

Lógica de Programação - Roteiro 8 - 2º Semestre
Prof. Sara.

Link de acesso: <https://codepen.io/pen>

HTML

```
<div class="aplicativo">
  <header class="cabecalho">
    <div>
      <p class="subtitulo">Lógica de Programação e Inteligência Artificial</p>
      <h1>EduBot IA</h1>
      <p class="descricao">
        Um assistente inteligente para estudar programação.
      </p>
    </div>

    <div class="status">
      <span class="status-bolinha"></span>
      Sistema ativo
    </div>
  </header>

  <main>
    <section class="painel-informacoes">
      <div class="cartao">
        <span class="icone"><img alt="speech bubble icon" data-bbox="358 665 385 682"/></span>
        <div>
          <strong id="totalPerguntas">0</strong>
          <p>Perguntas realizadas</p>
        </div>
      </div>

      <div class="cartao">
        <span class="icone"><img alt="brain icon" data-bbox="358 805 385 822"/></span>
        <div>
          <strong id="temaAtual">Nenhum</strong>
          <p>Último tema identificado</p>
        </div>
      </div>
    </section>
  </main>
</div>
```

```
<div class="cartao">
  <span class="icone">★</span>
  <div>
    <strong id="pontuacao">0</strong>
    <p>Pontos de aprendizagem</p>
  </div>
</div>
</section>
```

```
<section class="chat">
  <div class="chat-topo">
    <div>
      <h2>Converse com o EduBot</h2>
      <p>
        Pergunte sobre variáveis, condições, funções, HTML, CSS,
        JavaScript ou Inteligência Artificial.
      </p>
    </div>

    <button id="botaoLimpar" class="botao-secundario">
      Limpar conversa
    </button>
  </div>
```

```
<div id="mensagens" class="mensagens" aria-live="polite">
  <div class="mensagem bot">
    <div class="avatar">🤖</div>

    <div class="balao">
      <strong>EduBot</strong>
      <p>
        Olá! Eu sou o EduBot. Digite uma pergunta para começarmos.
      </p>
    </div>
  </div>
</div>
```

```
<div class="sugestoes">
  <button class="sugestao" data-pergunta="O que é uma variável?">
    O que é uma variável?
  </button>

  <button class="sugestao" data-pergunta="Como funciona uma condição if?">
    Como funciona o if?
  </button>

  <button class="sugestao" data-pergunta="O que é inteligência artificial?">
    O que é IA?
```

</button>

<button class="sugestao" data-pergunta="Qual a diferença entre HTML, CSS e JavaScript?">

HTML, CSS e JavaScript

</button>

</div>

<form id="formularioChat" class="formulario">

<label for="campoPergunta" class="sr-only">

Digite sua pergunta

</label>

<input

id="campoPergunta"

type="text"

placeholder="Digite uma pergunta..."

autocomplete="off"

maxlength="200"

/>

<button type="submit" class="botao-enviar">

Enviar

</button>

</form>

</section>

<section class="desafio">

<div>

<p class="subtitulo">Desafio de programação</p>

<h2>Amplie a inteligência do EduBot</h2>

<p>

Acrescente novas palavras-chave e respostas no JavaScript.

Depois, personalize o visual utilizando CSS.

</p>

</div>

<button id="botaoDica" class="botao-dica">

Mostrar dica

</button>

</section>

<div id="modalDica" class="modal escondido">

<div class="modal-conteudo">

<button id="fecharModal" class="fechar-modal">×</button>

<h2>💡 Dica para o desafio</h2>

```
<p>
  No JavaScript, procure a constante
  <code>baseDeConhecimento</code>.
</p>
```

```
<p>Adicione um novo objeto seguindo este modelo:</p>
```

```
<pre>
{
  tema: "Scratch",
  palavras: ["scratch", "blocos", "sprite"],
  resposta: "Scratch é uma linguagem visual..."
}
</pre>
</div>
</div>
</main>
```

```
<footer>
<p>
  Projeto educacional desenvolvido para aulas de Lógica de Programação e IA.
</p>
</footer>
</div>
```

CSS

```
* {
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}

:root {
  --cor-primaria: #5b4bdb;
  --cor-primaria-escura: #4134b3;
  --cor-secundaria: #15a085;
  --cor-fundo: #f3f5fb;
  --cor-cartao: #ffffff;
  --cor-texto: #25283d;
  --cor-texto-claro: #6b6f86;
  --cor-borda: #dfe3ef;
  --cor-bot: #ecefdf;
  --cor-usuario: #5b4bdb;
  --sombra: 0 12px 30px rgba(42, 45, 80, 0.12);
}

body {
```

```
min-height: 100vh;
background: var(--cor-fundo);
color: var(--cor-texto);
font-family: Arial, Helvetica, sans-serif;
}
```

```
button,
input {
  font: inherit;
}
```

```
button {
  cursor: pointer;
}
```

```
.aplicativo {
  width: min(1100px, 94%);
  margin: 30px auto;
}
```

```
.cabecalho {
  display: flex;
  align-items: center;
  justify-content: space-between;
  gap: 30px;
  padding: 30px;
  color: white;
  background: linear-gradient(
    135deg,
    var(--cor-primaria),
    var(--cor-primaria-escuro)
  );
  border-radius: 22px;
  box-shadow: var(--sombra);
}
```

```
.cabecalho h1 {
  margin: 6px 0;
  font-size: clamp(2rem, 5vw, 3.4rem);
}
```

```
.subtitulo {
  font-size: 0.85rem;
  font-weight: bold;
  letter-spacing: 0.08rem;
  text-transform: uppercase;
}
```

```
.descricao {  
  line-height: 1.5;  
  opacity: 0.9;  
}
```

```
.status {  
  display: flex;  
  align-items: center;  
  gap: 9px;  
  min-width: max-content;  
  padding: 10px 15px;  
  background: rgba(255, 255, 255, 0.14);  
  border: 1px solid rgba(255, 255, 255, 0.25);  
  border-radius: 999px;  
}
```

```
.status-bolinha {  
  width: 11px;  
  height: 11px;  
  background: #6dffb0;  
  border-radius: 50%;  
  box-shadow: 0 0 10px #6dffb0;  
}
```

```
.painel-informacoes {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  gap: 18px;  
  margin: 22px 0;  
}
```

```
.cartao {  
  display: flex;  
  align-items: center;  
  gap: 16px;  
  min-height: 105px;  
  padding: 20px;  
  background: var(--cor-cartao);  
  border: 1px solid var(--cor-borda);  
  border-radius: 18px;  
  box-shadow: 0 6px 18px rgba(42, 45, 80, 0.06);  
}
```

```
.icone {  
  display: grid;  
  width: 52px;  
  height: 52px;  
  place-items: center;
```

```
background: var(--cor-bot);
border-radius: 15px;
font-size: 1.6rem;
}
```

```
.cartao strong {
display: block;
margin-bottom: 5px;
color: var(--cor-primaria);
font-size: 1.35rem;
}
```

```
.cartao p {
color: var(--cor-texto-claro);
font-size: 0.9rem;
}
```

```
.chat {
overflow: hidden;
background: var(--cor-cartao);
border: 1px solid var(--cor-borda);
border-radius: 22px;
box-shadow: var(--sombra);
}
```

```
.chat-topo {
display: flex;
align-items: center;
justify-content: space-between;
gap: 20px;
padding: 24px;
border-bottom: 1px solid var(--cor-borda);
}
```

```
.chat-topo h2 {
margin-bottom: 6px;
}
```

```
.chat-topo p {
color: var(--cor-texto-claro);
line-height: 1.5;
}
```

```
.botao-secundario {
min-width: max-content;
padding: 10px 14px;
color: var(--cor-primaria);
background: transparent;
```

```
border: 1px solid var(--cor-primaria);
border-radius: 10px;
transition: 0.2s;
}
```

```
.botao-secundario:hover {
  color: white;
  background: var(--cor-primaria);
}
```

```
.mensagens {
  height: 390px;
  padding: 24px;
  overflow-y: auto;
  background: linear-gradient(180deg, #fafbff, #f5f6fc);
}
```

```
.mensagem {
  display: flex;
  align-items: flex-start;
  gap: 12px;
  max-width: 78%;
  margin-bottom: 18px;
  animation: aparecer 0.25s ease;
}
```

```
.mensagem.usuario {
  flex-direction: row-reverse;
  margin-left: auto;
}
```

```
.avatar {
  display: grid;
  flex-shrink: 0;
  width: 42px;
  height: 42px;
  place-items: center;
  background: white;
  border: 1px solid var(--cor-borda);
  border-radius: 50%;
}
```

```
.balao {
  padding: 14px 16px;
  background: var(--cor-bot);
  border-radius: 4px 16px 16px 16px;
  line-height: 1.55;
}
```

```
.balao strong {
  display: block;
  margin-bottom: 5px;
  color: var(--cor-primaria);
  font-size: 0.85rem;
}

.mensagem.usuario .balao {
  color: white;
  background: var(--cor-usuario);
  border-radius: 16px 4px 16px 16px;
}

.mensagem.usuario .balao strong {
  color: white;
}

.sugestoes {
  display: flex;
  gap: 8px;
  padding: 15px 24px 0;
  overflow-x: auto;
}

.sugestao {
  flex-shrink: 0;
  padding: 9px 12px;
  color: var(--cor-primaria);
  background: white;
  border: 1px solid var(--cor-borda);
  border-radius: 999px;
}

.sugestao:hover {
  background: var(--cor-bot);
  border-color: var(--cor-primaria);
}

.formulario {
  display: flex;
  gap: 12px;
  padding: 18px 24px 24px;
}

.formulario input {
  flex: 1;
  min-width: 0;
```

```
padding: 14px 16px;
border: 1px solid var(--cor-borda);
border-radius: 12px;
outline: none;
}

.formulario input:focus {
border-color: var(--cor-primaria);
box-shadow: 0 0 0 3px rgba(91, 75, 219, 0.12);
}

.botao-enviar,
.botao-dica {
padding: 13px 20px;
color: white;
background: var(--cor-primaria);
border: none;
border-radius: 12px;
font-weight: bold;
transition: transform 0.2s, background 0.2s;
}

.botao-enviar:hover,
.botao-dica:hover {
background: var(--cor-primaria-escura);
transform: translateY(-2px);
}

.desafio {
display: flex;
align-items: center;
justify-content: space-between;
gap: 25px;
margin-top: 22px;
padding: 25px;
background: #eaf8f4;
border: 1px solid #b8e8dc;
border-radius: 20px;
}

.desafio h2 {
margin: 5px 0 8px;
}

.desafio p {
max-width: 720px;
line-height: 1.5;
}
```

```
.botao-dica {  
  min-width: max-content;  
  background: var(--cor-secundaria);  
}
```

```
.modal {  
  position: fixed;  
  inset: 0;  
  z-index: 10;  
  display: grid;  
  padding: 20px;  
  place-items: center;  
  background: rgba(25, 27, 45, 0.68);  
}
```

```
.modal-conteudo {  
  position: relative;  
  width: min(560px, 100%);  
  padding: 28px;  
  background: white;  
  border-radius: 18px;  
  box-shadow: var(--sombra);  
}
```

```
.modal-conteudo h2 {  
  margin-bottom: 16px;  
}
```

```
.modal-conteudo p {  
  margin: 10px 0;  
  line-height: 1.5;  
}
```

```
.modal-conteudo pre {  
  margin-top: 15px;  
  padding: 15px;  
  overflow-x: auto;  
  background: #242638;  
  color: #ffffff;  
  border-radius: 12px;  
  font-size: 0.85rem;  
}
```

```
.fechar-modal {  
  position: absolute;  
  top: 12px;  
  right: 15px;
```

```
color: var(--cor-texto-claro);
background: transparent;
border: none;
font-size: 1.8rem;
}
```

```
.escondido {
  display: none;
}
```

```
footer {
  padding: 25px;
  color: var(--cor-texto-claro);
  text-align: center;
  font-size: 0.9rem;
}
```

```
.sr-only {
  position: absolute;
  width: 1px;
  height: 1px;
  overflow: hidden;
  clip: rect(0, 0, 0, 0);
}
```

```
@keyframes aparecer {
  from {
    opacity: 0;
    transform: translateY(8px);
  }
```

```
  to {
    opacity: 1;
    transform: translateY(0);
  }
}
```

```
@media (max-width: 760px) {
  .cabecalho,
  .chat-topo,
  .desafio {
    align-items: flex-start;
    flex-direction: column;
  }
```

```
.painel-informacoes {
  grid-template-columns: 1fr;
}
```

```

.mensagem {
  max-width: 92%;
}

.formulario {
  flex-direction: column;
}

.botao-enviar,
.botao-dica,
.botao-secundario {
  width: 100%;
}
}

```

JS

// Seleção dos elementos da página

```

const formularioChat = document.querySelector("#formularioChat");
const campoPergunta = document.querySelector("#campoPergunta");
const mensagens = document.querySelector("#mensagens");
const totalPerguntas = document.querySelector("#totalPerguntas");
const temaAtual = document.querySelector("#temaAtual");
const pontuacao = document.querySelector("#pontuacao");
const botaoLimpar = document.querySelector("#botaoLimpar");
const botoesSugestao = document.querySelectorAll(".sugestao");
const botaoDica = document.querySelector("#botaoDica");
const modalDica = document.querySelector("#modalDica");
const fecharModal = document.querySelector("#fecharModal");

```

// Recupera os dados salvos no navegador

```

let quantidadeDePerguntas =
  Number(localStorage.getItem("quantidadeDePerguntas")) || 0;

```

```

let pontosDeAprendizagem =
  Number(localStorage.getItem("pontosDeAprendizagem")) || 0;

```

// Atualiza os cartões quando o programa é iniciado

```

totalPerguntas.textContent = quantidadeDePerguntas;
pontuacao.textContent = pontosDeAprendizagem;

```

/*

Base de conhecimento do chatbot.

Cada objeto possui:

- tema: assunto identificado;
- palavras: termos que o programa deve procurar;

- resposta: texto apresentado ao usuário.

*/

```
const baseDeConhecimento = [
```

```
{
```

```
  tema: "Variáveis",
```

```
  palavras: ["variável", "variaveis", "variáveis", "let", "const", "var"],
```

```
  resposta:
```

"Uma variável é um espaço usado para armazenar informações. Em JavaScript, podemos criar variáveis com let ou const. Exemplo: let idade = 16;"

```
},
```

```
{
```

```
  tema: "Condições",
```

```
  palavras: ["if", "else", "condição", "condicao", "decisão", "decisao"],
```

```
  resposta:
```

"As estruturas condicionais permitem que o programa tome decisões. O if executa um código quando uma condição é verdadeira, enquanto o else pode ser executado quando ela é falsa."

```
},
```

```
{
```

```
  tema: "Funções",
```

```
  palavras: ["função", "funcao", "funções", "function", "retorno"],
```

```
  resposta:
```

"Uma função é um bloco de código criado para realizar uma tarefa. Ela pode receber dados, processá-los e devolver um resultado."

```
},
```

```
{
```

```
  tema: "Repetição",
```

```
  palavras: ["for", "while", "repetição", "repeticao", "laço", "loop"],
```

```
  resposta:
```

"Os laços de repetição executam um conjunto de instruções várias vezes. O for é útil quando sabemos quantas repetições serão realizadas."

```
},
```

```
{
```

```
  tema: "HTML",
```

```
  palavras: ["html", "estrutura", "página", "pagina", "site"],
```

```
  resposta:
```

"HTML é a linguagem responsável pela estrutura da página. Ele define títulos, parágrafos, imagens, botões, formulários e outros elementos."

```
},
```

```
{
```

```
  tema: "CSS",
```

```
  palavras: ["css", "estilo", "cor", "cores", "design", "aparência"],
```

```
  resposta:
```

"CSS é utilizado para controlar a aparência da página. Com ele, podemos alterar cores, fontes, tamanhos, espaçamentos e posições."

```
},
```

```
{
```

```
  tema: "JavaScript",
```

palavras: ["javascript", "js", "interatividade", "evento", "dom"],

resposta:

"JavaScript é a linguagem que adiciona comportamento e interatividade às páginas. Ele permite responder a cliques, alterar elementos e processar dados."

},

{

tema: "Inteligência Artificial",

palavras: [

"inteligência artificial",

"inteligencia artificial",

"ia",

"machine learning",

"aprendizado de máquina",

"aprendizado de maquina"

],

resposta:

"Inteligência Artificial é uma área da computação que desenvolve sistemas capazes de realizar tarefas associadas à inteligência humana, como reconhecer padrões, classificar informações e gerar respostas."

},

{

tema: "Algoritmos",

palavras: ["algoritmo", "passos", "sequência", "sequencia", "instruções"],

resposta:

"Um algoritmo é uma sequência organizada de passos para resolver um problema. Uma receita culinária e as instruções de um jogo são exemplos de algoritmos."

},

{

tema: "Arrays",

palavras: ["array", "lista", "vetor", "elementos"],

resposta:

"Um array é uma estrutura que armazena vários valores em uma única variável.

Exemplo: const notas = [8, 7, 9, 10];"

},

{

tema: "Objetos",

palavras: ["objeto", "objetos", "propriedade", "propriedades"],

resposta:

"Um objeto reúne informações relacionadas por meio de propriedades. Exemplo: const aluno = { nome: 'Ana', idade: 16 };"

},

{

tema: "CodePen",

palavras: ["codepen", "pen", "executar código", "executar codigo"],

resposta:

"O CodePen permite testar projetos de front-end. O código HTML deve ser colocado no painel HTML, o estilo no CSS e a programação no JavaScript."

}

```
];
```

```
// Remove acentos e transforma o texto em letras minúsculas
```

```
function normalizarTexto(texto) {  
  return texto  
    .toLowerCase()  
    .normalize("NFD")  
    .replace(/[\u0300-\u036f]/g, "");  
}
```

```
// Procura palavras-chave na pergunta
```

```
function identificarResposta(pergunta) {  
  const perguntaNormalizada = normalizarTexto(pergunta);  
  
  for (const conhecimento of baseDeConhecimento) {  
    const encontrouPalavra = conhecimento.palavras.some((palavra) => {  
      const palavraNormalizada = normalizarTexto(palavra);  
      return perguntaNormalizada.includes(palavraNormalizada);  
    });  
  
    if (encontrouPalavra) {  
      return conhecimento;  
    }  
  }  
  
  return {  
    tema: "Tema não identificado",  
    resposta:  
      "Ainda não tenho uma resposta específica para essa pergunta. Tente utilizar palavras  
      como variável, algoritmo, função, HTML, CSS, JavaScript ou inteligência artificial."  
  };  
}
```

```
// Cria uma nova mensagem no chat
```

```
function adicionarMensagem(texto, tipo) {  
  const mensagem = document.createElement("div");  
  mensagem.classList.add("mensagem", tipo);  
  
  const avatar = document.createElement("div");  
  avatar.classList.add("avatar");  
  avatar.textContent = tipo === "usuario" ? "👤" : "🤖";  
  
  const balao = document.createElement("div");  
  balao.classList.add("balao");  
  
  const nome = document.createElement("strong");  
  nome.textContent = tipo === "usuario" ? "Você" : "EduBot";
```

```

const conteudo = document.createElement("p");
conteudo.textContent = texto;

balao.appendChild(nome);
balao.appendChild(conteudo);

mensagem.appendChild(avatar);
mensagem.appendChild(balao);

mensagens.appendChild(mensagem);

// Move a barra de rolagem para a última mensagem
mensagens.scrollTop = mensagens.scrollHeight;
}

// Atualiza e salva as informações da aprendizagem
function atualizarEstatisticas(tema) {
    quantidadeDePerguntas++;
    pontosDeAprendizagem += 10;

    totalPerguntas.textContent = quantidadeDePerguntas;
    pontuacao.textContent = pontosDeAprendizagem;
    temaAtual.textContent = tema;

    localStorage.setItem(
        "quantidadeDePerguntas",
        quantidadeDePerguntas
    );

    localStorage.setItem(
        "pontosDeAprendizagem",
        pontosDeAprendizagem
    );
}

// Processa a pergunta enviada pelo estudante
function enviarPergunta(pergunta) {
    const perguntaLimpa = pergunta.trim();

    if (perguntaLimpa === "") {
        campoPergunta.focus();
        return;
    }

    adicionarMensagem(perguntaLimpa, "usuario");

    const resultado = identificarResposta(perguntaLimpa);

```

```

atualizarEstatisticas(resultado.tema);

// Simula um pequeno tempo de processamento da IA
setTimeout(() => {
  adicionarMensagem(resultado.resposta, "bot");
}, 500);

campoPergunta.value = "";
campoPergunta.focus();
}

// Evento de envio do formulário
formularioChat.addEventListener("submit", function (evento) {
  evento.preventDefault();
  enviarPergunta(campoPergunta.value);
});

// Eventos dos botões com perguntas prontas
botoesSugestao.forEach(function (botao) {
  botao.addEventListener("click", function () {
    const pergunta = botao.dataset.pergunta;
    enviarPergunta(pergunta);
  });
});

// Limpa a conversa e reinicia as estatísticas
botaoLimpar.addEventListener("click", function () {
  const desejaLimpar = confirm(
    "Deseja apagar a conversa e reiniciar a pontuação?"
  );

  if (!desejaLimpar) {
    return;
  }

  mensagens.innerHTML = `
    <div class="mensagem bot">
      <div class="avatar"><img alt="Avatar do bot" data-bbox="340 740 365 755"/></div>

      <div class="balao">
        <strong>EduBot</strong>
        <p>A conversa foi reiniciada. Faça uma nova pergunta!</p>
      </div>
    </div>
  `;

  quantidadeDePerguntas = 0;
  pontosDeAprendizagem = 0;

```

```
totalPerguntas.textContent = 0;
pontuacao.textContent = 0;
temaAtual.textContent = "Nenhum";

localStorage.removeItem("quantidadeDePerguntas");
localStorage.removeItem("pontosDeAprendizagem");
});

// Abre o modal com a dica
botaoDica.addEventListener("click", function () {
  modalDica.classList.remove("escondido");
});

// Fecha o modal
fecharModal.addEventListener("click", function () {
  modalDica.classList.add("escondido");
});

// Fecha o modal ao clicar fora da caixa
modalDica.addEventListener("click", function (evento) {
  if (evento.target === modalDica) {
    modalDica.classList.add("escondido");
  }
});

// Permite fechar o modal com a tecla Escape
document.addEventListener("keydown", function (evento) {
  if (evento.key === "Escape") {
    modalDica.classList.add("escondido");
  }
});
```