

Lista POO Nível 4 Herança 2.0

Exercício 5)

a) Crie uma classe "Animal" com os atributos "nome" e "especie". Adicione um método "fazer_som" que retorna uma string com o som característico do animal.

b) Crie uma classe "Cachorro" que herda da classe "Animal". A classe "Cachorro" deve ter um atributo adicional chamado "raca". Sobrescreva o método "fazer_som" na classe "Cachorro" para incluir o som "au au!".

c) Crie uma classe "Gato" que também herda da classe "Animal". A classe "Gato" deve ter um atributo adicional chamado "cor_pelagem". Sobrescreva o método "fazer_som" na classe "Gato" para incluir o som "miau!".

d) Crie objetos das classes "Cachorro" e "Gato" e chame o método "fazer_som" para cada um deles.

Exercício 6)

a) Crie uma classe "Pessoa" com os atributos "nome" e "idade". Adicione um método "apresentar" que retorna uma string com a apresentação da pessoa (exemplo: "Olá, meu nome é [nome] e tenho [idade] anos.>").

b) Crie uma classe "Estudante" que herda da classe "Pessoa". A classe "Estudante" deve ter um atributo adicional chamado "curso". Sobrescreva o método "apresentar" na classe "Estudante" para incluir o curso na apresentação.

c) Crie uma classe "Professor" que também herda da classe "Pessoa". A classe "Professor" deve ter um atributo adicional chamado "disciplina". Sobrescreva o método "apresentar" na classe "Professor" para incluir a disciplina na apresentação.

d) Crie objetos das classes "Estudante" e "Professor" e chame o método "apresentar" para cada um deles.

Exercício 7)

a) Crie uma classe "Ponto" com os atributos "x" e "y". Adicione um método "mover" que recebe as coordenadas "dx" e "dy" e atualiza os valores de "x" e "y" somando as coordenadas recebidas.

b) Crie uma classe "Ponto3D" que herda da classe "Ponto". A classe "Ponto3D" deve ter um atributo adicional chamado "z". Sobrescreva o método "mover" na classe "Ponto3D" para receber uma terceira coordenada "dz" e atualizar também o valor de "z".

c) Crie objetos das classes "Ponto" e "Ponto3D" e teste o método "mover" para cada um deles.

Exercício 8)

a) Crie uma classe "Funcionario" com os atributos "nome" e "salario". Adicione um método "calcular_salario_anual" que retorna o salário anual multiplicando o salário mensal por 12.

b) Crie uma classe "Gerente" que herda da classe "Funcionario". A classe "Gerente" deve ter um atributo adicional chamado "bonus_anual". Sobrescreva o método "calcular_salario_anual" na classe "Gerente" para incluir o bônus anual no cálculo.

c) Crie uma classe "Estagiario" que também herda da classe "Funcionario". A classe "Estagiario" deve ter um atributo adicional chamado "meses_contrato". Sobrescreva o método "calcular_salario_anual" na classe "Estagiario" para considerar o número de meses do contrato no cálculo do salário anual.

Desafio:

a) Crie uma classe "Tarefa" com os atributos "titulo" e "prioridade" (sendo a prioridade um valor inteiro de 1 a 5, onde 1 é a mais alta e 5 é a mais baixa). Adicione um método "exibir_tarefa" que exibe os detalhes da tarefa.

b) Crie uma classe "TarefaSimples" que herda da classe "Tarefa". A classe "TarefaSimples" deve ter um atributo adicional chamado "descricao". Sobrescreva o método "exibir_tarefa" na classe "TarefaSimples" para exibir todos os detalhes da tarefa, incluindo a descrição.

c) Crie uma classe "TarefaComPrazo" que também herda da classe "Tarefa". A classe "TarefaComPrazo" deve ter um atributo adicional chamado "prazo". Sobrescreva o método "exibir_tarefa" na classe "TarefaComPrazo" para exibir todos os detalhes da tarefa, incluindo o prazo.

d) Crie uma classe "TarefaRecorrente" que também herda da classe "Tarefa". A classe "TarefaRecorrente" deve ter um atributo adicional chamado "frequencia" (por exemplo, diária, semanal, mensal). Sobrescreva o método "exibir_tarefa" na classe "TarefaRecorrente" para exibir todos os detalhes da tarefa, incluindo a frequência.

e) Crie um sistema que permita ao usuário cadastrar tarefas do tipo "TarefaSimples", "TarefaComPrazo" e "TarefaRecorrente". O usuário deve ser capaz de visualizar todas as tarefas cadastradas, bem como exibir os detalhes de cada uma delas usando o método "exibir_tarefa".