar a preparação necessária para realizá-lo. Este documento será atualizado sempre que houver alterações em um exame (e quando essas alterações entrarem em vigor) para que você possa estar preparado. **Esta versão aborda os exames atualmente ativos a partir de 1° de janeiro de 2024. Não deixe de conferir duas semanas antes de fazer o exame para ter certeza de que você está usando a versão mais atual.**

Descrição do público

O exame de certificação Databricks Certified Data Engineer Associate avalia a capacidade de uma pessoa de usar a plataforma lakehouse do Databricks para executar tarefas introdutórias de data engineering. Isso inclui uma compreensão da plataforma lakehouse e do workspace, da arquitetura e das capacidades. Ele também avalia a capacidade de executar tarefas ETL de arquitetura multi-hop usando o Apache Spark SQL e o Python em paradigmas processado em lote e incrementalmente. Por fim, o exame avalia a capacidade do avaliado de colocar pipelines ETL básicos e queries e painéis do Databricks SQL em produção, mantendo as permissões da entidade. Espera-se que as pessoas aprovadas neste exame de certificação executem tarefas básicas de data engineering usando o Databricks e as ferramentas associadas.

Sobre o exame

- Número de itens: 45 perguntas de múltipla escolha
- Limite de tempo: 90 minutos
- Taxa de inscrição: US$ 200, mais impostos aplicáveis, conforme exigido pela legislação local
- Método de entrega: supervisionado on-line
- Materiais de auxílio durante o teste: nenhum permitido.
- Pré-requisito: nenhum exigido; participação no curso e seis meses de experiência prática em Databricks são recomendados
- Validade: 2 anos

- Conteúdo sem pontuação: os exames podem incluir itens não pontuados para coletar informações estatísticas para uso futuro. Esses itens não são identificados no formulário e não impactam a sua pontuação. Levamos em consideração o tempo adicional para esse conteúdo.

Treinamento recomendado

- Conduzido por instrutor: Data Engineering com Databricks
- Individual: Data Engineering com Databricks (disponível na Databricks Academy)

Visão geral do exame

Seção 1: Plataforma lakehouse do Databricks

- Descreva o relacionamento entre o data lakehouse e o data warehouse.
- Identifique a melhoria na qualidade dos dados no data lakehouse em relação ao data lake.
- Compare e contraste as tabelas silver e gold, quais cargas de trabalho usarão uma tabela bronze como origem e quais cargas de trabalho usarão uma tabela gold como origem.
- Identifique elementos da arquitetura da plataforma Databricks, como o que está localizado no plano de dados versus o plano de controle e o que reside na conta na cloud do cliente
- Diferencie clusters all-purpose e clusters de jobs.
- Identifique como o software de cluster é versionado usando o Databricks Runtime.
- Identifique como os clusters podem ser filtrados para visualizar aqueles que são acessíveis ao usuário.
- Descreva como os clusters são encerrados e o impacto do encerramento de um cluster.
- Identifique um cenário em que reiniciar o cluster será útil.
- Descreva como usar várias linguagens no mesmo notebook.
- Identifique como executar um notebook de dentro de outro notebook.
- Identifique como os notebooks podem ser compartilhados com outras pessoas.
- Descreva como o Databricks Repos permite fluxos de trabalho de CI/CD no Databricks.
- Identifique as operações Git disponíveis por meio do Databricks Repos.
- Identifique limitações na funcionalidade de controle de versão do Databricks Notebooks em relação ao Repos.

Seção 2: ELT com Apache Spark

- Extraia dados de um único arquivo e de um diretório de arquivos.
- Identifique o prefixo incluído após a palavra-chave FROM como o tipo de dados.
- Crie uma view, uma view temporária e um CTE como referência a um arquivo.
- Identifique que as tabelas de fontes externas não são tabelas do Delta Lake.
- Crie uma tabela a partir de uma conexão JDBC e de um arquivo CSV externo.
- Identifique como podem ser usadas a função count_if e a contagem, onde x é nulo.
- Identifique como count(row) ignora valores NULL.

- Desduplique linhas de uma tabela do Delta Lake existente.
- Crie uma nova tabela a partir de uma tabela existente enquanto remove linhas duplicadas.
- Desduplique uma linha com base em colunas específicas.
- Valide se a chave primária é exclusiva em todas as linhas.
- Valide se um campo está associado a apenas um valor exclusivo em outro campo.
- Valide se um valor não está presente em um campo específico.
- Converta uma coluna em um timestamp.
- Extraia dados do calendário de um timestamp.
- Extraia um padrão específico de uma coluna de string existente.
- Utilize a sintaxe de ponto para extrair campos de dados aninhados.
- Identifique os benefícios do uso de funções de array.
- Converta strings JSON em estruturas.
- Identifique qual resultado será retornado com base em uma query com join.
- Identifique um cenário para usar a função explode versus a função flatten.
- Identifique a cláusula PIVOT como uma forma de converter dados de formato amplo para formato longo.
- Defina uma SQL UDF.
- Identifique a localização de uma função.
- Descreva o modelo de segurança para compartilhar SQL UDFs.
- Use CASE/WHEN no código SQL.
- Use CASE/WHEN para fluxo de controle personalizado.

Seção 3: Processamento Incremental de dados

- Identifique onde o Delta Lake fornece transações ACID.
- Identifique os benefícios das transações ACID.
- Identifique se uma transação é compatível com ACID.
- Compare e contraste dados e metadados.
- Compare e contraste tabelas gerenciadas e externas.
- Identifique um cenário para usar uma tabela externa.
- Crie uma tabela gerenciada.
- Identifique a localização de uma tabela.
- Inspecione a estrutura de diretórios dos arquivos do Delta Lake.
- Identifique quem escreveu versões anteriores de uma tabela.
- Revise um histórico de transações da tabela.
- Reverta uma tabela para uma versão anterior.
- Identifique se uma tabela pode ser revertida para uma versão anterior.
- Consulte uma versão específica de uma tabela.
- Identifique por que o Zordering é benéfico para as tabelas do Delta Lake.
- Identifique como o vacuum confirma exclusões.
- Identifique os tipos de arquivos compactados pelo Optimize.
- Identifique CTAS como uma solução.

- Crie uma coluna gerada.
- Adicione um comentário à tabela.
- Use CREATE OR REPLACE TABLE e INSERT OVERWRITE
- Compare e contraste CREATE OR REPLACE TABLE e INSERT OVERWRITE
- Identifique um cenário em que MERGE deve ser usado.
- Identifique MERGE como um comando para desduplicar dados durante a gravação.
- Descreva os benefícios do comando MERGE.
- Identifique por que uma instrução COPY INTO não está duplicando dados na tabela de destino.
- Identifique um cenário em que COPY INTO deve ser usado.
- Use COPY INTO para inserir dados.
- Identifique os componentes necessários para criar um novo pipeline DLT.
- Identifique a finalidade do destino e das bibliotecas de notebook na criação de um pipeline.
- Compare e contraste pipelines acionados e contínuos em termos de custo e latência.
- Identifique qual local de origem está utilizando o Auto Loader.
- Identifique um cenário em que o Auto Loader seja benéfico.
- Identifique por que o Auto Loader inferiu que todos os dados são STRING de uma fonte JSON.
- Identifique o comportamento padrão de uma violação de restrição.
- Identifique o impacto de ON VIOLATION DROP ROW e ON VIOLATION FAIL UPDATE para uma violação de restrição.
- Explique a captura de dados de alterações e o comportamento de APPLY CHANGES INTO.
- Consulte o log de eventos para obter métricas, fazer log de auditoria e examinar a linhagem.
- Solucione problemas de sintaxe DLT: identifique qual notebook em um pipeline DLT produziu um erro, identifique a necessidade de utilizar LIVE na instrução create, identifique a necessidade de utilizar STREAM na cláusula from.

Seção 4: Pipelines de produção

- Identifique os benefícios de usar várias tarefas em Jobs.
- Configure uma tarefa predecessora em Jobs.
- Identifique um cenário no qual uma tarefa predecessora deve ser configurada.
- Revise o histórico de execução de uma tarefa.
- Identifique o CRON como uma oportunidade de agendamento.
- Depure uma tarefa com falha.
- Configure uma política de novas tentativas em caso de falha.
- Crie um alerta no caso de uma tarefa com falha.
- Identifique se um alerta pode ser enviado por email.

Seção 5: Governança de dados

- Identifique uma das quatro áreas de governança de dados.
- Compare e contraste metastores e catálogos.
- Identifique os protegíveis do Unity Catalog.
- Defina uma entidade de serviço.

- Identifique os modos de segurança de cluster compatíveis com o Unity Catalog.
- Crie um cluster all-purpose habilitado para UC.
- Crie um DBSQL warehouse.
- Identifique como consultar um namespace de três camadas.
- Implemente controle de acesso a objetos de dados.
- Identifique a colocação de metastores com um workspace como prática recomendada.
- Identifique o uso de entidades de serviço para conexões como prática recomendada.
- Identifique a segregação de unidades de negócios no catálogo como prática recomendada.

Exemplos de perguntas

Estas perguntas foram retiradas de uma versão anterior do exame. O propósito é mostrar os objetivos conforme estão indicados no guia do exame e oferecer um exemplo de pergunta que se alinhe ao objetivo. O guia do exame lista os objetivos que podem ser abordados em um exame. A melhor maneira de se preparar para um exame de certificação é revisar o resumo no guia do exame.

Pergunta 1

Objetivo: Descrever os benefícios de um data lakehouse em comparação a um data warehouse tradicional.

Qual é a vantagem de um data lakehouse que não está disponível em um data warehouse tradicional?

A. Um data lakehouse oferece um sistema relacional de gerenciamento de dados.
B. Um data lakehouse captura snaphots de dados para fins de controle de versão.
C. Um data lakehouse une armazenamento e computação para controle total.
D. Um data lakehouse utiliza formatos de armazenamento proprietários para dados.
E. Um data lakehouse permite batch e streaming analíticos.

Pergunta 2

Objetivo: Identificar técnicas de otimização de query

Uma equipe de engenharia de dados precisa consultar uma tabela Delta para extrair linhas que atendam à mesma condição. No entanto, a equipe percebeu que a query está lenta. A equipe já ajustou o tamanho dos arquivos de dados. Após a investigação, a equipe concluiu que as linhas que atendem à condição estão localizadas esparsamente em cada um dos arquivos de dados.

Quais técnicas de otimização poderiam acelerar a query?

A. Data skipping
B. Z-Ordering
C. Bin-packing
D. Gravar como um arquivo Parquet
E. Ajustar o tamanho do arquivo

Pergunta 3
Objetivo: Identificar cargas de trabalho de dados que utilizam uma tabela Silver como fonte.

Qual carga de trabalho de dados utilizará uma tabela Silver como fonte?

A. Um job que enriquece os dados convertendo seus timestamps em um formato legível por humanos
B. Um job que consulta dados agregados que já alimentam um painel
C. Um job que ingere dados brutos de uma fonte de streaming no Lakehouse
D. Um job que agrega dados limpos para criar estatísticas de resumo padrão
E. Um job que limpa dados removendo registros mal formatados

Pergunta 4
Objetivo: Descrever como configurar uma atualização de agendamento

Um gerente de engenharia usa uma query do Databricks SQL para monitorar o progresso de sua equipe em
correções relacionadas a bugs relatados pelo cliente. O gerente verifica os resultados da query todos os
dias, mas tem que executar manualmente a consulta todos os dias e aguardar os resultados.

Como a query deve ser agendada para garantir que os resultados da consulta sejam atualizados todos os dias?

A. Para atualizar a cada 12 horas na página da query do Databricks SQL.
B. Para atualizar a cada 1 dia na página da query do Databricks SQL.
C. Para executar a cada 12 horas na IU de jobs.
D. Para atualizar a cada 1 dia na página de SQL warehouse do Databricks SQL.
E. Para atualizar a cada 12 horas na página de SQL warehouse do Databricks SQL.

Pergunta 5
Objetivo: Identificar comandos para conceder permissões apropriadas

Um novo engenheiro de dados começou a trabalhar em uma empresa. O engenheiro de dados foi recentemente adicionado ao Databricks Workspace da empresa como new.engineer@company.com. O engenheiro de dados precisa ser capaz de consultar a tabela sales (vendas) no banco de dados retail (varejo). O novo engenheiro de dados já recebeu USAGE (permissão de uso) para o banco de dados retail.

Qual comando deve ser usado para conceder as permissões apropriadas ao novo engenheiro de dados?

A. GRANT USAGE ON TABLE sales TO new.engineer@company.com;
B. GRANT CREATE ON TABLE sales TO new.engineer@company.com;
C. GRANT SELECT ON TABLE sales TO new.engineer@company.com;
D. GRANT USAGE ON TABLE new.engineer@company.com TO sales;
E. GRANT SELECT ON TABLE new.engineer@company.com TO sales;

Respostas
Pergunta 1: E
Pergunta 2: B
Pergunta 3: D
Pergunta 4: B
Pergunta 5: C