ALUNO: CAUÊ CARLOS DE AZEVEDO GUEDES, TURNO: MANHÃ, TDS3

Atividade SOO

Perguntas reflexiva 1:

Como você modelaria um sistema de biblioteca usando objetos?

Identifique:

1. ▪ Entidades físicas (livro, usuário)
2. ▪ Conceitos (empréstimo, reserva)
3. ▪ Identidades únicas (ISBN, ID do usuário)

1. Livro: título, autor, ISBN (identificador único), status (disponível ou emprestado).
   Usuário: nome, ID, lista de livros emprestados.
   Biblioteca: catálogo de livros e lista de usuários.

2. Empréstimo: Quando um usuário pega um livro por um período determinado.
   Reserva: Quando um usuário solicita um livro que está emprestado para garantir que poderá pegá-lo assim que for devolvido.

3. ISBN: Cada livro tem um código ISBN único.
   ID do Usuário: Cada usuário tem um identificador único para gerenciar seus empréstimos e reservas.

Pergunta Reflexiva 2:

1. "Como você projetaria uma classe Paciente para um sistema hospitalar, equilibrando

abstração (dados relevantes) e encapsulamento (proteção de dados sensíveis)?"

1.1 Abstração (Dados Relevantes)

Dados essenciais: Nome, data de nascimento, ID do paciente.

Informações médicas: Histórico de consultas, diagnósticos, tratamentos.

1.2 Encapsulamento (Proteção de Dados Sensíveis)

Dados pessoais protegidos: O histórico médico não deve ser acessado diretamente.

Métodos controlados: Apenas médicos ou funcionários autorizados podem atualizar o histórico

1.3 Como isso funciona?

O sistema permite que um paciente veja seus dados básicos, mas protege informações médicas.

Apenas usuários com permissão podem modificar o histórico.

Pergunta Reflexiva 3:

"Como você projetaria os atributos de uma classe Voo para um sistema de companhia

aérea, considerando restrições como capacidade máxima de passageiros e horários

válidos?"

1. Atributos Essenciais:

Número do voo (identificador único).

Origem e destino (cidades de partida e chegada).

Horário de partida e chegada (deve ser válido, sem horários no passado).

Capacidade máxima (quantidade de passageiros permitida).

Lista de passageiros (registro das pessoas no voo).

2. Restrições Importantes:

Capacidade Máxima: O número de passageiros não pode ultrapassar o limite do avião.

Horários Válidos: O voo não pode ser agendado com horários errados (ex: chegada antes da partida).

3. Benefícios:

Evita overbooking (excesso de passageiros).

Mantém dados organizados e seguros.

Garante que os voos tenham horários realistas.

Pergunta Reflexiva 4:

"Como você projetaria os métodos de uma classe SensorIoT para medir

temperatura, garantindo que leituras inválidas (ex: -300°C) não alterem o

estado interno do objeto?"

1. Métodos Essenciais:

RegistrarLeitura(temperatura): Recebe um valor e armazena apenas se for válido.

ObterUltimaLeitura(): Retorna a última temperatura registrada.

Calibrar(): Ajusta o sensor caso necessário.

2. Validações Importantes:

Temperatura dentro do intervalo realista:

Aceitar apenas valores entre, por exemplo, -100°C e 100°C.

Se o valor for inválido, a leitura é rejeitada.

3. Benefícios:

Evita dados errados ou incoerentes no sistema.

Garante que as leituras sejam confiáveis.

Mantém o sensor seguro e eficiente.

Pergunta Reflexiva 5:

"Como você projetaria o fluxo de mensagens em um aplicativo de mensagens

instantâneas, onde Usuario envia MensagemTexto para Grupo, que notifica todos os

Membros?"

1. Objetos Principais:

    Usuário: Pessoa que envia e recebe mensagens.

    MensagemTexto: Conteúdo da mensagem enviada.

    Grupo: Conjunto de usuários que compartilham mensagens.

2. Fluxo de Comunicação:

    Usuário cria e envia uma MensagemTexto para um Grupo.

    O Grupo recebe a mensagem e a encaminha para todos os Membros.

    Cada Membro recebe uma notificação e pode visualizar a mensagem.

3. Benefícios:

    Entrega eficiente: Todos os membros recebem a mensagem rapidamente.

    Organização: Cada grupo mantém um histórico das mensagens trocadas.

    Interatividade: Usuários podem responder e continuar a conversa.

EXERCÍCIO 1. ContaBancaria.

```
using System;

public class ContaBancaria {
    private decimal _saldo;
    public string? numeroConta { get; set; }
    public ContaBancaria(decimal saldoInicial) {
        _saldo = saldoInicial;
    }
    public void SacarDinheiro(decimal valor) {
        if (valor <= 0) {
            Console.WriteLine("|-----------------------------------------------------|");
            Console.WriteLine("|           O valor do saque deve ser positivo.       |");
            Console.WriteLine("|=====================================================|");
            return;
        }

        if (valor <= _saldo) {
            _saldo -= valor;
            Console.WriteLine("|-----------------------------------------------------|");
            Console.WriteLine($"|         Saque de R${valor} realizado com sucesso. ");
            Console.WriteLine("|=====================================================|");
        } else {
            Console.WriteLine("|-----------------------------------------------------|");
            Console.WriteLine("|                  Saldo insuficiente.                |");
            Console.WriteLine("|=====================================================|");
        }
    }
    public void DepositarDinheiro(decimal valor) {
        if (valor <= 0) {
            Console.WriteLine("|-----------------------------------------------------|");
            Console.WriteLine("|        O valor do depósito deve ser positivo.       |");
            Console.WriteLine("|=====================================================|");
```

```
            return;
        }

        _saldo += valor;
        Console.WriteLine("|-------------------------------------------------|");
        Console.WriteLine($"|  Depósito de R${valor} realizado com sucesso. ");
        Console.WriteLine("|=================================================|");
    }
    public decimal VerificarSaldo() {
        return _saldo;
    }
    public void Selecao() {
        int escolha;
        decimal valor;

        do {
            Console.WriteLine("|=========================================|");
            Console.WriteLine($"|           Número da Conta: {numeroConta}     ");
            Console.WriteLine("|-----------------------------------------|");
            Console.WriteLine("|         Qual ação deseja realizar?      |");
            Console.WriteLine("| 1) Saque                                |");
            Console.WriteLine("| 2) Depósito                             |");
            Console.WriteLine("| 3) Verificar saldo                      |");
            Console.WriteLine("| 4) Sair                                 |");
            Console.WriteLine("|-----------------------------------------|");
            Console.Write("| Sua escolha: ");
            escolha = Convert.ToInt32(Console.ReadLine());
            Console.Clear();

            switch (escolha) {
                case 1:
                    Console.WriteLine("|=================================================|");
                    Console.WriteLine("|               Escolha realizada: SAQUE          |");
                    Console.WriteLine("|-------------------------------------------------|");
                    Console.Write("| Insira o valor desejado: ");
                    valor = Convert.ToDecimal(Console.ReadLine());
                    SacarDinheiro(valor);
                    System.Threading.Thread.Sleep(2700);
                    Console.Clear();
                    break;

                case 2:
                    Console.WriteLine("|=================================================|");
                    Console.WriteLine("|            Escolha realizada: DEPÓSITO          |");
                    Console.WriteLine("|-------------------------------------------------|");
                    Console.Write("| Insira o valor desejado: ");
                    valor = Convert.ToDecimal(Console.ReadLine());
                    DepositarDinheiro(valor);
                    System.Threading.Thread.Sleep(2700);
                    Console.Clear();
                    break;

                case 3:
                    Console.WriteLine("|=================================================|");
```

```
                Console.WriteLine("|      Escolha realizada: VERIFICAR SALDO           |");
                Console.WriteLine("|---------------------------------------------------|");
                Console.Write("| Saldo disponível: ");
                Console.WriteLine(VerificarSaldo());
                System.Threading.Thread.Sleep(2700);
                Console.Clear();
                break;

            case 4:
                Console.WriteLine("|===================================================|");
                Console.WriteLine("|             Saindo... Até logo!                   |");
                Console.WriteLine("|===================================================|");
                break;

            default:
                Console.WriteLine("|---------------------------------------------------|");
                Console.WriteLine("|              Opção inválida. Tente novamente.     |");
                Console.WriteLine("|===================================================|");
                break;
        }
    } while (escolha != 4);
}
public void saudacaoInicial() {
    Console.WriteLine("|===================================================|");
    Console.WriteLine("|         Seja bem-vindo(a) ao sistema bancário     |");
    Console.WriteLine("|---------------------------------------------------|");
    Console.Write("| Insira o numero da sua conta: ");
    numeroConta = Console.ReadLine();
    Console.WriteLine("|---------------------------------------------------|");
    Console.WriteLine("|              Acesso feito com sucesso!            |");
    Console.WriteLine("|===================================================|");
    System.Threading.Thread.Sleep(2700);
    Console.Clear();
}
static void Main() {
    Console.Clear();
    ContaBancaria conta = new ContaBancaria(1000);
    conta.saudacaoInicial();
    conta.Selecao();

    Console.WriteLine("\nSaldo final: " + conta.VerificarSaldo());
}
}
```

EXERCÍCIO 2. Carro

```
using System;

public class Carro {
    private int velocidadeAtual;
    private int marchaAtual;

    public Carro() {
        velocidadeAtual = 0;
        marchaAtual = 1;
    }

    public void acelerar(int incremento) {
        if (incremento > 0) {
            velocidadeAtual += incremento;
            Console.WriteLine("|=====================================================|");
            Console.WriteLine($"|            O carro acelerou para {velocidadeAtual} km/h.          |");
            Console.WriteLine("|=====================================================|");
        } else {
            Console.WriteLine("|=====================================================|");
            Console.WriteLine("|       O valor de incremento deve ser positivo.      |");
            Console.WriteLine("|=====================================================|");
        }
        System.Threading.Thread.Sleep(900);
        Console.Clear();
    }

    public void frear(int decremento) {
        if (decremento > 0) {
            velocidadeAtual -= decremento;
            if (velocidadeAtual < 0) velocidadeAtual = 0;
            Console.WriteLine("|=====================================================|");
            Console.WriteLine($"|            O carro reduziu para {velocidadeAtual} km/h.          |");
            Console.WriteLine("|=====================================================|");
        } else {
            Console.WriteLine("|=====================================================|");
            Console.WriteLine("|      O valor de decremento deve ser positivo.      |");
            Console.WriteLine("|=====================================================|");
        }
        System.Threading.Thread.Sleep(900);
        Console.Clear();
    }

    public void trocarMarcha(int novaMarcha) {
        if (novaMarcha > 0) {
            marchaAtual = novaMarcha;
            Console.WriteLine("|=====================================================|");
            Console.WriteLine($"|              Marcha alterada para {marchaAtual}.              |");
            Console.WriteLine("|=====================================================|");
        } else {
            Console.WriteLine("|=====================================================|");
            Console.WriteLine("|        A marcha deve ser um valor positivo.        |");
            Console.WriteLine("|=====================================================|");
```

```
        }
        System.Threading.Thread.Sleep(900);
        Console.Clear();
    }

    public void verificarVelocidadeAtual() {
        Console.WriteLine("|=================================================|");
        Console.WriteLine($"|            Velocidade atual: {velocidadeAtual} km/h             |");
        Console.WriteLine("|=================================================|");
        System.Threading.Thread.Sleep(900);
        Console.Clear();
    }

    public void menu() {
        int opcao;

        do {
            Console.WriteLine("|=================================================|");
            Console.WriteLine("|          Qual ação deseja realizar?             |");
            Console.WriteLine("|-------------------------------------------------|");
            Console.WriteLine("| 1) Acelerar                                     |");
            Console.WriteLine("| 2) Frear                                        |");
            Console.WriteLine("| 3) Trocar marcha                                |");
            Console.WriteLine("| 4) Verificar velocidade                         |");
            Console.WriteLine("| 5) Sair                                         |");
            Console.WriteLine("|=================================================|");
            Console.Write("| Escolha uma opção: ");
            opcao = int.Parse(Console.ReadLine());

            switch (opcao) {
                case 1:
                    Console.Write("| Digite o valor para acelerar: ");
                    acelerar(int.Parse(Console.ReadLine()));
                    break;
                case 2:
                    Console.Write("| Digite o valor para frear: ");
                    frear(int.Parse(Console.ReadLine()));
                    break;
                case 3:
                    Console.Write("| Digite a nova marcha: ");
                    trocarMarcha(int.Parse(Console.ReadLine()));
                    break;
                case 4:
                    verificarVelocidadeAtual();
                    break;
                case 5:
                    Console.WriteLine("|=================================================|");
                    Console.WriteLine("|                    Saindo...                    |");
                    Console.WriteLine("|=================================================|");
                    break;
                default:
                    Console.WriteLine("|=================================================|");
                    Console.WriteLine("|          Opção inválida. Tente novamente.       |");
                    Console.WriteLine("|=================================================|");
```

```
                    break;
            }

            System.Threading.Thread.Sleep(900);
            Console.Clear();

        } while (opcao != 5);
    }

    static void Main() {
        Console.Clear();
        Carro carro = new Carro();
        carro.menu();
    }
}
```