

Pratique – Valor 3.0 pontos

1- Crie um novo projeto com nome no Eclipse. No projeto, adicione 4 pacotes:

- br.com.concessionaria.veiculos
- br.com.concessionaria.oficina
- br.com.concessionaria.funcionarios
- br.com.concessionaria.estrutura

2- Implementar a classe no referido pacote, conforme mostrado na tabela abaixo:

Classe	Pacote
Veiculo	br.com.concessionaria.veiculos
Oficina	br.com.concessionaria.oficina
Mecanico	br.com.concessionaria.funcionarios
Peca	br.com.concessionaria.estrutura

3- Atributos das classes:

Veiculo	Oficina	Peca	Mecanico
+ placa: String	+ nomeOficina: String	+ nome: String	+ nome: String
+ ano: int	+ listaMecanicos: lista <Mecanico>	+ valor: float	+ anosExperiencia: int
+ modelo: int	+ listaPecas: lista <Peca>	+ dataCompra: String	+ carrosSimultaneos: int
+ numeroChassi: String	+ listaVeiculos: lista <Veiculo>		
+ cor: String	+ pecasNecessarias: int		
+ kmRodados: int			

4- Na tabela abaixo são definidos os parâmetros obrigatórios no construtor de cada classe:

Classe	Parâmetros do Construtor
Veiculo	Todos os atributos da classe
Oficina	Nome da oficina
Mecanico	Todos os atributos da classe
Peca	Todos os atributos da classe

5- Na tabela abaixo são definidos os métodos que classe possui, com os parâmetros necessários em cada método:

Classe	Método	Parâmetros	Retorno
Veiculo	informarQuilometragem	quilometragem: int	-----
Oficina	adicionarMecanico	mecanico: Mecanico	-----
	removerMecanico	mecanico: Mecanico	-----
	adicionarPeca	peca: Peca	-----
	removerPeca	peca: Peca	-----
	adicionarVeiculo	veiculo: Veiculo	
	removerVeiculo	veiculo: Veiculo	
	verificarPecasRevisao	-----	int: N° aleatório (de 1 a 3) de peças a serem utilizadas
	realizarRevisaoVeiculos	-----	boolean: informando se foi possível fazer simultaneamente as revisões
	valorTotalPecas	-----	float: retornar a soma do valor das peças na oficina. Peças necessárias x valor de cada peça.

No objeto Oficina:

- Utilizar o método verificarPecasRevisao e realizarRevisaoVeiculos. Se não houverem carros a serem revisados, de acordo com a lista de veículos no atributo da classe, informar através de uma mensagem;
- Se houver um ou mais carros, o método verificarPecasRevisao deve armazenar no atributo pecasNecessarias e retornar o total de peças necessárias para realizar as revisões, além de exibir uma mensagem com o total de peças que serão necessárias. Todas as peças, apesar de ter nomes e valor diferentes serão consideradas iguais. Para retornar o total de peças, para cada veículo que estiver na lista do atributo da classe, gere um número aleatório (de 1 a 3) de peças e some no atributo pecasNecessarias da classe.
- Ao utilizar o método realizarRevisaoVeiculos, deve-se verificar:
 - Se há veículos a serem revisados;
 - Se no atributo de lista de peças (listaPecas) há número suficiente para realizar as revisões de todos os veículos simultaneamente (comparar o atributo pecasNecessarias com a listaPecas).
 - Se há mecânicos suficientes, sendo o atributo carrosSimultaneos da classe Mecanico o meio de saber quantos mecânicos são necessários para atender os veículos na oficina.

- Caso haja número suficiente de peças e mecânicos, exibir as placas dos carros, os nomes de todos os mecânicos na oficina, e os dados de todas as peças na oficina.
- Ao adicionar ou remover um veículo da oficina, além de adicionar/remover o veículo da lista de veículos, deve-se adicionar/remover um número aleatório (de 1 a 3) de peças na lista de peças.

6- Criar uma classe Main com um método main. Dentro do main realizar as seguintes ações

- Declarar 3 objetos diferentes da classe Veiculo.
- Declarar 10 objetos diferentes da classe Peca.
- Declarar 3 objetos diferentes da classe Mecanico.
- Declarar 1 objeto da classe Oficina.
- Adicione no objeto Oficina: veículos, motoristas e peças e utilize os métodos verificarPecasRevisao e realizarRevisaoVeiculos.

Método de Avaliação:

1. (0.5 pontos) se todos os pacotes e todas as classes estiverem criados
2. (0.5 pontos) se as classes contiverem todos os atributos, incluindo os getters e setters, e os métodos solicitados com os parâmetros exigidos.
3. (1.0 pontos) se o método verificarPecasRevisao realizar todas as verificações solicitadas e retornar o número correto de peças.
4. (1.0 pontos) se o método realizarRevisaoVeiculos realizar todas as verificações solicitadas e exibir as mensagens de acordo com as validações efetuadas.

Para compensar o valor não obtido nas avaliações de 1 a 4, a questão 5 será utilizada para ajudar na pontuação. Quem conseguir os 3.0 pontos nas questões anteriores não terá direito à pontuação extra desta questão.

5. (0.5 pontos) no método valorTotalPecas da classe Oficina, exibir o nome e o valor de cada peça constante no atribuo listaPecas da classe. Retornar o valor total de peças na oficina, somando o valor de cada peça da lista.