

Coleção de Dicas para certificação Microsoft DP-900

Fonte: ExamTopics / Jan/2022 updated questions

- 1 - Análise descritiva diz o que aconteceu no passado.
- 2 - Normalização não envolve eliminar relacionamento entre tabelas, reduz a redundância e melhora a integridade dos dados.
- 3 - Um processo de ETL requer um repositório de dados poderoso suficiente para transformar os dados, sendo ele um banco de dados, sistema de arquivos ou diretório. A transformação ocorre no repositório apontado para armazenamento dos dados e o mesmo precisa ser poderoso para ser eficiente.
- 4 - Os dados não precisam ser totalmente processados. Aliás, frequentemente as três fases do ETL ocorrem em paralelo para otimizar tempo.
- 5 - Um processo de batch é quando um computador realiza requisições completas de jobs, frequentemente de forma simultânea, contínua e em ordem sequencial.
- 6 - Em um processo de batching(processamento de arquivos em lote) a latência é esperada.
- 7 - Transcrever um arquivo de áudio é um exemplo ou criar um texto de closed caption são tipos de análise/serviços cognitivos.
- 8 - A pergunta "Por que as vendas aumentaram no mês passado?" é um tipo de análise de diagnóstico, pois os dados históricos podem ser usados contra outros dados para responder o motivo de algo ter ocorrido no passado. Esse é o processo de interpretação de diferentes dados, anomalias, padrões e relacionamentos.
- 9 - A pergunta "Como eu reservo meu orçamento para comprar diferentes itens do inventário?" É um tipo de análise prescritiva, pois é uma combinação de dados, modelos matemáticos e diversas regras de negócio para inferir ações que influenciam em desejáveis resultados futuros. Análise preditiva, falando superficialmente, é uma categoria de Business Intelligence que usa variáveis descritivas e preditivas do passado para analisar e identificar a plausibilidade de um resultado futuro desconhecido.
- 10 - A pergunta "Quais pessoas são mencionadas nos documentos de uma companhia?" é uma análise descritiva, pois questiona o que está acontecendo.
- 11 - A Análise de Dados, de maneira geral, compõe-se de quatro tipos de análises: Descritiva, que responde o que está acontecendo; Diagnóstica, que responde porque está acontecendo; Preditiva, que responde o que acontecerá; Prescritiva, que responde quês ações devemos tomar.

12 - Em um arquivo JSON contendo as informações de um "customer", havendo entre seus dados de "social media" duas opções declaradas distintamente e, entre os dados dentro de "customer" um total de três "address", podemos afirmar que: "Customer" é um objeto raiz, pois é a estrutura que encabeça os outros dados a seguir; "Address" é um objeto aninhado, pois dentro de "customer" há ainda três campos dentro de "address"; "Social media" é um vetor aninhado, pois dentro dele há dois objetos distintos para cada posição.

13 - Em um modelo dimensional havendo uma tabela central referenciada como "Sales" referenciando outras quatro tabelas compostas por Chaves Primárias sem chaves estrangeiras, temos um modelo de dados do tipo Estrela ou "star schema", organizadas por tabelas Fatos e Dimensões. Dado o exemplo, a tabela "Sales" é a tabela Fato e as demais, sejam quais forem, são as tabelas Dimensão.

14 - No Azure Synapse Analytics, a engine MPP distribui processamento entre nós de computação.

15 - Um índice clusterizado é um objeto associado a uma tabela que sorteia e armazena as linhas dos dados baseada em suas chaves-valor.

16 - Um banco de dados relacional é apropriado para cenários que envolvam um grande volume de escritas transacionais. Utilize um banco de dados relacional quando os dados que você trabalha são estruturados e sua estrutura não se sujeita a alterações frequentes.

17 - Como solução Big Data, um processo batch pode exportar um arquivo para análise de um armazenamento para outro, como um sistema de arquivos escalável, mas não pode exportar para um banco de dados relacional e nem para um banco de dados NoSQL.

18 - Se a sua companhia planeja realizar um processo de ELT extraindo e tratando os dados de um sistema CRM para um Data Warehouse, consideramos que: O Sistema CRM é o local de extração, a ferramenta de integração dos dados é o local de transformação e o Data Warehouse o local de carregamento.

19 - A visualização que mostra o atual status do volume de matrículas de uma universidade versus a capacidade máxima é um exemplo de análise descritiva.

20 - Sobre cartões de visualização do Power BI, um cartão Treemap é multicolorido e mostra o aninhamento de retângulos que mostram individualmente os dados representados pelo seu tamanho e cor relativo. O cartão Key Influencer mostra a maioria dos contribuintes de um resultado ou valor selecionado. Já o cartão Scatter e/ou Bubble mostra a relação entre dois valores numéricos.

21 - Você precisa criar um Azure Storage Account. Os dados na conta precisam ser replicados fora da região Azure automaticamente. Os dois tipos de replicação que você pode usar para um Storage Account são a RA-GRS "read-access geo-redundant storage" e a GRS "geo-redundant storage".

22 - Um banco de dados PaaS oferecido no Azure tem menos requisitos de configuração em comparação a um banco de dados IaaS. Um banco de dados PaaS também provê aos administradores a habilidade de controlar e atualizar o versionamento do sistema operacional, mas nenhum banco relacional ou não-relacional PaaS oferecidos no Azure podem ser pausados para redução de custos.

23 - As declarações INSERT, BULK INSERT, DELETE, SELECT, UPDATE e MERGE são do tipo Data Manipulation Language (DML).

24 - Você possui uma consulta SQL que combina os dados de pedido e dados de um cliente. A consulta também inclui colunas calculadas. Caso precise criar um objeto no banco de dados que permitiria a outros usuários reutilizar a mesma consulta SQL, você deve criar uma view, não um índice, função escalar ou tabela.

25 - O uso de chave-valor armazenada é otimizado para pesquisas simples.

26 - O local correto para o armazenamento de arquivos de imagem é o Azure Blob Storage, já para pares de chave-valor o Azure Cosmos DB Gremlin API e o local onde armazenar o relacionamento entre funcionários é o Azure Table Storage por se tratar de um relacionamento um-para-muitos e no Azure Table Service há diversas maneiras de implementar tais relacionamentos.

27 - É possível descrever o nível de hierarquia de um Data Lake Storage Gen2 por Azure Resource Group > Azure Storage account > File share > Folders > Files. Contêineres não são parte da hierarquia de um Data Lake Storage Gen2.

28 - As duas características do processamento de dados em tempo real são a baixa latência esperada e o processamento de um dado assim que o mesmo é criado.

29 - Sobre componentes do Azure Data Factory, um dataset é a representação da estrutura de dados dentro do armazenamento de dados, os Linked Services são a informação usada para conectar a recursos externos e as pipelines são um grupo lógico de Activities que performam a unidade de trabalho e podem ser agendadas para execução.

30 - Sobre tipos de cargas de trabalho, o Batch serve para processar lote de dados para um catálogo de produtos e carregá-los a cada 12 horas para um Data Warehouse, assim como também pode efetuar a atualização desse inventário a cada 1 milhão de transações, por exemplo. Batch tecnicamente se traduz por "lote" e a periodicidade de INSERTS e UPDATES pode ser realizada por hora ou volume. A carga de trabalho de streaming pode enviar para o Data Warehouse um determinado volume de dados por segundo, como no caso de compras online que serão inseridas e registradas em tempo real.

31 - Em um cartão de Power BI representado por gráfico, uma linha temporal exibe o volume de vendas de um quadrimestre. O tipo de análise realizado é a preditiva, que responde a pergunta "O que acontecerá?" para que previsões sejam realizadas com base nos dados recentes "do que vem acontecendo".

32 - Você precisa criar um recurso dentro do Azure para armazenar os dados em Azure Table Storage. De acordo com a comunidade, o comando que você deve utilizar é o `az storage account create`, mas a microsoft preconiza que o comando é `az storage container create`. O argumento da comunidade é que não é possível criar um Azure Table Storage sem antes criar uma Storage Account.

33 - O Azure Cosmos DB Core (SQL) API permite que se use a declaração `SELECT` para requisitar documentos do Azure Cosmos DB.

34 - Você precisa modificar uma view em um banco de dados relacional adicionando uma nova coluna. A declaração que se deve utilizar é `ALTER` para adicionar novas colunas. Para adicionar novas linhas se deve utilizar `UPDATE`.

35 - O processo ELT pode reduzir a transferência de dados sensíveis para um sistema de destino.

36 - O processo ELT não transforma os dados usando o recurso de computação, seja da origem ou destino do sistema.

37 - O processo ELT minimiza o tempo que se leva para copiar um grande volume de dados para o destino dos sistemas.

38 - Você planeja o deploy de um banco de dados PostgreSQL no Azure. O modelo correspondente às opções de deployment são: PostgreSQL para VM's do Azure: IaaS e Azure Database para PostgreSQL: PaaS.

39 - Bancos de Dados de grafos suportam remotamente a análise de relacionamentos entre entidades. Bancos de dados do tipo "column family" não.

40 - A solução Azure Data Lake Storage Gen2 provém suporte nativo para listas de controle de acesso (ACL) do tipo POSIX, uma interface portátil de sistemas operacionais que define basicamente o conjunto de funções de biblioteca adotadas.

41 - Dados relacionais são armazenados de forma tabular em linhas e colunas.

42 - Um índice clusterizado fisicamente organiza os dados em uma tabela baseada nos valores de uma coluna específica.

43 - Um bloco de código, ou algoritmo, que roda em um banco de dados é chamado de procedimento armazenado (stored procedure).

44 - Você tem um banco de dados de gerenciamento de inventário que contém uma tabela. Nela, duas colunas "product name" e "quantity" com três linhas cada, sendo a quantidade para cada produto respectivamente de valores "100", "129" e "176". A declaração SQL que você deve utilizar em uma consulta para alterar a quantidade do produto mencionado na primeira linha para "270" deve ser UPDATE, pois se trata de uma alteração nos dados de uma linha, ainda que haja uma coluna como referência. Se fosse uma operação para adicionar, alterar ou remover coluna, seria ALTER.

45 - Sua companhia precisa implementar um banco de dados relacional em Azure. A solução deve ser minimizar a manutenção recorrente. O serviço Azure que você deve implementar é o Azure SQL Database.

46 - Você está escrevendo um conjunto de consultas SQL que administradores utilizarão para resolver problemas no Azure SQL Database. É preciso anexar documentos e resultados de consultas em um notebook SQL. O serviço que você deve utilizar é o Azure Data Studio, que tem suporte a notebooks.

47 - Um objeto de banco de dados que registra dados é uma tabela, o objeto que armazena dados definidos por uma consulta é uma View e o objeto que ajuda a aprimorar a velocidade de retorno dos dados é um índice.

48 - Você possui uma aplicação de e-commerce que lê e escreve dados para um Azure SQL Database. O tipo de processo que a aplicação usa é um OLTP (Online Transaction Processing), uma prática de captura, armazenamento e processamento de dados transacionais em tempo real.

49 - Você pode utilizar um template do Azure Resource Manager (ARM) para automatizar a criação de um grupo de recursos independente de forma recorrente especificando em um documento JSON os recursos que você deseja gerar. Esse processo pode ser replicado, se necessário, para o deploy de recursos para uma diferente região.

50 - Você possui um Azure SQL Database que você acessa diretamente da internet. Recentemente seu endereço IP externo foi alterado. Após a mudança de IP, você não tem mais acesso ao banco de dados, mas consegue conectar a outros recursos do Azure. Uma das causas possíveis do problema é um firewall configurado no Azure SQL Database que determina quais endereços de IP podem ou não acessar o servidor ou o banco de dados.

51 - O Microsoft SQL Server Management Studio (SSMS) é uma ferramenta gráfica para o gerenciamento do SQL Server ou Azure SQL que suporta acesso, configuração, gerenciamento e administração de tarefas.

52 - O Microsoft Visual Studio Code é um editor super leve de código fonte com uma extensão MSSQL que suporta conexões a um SQL Server e uma rica experiência de edição para T-SQL (Transact SQL).

53 - O Azure Data Studio é uma ferramenta super leve que pode rodar consultas e views SQL sob demanda, salvar os resultados como texto, JSON ou arquivos Microsoft Excel.

54 - O Microsoft SQL Server Data Tools (SSDT) é uma ferramenta de desenvolvimento para construção de bancos de dados relacionais como Azure SQL e SQL Server, modelos de dados SQL Server Analysis Services (SSAS), pacotes SQL Server Integration Services (SSIS) e relatórios SQL Server Reporting Services (SSRS).

55 - Bancos de dados relacionais contém colunas e linhas. Índices em um banco de dados relacional não descrevem o tipo de dados em uma tabela. Uma view em um banco de dados é uma tabela virtual (ou temporária) que seu conteúdo é definido em uma consulta SQL.

56 - A ferramenta de linha de comando que você pode utilizar para iniciar uma consulta no Azure SQL é o "sqlcmd", que inicializa as declarações transacionais do T-SQL, procedimentos de sistema e arquivos de script.

57 - O Azure SQL Database inclui um serviço de backup gerenciado, um recurso embutido de alta disponibilidade e pode usar o Azure Defender.

58 - Você pode usar o Azure Data Studio para consultar um cluster de Big Data em Microsoft SQL Server, o Microsoft SQL Server Management Studio (SSMS) para consultar um Data Warehouse do Azure Synapse Analytics e você pode também usar o MySQL Workbench em consultas do Azure Database para bancos de dados MariaDB.

59 - Um banco de dados PaaS oferecido no Azure provê alta disponibilidade embutida, opções de escalabilidade configuráveis e redução de overhead (sobrecarga) administrativo para gerenciamento de hardware.

60 - Você tem a seguinte consulta SQL: "INSERT INTO dbo.Products (ProductID, ProductName, Price, ProductDescription) VALUES (1, 'Clamp', 12.48, 'Workbench clamp') ;". Para não haver dúvidas, "dbo.Products" é o nome da tabela e "ProductName" uma coluna.

61 - Você NÃO precisa aplicar updates do sistema operacional ao banco de dados relacional do Azure SQL regularmente, nem precisa utilizar uma assinatura Microsoft 365 para criar um banco de dados Azure SQL, mas PODE UTILIZAR uma licença existente do Microsoft SQL Server para reduzir o custo de um banco de dados Azure SQL.

62 - CREATE, ALTER, DROP, RENAME, TRUNCATE TABLE entre outras são exemplos de declaração DDL (Data Definition Language).

63 - O Azure Data Studio pode ser utilizado para consultas no banco de dados Azure SQL a partir de um dispositivo que roda no sistema operacional macOS, assim como também pode ser utilizado para restaurar um banco de dados. O Microsoft SQL Server Management Studio (SSMS) não permite que usuários criem e usem notebooks SQL como é feito no Data Studio.

64 - Você está realizando o deploy de uma aplicação SaaS que requer um banco de dados relacional do tipo OLTP (Online Transaction Processing). O serviço Azure que deve ser utilizado em suporte a essa aplicação é o Azure SQL Database, já que é um banco de dados relacional e um serviço gerenciado. Cosmos DB e HDInsight são bancos não-relacionais e o Synapse Analytics serve para Data Warehousing, não para OLTP.

65 - Quais são os dois benefícios de um banco de dados PaaS relacional oferecido no Azure, como por exemplo o Azure SQL Database? Primeiro o acesso aos últimos recursos e features lançados, depois a redução de carga de trabalho administrativo para o gerenciamento da infraestrutura do servidor. O controle sobre backups e restauração de processos são automaticamente providos pela Microsoft, não sendo um benefício de escolha.

66 - Se você tem um banco de dados PaaS no Azure, você não é responsável por aplicar atualizações de sistemas operacionais, pois a tarefa é gerenciada pela própria Microsoft. O mesmo se aplica para upgrade da engine do banco de dados. Da mesma forma, os backups são performados automaticamente.

67 - Se você deseja criar uma consulta completa para criar uma view da média de vendas de uma tabela chamada "Sales" contendo as colunas "SalesDate", "SalesAmount" e "ProductID" agrupando e ordenando essa média pela data das transações, utilize a consulta completa "SELECT SalesDate, AVG(SalesAmount) FROM Sales GROUP BY SalesDate ORDER BY SalesDate".

68 - Quando você cria um banco de dados Azure SQL, o login da conta de administrador do servidor lógico será a conta com a qual sempre será possível se conectar ao banco de dados.

69 - Um time de desenvolvedores têm computadores que rodam Windows 10 e Ubuntu Desktop. Os desenvolvedores precisam conectar e consultar um banco de dados Azure SQL a partir de seus dispositivos. Eles precisam de features de assistência de código como o IntelliSense. Para tal, os desenvolvedores precisam utilizar o Azure Data Studio, que possui suporte ao download de extensões e suporta ambos sistemas operacionais. O SQL Server Management Studio não possui suporte ao Ubuntu, portanto não seria uma solução viável.

70 - O TDE (Transparent Data Encryption) criptografa o banco de dados para proteção dos dados ociosos de ameaças maliciosas offline.

71 - Você precisa assegurar que usuários utilizem o MFA (Multi-factor Authentication) quando conectando ao banco de dados Azure SQL. O tipo de autenticação que deve ser utilizada é o Azure Active Directory (Azure AD).

72 - Um benefício em hospedar um banco no Azure SQL Managed Instance em comparação ao Azure SQL Database é o suporte nativo das transações e consultas entre bancos, chamado de "cross-database".

73 - Por padrão, cada banco de dados Azure SQL é protegido por um firewall de hierarquia "server-level", o mesmo que eventualmente pode travar o acesso a novos IPs de rede que não estejam previamente autorizados.

74 - Você precisa desenhar e modelar um banco de dados usando uma ferramenta gráfica que suporte desenvolvimento off-line orientado a projeto. O software adequado para tal é o Microsoft SQL Server Data Tools (SSDT).

75 - Um firewall previne acesso a um banco de dados Azure SQL a partir de outra rede, a autenticação suporta acessos múltiplos ao Azure Active Directory (Azure AD) e a criptografia garante que dados sensíveis nunca apareçam textualmente em um banco de dados Azure SQL.

76 - Você possui uma aplicação transacional que armazena dados em um Azure SQL Managed Instance. Você deve implementar uma réplica desse banco no modo somente leitura quando precisar gerar relatórios sem afetar a carga de trabalho transacional alimentada em tempo real.

77 - Um banco de dados relacional precisa ser usado quando é requerido forte consistência e garantia.

78 - Dados relacionais usam CHAVES para forçar relacionamentos entre tabelas diferentes, como por exemplo as Chaves Primárias e Chaves Estrangeiras.

79 - Uma View é uma tabela virtual preenchida por conteúdo definido por uma consulta.

80 - Você precisa consultar uma tabela chamada "Products" em um banco de dados Azure SQL. Há três requerimentos a serem preenchidos para acessar esse banco pela internet. São eles: Você obviamente precisa utilizar a declaração SELECT para acessar a tabela "Products", você também precisa ter um usuário no banco de dados e por fim, seu endereço de IP precisa estar com acesso autorizado no server-level firewall.

81 - Sobre os tipos de dados que podem ser armazenados em um repositório, podemos dizer que: Usuários de aplicação e suas linguagens padrão são dados do tipo chave-valor, imagens médicas e seus metadados associados são do tipo objeto e os dados de funcionários assim como o relacionamento entre eles são do tipo grafos.

82 - Você possui uma conta Azure Cosmos DB que utiliza a Core (SQL) API. No nível de hierarquia container-level é possível configurar a taxa de transferência (throughput) e chave de partição (partition key).

83 - Sua companhia está desenhando um data store(repositório de dados) que contém os dados dos estudantes. Os dados têm o formato: Duas colunas "StudentNumber" e "StudentInformation" e nelas estão armazenados o número da matrícula e as informações pessoais dos mesmos. O tipo de repositório de dados que deve ser utilizado é o colunar.

84 - O Azure Data Lake Storage é a solução de armazenamento que suporta controle de acesso regrado (RBAC) nos níveis de hierarquia de arquivos e pastas.

85 - Você precisa armazenar dados em um Azure Blob Storage por sete anos para atender os requisitos de compliance da sua empresa. O tempo de recuperação dos dados é irrelevante. A solução precisa minimizar os custos de armazenamento. O Tier (camada) do armazenamento que se deve utilizar é o Archive (arquivo). Para dados acessados frequentemente recomenda-se o tier Hot, já para dados acessados em intervalos regulares acima dos 30 dias, é recomendado o tier Cool.

86 - O documento é um tipo de repositório não-relacional que suporta esquemas flexíveis, armazena dados como arquivos JSON e também os dados para uma entidade no mesmo documento. Isto pois, JSON por si só é um documento e esse suporte é auto-explicável na definição do tipo de repositório com tais suportes.

87 - Alguns bancos de dados possuem API para ingestão de dados de alguns tipos específicos. São eles: Gremlin API para dados do tipo grafo, MongoDB API para dados do tipo documentos JSON e Table API para dados do tipo chave-valor.

Fonte: 200 practice questions for azure dp-900 certificate / Medium.com

88 - Para configurar um conta Azure Storage que suporte segurança com hierarquia de nível folder-level e também manipulação atômica de diretórios, habilite o namespace hierárquico.

89 - Dados são classificados como estruturados, semi-estruturados e não-estruturados. Os dados estruturados são tipicamente representados de forma tabular em colunas e linhas de um banco de dados. Dados semi-estruturados são informações que não residem em um banco relacional mas ainda possuem alguma estrutura, como documentos guardados em JSON, dados do tipo chave-valor e grafos. Os dados não-estruturados são do tipo arquivos de áudio e vídeo e arquivos binários.

90 - É verdade que o Azure provém diferentes tipos de serviços de armazenamento baseados no tipo de dados. Para dados estruturados, bancos relacionais como SQL Server ou Azure SQL Database, para dados semi-estruturados como documentos, um banco não-relacional com o Azure Cosmos DB, já para dados não-estruturados é possível utilizar o Azure Blob Storage.

91 - Provisionamento, por definição, é o ato de disponibilizar o recurso ou serviço de um servidor e um banco de dados.

92 - Os níveis de acesso que são possíveis definir no Azure são somente leitura, leitura/escrita e privilégio de proprietário. Para somente leitura, o usuário pode visualizar mas não pode modificar os dados. Para leitura/escrita, usuários têm a habilidade de visualizar e modificar os dados. Já para o proprietário, há acesso integral incluindo o gerenciamento de segurança como a possibilidade de adição ou remoção dos privilégios de acesso a usuários existentes, assim como definir o usuário que pode ter acesso aos dados prioritariamente.

93 - Existem dois tipos de soluções para o processamento de dados, o OLTP (Transactional) e o OLAP (Analítico). O transacional registra transações financeiras em tempo real e rastreia pagamentos. O analítico se concentra na captura do dado bruto utilizado para gerar insights para decisões de negócios e determinar foco de manufatura, campanhas e lucratividade.

94 - As tarefas envolvidas em um sistema analítico (OLAP) são a ingestão dos dados, transformação ou processamento de dados, consulta dos dados e visualização. O processo de ETL se dá pelas etapas de ingestão, transformação e visualização, sendo a consulta e visualização etapas que podem, por exemplo, ser realizadas no Microsoft Power BI.

95 - No processo de transformação do ETL, algumas das operações que podem ser realizadas são a limpeza de operações, remoção de dados questionáveis ou inválidos, realizar agregações como cálculo de lucro, margens e outras métricas-chave de performance, chamadas de KPI. As KPI's são como os negócios são medidos para crescimento e performance.

96 - Normalização é o processo de divisão em um grande número de tabelas magras (com poucas colunas) e bem definidas, com referências de uma tabela a outra.

97 - Você tem um grande volume de dados de clientes e decidiu armazenar esses dados em um banco relacional. A primeira coisa que você deve fazer é o processo de normalização.

98 - A desvantagem da normalização é o processo de desnormalizar, que é reunir todas as informações divididas na normalização através de comandos JOINS. Essas consultas por vezes podem se tornar custosas.

99 - Bancos de dados não-relacionais permitem que você armazene dados em um formato mais próximo a estrutura original, mas a desvantagem disso é que parte desses dados fica duplicada no banco de dados. Essa duplicação não apenas aumenta o requisito de armazenamento, mas também força manutenções mais complexas pois é preciso fazer as modificações em todos os locais.

100 - Os princípios ACID são: Atomicidade, Consistência, Isolamento e Durabilidade.

101 - Atomicidade garante que cada transação é tratada como uma unidade singular, que sucede perfeitamente ou falha completamente. Se qualquer uma das etapas que constituem uma transação falha, então a transação falha por completo e o banco segue sem alterações.

102 - Consistência garante que a transação pode levar os dados no banco de dados de um estado válido ao outro. Um banco consistente nunca deve perder ou criar dados de maneira que ele não possa ser contabilizado. Por exemplo, se você adicionar fundos a uma conta bancária, então deve haver a dedução dos fundos de uma conta e um banco de origem ou um o registro que descreva a origem dos fundos, sem a possibilidade de simplesmente criar dados que inexistem.

103 - Isolamento garante que a execução concorrente de uma transação deixe o banco de dados no mesmo estado que ele estaria se tivesse recebido dados de forma sequencial. Uma transação ou registro de dados em tempo real que ocorra paralelamente não pode invalidar o registro de dados de outra origem causando inconsistência.

104 - A durabilidade garante que, uma vez que a transação seja concluída, ela se manterá concluída mesmo que haja uma falha no sistema como um bug, crash ou falha de energia.

105 - Um banco de dados transacional precisa aderir as propriedades ACID (Atomicidade, consistência, isolamento e durabilidade) para garantir que o banco se mantenha consistente ao longo do processamento das transações.

106 - Consistência eventual é quando se aplicam travas na implementação do banco para que elas hajam quando o mesmo está atualizado. A trava previne outro processo de fazer a leitura dos dados até que a trava seja removida. Isto ocorre, por exemplo, em bancos de dados distribuídos, onde os dados são armazenados em diferentes locais físicos ou dispersos ao longo de redes de computadores interconectados em um mesmo local.

107 - O processamento de dados é simplesmente a conversão do dado bruto para uma informação relevante através de um processo. O processar dados assim que o mesmo chega é chamado de streaming. Carregar e processar os dados em grupos é chamado de batch, ou arquivo em lotes.

108 - No processamento de arquivo em lotes (batch) há vantagens e desvantagens. Entre as vantagens, o grande volume de dados que podem ser processados em um momento conveniente, inclusive sendo agendado para ser realizado quando computadores ou sistemas estiverem ociosos ou com mais memória e processamento disponíveis, como durante a madrugada. Entre desvantagens, o tempo perdido entre a ingestão dos dados e obtenção dos resultados. A entrada de todos os lotes de jobs (processamentos) precisam estar prontas antes de um lote ser processado. Mesmo o menor dos erros nesses dados, como erros de tipo de dados em datas, podem impedir que um batch ocorra adequadamente.

109 - O website de uma corretora imobiliária que rastreia os dados de clientes através de um dispositivo mobile e sugere em tempo real recomendações de propriedades para visita baseado em sua geolocalização utiliza processamento de dados por streaming.

110 - As diferenças entre o processamento em batch e steaming são escopo, tamanho, performance e análise. Escopo: um batch pode processar todos os dados de um dataset, mas os stream tipicamente só tem acesso ao dado mais recente recebido ou em uma janela de tempo como por exemplo dos últimos 30 segundos. Tamanho: um batch pode ser dirigido a grandes datasets de forma eficiente, mas o processamento por stream é adequado para registro individual ou micro lotes que consistem de pequenos registros. Performance: a latência de um batch é tipicamente algumas horas, já o stream processa tipicamente de forma imediata, com latência na ordem de milissegundos, que é o tempo entre o registro e a recepção e processamento. Análise: você tipicamente utiliza um batch para performar análises complexas, já o stream é utilizado para funções simples, agregações ou cálculo de médias.

111 - Um banco de dados relacional é organizado em colunas e linhas.

112 - Áudio e vídeo são tipos de dados não-estruturados.

113 - Dados do feed do twitter são exemplos de dataset por streaming.

114 - No mundo dos dados, as funções são: Azure Database Administrator, Engenheiro de Dados e Analista de Dados. O Administrador é responsável pelo design, implementação, manutenção e aspectos operacionais on-premise e soluções cloud. O Engenheiro de Dados colabora com os stakeholders para desenhar e implementar bens relacionados aos dados que incluem pipelines de ingestão de dados, atividades de limpeza e transformação de dados e repositórios para cargas de trabalho para análise. É utilizado uma pequena variedade de plataforma de dados, incluindo bancos relacionais e não-relacionais, sistemas de arquivos e streaming de dados. Análise de Dados: Um analista habilita o business a maximizar o valor de seus dados. São responsáveis por desenhar e construir modelos escaláveis, limpeza e transformação de dados e habilitar a capacidade analítica avançada através de relatórios e visualizações.

115 - Azure Data Studio disponibiliza uma interface gráfica para o gerenciamento de diferentes bancos de dados. Provém conexões a bancos de dados SQL Server on-premise, Azure SQL Database, PostgreSQL, Azure SQL Data Warehouse e clusters Big Data de SQL Server. É uma ferramenta extensível, é possível baixar ferramentas de assistências que se conectam a outros sistemas ou fornece tutoriais para ajudar na automação de muitas tarefas administrativas.

116 - O SQL Server Management Studio disponibiliza interface gráfica, habilitando você a consultar dados, performar tarefas de administração geral de um banco de dados e gera scripts para manutenção automatizada e operações de suporte.

117 - Ferramentas comuns utilizadas por engenheiros de dados são, por exemplo, sqlcmd utility, Azure Databricks, Azure Data Factory e Azure HDInsight.

118 - Uma das ferramentas mais comuns utilizadas pelo analista de dados é o Power BI.

119 - Uma das tarefas rotineiras na função de um administrador de banco de dados Azure é a restauração e backup.

120 - As características principais de um dado relacional são: Tabulares, todas as linhas possuem o mesmo conjunto de colunas, a tabela pode conter qualquer número de linhas, a chave primária identifica unicamente cada linha na tabela e duas linhas não podem compartilhar a mesma chave primária, além da chave estrangeira referenciar linhas em outra tabela relacionada. Para cada valor em uma coluna da chave estrangeira deve haver uma linha com o mesmo valor na coluna de chave primária correspondente em outra tabela.

121 - Chaves primárias indicam a coluna ou combinação de colunas que identificam unicamente cada linha. Cada tabela deve ter uma chave primária.

122 - Colunas marcadas como FK (foreign keys), que são as chaves estrangeiras, referenciam ou ligam a uma chave primária de outra tabela e são usadas para manter o relacionamento entre tabelas. A chave estrangeira também ajuda a identificar e prevenir anomalias, como pedidos de clientes que não existem em uma tabela "Clientes".

123 - Como consultar um dado relacional? A maioria dos bancos de dados relacionais

suportam o SQL, você pode usar o SQL para criar tabelas, INSERT, UPDATE e DELETE nas linhas da tabela para consultar ou modificar os dados.

124 - Um exemplo de SQL é o código de consulta: "SELECT CustomerID, CustomerName, CustomerAddress FROM Customers"

125 - Sobre os JOINS nas consultas SQL: você pode combinar os dados de múltiplas tabelas em uma consulta utilizando a operação JOIN. Ela cria a relação entre tabelas, permitindo retornar os dados de uma ou mais tabelas ao mesmo tempo. Um exemplo: "SELECT Customers.CustomerName, Orders.QuantityOrdered, Products.ProductName FROM Customers JOIN Orders ON Customers.CustomerID = Orders.CustomerID JOIN Products ON Orders.ProductID = Products.ProductID".

126 - Os casos mais comuns de uso de um banco de dados relacional são as aplicações do tipo OLTP, que utilizam bancos relacionais como soluções bancárias, aluguéis online, reserva de vôos e compras online.

127 - Quando você cria um índice em um banco de dados, você especifica uma coluna de uma tabela e o índice contém a cópia desse dado específico em ordenadamente, com indicadores que correspondem a linhas na tabela. Quando um usuário roda uma consulta que especifica essa coluna na cláusula WHERE, o sistema gerenciador de banco de dados pode utilizar esse índice para requisitar o dado mais rapidamente do que se precisasse escanear toda a tabela linha por linha.

128 - Um índice pode consumir espaço de armazenamento adicional cada vez que você insere, atualiza ou deleta um dado na tabela. Os índices para tabela precisam ser mantidos. Esse trabalho adicional pode reduzir a velocidade dos inserts, updates e deletes ou incorrer em custos de processamento.

129 - Você tem uma tabela que frequentemente é lida, mas que raramente recebe INSERTS ou UPDATES. Como melhorar a performance das consultas? Criando um índice.

130 - Uma view é uma tabela virtual baseada num conjunto de resultados de uma consulta. Em um caso simples, você pode pensar na view como uma janela de linhas específicas sobrepostas a tabela original, como numa lupa em uma interface gráfica.

131 - Você pode consultar uma view e filtrar os dados em grande parte como em uma tabela.

132 - O IaaS é um acrônimo para Infraestrutura-como-serviço. O Azure permite que você crie

infraestrutura virtual na cloud que espelha a mesma forma como bancos de dados trabalham em um ambiente on-premise. Você continua responsável pela maior parte das operações de rotina, como instalação e configuração de software, patching, backups e restauração de dados quando necessário. O IaaS é a melhor solução para migração e aplicações que requerem o acesso de hierarquia system-level. Máquinas virtuais SQL são lift-and-shift, isso significa que você pode copiar o seu ambiente on-premise diretamente para uma máquina virtual na cloud. O sistema deve trabalhar mais ou menos exatamente como antes em sua nova localização, exceto por algumas pequenas alterações nas configurações (como por exemplo, mudanças no endereço de rede) para contabilidade das mudanças no ambiente.

133 - O PaaS é o acrônimo para plataforma-como-serviço. Em vez de criar uma infraestrutura virtual, instalar e gerenciar o banco de dados por si mesmo, a solução PaaS faz isso por você. Simplesmente você especifica os recursos que requer (baseado em volume que você acha que seu banco de dados deve ter, número de usuários e performance) e o Azure automaticamente cria as máquinas virtuais necessárias, redes e outros dispositivos para você.

134 - O principal benefício de um serviço PaaS ao invés de um sistema on-premise para rodar um sistema de gerenciamento de banco de dados é a escalabilidade. A solução PaaS habilita você a aumentar exponencialmente a capacidade do seu banco de dados sem precisar investir em expansão do seu próprio hardware.

135 - As características-chaves de um dado não-relacional é que ele permite que você armazene os dados de uma maneira muito flexível. Bancos não-relacionais não impõem um esquema ao dado. Ao invés disso, ele foca no dado por si só e não como ele é estruturado. Essa abordagem significa que você pode armazenar informação em um formato natural, que espelha a forma como você vai consumir, consultar e utilizar esses dados.

136 - Sistemas não-relacionais como o Azure Cosmos DB, suportam a indexação mesmo quando a estrutura de indexação dos dados pode variar de registro para registro.

137 - Quais são os casos de uso dos bancos de dados não-relacionais? IoT e telemetria: Esses sistemas tipicamente fazem a ingestão de grandes volumes de dados em frequentes disparos de atividade. Bancos não-relacionais podem armazenar essa informação muito rapidamente. Os dados podem ser utilizados para serviços analíticos como Azure Machine Learning, Azure HDInsight e Power BI. Varejo e marketing: A Microsoft usa o CosmosDB para sua própria plataforma de comércio eletrônico que parcialmente como Windows Store e Xbox Live. Também é usada na indústria varejista para armazenar dados de catálogo e para processar pipelines. Jogos: As camadas do banco de dados são um ponto crucial no componente de aplicações para jogos. Games modernos performam processamento gráfico em clients mobile/console, mas é na cloud que entregam conteúdo customizado como os stats ingame, integração com mídias sociais e ranking de score. Web e aplicações mobile: Um banco de dados não-relacional como Azure Cosmos DB é comumente usado em aplicações web e mobile e são muito bem adequados para a modelagem de interações sociais, integração com serviços externos (chamados third-party) e para construção de ricas experiências personalizadas. Os SDK (software development kit) do Cosmos DB podem ser utilizados para criação de aplicações iOS e Android utilizando o framework Xamarin.

138 - Os formatos mais comuns de dados semi-estruturados são JSON, Avro, ORC e Parquet.

139 - Um documento JSON é aberto e fechado entre chaves "{ }". Cada campo tem um nome, seguido por colônia e depois o valor do campo. Esses campos podem conter valores simples ou subdocumentos (cada um começando e terminando com as "{ }". Eles também podem ter múltiplos valores, armazenados como vetores e cercado com colchetes "[]". Strings são cercadas por aspas e campos são separados por vírgulas.

140 - Um documento Avro é baseado em linhas e foi criado pela Apache. Cada registro contém um título que descreve a estrutura dos dados no registro. Esse título é armazenado como JSON. O dado é armazenado como informação binária. Uma aplicação usa a informação no campo de título para analisar o dado binário e extrair os campos que ele possui. Avro é um formato muito bom para comprimir dados e minimizar requisitos de banda larga e armazenamento.

141 - ORC é um acrônimo para "Optimized Row Columnar format" e organiza os dados em colunas ao invés de linhas, o oposto do Avro. Ele foi desenvolvido pela HortonWorks para otimizar a leitura e escrita das operações no Apache Hive. O Hive é um Data Warehouse que suporta rápida sumarização e consulta ao longo de datasets muito grandes. Hive suporta consultas do tipo SQL por dados não-estruturados. Um arquivo ORC contém listras de dados e cada listra possui o dado para uma colônia ou um conjunto de colunas. Uma listra contém um índice nas linhas de uma listra, os dados para cada linha e um rodapé que possui as informações estatísticas (contagem, soma, máximo, mínimo e etc) para cada colônia.

142 - O Parquet é outro formato de dados colunares. Ele foi criado pela Cloudera e Twitter e contém grupos de linhas. Os dados para cada colônia são armazenados juntamente no mesmo grupo de linhas. Cada grupo de linhas contém um ou mais pedaços de dados. Um arquivo Parquet inclui metadados que descrevem o conjunto de linhas achadas em cada pedaço. Uma aplicação pode usar esses metadados para encontrar rapidamente o pedaço correto para o conjunto de linhas dadas e retornar o dado na colônia específica para essas linhas. O Parquet é especializado em armazenar e processar tipos de dados aninhados eficientemente e suporta compressão e esquemas de codificação de altíssima eficiência.

143 - Sobre bancos de dados NoSQL, geralmente recaem em quatro categorias: repositórios chave-valor, bancos de documentos, bancos de família colunar e bancos de dados grafos.

144 - Um repositório chave-valor é o mais simples tipo de banco NoSQL para inserção e consulta de dados. Cada item do dado em um repositório chave-valor possui dois elementos, uma chave e um valor. A chave identifica o item unicamente e o valor possui o dado para cada item. O valor é opaco ao sistema de gerenciamento de banco de dados e os itens são armazenados de forma ordenada pelas chaves.

145 - Um banco de dados de documentos representa o oposto do espectro NoSQL de um repositório chave-valor. Em um banco de documentos, cada documento possui um único ID, mas os campos dos documentos são transparentes ao sistema de gerenciamento de banco de dados. Bancos de documentos tipicamente armazenam os dados em formato JSON e podem ser codificados usando outros formatos como XML, YAML, JSON e BSON.

146 - Bancos de dados colunares organizam os dados em linhas e colunas. Exemplos dessa

estrutura incluem arquivos ORC e Parquet. De uma forma simples, um banco de família de colunas pode parecer muito similar a um banco relacional, ao menos conceitualmente. O verdadeiro poder de um banco de família de colunas está em sua abordagem desnormalizada para estruturar dados escassos.

147 - Bancos de dados grafos permitem que você armazene entidades, mas o foco principal é o relacionamento que essas entidades possuem uma com as outras. Um banco de grafos armazena dois tipos de informações: nós que você pode pensar como instâncias de entidades e arestas que especificam o relacionamento entre tais nós. Ambos, nós e arestas, podem possuir propriedades que disponibilizam informação sobre esse nó ou aresta (como colunas em uma tabela). Adicionalmente, arestas podem ter uma direção indicando a natureza desse relacionamento.

148 - As principais características de um repositório chave-valor são: a consulta especifica as chaves para identificar os itens que serão retornados, você não pode procurar por valores pois a aplicação que retorna os dados de um repositório chave-valor é responsável pela análise do conteúdo dos valores retornados. O valor é opaco ao sistema de gerenciamento de banco de dados, as operações de escrita são restritas a INSERT e DELETE e se você precisar realizar um UPDATE em um item, precisa retornar este item, modificá-lo na memória (na aplicação) e depois escrever este item novamente para o banco de dados, sobrescrevendo o original (efetivamente um DELETE e INSERT).

149 - Um caso de uso do repositório chave-valor reside na habilidade focada do mesmo em ler e escrever dados bem rapidamente. As capacidades de pesquisa são secundárias. Um repositório chave-valor é uma excelente escolha para ingestão de dados quando um grande volume de dados chega de forma contínua por stream e precisa ser armazenado imediatamente.

150 - Se você estiver construindo um sistema que monitora a temperatura através de um conjunto de blocos empresariais e determina as configurações térmicas dos aparelhos de ar condicionado em cada sala de cada bloco, seu sistema tem que gerenciar aparelhos em centenas de edifícios espalhados ao longo de um país ou região, sendo que, cada prédio tipicamente contém pelo menos 100 salas de temperatura controlada por aparelhos de ar condicionado. O tipo de repositório de dados NoSQL (como o Cosmos DB, por exemplo) é mais apropriado para captura dos dados de temperatura para permitir que isso seja rapidamente processado é um repositório ou banco de dados chave-valor. Você pode processar os dados utilizando o Azure Functions.

151 - O que é um Data Wrangling (disputa de dados)? O processo de wrangling é pelo qual você transforma e mapeia os dados brutos para um formato útil para análise. Isso pode envolver escrever códigos para captura, filtragem, limpeza, combinação e agregação de dados oriundos de diversas fontes.

152 - Dois estágios importantes da análise de dados são a ingestão e o processamento de dados.

153 - O processo de ingestão de dados trata-se de obter e importar os dados para uso imediato ou armazenamento em um banco de dados. Os dados podem chegar como um stream contínuo ou em lotes, dependendo da fonte. O propósito de um processo de ingestão é a captura e armazenamento desses dados. Esses dados brutos podem ser armazenados em um repositório como um sistema de gerenciamento de banco de dados, um conjunto de arquivos ou outro tipo de armazenamento que seja rápido e de fácil acesso.

154 - O estágio de processamento de dados ocorre após os dados serem ingeridos e coletados. O processamento de dados pega os dados em sua forma bruta, limpa e converte-os em um formato mais relevante (tabelas, grafos, documentos e etc). O resultado é um banco com dados que você pode usar para realizar consultas e gerar visualizações, dando a forma e o contexto necessários para serem interpretados por computadores e utilizados por funcionários através de uma organização.

155 - ETL significa extrair, transformar e carregar. O dado bruto é recebido e transformado antes de ser salvo. As etapas de extração, transformação e carregamento podem ser realizadas como uma pipeline de operação contínua. É recomendado para sistemas que apenas requerem modelos simples de baixa dependência entre itens.

156 - ELT significa extrair, carregar e transformar. O processo se difere do ETL pois o dado é armazenado antes de ser transformado. O processamento desses dados pode ser feito com uma abordagem iterativa, recebendo e processando o dado vindo do armazenamento antes de escrever o dado transformado e modelos de volta ao banco. O processo de ELT é recomendável para construção de modelos complexos que dependem de múltiplos itens do banco de dados, por vezes utilizando processamento de lotes periódicos.

157 - Um relatório é um processo de organizar os dados em um sumário informativo para monitorar como as diferentes áreas de uma organização são realizadas. Um relatório ajuda as companhias a monitorar seu negócio online e saber quando um dado fica fora de seu alcance. Bons relatórios devem levantar questões sobre os negócios a partir do seu usuário final. Relatórios te mostram o que aconteceu, enquanto análise foca em explicar o motivo de ter acontecido e o que se pode ser feito a respeito.

158 - Business Intelligence, ou Inteligência de Mercado se refere a tecnologias, aplicações e práticas para coletar, integrar, analisar e apresentar informações sobre negócios. O propósito da inteligência de mercado é dar suporte à melhor tomada de decisão.

159 - A visualização de dados é uma representação gráfica da informação de dados. Utilizando elementos visuais como cartões, gráficos e mapas, as ferramentas de visualização de dados fornecem uma forma acessível de observar e entender tendências, outliers (informações destoantes) e padrões de dados.

160 - As formas mais comuns de visualização de dados são: Bar and column charts, line charts, matrix, key influencers, treemap, scatter e filled map.

161 - Os Bar and column charts permitem que você veja como um conjunto de variáveis muda ao longo de diferentes categorias.

162 - Line charts dão ênfase ao formato geral de uma série inteira de valores, usualmente ao longo do tempo.

163 - A visualização matrix é uma estrutura tabular que sumariza os dados. Eventualmente, designers de relatórios incluem matrix em relatórios e dashboards para permitir aos usuários selecionar um ou mais elementos (linhas, colunas e células) em sua matrix para destacar outras visualizações na página de relatório.

164 - Key influencers chart trata-se de uma visualização que mostra os maiores contribuintes de um resultado ou valor selecionado. Key influencers são a grande escolha para ajudá-lo a entender os fatores que influenciam uma métrica chave.

165 - Treemaps são cartões de retângulos coloridos, com tamanho representando o valor relativo de cada item. Eles podem ser hierárquicos com retângulos aninhados dentro de outros retângulos principais.

166 - Scatter são cartões que mostram o relacionamento entre dois valores numéricos. Um cartão do tipo bubble é uma visualização em scatter que substitui pontos de dados com bolhas, com o tamanho dessas bolhas representando uma terceira dimensão adicional.

167 - Filled maps servem se você tiver dados geográficos, então é possível usar um filled map para mostrar como um valor difere em proporções ao longo de uma geografia ou região.

168 - Os bancos relacionais atualmente disponíveis no Azure são SQL Database, MySQL Servers, MariaDB Servers e PostgreSQL Servers.

169 - Utilizar o Azure Data Services reduz o montante de tempo investido em administrar um sistema de gerenciamento de banco de dados.

170 - A disponibilidade dos serviços do Azure Data Services são garantidos pelo menos por 99.99% do tempo.

171 - O preço base de cada serviço do Azure Data Services cobre de forma adjacente a infraestrutura e licenciamento, assim como os custos de administração. Adicionalmente esses serviços são desenhados para estar sempre online. Isso significa que você não pode desligar um banco de dados e reiniciá-lo posteriormente.

172 - A limitação dos serviços Azure é que nem todos os serviços de gerenciamento estão disponíveis, pois o Azure pega a tarefa de gerenciar o sistema e mantê-lo rodando usando um hardware situado em um Datacenter, exposto a certas funções administrativas que podem tornar a plataforma subjacente vulnerável a uso indevido ou até mesmo causar preocupações quanto a segurança. Você não tem controle direto da plataforma onde os serviços funcionam.

173 - Para ter mais controle sobre o Azure Data Services do que é normalmente permitido, você pode instalar seu gerenciador de banco de dados em uma máquina virtual que rode o Azure.

174 - A migração de um sistema rodando on-premise para uma máquina virtual Azure não é diferente de mover um banco de dados de um servidor on-premise para o outro. Por exemplo, um SQL Server rodando em uma máquina virtual Azure replica eficientemente o banco de dados rodando em um hardware on-premise.

175 - Quando você está rodando o SQL Server em uma máquina virtual, quem cuida da manutenção do software SQL Server e executa várias tarefas administrativas para manter o banco de dados rodando diariamente é o cliente. Essa é uma característica de on-premise.

176 - Há situações onde você deve rodar o Microsoft SQL Server em uma máquina virtual. Essa abordagem é otimizada para migração de aplicações existentes para o Azure ou estendendo aplicações on-premise para a cloud em deployments híbridos.

177 - Entre as capacidades de rodar um SQL Server em uma máquina virtual, estão a criação de cenários de rápido desenvolvimento e testes quando você não deseja comprar hardwares on-premises SQL Server que não produzem, tornar-se pronto para lift-and-shift no caso de aplicações que requerem rápida migração para a cloud com o mínimo ou nenhuma mudança e escalar a plataforma onde o SQL Server esteja rodando reservando mais memória, poder de processamento, espaço em disco para a máquina virtual. Você pode rapidamente redimensionar uma máquina virtual Azure sem o requisito de instalar novamente o software que está rodando nele.

178 - As opções disponíveis quando escolhendo o tipo de Azure SQL Database, sendo ele uma plataforma PaaS, são: Single Database, Elastic Pool e Manage Instance.

179 - O Single Database é uma opção que permite rapidamente configurar e rodar um único banco de dados SQL Server. Você pode criar e rodar um servidor na cloud e você acessa o seu banco de dados através desse servidor. A Microsoft gerencia o servidor, então tudo que você precisa fazer é configurar o banco de dados, criar suas tabelas e popular com seus dados. Você pode escalar o banco de dados se precisar de espaço de armazenamento, memória ou poder de processamento adicional.

180 - Elastic Pool é uma opção similar ao Single Database, exceto que por padrão múltiplos bancos de dados podem compartilhar o mesmo recursos como memória, espaço de armazenamento de dados e poder de processamento. Os recursos são referidos como a "pool". Você pode criar a pool e apenas seus bancos de dados podem usar a pool. Esse modelo é útil se você tem bancos de dados com requisitos de recursos que variam ao longo do tempo e isso pode ajudá-lo a reduzir custos.

181 - O tipo Managed Instance eficientemente roda uma instância totalmente controlada do SQL Server na cloud. Você pode instalar múltiplos bancos de dados na mesma instância. Você possui total controle dessa instância, muito do que você faria e teria se fosse um servidor on-premise.

182 - Aplicações cloud modernas necessitam o uso das mais recentes e estáveis features do

SQL Server, alta disponibilidade e diversas fontes de origem que precisam de bancos de dados rapidamente escaláveis e dimensionáveis são os casos onde o Azure SQL Database é uma solução.

183 - Azure SQL Database automaticamente realiza o update e patches no SQL Server para garantir que você esteja sempre rodando a última e mais segura versão do serviço.

184 - A escalabilidade do Azure SQL Database garante que você possa aprimorar os recursos disponíveis para armazenar e processar dados sem que precise executar um custoso upgrade manual.

185 - O serviço disponibiliza alta disponibilidade para garantir que o banco de dados esteja disponível pelo menos 99.99% do tempo.

186 - O Azure SQL Database suporta restauração ponto-a-ponto, permitindo que você restaure um banco de dados ao estado que ele estava em qualquer ponto do passado.

187 - Bancos de dados podem ser replicados para diferentes regiões para prover segurança adicional e recuperação em casos de desastres.

188 - A proteção avançada contra ameaças disponibiliza avançadas capacidades de segurança como avaliação de vulnerabilidade para ajudar a detectar e remediar potenciais problemas de segurança com seu banco de dados.

189 - O Azure SQL Database possui monitoramento contínuo do seu banco de dados para atividades suspeitas e disponibiliza alertas de segurança imediatos para potenciais vulnerabilidades, ataques injetados por SQL e anomalias no padrão de acesso ao banco de dados.

190 - O banco de dados SQL ajuda a assegurar seus dados disponibilizando criptografia. Para dados em movimento, utiliza uma camada de segurança para o transporte. Para dados ociosos, utiliza uma criptografia de dados transparente.

191 - Um dos diferenciais do MariaDB é o suporte nativo a dados temporais. Uma tabela pode armazenar muitas versões de dados, permitindo aplicações a consultar o dado assim que ele aparece em algum ponto no passado.

192 - O MySQL se tornou popular pela simplicidade de uso e com um sistema de gerenciamento open-source para aplicações web rodando Linux, mas também com versões para Windows.

193 - O PostgreSQL é um banco de dados híbrido relacional-objeto. Você pode armazenar dados em tabelas relacionais, mas o banco de dados PostgreSQL também permite que você armazene dados de tipos customizados com suas propriedades não-relacionais. Você também pode baixar extensões e adicionar módulos de código que podem rodar por consultas. Outro recurso chave é a habilidade de armazenar e manipular dados geométricos, como linhas, círculos e polígonos.

194 - Entre as opções de deployment para o Azure Database para PostgreSQL estão o Single Server e o Hyperscale. O Single Server possui benefícios similares ao Azure Database para MySQL. Você escolhe entre três preços por tier: Básico, General Purpose e Memory Optimized. Cada tier suporta diferentes números de CPU, memórias e tamanhos de armazenamento. O Hyperscale (Citrus) é uma opção de deployment que escala consultas ao longo de múltiplos nós de servidor para dar suporte a grandes cargas de banco de dados. Seu banco de dados é dividido por nós. Os dados são divididos em pedaços baseados no valor da chave partição ou fragmento partição. Considere utilizar essa opção para os maiores deployments do PostgreSQL na cloud do Azure.

195 - Havendo necessidade de re-escalonar um banco de dados SQL, ele não será reinicializado para tal ação.

196 - Utilizar o Azure Database Migration Services é a melhor forma de transferir dados em um banco de dados PostgreSQL rodando on-premises para um serviço de banco de dados Azure Database rodando PostgreSQL.

197 - Backups são automaticamente dirigidos quando utilizando um Azure SQL Database Managed Instance, tornando-se a forma mais simples de implementação de backups.

198 - Você é responsável por toda a instalação, manutenção e execução dos backups quando utilizando SQL Server em uma máquina virtual.

199 - O dialeto utilizado pela Oracle é o PL/SQL.

200 - O dialeto utilizado pelo PostgreSQL é o pgSQL.

201 - As cláusulas DDL são para a definição de uma tabela e as DML são para modelagem das mesmas, sendo por tanto a lista de cláusulas listadas à seguir: (DDL: create, alter, drop, rename) / (DML: select, insert, update, delete).

202 - Você deve utilizar as cláusulas DML para manipular linhas em uma tabela relacional. Essas cláusulas permitem que você retorne uma consulta de dados, insira novas linhas ou edite linhas existentes. Você também pode deletar linhas se não as quiser mais.

203 - Por padrão, o SELECT, UPDATE e o DELETE são cláusulas aplicadas para cada linha em uma tabela.

204 - A cláusula WHERE deve ser utilizada em conjunto com as cláusulas SELECT, UPDATE ou DELETE para aplicar mudanças para linhas específicas.

205 - A cláusula ORDER BY deve ser usada para o ordenamento dos dados selecionados na consulta.

206 - Você deve utilizar a cláusula JOIN para retornar os dados relativos às múltiplas tabelas de interesse. Uma condição de união define a forma como duas tabelas se relacionam em uma consulta especificando a coluna de cada tabela a ser usada na operação JOIN. Uma condição típica de um JOIN especifica a chave estrangeira de uma tabela e a chave-primária associada a ela na outra tabela. Outra condição é especificar o operador lógico, por exemplo “= ou <>” a ser usado em comparação aos valores das colunas.

207 - O serviço Azure que implementa um modelo NoSQL de chave-valor é o Azure Table Storage.

208 - O Azure Table Storage é um repositório de chave-valor escalável que fica hospedado na cloud. Você cria uma tabela utilizando uma Azure Storage Account. Em uma tabela Azure Table Storage, os itens são referenciados como linhas e campos conhecidos como colunas.

209 - Você possui dados semi-estruturados e deseja armazenar esses dados em um banco de dados como pares de chave-valor onde a chave é única, as colunas podem variar e cada linha guardar os todos os dados para uma entidade lógica. O armazenamento que deve ser utilizado é o Azure Table Storage.

210 - As tabelas do Azure Table Storage não tem conceitos de relacionamento, stored procedures, índices secundários ou chaves primárias.

211 - O Azure Table Storage provê acesso muito rápido aos dados que você precisa porque eles estão disponíveis em uma única linha, sem precisar que você execute operações de JOIN entre os relacionamentos. Para garantir o acesso rápido, o Azure Table Storage divide a tabela em partições.

212 - O particionamento é um mecanismo para grupo de linhas relativas baseado em uma propriedade comum ou chave-partição. As linhas que compartilham a mesma chave-partição serão armazenadas juntas. O particionamento não ajuda somente a organizar os dados, ele também aprimora a escalabilidade e performance no Azure Table Storage.

213 - Essas partições podem crescer ou ser reduzidas? Sim. Partições são independentes uma da outra e podem crescer ou diminuir conforme linhas são adicionadas ou removidas de uma partição. Uma tabela pode conter qualquer número de partições.

214 - Um particionamento ajuda a aprimorar a performance pois, quando você procura por dados, também pode incluir a chave-partição no critério de busca. Isso ajuda diminuir o volume de dados a serem examinados e melhora a performance reduzindo a quantidade de I/O (leituras e escritas) necessárias para reservar o dado.

215 - As chaves de uma tabela no Azure Table Storage são compostas por dois elementos,

uma chave-partição que identifica a partição contendo a linha e uma chave-linha única para cada linha na mesma partição. Itens com a mesma partição são armazenados ordenadamente pela chave-linha. Se uma aplicação adiciona uma nova linha a uma tabela, o Azure garante que a linha é colocada na posição correta da tabela.

216 - No Azure Table Storage há dois tipos de métodos de consulta, uma pontual e outra por extensão, chamadas de “point query” e “range query”. Em um point query, quando uma aplicação retorna somente uma linha, a chave-partição habilita o Azure a apontar rapidamente para a partição correta, onde a chave-linha deixa o Azure identificar a linha em uma partição. Em uma range query, a aplicação procura por um conjunto de linhas em uma partição, especificando o início e o fim de um conjunto de chaves-linhas. Esse tipo de consulta é também muito rápida, contanto que você tenha desenhado suas chaves-linha de acordo com os requisitos de consultas performadas pela sua aplicação.

217 - Você não precisa fazer a definição de um schema para uma tabela do Azure Table Storage, pois elas são do tipo schemaless (sem esquema). É fácil adaptar seus dados conforme a necessidade da sua aplicação evolui.

218 - Entre as vantagens de usar o Azure Table Storage, ele é facilmente escalável. Ele leva o mesmo tempo para inserir dados em uma tabela vazia ou uma tabela com bilhões de entradas. Uma conta de armazenamento Azure pode armazenar um volume de dados superior a 500 terabytes. Uma tabela do Azure Table Storage pode armazenar dados semi-estruturados, não possuem a necessidade de mapeamento e manutenção de relacionamentos complexos tipicamente requeridos por bancos relacionais normalizados, a inserção de novas linhas é rápida e o retorno de consultas é rápido se você especificar a partição e as chaves-linha como critério em sua query.

219 - Entre as desvantagens de utilizar o Azure Table Storage, é preciso considerar a consistência, já que os updates transacionais ao longo de múltiplas entidades não é garantido. Não há integridade referencial, qualquer relacionamento entre linhas precisa ser mantido externamente à tabela. É difícil realizar o refinamento e ordenamento em dados sem chaves. Consultas que buscam por campos sem chave podem resultar em escaneamento completo da tabela.

220 - O caso de uso de uma Azure Table Storage é uma análise importante a se mencionar.

Armazenar terabytes de dados estruturados é uma dessas situações. Exemplos incluem catálogo de produtos para aplicações de ecommerce e informações de clientes onde os dados podem ser rapidamente identificados e ordenados por uma chave composta. No caso de um catálogo de produtos, a chave-partição pode ser a categoria do produto (como calçados, por exemplo) e a chave-linha identifica o produto específico na categoria (como botas para trilhas, por exemplo). Armazenar datasets não precisa de operações join complexas, nem chaves-estrangeiras ou stored procedures, e podem ser desnormalizadas para acesso rápido. Em um sistema IoT, você pode usar o Azure Table Storage para capturar dados de sensores de um dispositivo. Cada dispositivo deve ter sua própria partição e o dado pode ser ordenado pela data e horário na qual cada medida foi capturada. A captura de registro de atividades e dados de monitoramento de performance também é um desses casos. O registro de eventos e informação de performance tipicamente contêm dados que são estruturados de acordo com o tipo de evento ou métrica de performance registrada. O dado pode ser particionado por evento ou tipo de métrica, ordenado por data e horário que foi registrado. Alternativamente você pode particionar os dados por data se você precisar analisar e ordenar séries de eventos e métricas de performance cronologicamente. Se você quiser analisar dados por tipo e data/horário, então é preciso considerar o armazenamento de dados duas vezes, particionado por tipo e novamente por data. A escrita dos dados é rápida e os dados armazenados são estáticos.

221 - O Azure Table Storage é desejável para suportar grande volume de dados, acima de centenas de terabytes em volume.

222 - Você precisa criar uma conta de armazenamento antes de criar um Azure Table Storage.

223 - Um Azure Blob Storage é um serviço que habilita você a armazenar grande quantidade de dados não-estruturados, os chamados blobs, na cloud.

224 - Você precisa criar uma conta de armazenamento antes de criar um Azure Blob Storage.

225 - Os diferentes tipos de blobs que o Azure Blob Service suporta são: Block blobs, Page blobs e Append blobs.

226 - Um block blob é dirigido a conjunto de blocos. Cada bloco pode variar em tamanho, acima de 100mb. Um blob de bloco pode conter acima de 50000 (cinquenta mil) blocos, entregando no máximo o volume de 4.7 terabytes. O bloco é a menor quantidade de dados que pode ser lido ou escrito como uma unidade individual. Blobs de bloco são melhor utilizados para armazenar objetos discretos, grandes e binários que raramente sofrem mudanças.

227 - Um page blob é organizado como uma coleção de páginas de tamanho fixado em 512 bytes. Um blob de página é otimizado para suportar operações de leitura e escrita aleatórias. Você pode requisitar e armazenar dados de uma página específica se necessário. Um blob de página pode armazenar até 8 terabytes de dados. Azure utiliza blobs de páginas para implementar discos virtuais para máquinas virtuais.

228 - Um append blob é um blob de blocos otimizado para suportar operações de anexo. Você pode adicionar somente blocos ao fim de um append blob, atualizando e ou excluindo qualquer bloco que não seja suportado. Cada bloco pode variar em tamanho, acima de 4 megabytes. O tamanho máximo de um append blob é de apenas 195 gigabytes.

229 - Você controla quem pode ler e escrever blobs dentro de um contêiner no nível de hierarquia container-level.

230 - Você precisa salvar objetos em um Blob Storage com uma camada de acesso do tipo Hot e deseja mover esses arquivos depois de 6 meses para a camada Arquivo, já que não precisaremos mais acessá-los. Você pode criar políticas de gerenciamento de ciclos de vida para blobs em uma conta de armazenamento. Ela pode automaticamente mover um blob de uma camada Hot para a Cool e depois para a Archive conforme o tempo vai passando e os blobs são acessados de forma menos frequente. Essas políticas são definidas com base no intervalo de tempo desde a última modificação.

231 - Entre os casos de uso do Azure Blob Storage estão: Servir imagens ou documentos diretamente a um navegador web, na forma de um website estático. Armazenar arquivos para acesso distribuído. Streaming de áudio e vídeo. Armazenamento de dados para backup e restauração, recuperação e arquivamento em caso de desastres. Armazenamento de dados para análise por um serviço on-premise ou hospedado no Azure.

232 - Se você quiser manter e restaurar versões anteriores de um blob, então você precisa de versionamento, que é possível no Azure Blob Storage.

233 - O recurso que você deve habilitar no Azure Blob Storage se desejar em algum ponto do futuro recuperar um blob que tenha sido removido ou sobrescrito acidentalmente é o “Soft Delete”.

234 - Snapshots são versões do tipo somente leitura de um blob em um momento particular e específico do tempo.

235 - A opção “Change Feed” para um blob disponibiliza um registro ordenado do tipo somente leitura com os updates feitos nele.

236 - Um Azure File Storage permite que você crie arquivos compartilhados na cloud e os acesse de qualquer lugar com uma conexão a internet.

237 - Entre os recursos de um Azure File Storage, ele permite que você compartilhe mais de 100 terabytes de dados em uma única conta de armazenamento. O tamanho máximo de um arquivo é de 1 “TiB”, o tebibyte (medida matematicamente exata de uma unidade de terabyte). O Azure File Storage suporta acima de 2000 conexões paralelas por arquivo compartilhado.

238 - Você pode fazer o upload de arquivos no Azure File Storage utilizando o portal Azure ou as ferramentas como a utilidade AzCopy. Você pode também usar o serviço do Azure File Sync para sincronizar localmente cópias em cache de arquivos compartilhados em um Azure File Storage.

239 - O Azure File Storage oferece duas camadas de performance. A camada Standard utiliza discos rígidos (HDD) em um datacenter e a camada Premium utiliza discos sólidos (SSD). A camada Premium oferece maior transferência, mas é orçado em preços mais elevados.

240 - Alguns dos casos de uso de um Azure File Storage são: Migração, compartilhamento, integração e simplificação. Ele migra aplicações existentes na cloud, compartilha servidores entre ambientes on-premise e cloud, integra aplicações modernas e simplifica a hospedagem de cargas de dados de alta disponibilidade, as chamadas High Availability (HA).

241 - Não utilize o Azure File Storage para arquivos que possam ser escritos por múltiplos concorrentes (acessos paralelos) simultaneamente. Escritas múltiplas exigem sincronização cuidadosa, senão as mudanças feitas por um processo podem sobrescrever outra. A solução alternativa é travar (usar lock) o arquivo assim que ele é escrito e depois liberar a trava quando a operação de escrita for completa. Contudo, essa abordagem pode impactar severamente no acesso concorrente e limitar a performance.

242 - O Azure Cosmos DB é um sistema de gerenciamento de banco de dados NoSQL de multi-modelo. O Cosmos gerencia os dados como um conjunto particionado de documentos. Um documento é uma coleção de dados identificados pela chave. Os campos em cada documento podem variar e o campo pode conter "child documents". Muitos bancos de documentos usam o JSON para representar a estrutura de documentos. Nesse formato, os campos em um documento são registrados entre chaves "{" e cada campo é declarado como um prefixo do seu nome. O propósito principal das APIs é dar suporte a aplicações existentes. Se você estiver construindo uma nova aplicação e banco de dados, você deve utilizar a SQL API.

243 - O Cosmos DB suporta diferentes APIs, sendo uma delas a interface SQL API, que provê uma linguagem de consulta do tipo SQL para documentos, permitindo identificar e retornar documentos utilizando cláusulas SELECT.

244 - O Cosmos DB suporta diferentes APIs, sendo uma delas a interface Table API, que permite utilizar o Azure Table Storage API para armazenar e retornar documentos. A proposta dessa interface é permitir que você alterne entre o Table Storage e o Cosmos DB sem precisar que você modifique suas aplicações existentes.

245 - O Cosmos DB suporta diferentes APIs, sendo uma delas a interface Mongo DB API, que é outro banco de documentos muito conhecido com sua própria interface de programação. Muitas organizações rodam o Mongo on-premises. Você pode usar o Mongo DB API para Cosmos DB para permitir uma aplicação Mongo a rodar inalterado contra o banco de dados Cosmos.

246 - O Cosmos DB suporta diferentes APIs, sendo uma delas a interface Cassandra API, que é um sistema de gerenciamento de banco de dados do tipo família colunar. Esse é outro sistema de gerenciamento de banco de dados que muitas organizações rodam on-premises. A Cassandra API para Cosmos DB provê uma interface de programação semelhante ao Cassandra para o Cosmos DB. Os requisitos do Cassandra API são mapeados com os requisitos dos documentos no Cosmos DB.

247 - O Cosmos DB suporta diferentes APIs, sendo uma delas a interface Gremlin API, que implementa uma interface de banco de dados grafos ao Cosmos DB. Um grafo é uma coleção de objetos e relacionamentos diretos entre os dados. Os dados ainda são armazenados como um conjunto de documentos no Cosmos DB, mas o Gremlin API permite que você execute consultas de grafos pelos dados. Utilizando o Gremlin API você pode transitar entre objetos e relacionamentos nos grafos para descobrir todos os tipos de relacionamentos complexos.

248 - O Cosmos DB garante menos de 10 milissegundos (ms) de latência, seja para leitura (com índices) ou escrita, em 99% dos casos ao redor do mundo.

249 - As chaves-partição e as chaves-linha são os elementos de uma Azure Table Storage Key.

250 - As ferramentas mais utilizadas para provisionar serviços são o portal Azure, a interface de linha de comando Azure (CLI), o Azure Powershell e os modelos (templates) do Azure Resource Manager (ARM).

251 - Entre as opções de replicação que temos ao provisionar uma conta de armazenamento Azure estão: LRS, GRS e RA-GRS.

252 - Sobre a replicação, os dados em uma conta de armazenamento Azure Storage é sempre replicado três vezes na região que você especifica como localização primária para a conta.

253 - A replicação do tipo LRS, acrônimo para “Locally redundant storage” copia seus dados sincronizadamente três vezes dentro de uma localização física da região. O LRS é a opção de replicação de menor custo, mas não é recomendada para aplicações que requerem alta disponibilidade.

254 - A replicação do tipo GRS, acrônimo para “Geo-redundant storage” copia seus dados sincronizadamente três vezes dentro de uma localização física da região usando a replicação LRS. Depois, copia seus dados de forma assíncrona para um único local físico na região secundária. Dessa forma a replicação protege os dados de blecautes regionais.

255 - A replicação do tipo RA-GRS, acrônimo para “Read-access ge-redundant storage” é uma extensão da replicação GRS que provê acesso direto do tipo somente leitura aos dados na localização secundária. Em contraste, a opção GRS não expõe os dados da localização secundária, sendo utilizada somente para recuperação em caso de falha da localização primária. A replicação RA-GRS permite que você armazene uma cópia somente leitura dos dados próximo aos usuários que são localizados em um local geograficamente distante, ajudando a reduzir os tempos da latência de leitura.

256 - A conectividade padrão para o Azure Cosmos DB e Azure Storage tem o propósito de permitir o acesso ao mundo todo. Você pode conectar aos serviços de uma rede on-premise, internet ou dentro de uma máquina virtual Azure.

257 - As opções para proteger os recursos Azure como a conta de armazenamento, Cosmos DB e etc são: Azure Private Endpoint, firewalls e redes virtuais, configuração de autenticação, configuração do controle de acesso e configurações avançadas de segurança.

258 - Quando você configura o Cosmos DB para replicação, o comportamento padrão é que apenas a região onde você criou a conta suporte operações de escrita. Todas as réplicas são do tipo somente leitura.

259 - A replicação é assíncrona para o Cosmos DB. Portanto é comum que haja atraso entre uma mudança realizada em uma região e a mudança se tornar visível em outras regiões. O atraso e desordem na alteração dos dados por conta disso é chamado de consistência eventual ou “Eventual Consistency Option”.

260 - Ainda sobre replicações no Cosmos DB, a chamada “Consistent Prefix Option” é uma opção que garante que mudanças apareçam em ordem, entretanto pode haver um atraso até que elas sejam visíveis. Nesse período, aplicações enxergam dados antigos.

261 - Ainda sobre replicações no Cosmos DB, a chamada “Session Option” é quando uma aplicação realiza um número de alterações e todas elas se tornam visíveis a aplicação, e em ordem. Outras aplicações podem ver dados antigos, contudo qualquer alteração aparecerá em ordem, como no caso da “Consistent Prefix Option”. Essa forma de consistência é às vezes conhecida por “read your own writes”, ou “veja suas próprias escritas”.

262 - Ainda sobre replicações no Cosmos DB, a chamada “Staleness Option”, ou “Inatividade limitada” é quando há atraso entre a escrita e posterior leitura de dados atualizados. Você especifica essa inatividade assim como o período do tempo ou o número de versões prévias dos dados para os quais estão inconsistentes.

263 - Ainda sobre replicações no Cosmos DB, a chamada “Strong Consistency Option” determina que todas as escritas são visíveis aos clientes após as mudanças serem confirmadas como “escritas bem sucedidas” para todas as réplicas. Essa opção fica indisponível se você precisar distribuir seus dados ao longo de múltiplas regiões globais.

264 - A consistência eventual é a opção do Cosmos DB que provê menor latência e menor consistência.

265 - Você pode utilizar o SAS, acrônimo para “Shared Access Signatures” ou “assinaturas de acesso compartilhado” para permitir direitos limitados aos recursos na conta de armazenamento Azure por um período limitado. Essa feature permite aplicações a acessar recursos como blobs e arquivos sem a necessidade de primeiro realizar uma autenticação.

266 - Um objeto que representa um usuário, grupo, serviço ou identidade gerenciada que esteja requisitando acesso a um recurso Azure é chamado de diretor de segurança ou “security principal”.

267 - Uma vantagem de utilizar replicação multi regional no Cosmos DB é o acréscimo de disponibilidade.

268 - O operador que se utiliza como parte de uma cláusula SELECT para eliminar duplicatas em um dado resultante é o DISTINCT.

269 - As funções de agregação são: COUNT, SUM, AVG, MAX e MIN.

270 - A função de agregação COUNT(p) retorna a contagem do número de instâncias de um campo “p” no conjunto de resultados. Para contar todos os itens de um conjunto de resultados, determine “p” para um valor escalonável, como por exemplo 1.

271 - A função de agregação SUM(p) retorna a soma de todas as instâncias de um campo “p” no conjunto de resultados. O valor de “p” precisa ser numérico.

272 - A função de agregação AVG(p) retorna a média matemática de todas as instâncias do campo “p” em um conjunto de resultados. O valor de “p” precisa ser numérico.

273 - A função de agregação MAX(p) retorna o valor máximo do campo “p” no conjunto de resultados.

274 - A função de agregação MIN(p) retorna o valor mínimo do campo “p” no conjunto de dados.

275 - Um contêiner disponibiliza um modo conveniente de agrupamento de blobs relacionados juntos, e você pode organizar os blobs em uma hierarquia de pastas dentro de um contêiner, similar a um sistema de arquivos do disco.

276 - A utilidade AzCopy é otimizada para transferir grandes arquivos (e blobs) entre um local do computador e o Azure File Storage.

277 - Um Data Warehouse coleta dados de diferentes fontes dentro de uma organização. Esses dados são utilizados posteriormente como fonte de análise, relatórios e processamento analítico online (OLAP). O foco de um data warehouse é disponibilizar respostas para consultas complexas, diferente de um banco de dados relacional, que é focado em performance transacional.

278 - O Azure Data Factory é descrito como um serviço de integração de dados. O propósito do Data Factory é retornar os dados de uma ou mais fontes de dados e convertê-los em um formato que você pode processar.

279 - Seu dado pode conter datas e horários formatados de diferentes formas em diferentes fontes de dados. Você pode usar o Azure Data Factory para transformar esses itens em uma única estrutura uniforme.

280 - Em um Azure Data Factory, uma pipeline pode rodar continuamente conforme os dados são recebidos de diversas fontes. Você pode criar pipelines usando a interface gráfica do usuário disponibilizada pela Microsoft ou escrevendo seu próprio código JSON.

281 - Um Data Lake Storage é um repositório para um grande volume de dados brutos. Por serem esses dados brutos e sem processamento, é muito rápido carregar e atualizar, mas os dados não são colocados em uma estrutura adequada para análise eficiente. Você pode pensar em um Data Lake como um local provisório para você ingerir dados antes de serem tratados e convertidos em formatos adequados para performance das análises.

282 - Você está coletando os dados de diferentes fontes para o Data Warehouse e processamento. Os serviços que você deve utilizar na sequência da arquitetura após a coleta e ingestão desses dados são o Azure Data Factory e o Azure Data Lake Storage.

283 - O Azure Data Lake Storage é essencialmente uma extensão do Azure Blob Storage organizada como um sistema de arquivos quase infinito.

284 - Um Data Lake Storage organiza seus arquivos em diretórios e subdiretórios para aprimorar a organização dos arquivos. Um Blob Storage somente imita uma estrutura de diretórios.

285 - O Azure Databricks é um ambiente Apache Spark rodando no Azure para disponibilizar processamento de Big Data, streaming e machine learning.

286 - O Azure Synapse Analytics é uma engine de analytics. É desenhado para processar grande volume de dados muito rapidamente. Utilizando o Synapse Analytics, você pode ingerir dados de fontes externas, assim como arquivos simples, Azure Data Lake ou outro sistema de gerenciamento de banco de dados, para depois transformar e agregar esses dados em um formato adequado para processar o analytics. Você pode executar consultas complexas ao longo desses dados e gerar relatórios, gráficos e cartões no Power BI.

287 - O Azure Synapse Analytics alavanca uma arquitetura de processamento paralelo massivo, também conhecida por MPP.

288 - O Azure Synapse Analytics inclui um nó de controle e um pool de nós de computação.

289 - O nó de controle é o cérebro da arquitetura. Ele é o frontend que interage com todas as aplicações. A engine MPP roda nesse nó de controle para otimizar e coordenar as consultas paralelas. Quando você submete uma solicitação de processamento, o nó de controle transforma isso em solicitações menores que são executadas em um subconjunto distinto dos dados em paralelo.

290 - Um nó de computação fornece poder computacional. O dado a ser processado é distribuído eventualmente pelos nós. Usuários e aplicações enviam solicitações de processamento ao nó de controle. O nó de controle envia as consultas para os nós de computação, os quais executam as consultas pela porção de dados que cada nó armazenou. Quando cada nó conclui seu processamento, os resultados são enviados de volta ao nó de controle onde foram combinados em um resultado de parâmetro geral.

291 - Azure Synapse Analytics suporta dois modelos computacionais, o SQL pools e o Spark pools.

292 - Nessa arquitetura em uma hipotética ilustração diagramada, o Azure Synapse Analytics legenda todo o contexto que engloba os fatores e indivíduos que representam os nós de controle e de computação até as contas de armazenamento em uma pool. Já o Azure Storage legenda esse pool de armazenamento contida na pool de Analytics.

293 - O Azure Synapse Analytics utiliza a tecnologia chamada PolyBase, que habilita você a retornar os dados de fontes relacionais ou não-relacionais, assim como arquivos de texto com delimitadores, Azure Blob Storage ou Azure Data Lake Storage. Você pode salvar os dados lidos em tabelas SQL dentro do Synapse Analytics Service.

294 - Em uma Spark pool, os nós são substituídos por um cluster spark.

295 - O Azure Analysis Service permite que você construa modelos tabulares para dar suporte às consultas OLAP. Você pode combinar dados de múltiplas fontes, incluindo Azure SQL Database, Azure Synapse Analytics, Azure Data Lake, Azure Cosmos DB e muitos outros. Você pode usar essas fontes de dados para construir modelos que incorporem sua expertise de negócios. O modelo é essencialmente um conjunto de consultas e expressões que retornam dados de várias fontes e gera resultados. Os resultados podem ser armazenados memória cache para uso posterior ou podem ser calculados dinamicamente, diretamente de uma subcamada de fontes de dados.

296 - A diferença objetiva entre Analysis Service e Synapse Analytics é que o primeiro possui significativa sobreposição funcional com o segundo, porém é mais adequado para processamento em escalas menores.

297 - Utilize o Azure Synapse Analytics para datasets com altíssimos volumes de dados (multi-terabytes e petabytes) e consultas e agregações de altíssima complexidade.

298 - Utilize o Azure Analysis Services para pequenos volumes de dados (alguns poucos terabytes) e múltiplas fontes que possam ser correlacionadas.

299 - É possível combinar Analysis Services com Synapse Analytics. Muitos cenários podem se beneficiar utilizando ambos os serviços juntos. Se você possui um grande montante de dados ingeridos que necessitam de pré-processamento, você pode utilizar o Synapse Analytics para ler esses dados e manipulá-los em um modelo que contenha informação de negócios ao invés de grande volume de dados brutos. A escalabilidade do Synapse Analytics te dá a habilidade de processar e reduzir muitos terabytes de dados em datasets menores e mais sucintos que sumarizam e agregam muito desses dados. Você pode depois usar o Analysis Services para executar a interrogação dessas informações detalhadamente e visualizar os resultados dessas perguntas com o Power BI.

300 - Azure HDInsight é um serviço de processamento Big Data que disponibiliza a plataforma para tecnologias como no Spark em um ambiente Azure. HDInsight implementa um modelo clusterizado que distribui processamento ao longo de um conjunto de computadores. Esse modelo é similar ao utilizado pelo Synapse Analytics, exceto que por estarem esses nós rodando no processamento do Spark ao invés do Azure SQL Database.

301 - Você deve utilizar o Azure Synapse Analytics para executar consultas e agregações muito complexas.

302 - O propósito da ingestão de dados é a captura do fluxo de dados para um sistema Data Warehouse o mais rapidamente possível.

303 - A diferença primária entre um Data Lake e Data Warehouse é que um Data Lake armazena dados brutos, enquanto o Data Warehouse armazena informação estruturada.

304 - O componente de um Azure Data Factory que pode ser engatilhado para rodar tarefas de ingestão de dados é uma pipeline.

305 - Você pode utilizar a PolyBase para consultar dados de uma fonte externa a partir do Azure Synapse Analytics.

306 - Você pode utilizar o Azure Data Factory para ingerir dados no Azure Synapse Analytics.

307 - Você possui um grande volume de dados armazenados em arquivos no Azure Data Lake Storage. Você deseja retornar os dados nesses arquivos e usá-los para popular tabelas armazenadas no Azure Synapse Analytics. A opção de processamento mais apropriada para tal é o Synapse SQL pool.

308 - O componente do Azure Synapse Analytics que permite o treinamento de modelos de inteligência artificial utilizando o Azure Machine Learning é o Synapse Spark.

309 - Você pode alterar a linguagem utilizada na cédula de um notebook no Azure Databricks utilizando o sinal de porcentagem no início dela, como por exemplo “%scala” ou “%sql” e “%python”.

310 - Trazer dados para o Power BI Desktop e criar um relatório, compartilhá-lo para um serviço Power BI, visualizar e interagir com esses relatórios e dashboards no serviço e no Power BI Mobile são os fluxos de atividades mais comuns da plataforma Power BI.

311 - Visualizações, datasets, relatórios, dashboards e títulos são chamados de blocos de construção do Power BI.

312 - Uma coleção de visuais pré-prontos e pré-arranjados em dashboards e relatórios é chamada de “app” no Power BI.

313 - Power BI consiste em três elementos principais. São eles o Power BI Desktop, Power BI service e o Power BI Mobile.

314 - Sobre os “building blocks”, ou blocos de construção do Power BI, podemos dizer que as visualizações são representações visuais de dados, por vezes chamadas de “visuals”.

315 - Ainda sobre os building blocks do Power BI, os datasets são coleções de dados que o Power BI usa para criar visualizações.

316 - Ainda sobre os building blocks do Power BI, os relatórios são coleções de visuais de um dataset expostos em uma ou mais abas ou páginas.

317 - Ainda sobre os building blocks do Power BI, os dashboards são coleções de visuais de página única construídos a partir de um relatório.

318 - Ainda sobre os building blocks do Power BI, os títulos são uma visualização singular em um relatório ou dashboards.

319 - Um “app” é uma coleção de presets, visuais e relatórios pré-prontos que são compartilhados com uma organização inteira.

320 - O fluxo mais comum usando o Power BI pode ser descrito pela seguinte sequência: Trazer dados para dentro do Power BI Desktop e criar um relatório. Publicar para o Power BI service, onde você pode criar novas visualizações ou construir dashboards. Compartilhar dashboards com outros, especialmente pessoas que estão integradas a esse movimento como consultores e clientes. Visualizar e interagir com dashboards compartilhados e relatórios nos aplicativos Power BI mobile.

Conclusão

O exame Data Fundamentals é de múltipla escolha, múltiplas respostas, baseadas em texto, drag-and-drop e preenchimento de lacunas em branco. Essa lista definitivamente ajuda você a se preparar para a certificação. Recomendaria que também não deixasse de conferir primeiro a documentação sugerida pela própria Microsoft em seu idioma de origem.