



SOFTEX
PERNAMBUCO

 **Softex**

MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÃO

GOVERNO FEDERAL
BRASIL
UNIÃO E RECONSTRUÇÃO



Olá pessoal, seguem nos próximos slides dois exercícios práticos para a nossa 1a aula de reposição para o dia 25/09/25.





Enunciado do Problema – Controle de Pacientes em Clínica

Uma clínica médica precisa organizar melhor o **cadastro de seus pacientes**. Atualmente, os registros são feitos em anotações manuais, dificultando a consulta rápida das informações. Para resolver esse problema, a direção da clínica solicitou a criação de um **sistema simples em JavaScript**, utilizando **Programação Orientada a Objetos (POO)**, para cadastrar e listar pacientes.

Requisitos do sistema:

1. Criar uma classe chamada `Paciente` que deve possuir os seguintes **atributos**:

- `nome` (texto)
- `idade` (número)
- `telefone` (texto)

Essa classe deve conter um **método** chamado `exibirInfo()`, que mostre as informações do paciente no console.

2. Criar uma classe chamada `Clínica` que deve possuir os seguintes **atributos**:

- `nome` (texto)
- `pacientes` (lista para armazenar objetos do tipo `Paciente`).

Essa classe deve conter os seguintes **métodos**:

- `adicionarPaciente(paciente)` → adiciona um paciente à lista.
- `listarPacientes()` → exibe no console todos os pacientes cadastrados na clínica.

3. No programa principal:

- Criar uma clínica chamada "**Clínica Bem Estar**".
- Cadastrar pelo menos **três pacientes diferentes** com nome, idade e telefone.
- Exibir no console a confirmação do cadastro de cada paciente.
- Listar todos os pacientes cadastrados na clínica.





Enunciado do Problema – Sistema de Pontuação em Jogo

Um estúdio de jogos deseja desenvolver um sistema simples de **pontuação para jogadores**, a fim de controlar quem está na frente em um jogo competitivo. Para isso, eles contrataram você para implementar o sistema em **JavaScript utilizando Programação Orientada a Objetos (POO)**.

Requisitos do sistema:

1. Criar uma classe chamada `Jogador`, que deve possuir:
 - **Atributos:**
 - `nome` (texto)
 - `pontuacao` (número, inicia em 0)
 - **Métodos:**
 - `adicionarPontos(pontos)` : adiciona pontos à pontuação do jogador.
 - `removerPontos(pontos)` : remove pontos da pontuação (sem permitir valores negativos).
 - `exibirStatus()` : mostra no console o nome e a pontuação atual do jogador.
2. Criar uma classe chamada `Jogo`, que deve possuir:
 - **Atributos:**
 - `nome` (texto)
 - `jogadores` (lista para armazenar jogadores).
 - **Métodos:**
 - `adicionarJogador(jogador)` : adiciona um jogador ao jogo.
 - `mostrarPontuacoes()` : exibe no console a pontuação de todos os jogadores cadastrados.
3. No programa principal:
 - Criar um jogo chamado **"Aventura Épica"**.
 - Criar pelo menos **três jogadores diferentes**.
 - Adicionar os jogadores ao jogo.
 - Simular rodadas de pontuação, onde jogadores ganham e perdem pontos.
 - Exibir o **placar final de todos os jogadores**.





SOFTEX
PERNAMBUCO

 **Softex**

GOVERNO FEDERAL
MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÃO

UNIÃO E RECONSTRUÇÃO