

Iniciado em quarta, 11 nov 2020, 19:08

Estado Finalizada

Concluída em quarta, 11 nov 2020, 19:08

**Tempo
empregado** 51 segundos

Avaliar 10,00 de um máximo de 10,00(100%)

Questão 1

Completo

Atingiu 2,00 de 2,00

Considerando a constelação de fatos da **BI Solutions**, qual a ordem de processamento lógico da seguinte consulta?

```
SELECT dataMes, SUM(quantidadeLancamentos)
FROM data, pagamento
WHERE data.dataPK = pagamento.dataPK AND dataAno = 2020
GROUP BY dataMes
HAVING SUM(quantidadeLancamentos) > 5
```

Escolha uma opção:

- ☒ a. FROM - WHERE - GROUP BY - HAVING - SELECT
- ☐ b. FROM - SELECT - WHERE - GROUP BY - HAVING
- ☐ c. FROM - GROUP BY - HAVING - WHERE - SELECT
- ☐ d. SELECT - FROM - WHERE - GROUP BY - HAVING
- ☐ e. SELECT - FROM - GROUP BY - HAVING - WHERE

Considere as seguintes afirmações sobre as cláusulas da consulta da linguagem SQL.

I. A especificação de duas relações na cláusula FROM sem a especificação de uma condição de junção referente a essas duas relações na cláusula WHERE gera um produto cartesiano das relações.

II. É possível especificar várias condições de seleção na cláusula WHERE usando os operadores lógicos booleanos AND e/ou OR, sendo que essas condições devem ser referentes a atributos presentes nas relações especificadas na cláusula FROM.

III. A cláusula HAVING, assim como a cláusula WHERE, filtra os resultados finais a serem exibidos como resposta à consulta, porém a cláusula HAVING é definida sobre grupos de conjuntos de tuplas, ao invés de ser aplicada sobre tuplas individuais como a cláusula WHERE.

IV. Na cláusula SELECT podem ser especificados um ou mais atributos, sendo que a ordem de apresentação desses atributos na resposta da consulta é a ordem na qual eles são especificados.

V. A cláusula ORDER BY ordena os dados exibidos como resposta à consulta de modo crescente por padrão.

Indique se cada uma das afirmações é V (Verdadeira) ou F (Falsa).

Escolha uma opção:

- ☐ a. F - V - F - F - V
- ☐ b. V - F - F - V - V
- ☐ c. V - F - V - V - F
- ☒ d. V - V - V - V - V
- ☐ e. V - V - V - F - F

Considere as seguintes afirmações sobre as extensões ROLLUP, CUBE e GROUPING SETS.

I. Todas as três extensões foram introduzidas na proposta original da linguagem SQL e têm como objetivo a criação de subtotais referentes a diferentes níveis de agregação.

II. Considerando a constelação de fatos da BI Solutions, as consultas A e B a seguir produzem as mesmas agregações.

A. SELECT dataPK, equipePK, clientePK, SUM(receita)

FROM negociacao

GROUP BY ROLLUP (dataPK, equipePK, clientePK)

B. SELECT dataPK, equipePK, clientePK, SUM(receita)

FROM negociacao

GROUP BY GROUPING SETS ((dataPK, equipePK, clientePK), (dataPK, equipePK), (dataPK, clientePK), (dataPK))

III. As extensões ROLLUP e CUBE utilizam como base uma lista que especifica cada nível de agregação desejado e criam, a partir desta lista, subtotais que correspondem desde o nível inferior até o nível totalmente agregado.

IV. Considerando a constelação de fatos da BI Solutions, as consultas A e B a seguir produzem as mesmas agregações.

A. SELECT dataPK, equipePK, SUM(receita)

FROM negociacao

GROUP BY CUBE (dataPK, equipePK)

B. SELECT dataPK, equipePK, SUM(receita)

FROM negociacao

GROUP BY GROUPING SETS ((dataPK, equipePK), (dataPK), (equipePK), ())

V. Os subtotais criados pela extensão CUBE podem ser equivalentes aos subtotais criados pelas extensões ROLLUP e GROUPING SETS, mas também podem ser equivalentes a outros subtotais.

Marque:

Escolha uma opção:

- ☐ a. Se somente a afirmativa II está correta.
- ☒ b. Se somente a afirmativa IV está correta.
- ☐ c. Se as afirmativas II e IV estão corretas.
- ☐ d. Se somente a afirmativa V está correta.
- ☐ e. Se as afirmativas I e III estão corretas.

Considerando a constelação de fatos da BI Solutions, assinale a alternativa que corresponde à consulta SQL para: "Qual a quantidade de funcionários que tiveram salários lançados em outubro de 2020, por sexo? Ordene o resultado final pela quantidade de lançamentos"

Escolha uma opção:

- ☐ a.

```
SELECT funcSexo, SUM(quantidadeLancamentos)
FROM funcionario JOIN pagamento JOIN data
WHERE dataMes = 10 AND dataAno = 2020
GROUP BY funcSexo
ORDER BY SUM(quantidadeLancamentos)
```
- ☐ b.

```
SELECT funcSexo, SUM(quantidadeLancamentos)
FROM funcionario JOIN pagamento ON funcionario.funcPK = pagamento.funcPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataMes = 10 AND dataAno = 2020
GROUP BY funcSexo
ORDER BY funcSexo
```
- ☒ c.

```
SELECT funcSexo, SUM(quantidadeLancamentos)
FROM funcionario JOIN pagamento ON funcionario.funcPK = pagamento.funcPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataMes = 10 AND dataAno = 2020
GROUP BY funcSexo
ORDER BY SUM(quantidadeLancamentos)
```
- ☐ d.

```
SELECT funcSexo, SUM(quantidadeLancamentos)
FROM funcionario JOIN pagamento ON funcionario.funcPK = pagamento.funcPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataMes = 10 AND dataAno = 2020
ORDER BY SUM(quantidadeLancamentos)
```
- ☐ e.

```
SELECT SUM(quantidadeLancamentos)
FROM funcionario JOIN pagamento ON funcionario.funcPK = pagamento.funcPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataMes = 10 AND dataAno = 2020
GROUP BY funcSexo
ORDER BY SUM(quantidadeLancamentos)
```

Considerando a constelação de fatos da BI Solutions, assinale a alternativa que corresponde à consulta SQL para: “Qual a média de salários no ano de 2020, considerando cada cargo e seus respectivos níveis? Ordene o resultado final pela média de salários, da maior média para a menor média.”

Escolha uma opção:

- ☐ a. SELECT cargoNome, AVG(salario)
FROM cargo JOIN pagamento ON cargo.cargoPK = pagamento.cargoPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataAno = 2020
GROUP BY cargoNome
ORDER BY AVG(salario)
- ☒ b. SELECT cargoNome, cargoNivel, AVG(salario)
FROM cargo JOIN pagamento ON cargo.cargoPK = pagamento.cargoPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataAno = 2020
GROUP BY cargoNome, cargoNivel
ORDER BY AVG(salario) DESC
- ☐ c. SELECT cargoNome, cargoNivel, AVG(salario)
FROM cargo JOIN pagamento ON cargo.cargoPK = pagamento.cargoPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataAno = 2020 AND cargoNivel = 'SENIOR'
GROUP BY cargoNome, cargoNivel
ORDER BY AVG(salario) DESC
- ☐ d. SELECT cargoNome, cargoNivel, AVG(salario)
FROM cargo JOIN pagamento ON cargo.cargoPK = pagamento.cargoPK
JOIN data ON data.dataPK = pagamento.dataPK
WHERE dataAno = 2020
GROUP BY cargoNome, cargoNivel
ORDER BY AVG(salario)
- ☐ e. SELECT cargoNome, cargoNivel, AVG(salario)
FROM cargo JOIN pagamento ON cargo.cargoPK = pagamento.cargoPK
WHERE dataAno = 2020
GROUP BY cargoNome, cargoNivel
ORDER BY AVG(salario) DESC

[◀ Exercícios - soluções](#)

Seguir para...



[Tarefa Substitutiva da Avaliação Semanal 7 ▶](#)