

Iniciado em	sábado, 2 dez. 2023, 14:37
Estado	Finalizada
Concluída em	sábado, 2 dez. 2023, 14:41
Tempo empregado	3 minutos 15 segundos
Avaliar	95 de um máximo de 100

Questão 1

Correto

Atingiu 20 de 20

Com relação aos conceitos de banco de dados de objetos, marque todas as alternativas corretas:

- ☒ a. Um problema conhecido como divergência de impedância ocorre quando o modelo de dados do banco de dados é diferente do modelo usado na linguagem de programação. Por exemplo, o problema ocorre quando um sistema é desenvolvido em uma linguagem de programação orientada a objetos e os dados estão em um banco de dados relacional. Com isso, é necessário se criar uma camada de software para fazer um mapeamento entre os dois modelos. Bancos de dados de objetos podem ser integrados diretamente à sistemas desenvolvidos usando linguagens de programação orientadas a objetos, evitando o problema de divergência de impedância. Resposta correta
- ☒ b. O *Object Data Management Group* criou o Padrão ODMG para o desenvolvimento de bancos de dados de objetos “puros”. Já os bancos de dados que usam um misto dos modelos relacional e orientados a objetos usam o padrão definido pela linguagem SQL. Resposta correta
- ☐ c. Os conceitos de herança e de polimorfismo são nativos em bancos de dados de objetos, mas podem também ser implementados facilmente em bancos de dados relacionais.
- ☐ d. Todo objeto armazenado tem uma identificação única (OID), a qual não pode ser modificada, e é usada para referenciar o objeto. Um objeto difere de um literal, o qual não tem um OID. Um literal fica armazenado dentro do objeto e não pode ser referenciado diretamente de outros objetos. Na maioria dos bancos de dados de objetos, um número inteiro ou real é armazenado como um objeto.
- ☒ e. Um atributo multivalorado pode ser implementado em um banco de dados de objetos por meio de um construtor do tipo Coleção. Ele inclui uma coleção de outros objetos ou valores, onde todos os elementos são do mesmo tipo. Os principais construtores do tipo Coleção são: set, list, bag, array e dictionary. Resposta correta

Sua resposta está correta.

As respostas corretas são:

Um problema conhecido como divergência de impedância ocorre quando o modelo de dados do banco de dados é diferente do modelo usado na linguagem de programação. Por exemplo, o problema ocorre quando um sistema é desenvolvido em uma linguagem de programação orientada a objetos e os dados estão em um banco de dados relacional. Com isso, é necessário se criar uma camada de software para fazer um mapeamento entre os dois modelos. Bancos de dados de objetos podem ser integrados diretamente à sistemas desenvolvidos usando linguagens de programação orientadas a objetos, evitando o problema de divergência de impedância..

O *Object Data Management Group* criou o Padrão ODMG para o desenvolvimento de bancos de dados de objetos “puros”. Já os bancos de dados que usam um misto dos modelos relacional e orientados a objetos usam o padrão definido pela linguagem SQL..

Um atributo multivalorado pode ser implementado em um banco de dados de objetos por meio de um construtor do tipo Coleção. Ele inclui uma coleção de outros objetos ou valores, onde todos os elementos são do mesmo tipo. Os principais construtores do tipo Coleção são: set, list, bag, array e dictionary.

Questão 2

Correto

Atingiu 20 de 20

Com relação aos conceitos de banco de dados de objetos, marque todas as alternativas corretas:

- ☒ a. Um objeto possui dois componente: estado, que define seu valor, e comportamento, que define as operações aplicadas ao objeto. Uma operação é definida em duas partes: assinatura (ou interface) e método (corpo). A assinatura especifica o nome da operação e seus parâmetros, e o método especifica sua implementação, a qual é escrita em alguma linguagem de programação. Resposta correta
- ☒ b. O conceito de encapsulamento de operações está relacionado ao conceito de tipo abstrato de dados e ocultação de informação. Usuários externos só conhecem a interface (assinatura) das operações, e não o método. Os detalhes de implementação ficam escondidos. Resposta correta
- ☐ c. Objetos persistentes são aqueles que só existem no programa em execução, e desaparecem quando o programa termina. Já os objetos transientes são aqueles que são armazenados no banco de dados e persistem após o término do programa.
- ☒ d. O termo classe se refere a uma definição de tipo, e possui um nome, uma lista de atributos e operações (métodos). Um objeto é sempre de um determinado tipo, e é, portanto, uma instância (como uma variável) de uma classe. Resposta correta
- ☐ e. Usando o conceito de herança, é possível se definir novos tipos com base em outros tipos predefinidos, o que pode levar a uma hierarquia de tipos. Um subtipo herda todos os atributos e operações do tipo predefinido. Quando se tem uma hierarquia de mais de dois níveis, chamamos isso de herança múltipla.

Sua resposta está correta.

As respostas corretas são:

Um objeto possui dois componentes: estado, que define seu valor, e comportamento, que define as operações aplicadas ao objeto. Uma operação é definida em duas partes: assinatura (ou interface) e método (corpo). A assinatura especifica o nome da operação e seus parâmetros, e o método especifica sua implementação, a qual é escrita em alguma linguagem de programação.,

O conceito de encapsulamento de operações está relacionado ao conceito de tipo abstrato de dados e ocultação de informação. Usuários externos só conhecem a interface (assinatura) das operações, e não o método. Os detalhes de implementação ficam escondidos.,

O termo classe se refere a uma definição de tipo, e possui um nome, uma lista de atributos e operações (métodos). Um objeto é sempre de um determinado tipo, e é, portanto, uma instância (como uma variável) de uma classe.

Questão 3

Correto

Atingiu 20 de 20

Com relação às diferenças entre projetos de Banco de Dados de Objetos (BDO) e de Banco de Dados Relacional (BDR), marque todas as alternativas corretas:

- ☒ a. Em um BDO, relacionamentos binários podem ser implementados de forma bidirecional, usando referências aos objetos relacionados em ambos os lados. Já em um BDR, os relacionamentos binários são unidirecionais, especificando-se as referências por meio de chaves estrangeiras. ✔ Resposta correta
- ☒ b. Em um BDR, uma chave estrangeira é adicionada na tabela que referencia apenas uma instância de outra tabela (ou da mesma). Assim, não é possível implementar relacionamentos binários M:N diretamente, necessitando então da criação de uma terceira tabela, onde ficam as chaves estrangeiras. Já em um BDO, pode-se usar o artifício de coleções para se criar uma referência para vários outros objetos e, assim, relacionamentos binários M:N podem ser implementados sem a necessidade de criação de uma terceira classe. ✔ Resposta correta
- ☐ c. O conceito de herança é implementado de forma nativa em um BDO. Já em um BDR, embora a herança não esteja implementada de forma nativa, ela pode ser facilmente simulada usando-se diversas formas de mapeamento. Inclusive, depois da estrutura de herança pronta, é também fácil aplicar o conceito de polimorfismo.
- ☒ d. Em um BDO, ao se especificar as classes, define-se os seus atributos e as suas operações, logo no início do projeto, de forma natural. Isso está embutido no conceito de orientação a objetos. Já na especificação de um BDR, é comum especificar-se apenas os atributos de cada tabela, deixando as operações para uma fase posterior do projeto. ✔ Resposta correta
- ☐ e. Em um BDO, as operações estão associadas aos objetos da classe em que foram definidas, e necessitam de um objeto para serem chamadas, tendo já o objeto como um parâmetro implícito. Já em um BDR, as operações são implementadas por meio de *store-procedures*, os quais estão associados a uma tabela e têm as suas instâncias como objetos implícitos.

Sua resposta está correta.

As respostas corretas são:

Em um BDO, relacionamentos binários podem ser implementados de forma bidirecional, usando referências aos objetos relacionados em ambos os lados. Já em um BDR, os relacionamentos binários são unidirecionais, especificando-se as referências por meio de chaves estrangeiras.,

Em um BDR, uma chave estrangeira é adicionada na tabela que referencia apenas uma instância de outra tabela (ou da mesma). Assim, não é possível implementar relacionamentos binários M:N diretamente, necessitando então da criação de uma terceira tabela, onde ficam as chaves estrangeiras. Já em um BDO, pode-se usar o artifício de coleções para se criar uma referência para vários outros objetos e, assim, relacionamentos binários M:N podem ser implementados sem a necessidade de criação de uma terceira classe.,

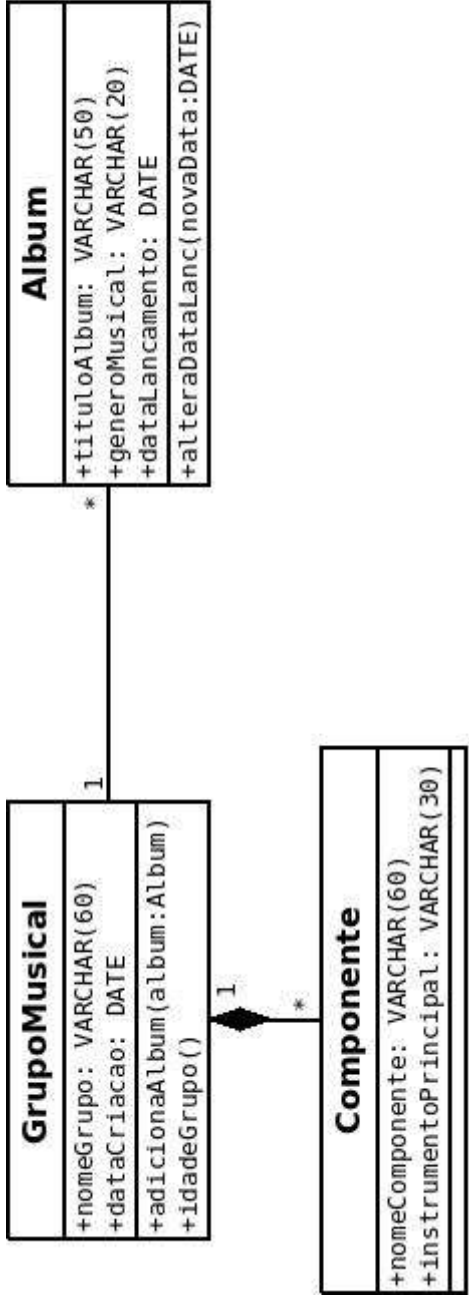
Em um BDO, ao se especificar as classes, define-se os seus atributos e as suas operações, logo no início do projeto, de forma natural. Isso está embutido no conceito de orientação a objetos. Já na especificação de um BDR, é comum especificar-se apenas os atributos de cada tabela, deixando as operações para uma fase posterior do projeto.

Questão 4

Correto

Atingiu 20 de 20

Considere o diagrama de classes abaixo, de um sistema de uma discoteca.



Com relação ao modelo de dados objeto-relacional e a linguagem SQL para definição de dados, marque todas as alternativas corretas:

- ☒ a. O tipo e a tabela para 'GrupoMusical' podem ser criados da seguinte forma:
CREATE TYPE TipoComponente AS (
 nomeComponente VARCHAR(60),
 instrumentoPrincipal VARCHAR(30),
 grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical)
INSTANTIABLE
NOT FINAL
- Resposta correta


```
REF IS SYSTEM GENERATED
```

```
);
```

```
CREATE TYPE TipoGrupoMusical AS (
```

```
    nomeGrupo VARCHAR(60),
```

```
    dataCriacao DATE,
```

```
    componentes REF (TipoComponente) SET SCOPE (Componente),
```

```
    INSTANTIABLE
```

```
    NOT FINAL
```

```
REF IS SYSTEM GENERATED
```

```
INSTANCE METHOD adicionaAlbum (TipoAlbum album) RETURNS BOOLEAN;
```

```
INSTANCE METHOD idadeGrupo () RETURNS INTEGER;
```

```
);
```

```
CREATE TABLE Componente OF TipoComponente;
```

```
CREATE TABLE GrupoMusical OF TipoGrupoMusical;
```

- b. Considerando que a classe para a tabela 'GrupoMusical' está criada com o nome 'TipoGrupoMusical', o tipo e a tabela para 'Album' podem ser criados da seguinte forma:

```
CREATE TYPE TipoAlbum AS (
```

```
    tituloAlbum VARCHAR(50),
```

```
    generoMusical VARCHAR(20),
```

```
    dataLancamento DATE,
```

```
    grupoMusical REF (TipoGrupoMusical) SET SCOPE (GrupoMusical),
```

```
    INSTANTIABLE
```

```
    NOT FINAL
```

```
REF IS SYSTEM GENERATED
INSTANCE METHOD alteraDataLanc (DATE novaData) RETURNS BOOLEAN;
);
```

```
CREATE TABLE Album OF TipoAlbum;
```



c.



No relacionamento entre 'GrupoMusical' e 'Album', criar um atributo em apenas um dos lados do relacionamento é suficiente para se resolver consultas envolvendo objetos dos dois tipos. Entretanto, quando as consultas são muito frequentes em cada lado, e é importante que elas sejam resolvidas muito rapidamente, pode-se criar um atributo de cada lado, implementando-se assim um relacionamento bidirecional. O código abaixo está correto para implementar um relacionamento bidirecional entre os objetos desses dois tipos. Observe o uso do tipo 'SET' para implementar a cardinalidade 'muitos' (indicada no diagrama pelo asterisco).

```
CREATE TYPE TipoGrupoMusical AS (
    ...
    albums REF (TipoAlbum) SET SCOPE (Album),
    ...
);
CREATE TYPE TipoAlbum AS (
    ...
    grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical),
    ...
);
```



d.

O método 'adicionaAlbum' deve ser chamado para adicionar um novo álbum a um grupo musical. Assim, a assinatura do método deve ser criada da seguinte forma em 'TipoGrupoMusical':

```
CREATE TYPE TipoGrupoMusical AS (
    ...
    INSTANCE METHOD adicionaAlbum (TipoAlbum álbum, TipoGrupoMusical grupo) RETURNS BOOLEAN;
);
```

Resposta
correta



- e. O relacionamento entre 'GrupoMusical' e 'Componente' é um relacionamento de composição. Nesse tipo de relacionamento, normalmente, a parte só é acessada de dentro do objeto todo. Assim, não é necessário que o tipo 'TipoComponente' seja instanciável (indicado pela palavra-chave INSTANTIABLE), a tabela 'Componente' não será criada. Além disso, ele não terá um atributo referenciando objetos do tipo 'TipoGrupoMusical'. Logo, o tipo 'TipoComponente' pode ser criado da seguinte forma:

```
CREATE TYPE TipoComponente AS (  
    nomeComponente VARCHAR(60),  
    instrumentoPrincipal VARCHAR(30)  
);
```

Resposta
correta

Sua resposta está correta.

As respostas corretas são:

O tipo e a tabela para 'GrupoMusical' podem ser criados da seguinte forma:

```
CREATE TYPE TipoComponente AS (  
    nomeComponente VARCHAR(60),  
    instrumentoPrincipal VARCHAR(30),  
    grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical)  
    INSTANTIABLE  
    NOT FINAL  
    REF IS SYSTEM GENERATED  
);
```

```
CREATE TYPE TipoGrupoMusical AS (  
    nomeGrupo VARCHAR(60),
```

```
dataCriacao DATE,  
componentes REF (TipoComponente) SET SCOPE (Componente),  
INSTANTIABLE  
NOT FINAL  
REF IS SYSTEM GENERATED  
INSTANCE METHOD adicionaAlbum (TipoAlbum album) RETURNS BOOLEAN;  
INSTANCE METHOD idadeGrupo () RETURNS INTEGER;  
);
```

```
CREATE TABLE Componente OF TipoComponente;  
CREATE TABLE GrupoMusical OF TipoGrupoMusical;;
```

No relacionamento entre 'GrupoMusical' e 'Album', criar um atributo em apenas um dos lados do relacionamento é suficiente para se resolver consultas envolvendo objetos dos dois tipos. Entretanto, quando as consultas são muito frequentes em cada lado, e é importante que elas sejam resolvidas muito rapidamente, pode-se criar um atributo de cada lado, implementando-se assim um relacionamento bidirecional. O código abaixo está correto para implementar um relacionamento bidirecional entre os objetos desses dois tipos. Observe o uso do tipo 'SET' para implementar a cardinalidade 'muitos' (indicada no diagrama pelo asterisco).

```
CREATE TYPE TipoGrupoMusical AS (  
...  
  albums REF (TipoAlbum) SET SCOPE (Album),  
  ...  
);  
CREATE TYPE TipoAlbum AS (  
...  
  grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical),  
  ...
```

```
);
```

O relacionamento entre 'GrupoMusical' e 'Componente' é um relacionamento de composição. Nesse tipo de relacionamento, normalmente, a parte só é acessada de dentro do objeto todo. Assim, não é necessário que o tipo 'TipoComponente' seja instanciável (indicado pela palavra-chave INSTANTIABLE), a tabela 'Componente' não será criada. Além disso, ele não terá um atributo referenciando objetos do tipo 'TipoGrupoMusical'. Logo, o tipo 'TipoComponente' pode ser criado da seguinte forma:

```
CREATE TYPE TipoComponente AS (  
    nomeComponente VARCHAR(60),  
    instrumentoPrincipal VARCHAR(30)  
);
```

Questão **5**

Parcialmente correto

Atingiu 15 de 20

Considere o código SQL abaixo, de criação do BDO de um sistema de uma discoteca.


```
CREATE TYPE TipoComponente AS (  
    nomeComponente VARCHAR(60),  
    instrumentoPrincipal VARCHAR(30)  
    grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical)  
    INSTANTIABLE  
    NOT FINAL  
    REF IS SYSTEM GENERATED  
);
```

```
CREATE TYPE TipoGrupoMusical AS (  
    nomeGrupo VARCHAR(60),  
    dataCriacao DATE,  
    componentes REF (TipoComponente) SET SCOPE (Componente),  
    INSTANTIABLE  
    NOT FINAL  
    REF IS SYSTEM GENERATED  
    INSTANCE METHOD adicionaAlbum (TipoAlbum album) RETURNS BOOLEAN;  
    INSTANCE METHOD idadeGrupo () RETURNS INTEGER;  
);
```

```
CREATE TYPE TipoAlbum AS (  
    tituloAlbum VARCHAR(50),  
    generoMusical VARCHAR(20),  
    dataLancamento DATE,  
    grupoMusical REF (TipoGrupoMusical) SCOPE (GrupoMusical),  
    INSTANTIABLE  
    NOT FINAL  
    REF IS SYSTEM GENERATED  
    INSTANCE METHOD alteraDataLanc (DATE novaData) RETURNS BOOLEAN;  
);
```

```
CREATE TABLE Componente OF TipoComponente;  
CREATE TABLE GrupoMusical OF TipoGrupoMusical;  
CREATE TABLE Album OF TipoAlbum;
```

Com relação à linguagem SQL para consulta a um BDO, marque todas as alternativas corretas:

- ☒ a.  Deseja-se recuperar o nome do componente e o instrumento musical principal de cada membro do grupo musical de nome 'Coldplay'. O comando abaixo está correto para executar a consulta. Observe o uso do símbolo de desreferência (->) no atributo 'grupoMusical', que é do tipo REF, uma referência para um objeto do tipo 'TipoGrupoMusical'. Assim, é possível acessar o valor de qualquer atributo ou método da classe 'TipoGrupoMusical'. Resposta correta
- ☐ b. Deseja-se recuperar o nome do componente e o instrumento musical principal de cada membro do grupo musical de nome 'Coldplay'. O comando abaixo está correto para executar a consulta. Observe o uso do símbolo de desreferência (->) no atributo 'componentes', que é do tipo REF SET, uma referência para um conjunto de objetos do tipo 'TipoComponente'. Assim, é possível
- ```
SELECT C.nomeComponente, C.instrumentoPrincipal
FROM Componente AS C
WHERE C.grupoMusical->nomeGrupo = 'Coldplay'
```

acessar o valor de qualquer atributo ou método da classe 'TipoComponente'.

```
SELECT G.componentes->nomeComponente, G.componentes->instrumentoPrincipal
FROM GrupoMusical AS G
WHERE G.nomeGrupo = 'Coldplay'
```



- c. Deseja-se recuperar o título do álbum e a data de lançamento de cada álbum lançado pelo grupo 'Goo Goo Dolls' após o ano 2000. O comando abaixo está correto para executar a consulta.



Resposta correta

```
SELECT A.tituloAlbum, A.dataLancamento
FROM Album AS A
WHERE A.grupoMusical->nomeGrupo = 'Goo Goo Dolls' AND A.dataLancamento > '31/12/2000'
```



- d. Deseja-se recuperar o nome do grupo, a sua data de criação e o seu tempo de existência (idade) para todos os grupos criados no século XXI. O comando abaixo está correto para executar a consulta. Observe que é possível usar um método da classe para obter o resultado de uma consulta.

```
SELECT G.nomeGrupo, G.dataCriacao, G.idadeGrupo()
FROM GrupoMusical G
WHERE G.dataCriacao >= '01/01/2001'
```



- e. Deseja-se recuperar o nome do grupo e o seu conjunto de componentes para todos os grupos que lançaram álbuns do gênero 'Pop Rock'. O comando abaixo está correto para executar a consulta. Observe que o conjunto de componentes (atributo 'componentes') é do tipo REF SET. Assim, não é possível visualizar na tela o nome dos componentes. Esse tipo de consulta é usado dentro de uma linguagem de programação, que depois irá usar esse conjunto em outra operação.



Resposta correta

```
SELECT DISTINCT A.grupoMusical->nomeGrupo, A.grupoMusical->componentes
FROM Album AS A
WHERE = A.generoMusical = 'Pop Rock'
```

Sua resposta está parcialmente correta.

Você selecionou corretamente 3.



As respostas corretas são:

Deseja-se recuperar o nome do componente e o instrumento musical principal de cada membro do grupo musical de nome 'Coldplay'. O comando abaixo está correto para executar a consulta. Observe o uso do símbolo de desreferência (->) no atributo 'grupoMusical', que é do tipo REF, uma referência para um objeto do tipo 'TipoGrupoMusical'. Assim, é possível acessar o valor de qualquer atributo ou método da classe 'TipoGrupoMusical'.

```
SELECT C.nomeComponente, C.instrumentoPrincipal
FROM Componente AS C
WHERE C.grupoMusical->nomeGrupo = 'Coldplay',
```

Deseja-se recuperar o título do álbum e a data de lançamento de cada álbum lançado pelo grupo 'Goo Goo Dolls' após o ano 2000. O comando abaixo está correto para executar a consulta.

```
SELECT A.tituloAlbum, A.dataLancamento
FROM Album AS A
WHERE A.grupoMusical->nomeGrupo = 'Goo Goo Dolls' AND A.dataLancamento > '31/12/2000',
```

Deseja-se recuperar o nome do grupo, a sua data de criação e o seu tempo de existência (idade) para todos os grupos criados no século XXI. O comando abaixo está correto para executar a consulta. Observe que é possível usar um método da classe para obter o resultado de uma consulta.

```
SELECT G.nomeGrupo, G.dataCriacao, G.idadeGrupo()
FROM GrupoMusical G
WHERE G.dataCriacao >= '01/01/2001',
```

Deseja-se recuperar o nome do grupo e o seu conjunto de componentes para todos os grupos que lançaram álbuns do gênero 'Pop Rock'. O comando abaixo está correto para executar a consulta. Observe que o conjunto de componentes (atributo 'componentes') é do tipo REF SET. Assim, não é possível visualizar na tela o nome dos componentes. Esse tipo de consulta é usado dentro de uma linguagem de programação, que depois irá usar esse conjunto em outra operação.

```
SELECT DISTINCT A.grupoMusical->nomeGrupo, A.grupoMusical->componentes
FROM Album AS A
WHERE = A.generoMusical = 'Pop Rock'
```