

Trabalho Prático do Módulo 2

Entrega 13 out em 21:00**Pontos** 25**Perguntas** 18**Disponível** até 13 out em 21:00**Limite de tempo** Nenhum

Instruções

O Trabalho Prático do Módulo 2 está disponível!

1. Instruções para realizar o trabalho prático

Consulte a data de entrega no teste e em seu calendário.

Reserve um tempo para realizar a atividade e leia as orientações e enunciados com atenção. Em caso de dúvidas, utilize o "Fórum de dúvidas sobre o Trabalho Prático do módulo 2".

Para iniciar o Trabalho Prático, clique em "Fazer teste". Você tem somente **uma** tentativa, e não há limite de tempo definido para realizá-la. Caso você precise interromper a atividade, apenas deixe a página e, ao retornar, clique em "Retomar teste".

Lembre-se de conferir todas as questões antes de enviar a atividade, e clique em "Enviar teste" **somente** quando você tiver terminado.

Uma vez terminado o prazo de entrega, a plataforma enviará as tentativas não finalizadas automaticamente. Por isso, fique atento ao prazo final.

Novas tentativas só serão concedidas perante a apresentação de atestado médico.

O gabarito será disponibilizado partir de sexta-feira, **14/10/2022**, às 23h59.

Bons estudos!

2. O arquivo abaixo contém o enunciado do trabalho prático:

[Enunciado do Trabalho Prático - Módulo 2 - Bootcamp Analista de Banco de dados.pdf](#)

Este teste foi indisponível 13 out em 21:00.

Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	<u>Tentativa 1</u>	4.340 minutos	25 de 25

Pontuação deste teste: **25** de 25

Enviado 13 out em 20:57

Esta tentativa levou 4.340 minutos.

Pergunta 1

1,37 / 1,37 pts

Ao executar o comando de criação do banco **BDProducoes**, passando apenas o parâmetro com o nome do banco, o SQL Server criará o banco de dados com as opções default (collation, tamanho, caminho dos datafiles etc.) da instância.

```
CREATE DATABASE BDProducoes
```

```
GO
```

☐ Falso.

☒ Verdadeiro.

Correto!

Pergunta 2

1,37 / 1,37 pts

Para criar a tabela **Autoria**, foi construído o script abaixo:

```
CREATE TABLE [dbo].[Autoria]
```

```
( [cod_Autoria] [int] IDENTITY(1,1) NOT NULL,
```

```
  [cod_titulo] [int] NOT NULL,
```

```
  [cod_pessoa] [int] NOT NULL,
```

```
  CONSTRAINT [PK_Autoria] PRIMARY KEY CLUSTERED
```

```
    ([cod_Autoria] ASC )
```

```
)
```

```
GO
```

Com base nele, podemos afirmar que:

Correto!

- ☐ Nenhuma das opções.
- ☐ A tabela será criada no banco BDProducoes.
- ☐ A tabela não será criada, pois há um erro no script.

- ☒ A tabela será criada no schema dbo dentro do banco que o usuário estiver usando.

Pergunta 3

1,37 / 1,37 pts

Para garantir que a tabela **Avaliacao** seja criada no banco **BDProducoes**, é preciso:

Executar o script inteiro abaixo:

```
USE [BDProducoes]

GO

CREATE TABLE [dbo].[Avaliacao]
(
    [cod_titulo] [int] NOT NULL,
    [classificacao_media] [int] NOT NULL,
    [qtd_votos] [int] NOT NULL
)
```

- ☐ GO

Usar *Full Qualifier Name* (FQN), como no exemplo abaixo:

- ☐ `CREATE TABLE [BDProducoes].[dbo].[Avaliacao] (.....`

Correto!

- ☒ Todas as afirmativas são corretas.
- ☐ Antes de executar o comando de criação da tabela, executar USE BDProducoes.

Pergunta 4**1,37 / 1,37 pts**

Na criação da tabela ***Titulo***, qual a parte do script abaixo, que contém o recurso de banco de dados, garantirá a NÃO existência de títulos com códigos repetidos?

```
CREATE TABLE [dbo].[Titulo]
(
    [cod_titulo] [int] NOT NULL,
    [nom_titulo] [varchar](1000) NULL,
    CONSTRAINT [PK_Titulo] PRIMARY KEY CLUSTERED
    ( [cod_titulo] ASC      )
)
GO
```

- ☐ [cod_titulo] [int] NOT NULL.
- ☐ NOT NULL.
- ☐ Nenhuma das opções.
- ☒ CONSTRAINT [PK_Titulo] PRIMARY KEY.

Correto!**Pergunta 5****1,37 / 1,37 pts**

Ao executar o comando script abaixo, o que está sendo feito a nível de schema físico?

```
ALTER TABLE [dbo].[Avaliacao] WITH CHECK
```

```
ADD CONSTRAINT [FK_Avaliacao_Titulo]
FOREIGN KEY([cod_titulo])
REFERENCES [dbo].[Titulo] ([cod_titulo])
GO
```



Implementação da garantia de que não seja armazenada a avaliação de um título que não esteja cadastrado.



Criação do relacionamento físico entre as tabelas Avaliacao e Titulo.



Criação de uma constraint.



Todas as opções.

Correto!

Pergunta 6

1,37 / 1,37 pts

A tabela **Pessoa** foi criada usando o script abaixo.

Ao executar o comando ***SELECT * FROM dbo.Pessoa***, é CORRETO afirmar que poderão ser encontradas pessoas sem o ano de nascimento, depois que a tabela for populada?

```
CREATE TABLE [dbo].[Pessoa]
(
    [cod_pessoa] [int] NOT NULL,
    [nom_pessoa] [varchar](500) NOT NULL,
    [ano_nascimento] [int] NULL,
    [ano_falecimento] [int] NULL,
    [dsc_profissao] [varchar](1000) NULL,
    CONSTRAINT [PK_Pessoa] PRIMARY KEY CLUSTERED
    (
        [cod_pessoa] ASC
    )
)
```

GO

Correto!

☒ Sim.

☐ Não.

Pergunta 7

1,37 / 1,37 pts

Considere a query abaixo:

```
SELECT COUNT(*)
```

```
FROM dbo.Titulo
```

```
WHERE nom_titulo IS NULL
```

GO

Ela sempre retornará a **zero (0) linhas**, se:

☐ A tabela estiver vazia.

☒ Todas as opções são corretas.

☐ A coluna nom_titulo for NOT NULL.

Correto!

A tabela tiver sido criada com o script abaixo:

```
CREATE TABLE [dbo].[Titulo]
(
    [cod_titulo] [int] NOT NULL,
    [nom_titulo] [varchar](1000) NULL,
    CONSTRAINT [PK_Titulo] PRIMARY KEY CLUSTERED
    ( [cod_titulo] ASC)
)
GO

ALTER TABLE dbo.Titulo
ALTER COLUMN nom_titulo VARCHAR(1000) NOT NULL
GO
```

Pergunta 8

1,37 / 1,37 pts

Estando a tabela **dbo.Pessoa** populada, NÃO é possível aumentar o tamanho do campo **nom_pessoa** para aceitar até 1.000 caracteres.

☐ Verdadeiro.

☒ Falso.

Correto!

Pergunta 9

1,37 / 1,37 pts

O que deve ser feito para incluir uma coluna como não nula (NOT NULL) em uma tabela já populada com dados?



Adicionar a coluna como nula, inserir valor na nova coluna e depois alterá-la para NOT NULL.



Renomear a tabela atual, criar uma nova tabela, onde a coluna já seja NOT NULL, e depois mover os dados da tabela renomeada para a nova tabela, definindo um valor para a nova coluna (não nula) em questão.

Correto!



Todas as opções são corretas.

Incluir a coluna já com um default, como no exemplo abaixo:

```
ALTER TABLE Titulo ADD ind_status char(1) NOT NULL  
CONSTRAINT DF_Titulo_ind_status2 DEFAULT 'A'
```



GO

Pergunta 10

1,37 / 1,37 pts

A query abaixo retornará à relação dos títulos (e seus detalhes) em **ordem alfabética**:

```
SELECT T.*,D.*  
FROM Titulo T JOIN Titulo_Detalhe D  
ON T.cod_titulo = D.cod_titulo  
ORDER BY T.nom_titulo DESC  
GO
```

Correto!



Falso.



Verdadeiro.

Pergunta 11**1,37 / 1,37 pts**

A query abaixo retornará à relação dos **100 títulos mais bem avaliados**, suas avaliações e total de votos.

```
SELECT TOP (100) T.*,A.*  
FROM Titulo T JOIN Avaliacao A  
ON T.cod_titulo = A.cod_titulo  
ORDER BY A.classificacao_media DESC  
GO
```

☐ Falso.☒ Verdadeiro.**Correto!****Pergunta 12****1,37 / 1,37 pts**

A query abaixo, quando executada, retornará:

```
SELECT T.*, P.nom_pessoa AS "Autor", P2.nom_pessoa AS "Diretor"  
FROM Titulo T LEFT JOIN Autoria A  
ON T.cod_titulo = A.cod_titulo  
LEFT JOIN Pessoa P  
ON A.cod_pessoa = P.cod_pessoa  
LEFT JOIN Direcao D  
ON T.cod_titulo = D.cod_titulo  
LEFT JOIN Pessoa P2  
ON D.cod_pessoa = P2.cod_pessoa  
ORDER BY T.nom_titulo
```

GO

Correto!



Relação dos títulos em ordem alfabética (e seus autores e diretores), existindo ou não essas duas últimas informações.



Somente os títulos que possuam autores e diretores.



Todos os autores e diretores.



Relação dos títulos em ordem alfabética (e seus autores e diretores), somente quando existir essas duas últimas informações.

Pergunta 13**1,37 / 1,37 pts**

Usando a query abaixo, é possível retornar a lista de **títulos sem avaliação**.

```
SELECT *
```

```
FROM Titulo T LEFT JOIN Avaliacao A
```

```
ON T.cod_titulo = A.cod_titulo
```

```
WHERE A.cod_titulo IS NULL
```

GO

Correto!



Verdadeiro.



Falso.

Pergunta 14**1,37 / 1,37 pts**

Qual das queries abaixo retornará os **títulos sem o detalhe da duração ou informação do gênero (dsc_genero)**?

SELECT *

FROM Titulo T JOIN Titulo_Detalhe D

ON T.cod_titulo = D.cod_titulo

☐ WHERE D.qtd_minutos IS NULL AND D.dsc_genero IS NULL

Correto!

SELECT *

FROM Titulo T JOIN Titulo_Detalhe D

ON T.cod_titulo = D.cod_titulo

WHERE D.qtd_minutos IS NULL OR D.dsc_genero IS NULL

☒ GO

☐

SELECT *

FROM Titulo T JOIN Titulo_Detalhe D

ON T.cod_titulo = D.cod_titulo

WHERE D.qtd_minutos IS NOT NULL OR D.dsc_genero IS NOT NULL

SELECT *

FROM Titulo T JOIN Titulo_Detalhe D

ON T.cod_titulo = D.cod_titulo

WHERE D.qtd_minutos = 0 OR D.dsc_genero = 0

☐ GO

Pergunta 15

1,37 / 1,37 pts

Qual das queries abaixo retornará os **títulos sem autor**?

```
SELECT *  
  
FROM Titulo T RIGHT JOIN Autoria A  
  
ON T.cod_titulo = A.cod_titulo  
  
WHERE A.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T FULL JOIN Autoria A  
  
ON T.cod_titulo = A.cod_titulo  
  
WHERE A.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T JOIN Autoria A  
  
ON T.cod_titulo = A.cod_titulo  
  
WHERE A.cod_titulo IS NULL
```

☐ GO

Correto!

```
SELECT *  
  
FROM Titulo T LEFT JOIN Autoria A  
  
ON T.cod_titulo = A.cod_titulo  
  
WHERE A.cod_titulo IS NULL
```

☒ GO

Pergunta 16

1,37 / 1,37 pts

Qual das queries abaixo retornará os **títulos sem diretor**?

Correto!

```
SELECT *  
  
FROM Titulo T FULL JOIN Direcao D  
  
ON T.cod_titulo = D.cod_titulo  
  
WHERE D.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T LEFT JOIN Direcao D  
  
ON T.cod_titulo = D.cod_titulo  
  
WHERE D.cod_titulo IS NULL
```

☒ GO

```
SELECT *  
  
FROM Titulo T RIGHT JOIN Direcao D  
  
ON T.cod_titulo = D.cod_titulo  
  
WHERE D.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T JOIN Direcao D  
  
ON T.cod_titulo = D.cod_titulo  
  
WHERE D.cod_titulo IS NULL
```

☐ GO**Pergunta 17****1,37 / 1,37 pts**

Qual das queries abaixo retornará os **títulos sem elenco**?

```
SELECT *  
  
FROM Titulo T FULL JOIN Elenco E  
  
ON T.cod_titulo = E.cod_titulo  
  
WHERE E.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T JOIN Elenco E  
  
ON T.cod_titulo = E.cod_titulo  
  
WHERE E.cod_titulo IS NULL
```

☐ GO

```
SELECT *  
  
FROM Titulo T RIGHT JOIN Elenco E  
  
ON T.cod_titulo = E.cod_titulo  
  
WHERE E.cod_titulo IS NULL
```

☐ GO

Correto!

```
SELECT *  
  
FROM Titulo T LEFT JOIN Elenco E  
  
ON T.cod_titulo = E.cod_titulo  
  
WHERE E.cod_titulo IS NULL
```

☒ GO

Pergunta 18

1,71 / 1,71 pts

Qual das queries abaixo retornará informações acerca dos títulos **ativos, ordenados alfabeticamente pelo nome do título**, com seu(s) autor(es) e diretor(es)?



```
SELECT T.nom_titulo AS "Nome do Título", UPPER (TD.tip_titulo) AS
"Tipo do Título", TD.ano_lancamento AS "Ano de
Lançamento", TD.qtd_minutos AS "Duração", TD.dsc_genero AS
"Gênero(s)", A.classificacao_media AS "Nota", P.nom_pessoa AS
"Autor", P2.nom_pessoa AS "Diretor"

FROM Titulo T LEFT JOIN Titulo_Detalhe TD

ON T.cod_titulo = TD.cod_titulo

LEFT JOIN Avaliacao A

ON T.cod_titulo = A.cod_titulo

LEFT JOIN Autoria E

ON T.cod_titulo = E.cod_titulo

LEFT JOIN Pessoa P

ON E.cod_pessoa = P.cod_pessoa

LEFT JOIN Pessoa P2

ON D.cod_pessoa = P2.cod_pessoa

WHERE T.ind_status = 'A'

ORDER BY T.nom_titulo

GO
```



```
SELECT T.nom_titulo AS "Nome do Título", TD.tip_titulo AS "Tipo do
Título", TD.ano_lancamento AS "Ano de
Lançamento", TD.qtd_minutos AS "Duração", TD.dsc_genero AS
"Gênero(s)", A.classificacao_media AS "Nota", P.nom_pessoa AS
"Autor", P2.nom_pessoa AS "Diretor"

FROM Titulo T LEFT JOIN Titulo_Detalhe TD

ON T.cod_titulo = TD.cod_titulo

LEFT JOIN Avaliacao A

ON T.cod_titulo = A.cod_titulo

LEFT JOIN Autoria E

ON T.cod_titulo = E.cod_titulo

LEFT JOIN Pessoa P

ON E.cod_pessoa = P.cod_pessoa

LEFT JOIN Direcao D

ON T.cod_titulo = D.cod_titulo

LEFT JOIN Pessoa P2

ON D.cod_pessoa = P2.cod_pessoa

WHERE T.ind_status = 'A'

ORDER BY T.nom_titulo DESC

GO
```




```
SELECT T.nom_titulo AS "Nome do Título", UPPER (TD.tip_titulo) AS
"Tipo do Título", TD.ano_lancamento AS "Ano de
Lançamento", TD.qtd_minutos AS "Duração", TD.dsc_genero AS
"Gênero(s)", A.classificacao_media AS "Nota", P.nom_pessoa AS
"Autor", P2.nom_pessoa AS "Diretor"

FROM Titulo T LEFT JOIN Titulo_Detalhe TD

ON T.cod_titulo = TD.cod_titulo

LEFT JOIN Avaliacao A

ON T.cod_titulo = A.cod_titulo

LEFT JOIN Autoria E

ON T.cod_titulo = E.cod_titulo

LEFT JOIN Pessoa P

ON E.cod_pessoa = P.cod_pessoa

LEFT JOIN Direcao D

ON T.cod_titulo = D.cod_titulo

LEFT JOIN Pessoa P2

ON D.cod_pessoa = P2.cod_pessoa

ORDER BY T.nom_titulo

GO
```

Correto!

```
SELECT T.nom_titulo AS "Nome do Título", UPPER (TD.tip_titulo) AS
"Tipo do Título", TD.ano_lancamento AS "Ano de
Lançamento", TD.qtd_minutos AS "Duração", TD.dsc_genero AS
"Gênero(s)", A.classificacao_media AS "Nota", P.nom_pessoa AS
"Autor", P2.nom_pessoa AS "Diretor"

FROM Titulo T LEFT JOIN Titulo_Detalhe TD

ON T.cod_titulo = TD.cod_titulo

LEFT JOIN Avaliacao A

ON T.cod_titulo = A.cod_titulo

LEFT JOIN Autoria E

ON T.cod_titulo = E.cod_titulo

LEFT JOIN Pessoa P

ON E.cod_pessoa = P.cod_pessoa

LEFT JOIN Direcao D

ON T.cod_titulo = D.cod_titulo

LEFT JOIN Pessoa P2

ON D.cod_pessoa = P2.cod_pessoa

WHERE T.ind_status = 'A'

ORDER BY T.nom_titulo

GO
```

Pontuação do teste: **25** de 25