



Roteiro de Estudo



Avaliações



Trabalhos



Livro da Disciplina



Central de Vídeos



Aula Ao Vivo



Tutoria



Fórum



Chat



Avisos

AVALIAÇÃO » NOVO



LEONAM UIRLEY DO NASCIMENTO CASSEMIRO - RU: 4672144 Nota: 100

PROTOCOLO: 2024051346721446B05B24

Disciplina(s):

Banco de Dados Relacional

Data de início: 13/05/2024 21:08

Prazo máximo entrega: -

Data de entrega: 13/05/2024 21:30

Atenção. Este gabarito é para uso exclusivo do aluno e não deve ser publicado ou compartilhado em redes sociais ou grupo de mensagens.

O seu compartilhamento infringe as políticas do Centro Universitário UNINTER e poderá implicar sanções disciplinares, com possibilidade de desligamento do quadro de alunos do Centro Universitário, bem como responder ações judiciais no âmbito cível e criminal.

Questão 1/10 - Banco de Dados Relacional

A cláusula *order by* é usada em consultas SQL (*Structured Query Language*) para classificar os resultados em ordem crescente ou decrescente.

Com base nessa informação, analise o seguinte enunciado:

Considere uma tabela "Pedido", a qual contém as seguintes colunas: "pedidoId", "clienteId", "produtoId", "dataPedido" e "valorPedido".

Assinale a alternativa em que cláusula *order by* é utilizada para classificar a tabela "Pedido" pelo valor do pedido em ordem crescente e, em caso de valores iguais, pela data do pedido em ordem decrescente.

Nota: 10.0

☐ A select * from Pedido order by valorPedido desc, dataPedido;

☒ B select * from Pedido order by valorPedido, dataPedido desc;

Você assinalou essa alternativa (B)

Você acertou!

Nesta consulta, a cláusula *order by* é usada para classificar a tabela "Pedido" pelo valor do pedido (valorPedido) em ordem crescente e, em caso de valores iguais, pela data do pedido (dataPedido) em ordem decrescente.

A cláusula *order by* segue a ordem das colunas especificadas na consulta. Logo, a ordem do valor do pedido vem antes da ordem da data do pedido. Isso significa que os valores do pedido serão classificados primeiro e, em caso de valores iguais, a data do pedido será usada como critério de desempate para classificar os resultados. Vale salientar que a classificação padrão é ascendente (*asc*) e, dessa forma, sua declaração é opcional.

Referência: Aula 4 - Manipulação de Estruturas de Dados; Tema 3 - Restrição de consultas; *Order by*.

☐ C select * from Pedido order by valorPedido desc, dataPedido desc;

☐ D select valorPedido, dataPedido from Pedido order by valorPedido desc, dataPedido;

☐ E select * from Pedido order by valorPedido, dataPedido;

Questão 2/10 - Banco de Dados Relacional

Subconsultas, ou *subqueries*, são linhas de comando *select* que estão incluídas (aninhadas) dentro da estrutura de outros comandos (*select*, *insert*, *update* ou *delete*), inclusive de outras *subqueries*, ou seja, a declaração de uma *query* dentro de outra *query*. A finalidade da *subquery* é retornar um conjunto de linhas para a *query* (consulta) ou comando principal.

Com relação às *subqueries* é correto afirmar que:

Nota: 10.0

☒ A O operador de igualdade, "=", não pode ser utilizado em uma consulta que contenha uma *subquery* que retorne múltiplas linhas.

Você assinalou essa alternativa (A)

Você acertou!

Uma *subquery* que retorna múltiplas linhas não pode ser comparada com um único valor usando o operador de igualdade (=), pois isso resulta em um erro. Nesse caso, outros operadores, como *in* ou *any/some*, devem ser usados para comparar o resultado da *subquery* com uma lista de valores.

Referência: Aula 5 - Comandos Avançados; Tema 1 - *Subqueries*; *Subqueries*.

☐ B A *subquery* não precisa estar incluída entre os parênteses.

- ☐ C A *subquery* deve sempre estar do lado esquerdo do operador de comparação.
- ☐ D A *subquery* pode conter a cláusula *order by*.
- ☐ E O operador *in* não pode ser utilizado em uma *subquery* que retorna múltiplas linhas.

Questão 3/10 - Banco de Dados Relacional

A formatação de dados numéricos é uma técnica utilizada para exibir dados numéricos em um formato específico, de acordo com as necessidades do usuário ou do projeto.


Assinale a alternativa que descreve como a função *mod()* atua em Banco de Dados relacional.

Nota: 10.0

- ☐ A Divide dois números e retorna o resultado inteiro.
- ☐ B Retorna o resultado da divisão de dois números.

☒ C **Retorna o resto da divisão de dois números.**

Você assinalou essa alternativa (C)

 **Você acertou!**

A função *mod()* tem como objetivo retornar o valor correspondente ao resto de uma divisão. Por exemplo, "*mod*(4, 2)" é equivalente a dividir o valor 4 pelo valor 2, o que resultará no valor 0 (zero), sendo, nesse caso, o resultado fornecido pela função *mod()*.

Referência: Aula 5 - Comandos Avançados; Tema 3 - Formatação de dados numéricos e temporais; *Mod()* e *div()*.

- ☐ D Arredonda o resultado da divisão de dois números.
- ☐ E Retorna o resultado da multiplicação de dois números.

Questão 4/10 - Banco de Dados Relacional

Em um Banco de Dados ocorre a execução de múltiplas transações simultaneamente, o que pode provocar diversos problemas de concorrência.

Com base nessa informação, considere o seguinte cenário: A transação T1 lê um dado que sofreu alteração pela transação T2. Porém, a transação T2 ainda não confirmou a modificação.

Assinale a alternativa que contém a nomenclatura desse tipo de problema.

Nota: 10.0

- ☐ A Leitura fantasma.
- ☐ B Leitura defeituosa.
- ☐ C Leitura não repetida.
- ☐ D Leitura duplicável.

☒ E **Leitura suja.**

Você assinalou essa alternativa (E)

 **Você acertou!**

Quando uma transação acessa um dado que foi alterado por outra transação, mas que ainda não foi confirmado (*commit*), ocorre o que é chamado de leitura suja, uma vez que a transação está manipulando uma informação que pode estar incorreta.

Referência: Aula 6 - Procedimentos Armazenados; Tema 1 - Índice e visão (*index* e *view*); Transações.

Questão 5/10 - Banco de Dados Relacional

Em Banco de Dados, uma *function* é, basicamente, um segmento de código ou rotina SQL (*Structured Query Language*) que recebe parâmetros de entrada, processa as instruções definidas e retorna ao ponto de chamada com um resultado. Além das vantagens de reutilização de código, facilidade no processo de manutenção e abstração de rotinas complexas de cálculo e lógica, sua aplicação também é importante em situações onde o MySQL não oferece suporte (de forma nativa) a determinadas situações.

Nesse contexto, sem levar em consideração o uso do delimitador de comando (*delimiter*), assinale a alternativa que apresenta a forma correta de criação de uma *function*.

Nota: 10.0

<input type="radio"/>	A	<code>declare nomeFuncao ([parametro1 tipoDado, ..., parametroN tipoDado]) returns tipoDado begin codigoFuncao end;</code>
<input checked="" type="radio"/>	B	<code>create function nomeFuncao ([parametro1 tipoDado, ..., parametroN tipoDado]) returns tipoDado begin codigoFuncao end;</code> <div>Você assinalou essa alternativa (B)</div> <div><p>Você acertou!</p><p>As palavras-chave <i>create function</i> são usadas para criar uma nova função no Banco de Dados. Logo após essas palavras, é fornecido o nome da função, que poderá conter, entre parênteses, uma lista de parâmetros (opcional), isso se a função precisar de entrada. O próximo passo é declarar a palavra-chave <i>returns</i>, que é usada para especificar o tipo de dado que a função retornará. Finalmente, o corpo da função é definido entre as palavras-chave <i>begin</i> e <i>end</i>. É nesse local que se escreve o código da função, a qual pode incluir instruções <i>select</i>, <i>if</i>, <i>while</i>, entre outras.</p><p>Referência: Aula 6 - Procedimentos Armazenados; Tema 4 - <i>Function</i>; Declaração de uma função.</p></div>
<input type="radio"/>	C	<code>declare nomeFuncao() returns tipoDado begin codigoFuncao end;</code>
<input type="radio"/>	D	<code>create function nomeFuncao() returns tipoDado begin codigoFuncao end;</code>
<input type="radio"/>	E	<code>declare nomeFuncao ([parametro1 tipoDado, ..., parametroN tipoDado]) begin codigoFuncao end;</code>

Questão 6/10 - Banco de Dados Relacional

Um *cursor* é uma estrutura que permite percorrer e manipular linhas em um conjunto de resultados retornados por uma consulta SQL (*Structured Query Language*), podendo acessar o conjunto de linhas retornadas e processar cada um desses conjuntos.

Assinale a alternativa que lista o comando que realiza essa função.

Nota: 10.0		
<input checked="" type="radio"/>	A	Fetch. <div>Você assinalou essa alternativa (A)</div> <div><p>Você acertou!</p><p>O comando <i>fetch</i> é utilizado em cursores em Banco de Dados para recuperar linhas específicas a partir da posição atual do cursor, lendo a linha do cursor que está alocada na memória.</p><p>Referência: Aula 6 - Procedimentos Armazenados; Tema 5 - <i>Cursor</i>, <i>Cursor</i>.</p></div>
<input type="radio"/>	B	Open.
<input type="radio"/>	C	Close.
<input type="radio"/>	D	Repeat.
<input type="radio"/>	E	Read.

Questão 7/10 - Banco de Dados Relacional

Nas atividades desenvolvidas em um Banco de Dados, é comum existirem *scripts* que são executados para diversas finalidades diferentes, os quais podem ser armazenados no Banco de Dados, a fim de estarem disponíveis para serem executados quando for necessário. Tais *scripts* são denominados *stored procedure*.

Assinale a alternativa que contém o comando utilizado para chamar ou executar uma *stored procedure* no MySQL.

Nota: 10.0		
<input type="radio"/>	A	<code>select * from NomeProcedimento (parâmetro(s));</code>
<input type="radio"/>	B	<code>execute NomeProcedimento (parâmetro(s));</code>

<input type="radio"/>	C	invoce NomeProcedimento (parâmetro(s));
<input checked="" type="radio"/>	D	call NomeProcedimento (parâmetro(s)); <div>Você assinalou essa alternativa (D)</div> <div><div>Você acertou!</div><p>O comando <i>call</i> é usado para chamar ou executar uma <i>stored procedure</i>. O nome da <i>stored procedure</i> e seus respectivos parâmetros são passados como argumentos para o comando <i>call</i>.</p><p>Referência: Aula 6 - Procedimentos Armazenados; Tema 3 - <i>Stored procedure</i>; Executar uma <i>procedure</i>.</p></div>
<input type="radio"/>	E	run NomeProcedimento (parâmetro(s));

Questão 8/10 - Banco de Dados Relacional

Em um Banco de Dados, geralmente, tem-se acesso a enormes quantidades de dados que estão em constante crescimento. Por essa razão, as funções de sintetização e de sumarização são fundamentais para o funcionamento de qualquer Banco de Dados, pois permitem agregar dados de várias fontes e transformá-los em informações mais concisas e fáceis de entender, tornando possível a tomada de decisões com base em dados precisos e atualizados, além de identificar padrões e tendências importantes.

Com base nessa informação, assinale a alternativa que corresponde a função que retorna o valor médio de um grupo de registros selecionados com a cláusula *select*.

Nota: 10.0	
<input checked="" type="radio"/>	A Avg. <div>Você assinalou essa alternativa (A)</div> <div><div>Você acertou!</div><p>A função <i>avg()</i> retorna a média dos valores de uma coluna predefinida. Com o uso da cláusula <i>where</i>, inclusive, é possível estabelecer uma condição para determinados valores da coluna.</p><p>Referência: Aula 5 - Comandos Avançados; Tema 4 - Agregação/Extração de dados; <i>Avg()</i>.</p></div>
<input type="radio"/>	B <i>Sum.</i>
<input type="radio"/>	C <i>Between.</i>
<input type="radio"/>	D <i>Distinct.</i>
<input type="radio"/>	E <i>Trim.</i>

Questão 9/10 - Banco de Dados Relacional

O comando *insert* é usado para inserir novos dados em uma tabela. Quando se está inserindo valores em uma coluna, é possível especificar um valor *null* para atribuição a coluna.

Com base nessa informação, o que acontecerá ao inserir um valor *null* em uma coluna com valor padrão definido?

Nota: 10.0	
<input checked="" type="radio"/>	A O valor <i>null</i> será inserido corretamente. <div>Você assinalou essa alternativa (A)</div> <div><div>Você acertou!</div><p>O valor <i>null</i> será inserido corretamente, pois o mesmo é considerado pelo Banco de Dados um valor indefinido, isto é, a coluna em questão não terá um valor definido, mas terá o valor <i>null</i>. Logo, o valor padrão só é inserido se não for especificado nenhum valor.</p><p>Referência: Aula 4 - Manipulação de Estruturas de Dados; Tema 1 - Inclusão de registros; Inclusão de registros.</p></div>
<input type="radio"/>	B Ocorre um erro de violação de chave estrangeira.
<input type="radio"/>	C Ocorre um erro de sintaxe.
<input type="radio"/>	D O valor inserido será o valor padrão definido para a coluna.
<input type="radio"/>	E Ocorre um erro de violação de chave primária.

Questão 10/10 - Banco de Dados Relacional

A cláusula *distinct*, quando usada na *select*, elimina as linhas repetidas, ou seja, se houver duas ou três linhas iguais, somente uma será apresentada no resultado.

Com base nessa informação, analise o seguinte enunciado:

Considere uma tabela "Pedido", a qual contém as seguintes colunas: "pedidoId", "clienteId", "produtoId" e "dataPedido".

Assinale a alternativa em que a cláusula *distinct* seleciona apenas os "clienteId" únicos da tabela "Pedido".

Nota: 10,0



A

select distinct clienteId from Pedido;

Você assinalou essa alternativa (A)



Você acertou!

A consulta, selecionando apenas a coluna "clienteId" e usando a cláusula *distinct*, garante que cada ID de cliente apareça apenas uma vez no resultado da consulta. Em outras palavras, está retornando uma lista de valores únicos da coluna "clienteId", como solicitado.

Referência: Aula 4 - Manipulação de Estruturas de Dados; Tema 5 - Outros comandos e recomendações; *Distinct*.



B

select distinct pedidoId, clienteId from Pedido;



C

select distinct * from Pedido;



D

select clienteId from Pedido unique by clienteId;



E

select distinct count(clienteId) from Pedido;