

DEVinHouse

Módulo 3 - Projeto Avaliativo 1

SUMÁRIO

1 INTRODUÇÃO	1
2 REQUISITOS DA APLICAÇÃO	1
3 ROTEIRO DA APLICAÇÃO	1
4 CRITÉRIOS DE AVALIAÇÃO	3
5 ENTREGA	4
6 PLANO DE PROJETO	5

1 INTRODUÇÃO

Considerando os assuntos estudados nas semanas anteriores do módulo 2, vamos aplicar todos os conceitos estudados resolvendo o problema abordado no desenvolvimento do software solicitado pelo cliente **DEVinCar**.

2 REQUISITOS DA APLICAÇÃO

A aplicação que deverá ser realizada **individualmente** deve contemplar os seguintes requisitos:

- O sistema deverá ser desenvolvido em Python 3.
- O sistema deve seguir o Roteiro da Aplicação.
- O desenvolvimento das mensagens de saída, esperando pela ação do usuário;
- Captura da interação do usuário via entrada padrão;
- O sistema deverá ser apresentado diretamente na linha de comando;

3 ROTEIRO DA APLICAÇÃO

A montadora/revenda de veículos **DEVinCar** deseja automatizar todo o sistema de armazenamento de informações referentes aos carros fabricados. O sistema deve conter os seguintes tipos de carros, cada um com suas características:

- Moto/Triciclo
 - Uma moto/triciclo possui como característica principal sua potência.

- Deve-se informar o total rodas da moto/triciclo.
- Carro
 - Carro possui como características principais:
 - Total de portas.
 - Se será flex ou somente gasolina.
 - Potência (em cavalos).
- Camionete:
 - Carro possui como características principais:
 - Total de portas.
 - Capacidade de carregamento na caçamba (em litros).
 - Potência (em cavalos).
 - Se será gasolina ou diesel.
 - Um detalhe importante é que a **DEVinCar** só fabrica caminhonetes na corte Roxa.

Todos os veículos devem ser derivadas da classe Veículo, que possui os seguintes atributos e métodos:

- Veículo
 - Atributos
 - Número do chassi (o sistema deverá gerar de forma automática e não poderá repetir)
 - Data de Fabricação
 - Nome
 - Placa
 - Valor
 - CPF do Comprador (quando cadastrado deve ficar em zero).
 - Cor
- Métodos
 - Vender veículo
 - Listar informações
 - Alterar informações:
 - Cor e valor

O **DEVinCar** também deseja manter um histórico das transferências que deverá armazenar (utilizar conceitos de composição):

- Dados do Veículo
- CPF do comprador
- Valor
- Data

O sistema também deverá apresentar os seguintes relatórios:

- Listar todos os carros. O usuário deverá informar quais veículos quais tipos de veículos deseja listar:
 - Motos/Triciclo
 - Carros
 - Camionete
 - Todos
- Carros disponíveis
- Carros vendidos
- Carro vendido com o maior preço
- Carro vendido com o menor preço

4 CRITÉRIOS DE AVALIAÇÃO

A tabela abaixo apresenta os critérios que serão avaliados durante a correção do projeto. O mesmo possui variação de nota de 0 (zero) a 10 (dez) como nota mínima e máxima, e possui peso de **45% sobre a avaliação do módulo**.

Serão **desconsiderados e atribuída a nota 0 (zero)** os projetos que apresentarem plágio de soluções encontradas na internet ou de outros colegas. Lembre-se: Você está livre para utilizar outras soluções como base, mas **não é permitida** a cópia.

Nº	Critério de Avaliação	0	0,5	1
1	O aluno seguiu as convenções de código da linguagem?	O aluno não deixou seu algoritmo organizado.	O aluno deixou seu algoritmo parcialmente organizado.	O aluno deixou seu algoritmo organizado.
2	O aluno apresentou um menu de fácil interação ao usuário?	O aluno não apresentou o menu.	O aluno apresentou o menu, contudo, de forma confusa.	O aluno apresentou o menu de forma clara para o usuário.
3	O aluno utilizou o conceito de heranças?	O aluno não utilizou o conceito de herança.	O aluno utilizou, parcialmente, os conceitos de heranças.	O aluno utilizou de forma correta o conceito de herança.
4	O aluno criou tratamentos de erros seguindo o descrito no roteiro da aplicação?	O aluno não realizou tratamento de erros	O aluno realizou o tratamento de erros, porém não de forma aderente aos princípios da linguagem e da	O aluno realizou o tratamento de erros corretamente por meio de estruturas

			orientação à objetos	específicas e reutilizáveis
Nº	Critério de Avaliação	0	1	2
5	O aluno implementou todas as classes com seus respectivos atributos e métodos?	O aluno não implementou as classes.	O aluno implementou de forma parcial as classes, atributos e métodos	O aluno implementou todas as classes, juntamente com seus atributos e métodos.
6	O aluno implementou e apresentou os relatórios solicitados?	O aluno não implementou os relatórios.	O aluno implementou e apresentou, de forma parcial, os relatórios.	O aluno implementou e apresentou todos os relatórios.
7	O aluno implementou todas as operações bancárias e a classe de transações?	O aluno não implementou as operações bancárias e nem a classe de transações.	O aluno implementou, de forma parcial, as operações bancárias e nem a classe de transações.	O aluno implementou todas as operações bancárias e nem a classe de transações.

5 ENTREGA

O código desenvolvido deverá ser submetido no [GitHub](#), e o link deverá ser disponibilizado na tarefa **Módulo 2 - Projeto Avaliativo 1**, presente na semana 6 do AVA até o dia **10/07/2022 às 23h55**.

O repositório deverá ser **privado**, com as seguintes pessoas adicionadas:

- Yan Esteves -
- Operação DEVinHouse - [devinhouse-operacao](#)

Não serão aceitos projetos submetidos após a data limite da atividade, e, ou alterados depois de entregues.

Importante:

1. Será considerado como data final de entrega a **última atualização** no repositório do projeto no GitHub. Lembre-se de não modificar o código até receber sua nota.
2. Não esqueça de submeter o link no AVA.

6 PLANO DE PROJETO

Ao construir a aplicação proposta, o aluno estará colocando em prática os aprendizados em:

- **Programação Orientada a Objetos:** Conceitos de POO, Classes, Objetos, Métodos de Classes e Métodos Estáticos, Encapsulamento, Herança, Polimorfismo, Tratamento de Exceções, Time e Decorators.
- **Modelagem:** Criação de Classes e Abstração
- **Python:** Ambiente virtual, Variáveis, Operadores lógicos, aritméticos e de comparação, Métodos, Listas, Tuplas, Sets, Condicionais, Strings, Loops, Compreensão de Listas, Dicionários e Funções