



UNIVERSIDADE FEDERAL DO CEARÁ
CURSO DE TECNÓLOGO EM SEGURANÇA DA INFORMAÇÃO

JOSÉ BASÍLIO BRANDÃO COELHO
RYAN GUILHERME

CRIAÇÃO DE UM CARRO ECOLÓGICO EM PROGRAMAÇÃO
ORIENTADA A OBJETO NA LINGUAGEM PYTHON

ITAPAJÉ
2023

Matéria: Desenvolvimento de Sistemas

Curso de Segurança da Informação

Semestre 2023.2

Relatório de Trabalho - Carro Ecológico

Autores:

Ryan Guilherme

Basílio Brandão

Resumo:

Este relatório descreve o desenvolvimento de uma classe em Python para um carro ecológico. O projeto envolve a implementação de métodos e atributos que permitem operações como entrar, sair, abastecer, dirigir, abrir e fechar o teto solar, buzinar e fornecer informações sobre o carro.

Introdução

Neste projeto, criamos uma classe denominada "Carro_Ecologico" que representa um veículo ecológico. A classe permite simular várias operações que um carro real pode realizar.

Desenvolvimento

Classe Carro_Ecologico

A classe "Carro_Ecologico" foi implementada com os seguintes atributos:

- tanque: Representa o nível do tanque de combustível.
- quilometragem: Registra a quilometragem percorrida.
- pessoas: Contabiliza o número de pessoas a bordo.
- max_pessoas: Define o número máximo de pessoas permitidas no carro.
- teto_solar_aberto: Indica se o teto solar está aberto.

O método construtor da classe inicia os valores iniciais dos atributos, com o tanque em 0, a quilometragem em 0, o número de pessoas em 0 e o valor máximo de pessoas definido como 2.

Métodos da Classe

- Método "entrar": Verifica se a quantidade de pessoas para entrar é menor que a quantidade limite de pessoas permitidas, se for menor, ele incrementa esse valor das pessoas.
- Método "sair": Se a quantidade de pessoas no carro for maior que 0, o valor de pessoas vai receber -1, se não, não existe ninguém dentro do carro.
- Método "abastecer(litros)": Ele pega o valor do atributo tanque e soma com a quantidade de litros que a pessoa quer abastecer, tendo o limite de até cem litros, caso ultrapasse o limite definido, a quantidade que passou vai ser descartada.
- Método "dirigir(distancia)": Inicia verificando a quantidade de pessoas no carro, se a quantidade de pessoas no carro for maior do que 0, ele vai começar a andar, caso a quantidade de pessoas seja 0, ele não vai andar, a contagem dos quilômetros ocorre se houver pelo menos 1 pessoa no carro, o método min é usado para determinar o valor mínimo entre o nível de água atual no tanque (self.tanque) e a distância desejada (distancia).

Isso garante que o carro não tente percorrer mais distância do que a quantidade de água disponível no tanque, a atualização do estado do carro com base na distância máxima calculada, primeiro subtrai-se o valor de `kilometros_maximos` com o nível do tanque (`self.tanque -= kilometros_maximos`) e depois a mesma quantidade de quilômetros é adicionada a quilometragem total percorrida (`self.kilometragem += kilometros_maximos`).

Após atualizar o tanque e a quilometragem, o código verifica se a distância máxima percorrida (`kilometros_maximos`) é menor do que a distância total desejada (`distancia`), `if kilometros_maximos < distancia`. Essa verificação indica que o carro não conseguiu percorrer toda a distância desejada com a água disponível no tanque. Nesse caso, o método retorna uma mensagem informando que a viagem não foi concluída totalmente e especifica quantos quilômetros foram possíveis percorrer. Se o carro conseguir percorrer a distância total desejada com a água

disponível no tanque, o método retornará uma mensagem indicando quantos quilômetros foram percorridos com sucesso.

- Método "abrir_teto_solar": É usada a condição if junto com o not para dizer que se o teto solar está fechado, ele vai abrir atribuindo o True para teto_solar_aberto, e vai receber a mensagem "Teto solar abriu", se não, a mensagem "Teto solar já está aberto".

- Método "fechar_teto_solar": Verificamos com a condição if se o teto_solar_aberto é igual a False, se sim recebe a mensagem "Teto solar fechou", se não, recebe a mensagem "Teto solar já está fechado".

- Método "buzinar": Caso alguém utilize a buzina, recebe a mensagem "Bip bip, sai da frente!!".

- Método "info": Fornecer informações detalhadas sobre o carro, incluindo o status do teto solar, o nível do tanque, a quilometragem e o número de pessoas a bordo.

Conclusão

O projeto de desenvolvimento de uma classe de carro ecológico foi concluído com êxito. Todos os métodos foram implementados e testados. O código demonstra o funcionamento adequado das operações relacionadas a um carro ecológico.

Referências

pythonando. O guia completo de POO com python. 2022.

YouTube. https://youtu.be/jeLeW6q9Mo4?si=2vVFFcSyWt35l2_m

Hashtag Programação. Como sair do zero em Classes no python - self e init explicados.2022.Youtube.

<https://youtu.be/gomDSZaay3E?si=G9tDdPVsW-XtxXYn>

pythonando. O guia completo de POO com python - staticmethods vs class methods. 2022. YouTube.

<https://youtu.be/gLaZQYZjcOE?si=8ZZwzscbY52yhYyp>

pythonando. O guia completo de POO com python - herança e sobreposição. 2022. YouTube. <https://youtu.be/Mjyw5-gv2kY?si=IkRHQ9iOAWjrP-Ra>

