

Avaliação II – PB2022 – JAVA – Compass.uol

1 – Um novo requisito de sistema foi criado e para isso, será necessário criar uma tabela de produtos no banco de dados. Os atributos são:

- id (primary key)
- nome
- descricao
- desconto
- data

No SQL, qual seria o script que seria capaz de criar a tabela acima?

2 – Supondo que a tabela de produtos existem registros, qual query seria capaz de retornar os registros que estão entre as datas 01/01/2020 e 15/08/2021?

3 – Faça uma consulta que retorne apenas os dados que não estão entre o intervalo de datas 01/01/2020 e 15/08/2021?

4 – Supondo que a tabela de produtos já está criada e com registros, houve a necessidade de criar uma tabela de notas fiscais. Crie a tabela de notas fiscais com os campos:

numero (primary key)

id (foreign key – id do produto)

valor

Qual seria o script que criaria a tabela de notas fiscais com a foreign key do id do produto?

5 – O vendedor da loja BobAlice Modas, precisa saber quais são os números das notas fiscais que possuem o id de produto 455644.

Sendo assim, considerando a tabela de produtos e notas fiscais com registros, qual query você faria para retornar essas notas?

6 – O requisito mudou e agora o campo “numero” da tabela de notas fiscais será renomeado para “order_id”. Sendo assim, qual script seria capaz de alterar o nome desse campo sem a necessidade de recriar a tabela toda?

7 – O produto 455644 foi descontinuado e o cliente solicitou que todos os produtos que tiverem o id 455644, serão alterados para o id de produto 243924. Como você faria essa query?

8 – Outro requisito chegou e foi verificado que as tabelas de produtos e notas fiscais não possuem constraints. Sendo assim, crie um script para cada tabela com uma constraint (com nome e no campo que acharem mais viável).

9 – Crie uma entidade produto, com os seguintes atributos:

- id
- nome
- descricao
- desconto
- data inicio

Sua aplicação deve oferecer no método main ao menos 5 opções no console usando a classe Scanner.

A primeira opção deve cadastrar ofertas e ao selecioná-la, deve cadastrar automaticamente 3 ofertas com valores diferentes;

A segunda opção é “*atualizar oferta*”, onde a atualização será feita através do id.

A terceira opção será “*excluir oferta*” pelo id;

A quarta opção deverá listar pelo menos 3 produtos , de acordo com a string passada pelo Scanner. Além disso, os produtos que serão mostrados, podem conter a palavra no início, meio o fim de um registro do banco. Se por exemplo eu passar “xpto notebook”, deveria mostrar algo assim:

id: 13 – notebook gamer

id 18 – tela de notebook gamer

id 22 – notebook de loja de geladeira

A quinta opção terá a função de “Sair” ao digitar uma tecla específica. Pode ser por exemplo o “0”.

Obrigatório: É importante ter uma **validação se o id já existe antes de executar qualquer operação**.

Por exemplo, se for update em um id que não existe, automaticamente deve se implementar um insert, já que o id passado não existe.

Se o id existir, segue a operação normalmente.

Exemplo de saída do console (não necessariamente precisa ser com o mesmo texto), porém, com todos os requisitos acima.

```
===== XPTO System =====
Digite a opção desejada:
1 - para INSERIR uma nova oferta
2 - para ATUALIZAR uma oferta
3 - para DELETAR uma nova oferta
4 - para listar as palavras que contem ?
0 - para SAIR
```

10 – Emoticons são símbolos usados para expressar o sentimento de quem escreve uma mensagem. Scott Fahlman, um professor de uma universidade americana, foi o primeiro a propor o uso das sequências de caracteres :-) e :-(, que viraram respectivamente símbolos para “divertido” e “chateado”.

Nosso cliente, está criando uma aplicação que basicamente quer saber como seus funcionários estão se sentindo baseado nos emoticons citados acima e digitados por eles.

Dada uma mensagem composta por uma cadeia de caracteres, escreva um programa para determinar o sentimento expresso na mensagem. O seu programa deve usar a classe Scanner para ler uma linha de entrada e a partir dessa linha devolver o sentimento expresso e salvar no banco de dados.

- a) neutro: se o número de símbolos “divertido” é igual ao número de símbolos “chateado”
- b) divertido: se o número de símbolos “divertido” é maior do que número de símbolos “chateado”
- c) chateado: se o número de símbolos “chateado” é maior do que número de símbolos “divertido”.

Exemplos de entrada/saída:

Entrada: Achei o filme muito divertido.
Saída: neutro

Entrada: :-):-(:-(:-)
Saída: neutro

Entrada: Sonhei com Java. :-((vou estudar).
Saída: chateado

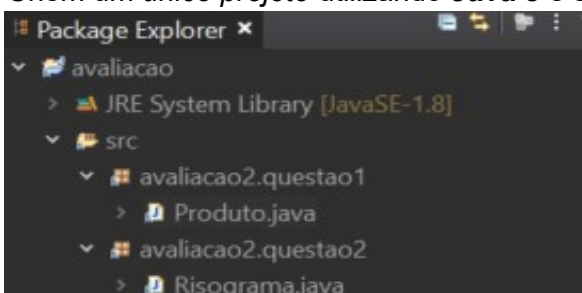
- Informações importantes e obrigatórias:

É **obrigatório** implementação com o **tratamento de exceção**.

Utilize POO para criar o código, utilizem as dicas dadas no feedback de melhoria.

Já implementou todos os requisitos e quer realizar um plus? Será muito bem recebido!

Criem um *único projeto* utilizando **Java 8** e separem as atividades por pacotes. Por exemplo:



Envie no corpo do e-mail o(s) arquivo(s) .sql das questões de 1 a 8 (pode ser feito em único .sql). O código deve ser hospedado no github pessoal e deve ser enviado no corpo do e-mail para a lista abaixo:

gionni.gomes@compasso.com.br; gabriel.sobreira@compasso.com.br;
giovanni.hoffmann@compasso.com.br; diego.scacinate@compasso.com.br;
jailson.lima@compasso.com.br

O **prazo final** para entrega da avaliação será até o dia 20/02 até as 23:59.