

Lista de exercícios 02: POO

Profº.: Cainã Antunes Silva

Faculdade de Tecnologia **SENAI Sorocaba**

Tecnólogo em Análise e Desenvolvimento de Sistemas (ADS)

O objetivo desta lista de exercícios é praticar a tarefa de converter as abstrações realizadas utilizando diagrama de classe em código, colocando em prática a essência do paradigma de programação orientada à objetos.

O paradigma de desenvolvimento de software intitulado Programação Orientada à Objetos é uma ferramenta poderosa que auxilia na construção de sistemas complexos. A abstração é um recurso indispensável para programadores que almejam dominar esta poderosa técnica de programação. Entretanto, é necessário dominar a habilidade de converter abstrações em código, ou seja, trazer classes e objetos do mundo das ideias para o mundo real.

Para mais informações acesse [Aula 01: Paradigma POO](#).

1. Projeto 01: Sistema de Conta Bancária

Contexto: Construa um sistema simples de gerenciamento de contas bancárias.

Descrição:

Account
<ul style="list-style-type: none"> - Number : int - Value : double - Name : string
<ul style="list-style-type: none"> + Deposit(double value, int number) : : bool + CashOut(double value) : : bool + CheckBalance() : : double

Instruções para Execução:

- Dentro do método main, crie uma instância da classe ContaBancaria, passando os parâmetros necessários para inicializá-la.
- Utilize os métodos dessa instância para realizar algumas operações, como depósito e saque.
- Após cada operação, utilize o método para verificar o saldo e exibi-lo na tela.

2. Projeto 02: Editor de arquivos de texto

Contexto: Construa um sistema simples de edição de arquivos de texto.

Descrição:

Arquivo
<ul style="list-style-type: none"> - Name : string - Size : int - Content : string
<ul style="list-style-type: none"> + Open() : : string + Edit(string newContent) : : void + Rename(string newName) : : bool + Clear() : : void

Instruções para execução:

- Dentro do método main, crie uma instância da classe Arquivo passando como parâmetro o nome do arquivo. Inicialmente o conteúdo do arquivo é vazio.

- O tamanho do arquivo é dado em bits e é proporcional ao número de caracteres do seu conteúdo, sendo que 1 byte = 1 caracter = 8 bits.
 - Abra o arquivo chamando o método abrir. Sempre que um arquivo é aberto o seu nome, conteúdo e tamanho devem ser exibidos.
 - Use o método editar para modificar o arquivo. A String passada como parâmetro para este método deve ser concatenada ao conteúdo.
 - Abra o arquivo novamente para visualizar as alterações.
 - Renomeie e limpe o arquivo
 - Abra o arquivo uma última vez para verificar as alterações.
-

3. Jogo de Luta

Contexto: Desenvolva um jogo simples onde os jogadores podem controlar personagens e lutar entre si.

Descrição:

Character
- Name : string - Life : int - Attack : int - Defense : int
+ Attack(Character opponent) : : void

Instruções:

- Implemente um construtor na classe Personagem que inicialize o nome e a vida do personagem (a vida inicia sempre com 100), e gere valores aleatórios para os atributos ataque (entre 20 e 30) e defesa (entre 10 e 15).
- Implemente um método ataque na classe Personagem, que recebe como parâmetro um objeto do tipo Personagem representando o adversário a ser atacado. O método deve calcular o dano causado ao adversário com base no poder de ataque do personagem atacante e na defesa do adversário. O dano será dado pela subtração do poder de ataque do atacante pelo poder de defesa do adversário.
- Dentro do método main, permita que o jogador instancie quantos personagens desejar, passando seus nomes como parâmetros para o construtor da classe Personagem.

- Permita que os personagens lutem entre si, chamando o método ataque de um personagem e passando como parâmetro o objeto do adversário a ser atacado. Repita o processo até que a vida de um dos personagens chegue a zero.