

ENUNCIADO DO EXERCÍCIO (para entregar aos alunos)

Projeto: Termómetro Digital Interativo (HTML + CSS + JavaScript)

Turma: 12MINF1/2

Instituição: Instituto Politécnico Privado Baptista da Paz

Duração estimada: 90 minutos

Descrição Geral

O termómetro inicia apagado e pode ser alterado através de interações do utilizador. Dependendo do estado seleccionado, a cor da barra do termómetro e a mensagem do ecrã mudam. O fundo da página também sofre uma transição suave para reforçar a percepção de temperatura.

REGRAS DO EXERCÍCIO — pedidos com propriedade e explicação

Estados do Termómetro

- **Criar variável de estado** — `let estado = "apagado"` (declara uma variável mutável em JavaScript para guardar o estado: "frio", "normal", "quente").

Comportamentos Obrigatórios

Botões

- **Botão "Frio"** — `<button id="btnFrio">` (tag HTML que cria um botão clicável).
- **Propriedade para cor azul** — `background-color: #4da6ff` (CSS — define a cor da barra para representar frio).
- **Evento para o botão** — `addEventListener("click", ...)` (JS — detecta cliques e executa a função que muda o estado).
- **Botão "Normal"** — `<button id="btnNormal">` (tag HTML).
- **Propriedade para cor verde** — `background-color: #5cd65c` (CSS — define a cor da barra para representar temperatura normal).
- **Evento para o botão** — `addEventListener("click", ...)` (JS).
- **Botão "Quente"** — `<button id="btnQuente">` (tag HTML).
- **Propriedade para cor vermelha** — `background-color: #ff5050` (CSS — define a cor da barra para representar calor).
- **Evento para o botão** — `addEventListener("click", ...)` (JS).

Efeito Hover

- **Simular “normal” no hover** — `addEventListener("mouseover", ...)` (JS — detecta passagem do mouse e aplica temporariamente o estado “normal”).
- **Reverter no mouseout** — `addEventListener("mouseout", ...)` (JS — restaura o estado anterior quando o mouse sai).

Duplo Clique

- **Apagar com duplo clique** — `addEventListener("dblclick", ...)` (JS — detecta duplo clique e define estado = "apagado"; faz reset visual).
- **Propriedade visual para apagar** — `height: 0px` (CSS via JS — reduz a altura da barra para mostrar “apagado”).

Elementos obrigatórios no HTML

- **Título do projecto** — `<h1>` (HTML — elemento de título principal).
- **Corpo do termómetro** — `<div id="termometro">` (HTML — contêiner externo para o termómetro).
- **Barra interna** — `<div id="barra">` (HTML — elemento cuja height e background-color mudam).
- **Área de texto do estado** — `<p id="estadoTexto">` ou `` (HTML — texto onde se escreve "Frio" / "Normal" / "Quente").
- **Botões** — `<button id="btnFrio">`, `<button id="btnNormal">`, `<button id="btnQuente">` (HTML).

Estilo obrigatório no CSS

- **Centralizar o termómetro** — `display: flex; justify-content: center; align-items: center;` (CSS — centralização horizontal e vertical).
- **Bordas arredondadas** — `border-radius: 12px;` ou `50%` para círculos (CSS — suaviza cantos).
- **Transições suaves (altura da barra)** — `transition: height 0.6s ease;` (CSS — anima a alteração da altura).
- **Transições suaves (cor da barra)** — `transition: background-color 0.6s ease;` (CSS — anima a mudança de cor).
- **Transição suave no fundo do body** — `transition: background-color 0.6s ease;` (CSS — anima a cor do fundo).

- **Área com aspecto digital** — font-family: "Digital-7", monospace; font-size: 1.2rem; (CSS — aplica estilo tipográfico tipo display digital).

JavaScript – Funcionalidades obrigatórias

- **Função única** — `function alterarEstado(estado) { ... }` (JS — recebe "frio"|"normal"|"quente"|"apagado" e atualiza interface).
- **Atualizar cor da barra** — `element.style.backgroundColor = "#..."` (JS — altera a cor dinamicamente).
- **Atualizar altura da barra** — `element.style.height = "XXpx" ou %` (JS — altera o nível do termómetro).
- **Atualizar texto exibido** — `document.getElementById("estadoTexto").textContent = "Normal"` (JS — muda o texto informativo).
- **Atualizar cor do fundo do body** — `document.body.style.backgroundColor = "#..."` (JS — altera ambiente visual da página).
- **Adicionar eventos (botões)** — `btn.addEventListener("click", () => alterarEstado("frio"))` (JS).
- **Eventos no termómetro** — `termometro.addEventListener("mouseover", ...), termometro.addEventListener("dblclick", ...)` (JS).
- **Selecionar elementos do DOM** — `document.getElementById()` / `document.querySelector()` (JS — obter referência aos elementos para manipular).

Requisitos pedagógicos (o que o aluno deve demonstrar)

- **Manipulação do DOM** — `document.getElementById()` / `querySelector()` (JS — seleccionar e alterar elementos).
- **Alterações de estilo via JavaScript** — `element.style.prop = value` (JS — mudar visual sem editar CSS).
- **Uso de classes (opcional)** — `element.classList.add("classe")` / `classList.remove()` (JS — activar estilos CSS predefinidos).
- **Uso de variáveis de estado** — `let estado = "..."` (JS — guardar e consultar estado atual).
- **Uso de funções** — `function nome(...) {}` (JS — organizar lógica).
- **Conhecimento de eventos** — "click", "mouseover", "dblclick" (JS — usar eventos apropriados).
- **Organização do código** — comentários, indentação, separação HTML/CSS/JS.

Observações finais

- O exercício deve ser implementado em **HTML + CSS + JS** separados.
- Use **transições** para suavizar mudanças visuais.
- A função `alterarEstado(estado)` deve ser a única responsável por colocar a interface no estado correcto (cores, altura, texto, fundo).