

vreau sa construiesc un site numit CleanPro pentru o firma de curatenie din Targu Mures. Scrie o descriere tehnica scurta a proiectului cu scop, pagini si tehnologii folosite (html5,css3,javascript), pe baza cerintelor mele.

Perfect 👍 lată o **descriere tehnică scurtă** pentru proiectul **CleanPro**, formulată clar și profesional, potrivită pentru documentație sau prezentare:

Descriere tehnică proiect – CleanPro

CleanPro este un site web de prezentare destinat unei firme de curătenie din **Târgu Mureș**, având ca scop promovarea serviciilor oferite și facilitarea contactului rapid cu potențialii clienți.

Scopul proiectului

Scopul site-ului este de a oferi informații clare și accesibile despre serviciile de curătenie, de a crește vizibilitatea firmei în mediul online și de a permite clienților să solicite oferte sau programări într-un mod simplu și eficient.

Structura site-ului (pagini)

- **Pagina principală (Home)** – prezentare generală a firmei, slogan, servicii principale și apel la acțiune
- **Servicii** – descriere detaliată a serviciilor de curătenie (rezidențială, birouri, industrială etc.)
- **Despre noi** – informații despre firmă, experiență și valori
- **Contact** – formular de contact, date de contact și locația firmei
- **Galerie / Portofoliu (optional)** – imagini cu lucrări realizate

Tehnologii utilizate

- **HTML5** – pentru structura semantică a paginilor web
- **CSS3** – pentru stilizare, layout responsive și design modern
- **JavaScript** – pentru interactivitate (formular de contact, animații, validări)

Site-ul va fi **responsive**, compatibil cu dispozitive mobile și optimizat pentru o experiență de utilizare intuitivă.

Dacă vrei, pot:

- să o fac **mai scurtă** (pentru proiect școlar),
- să o adaptez pentru **documentație tehnică / licență**,
- sau să continui cu **structura folderelor + fișierele HTML/CSS/JS** 



genereaza continutul fisierului home.html pentru site ul CleanPro care sa includa structura completa html5, <header> cu logo CleanPro si slogan, meniu cu linkuri catre home.html, servicii.html, preturi.html, contact.html, sectiune principala cu prezentare firma si lista de avantaje, plus <footer> cu urmatoarele date de contact
Adresa Bd-ul 1 Decembrie 1918, nr. 25, Targu Mures,
540525, contact@cleanpro.ro,
0712 345 678,
Program: Luni-Vineri 8:00-20:00, Sambata 9:00-15:00,
Duminica INCHIS,
© 2025 CleanPro. Toate drepturile rezervate.

Perfect  Mai jos ai **conținutul complet al fișierului** home.html , respectând structura HTML5 și toate cerințele tale. Poți copia direct fișierul și îl va valida corect în browser.

html

```
<!DOCTYPE html>
<html lang="ro">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>CleanPro - Firmă de Curățenie Târgu Mureș</title>
  <meta name="description" content="CleanPro - servicii profesionale de curățen"
```

```
<link rel="stylesheet" href="style.css">
</head>
<body>

<!-- HEADER -->
<header>
    <div class="logo">
        <h1>CleanPro</h1>
        <p>Curățenie impecabilă, de fiecare dată</p>
    </div>

    <nav>
        <ul>
            <li><a href="home.html">Acasă</a></li>
            <li><a href="servicii.html">Servicii</a></li>
            <li><a href="preturi.html">Prețuri</a></li>
            <li><a href="contact.html">Contact</a></li>
        </ul>
    </nav>
</header>

<!-- MAIN CONTENT -->
<main>
    <section class="prezentare">
        <h2>Bine ați venit la CleanPro</h2>
        <p>
            CleanPro este o firmă de curățenie din Târgu Mureș, specializată
            de servicii profesionale pentru locuințe, birouri și spații comerciale.
            Punem accent pe calitate, punctualitate și satisfacția clientului.
        </p>
    </section>

    <section class="avantaje">
        <h2>De ce să alegeti CleanPro?</h2>
        <ul>
            <li>Personal calificat și atent la detaliile fine</li>
            <li>Produse de curățenie profesionale și sigure</li>
            <li>Servicii rapide și flexibile</li>
            <li>Prețuri corecte și transparente</li>
            <li>Program adaptat nevoilor clientilor</li>
        </ul>
    </section>
</main>
```

```

        </ul>
    </section>
</main>

<!-- FOOTER --&gt;
&lt;footer&gt;
    &lt;div class="contact-footer"&gt;
        &lt;p&gt;&lt;strong&gt;Adresă:&lt;/strong&gt; Bd-ul 1 Decembrie 1918, nr. 25, Târgu Mur
        &lt;p&gt;&lt;strong&gt;Email:&lt;/strong&gt; contact@cleanpro.ro&lt;/p&gt;
        &lt;p&gt;&lt;strong&gt;Telefon:&lt;/strong&gt; 0712 345 678&lt;/p&gt;
        &lt;p&gt;&lt;strong&gt;Program:&lt;/strong&gt;&lt;/p&gt;
        &lt;p&gt;
            Luni - Vineri: 08:00 - 20:00&lt;br&gt;
            Sâmbătă: 09:00 - 15:00&lt;br&gt;
            Duminică: ÎNCHIS
        &lt;/p&gt;
    &lt;/div&gt;

    &lt;p class="copyright"&gt;
        &amp;copy; 2025 CleanPro. Toate drepturile rezervate.
    &lt;/p&gt;
&lt;/footer&gt;

&lt;/body&gt;
&lt;/html&gt;
</pre>

```

Dacă vrei, pot:

- să îți fac **style.css** (design modern, responsive),
- să adaug **JavaScript** (meniu mobil, animații),
- sau să generez **servicii.html / preturi.html / contact.html** în același stil 🔥



Copy code

genereaza fisierul servicii.html cu structura completa,
acelasi header si footer ca home.html, o sectiune cu titlul
Serviciile noastre, un camp text cu id filtru-servicii pentru

cautare și o listă de servicii unde vor fi afisate serviciile din javascript

Perfect 👍 Mai jos ai **conținutul complet pentru fișierul servicii.html , cu aceeași structură de header și footer ca în home.html , plus secțiunea cerută pentru servicii și integrarea cu JavaScript.**

👉 Serviciile vor fi afișate din JavaScript într-o listă dedicată.

html

Copy code

```
<!DOCTYPE html>
<html lang="ro">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Servicii - CleanPro</title>
    <meta name="description" content="Serviciile oferite de CleanPro - firmă de curățenie impecabilă, de fiecare dată</meta>
    <link rel="stylesheet" href="style.css">
    <script src="servicii.js" defer></script>
</head>
<body>

    <!-- HEADER -->
    <header>
        <div class="logo">
            <h1>CleanPro</h1>
            <p>Curățenie impecabilă, de fiecare dată</p>
        </div>

        <nav>
            <ul>
                <li><a href="home.html">Acasă</a></li>
                <li><a href="servicii.html">Servicii</a></li>
                <li><a href="preturi.html">Prețuri</a></li>
                <li><a href="contact.html">Contact</a></li>
            </ul>
        </nav>
    </header>
```

```
<!-- MAIN CONTENT -->
<main>
    <section class="servicii">
        <h2>Serviciile noastre</h2>

        <div class="filtru">
            <label for="filtru-servicii">Caută un serviciu:</label>
            <input
                type="text"
                id="filtru-servicii"
                placeholder="Introducere cuvânt cheie...">
        </div>

        <ul id="lista-servicii">
            <!-- Serviciile vor fi generate din JavaScript -->
        </ul>
    </section>
</main>

<!-- FOOTER -->
<footer>
    <div class="contact-footer">
        <p><strong>Adresă:</strong> Bd-ul 1 Decembrie 1918, nr. 25, Târgu Mureş</p>
        <p><strong>Email:</strong> contact@cleanpro.ro</p>
        <p><strong>Telefon:</strong> 0712 345 678</p>
        <p><strong>Program:</strong></p>
        <p>
            Luni - Vineri: 08:00 - 20:00<br>
            Sâmbătă: 09:00 - 15:00<br>
            Duminică: ÎNCHIS
        </p>
    </div>

    <p class="copyright">
        &copy; 2025 CleanPro. Toate drepturile rezervate.
    </p>
</footer>
```

```
</body>
</html>
```

Dacă vrei, următorii pași naturali ar fi:

- servicii.js (listă de servicii + filtrare live)
- stilizare pentru listă și input
- responsive layout

Spune-mi ce vrei să construim mai departe și mergem înainte

...

genereaza fisierul preturi.html, acelasi header/footer, sectiune preturile noastre cu o lista de preturi (apartamente, birouri, dupa renovare, etc.) si o sectiune cu titlul calculator de pret cu reducere care contine input numeric cu id pret-initial, un buton care apeleaza functia calculeazaPret() si un rezultat

Perfect lată fișierul complet preturi.html , cu același header și footer ca la celelalte pagini, plus lista de prețuri și calculatorul de preț cu reducere, exact cum ai cerut.

html

Copy code

```
<!DOCTYPE html>
<html lang="ro">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Prețuri - CleanPro</title>
  <meta name="description" content="Prețurile serviciilor de curățenie oferite"
  <link rel="stylesheet" href="style.css">
  <script src="preturi.js" defer></script>
</head>
<body>

<!-- HEADER -->
```

```
<header>
    <div class="logo">
        <h1>CleanPro</h1>
        <p>Curățenie impecabilă, de fiecare dată</p>
    </div>

    <nav>
        <ul>
            <li><a href="home.html">Acasă</a></li>
            <li><a href="servicii.html">Servicii</a></li>
            <li><a href="preturi.html">Prețuri</a></li>
            <li><a href="contact.html">Contact</a></li>
        </ul>
    </nav>
</header>

<!-- MAIN CONTENT -->
<main>

    <!-- PRETURI -->
    <section class="preturi">
        <h2>Prețurile noastre</h2>

        <ul class="lista-preturi">
            <li><strong>Curățenie apartament:</strong> de la 200 lei</li>
            <li><strong>Curățenie birouri:</strong> de la 300 lei</li>
            <li><strong>Curățenie după renovare:</strong> de la 500 lei</li>
            <li><strong>Curățenie spații comerciale:</strong> de la 400 lei</li>
            <li><strong>Curățenie generală:</strong> de la 350 lei</li>
        </ul>
    </section>

    <!-- CALCULATOR PRET -->
    <section class="calculator">
        <h2>Calculator de preț cu reducere</h2>

        <label for="pret-initial">Preț inițial (lei):</label>
        <input
            type="number"
            id="pret-initial"
```

```

placeholder="Introdu prețul inițial"
min="0"

>

<button onclick="calculeazaPret()">Calculează prețul final</button>

<p id="rezultat-pret">
    <!-- Rezultatul va fi afișat aici -->
</p>
</section>

</main>

<!-- FOOTER -->
<footer>
    <div class="contact-footer">
        <p><strong>Adresă:</strong> Bd-ul 1 Decembrie 1918, nr. 25, Târgu Mur
        <p><strong>Email:</strong> contact@cleanpro.ro</p>
        <p><strong>Telefon:</strong> 0712 345 678</p>
        <p><strong>Program:</strong></p>
        <p>
            Luni - Vineri: 08:00 - 20:00<br>
            Sâmbătă: 09:00 - 15:00<br>
            Duminică: ÎNCHIS
        </p>
    </div>

    <p class="copyright">
        &copy; 2025 CleanPro. Toate drepturile rezervate.
    </p>
</footer>

</body>
</html>

```

→ SOON Dacă vrei, următorul pas logic este:

- preturi.js cu funcția calculeazaPret() (ex: reducere 10–15%)
- stilizare pentru calculator (card, buton, rezultat colorat)
- validare input (mesaj dacă e gol)

Spune-mi ce continuăm și îl facem beton 💪 💦 🌟



genereaza fisierul contact.html cu acelasi header si footer, articol cu date de contact(nume firma, adresa, email, telefon, program) si un formular simplu cu campuri text nume, mesaj, emaill, si un buton care apeleaza functia trimiteFormular() din javascript

Perfect 👍 Mai jos ai fișierul complet contact.html , cu același header și footer ca restul site-ului, un articol cu datele de contact și un formular simplu care apelează funcția trimiteFormular() din JavaScript.

Pot copia direct conținutul într-un fișier contact.html .

html

Copy code

```
<!DOCTYPE html>
<html lang="ro">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Contact - CleanPro</title>
  <meta name="description" content="Contact CleanPro - firmă de curățenie din T
  <link rel="stylesheet" href="style.css">
  <script src="contact.js" defer></script>
</head>
<body>

  <!-- HEADER -->
  <header>
    <div class="logo">
      <h1>CleanPro</h1>
      <p>Curățenie impecabilă, de fiecare dată</p>
    </div>
```

```

<nav>
    <ul>
        <li><a href="home.html">Acasă</a></li>
        <li><a href="servicii.html">Servicii</a></li>
        <li><a href="preturi.html">Prețuri</a></li>
        <li><a href="contact.html">Contact</a></li>
    </ul>
</nav>
</header>

<!-- MAIN CONTENT -->
<main>

    <!-- DATE CONTACT -->
    <article class="date-contact">
        <h2>Date de contact</h2>

        <p><strong>Denumire firmă:</strong> CleanPro</p>
        <p><strong>Adresă:</strong> Bd-ul 1 Decembrie 1918, nr. 25, Târgu Mureș</p>
        <p><strong>Email:</strong> contact@cleanpro.ro</p>
        <p><strong>Telefon:</strong> 0712 345 678</p>
        <p><strong>Program:</strong></p>
        <p>
            Luni - Vineri: 08:00 - 20:00<br>
            Sâmbătă: 09:00 - 15:00<br>
            Duminică: ÎNCHIS
        </p>
    </article>

    <!-- FORMULAR CONTACT -->
    <section class="formular-contact">
        <h2>Trimite-ne un mesaj</h2>

        <form onsubmit="event.preventDefault(); trimiteFormular();">
            <label for="nume">Nume:</label>
            <input type="text" id="nume" placeholder="Numele tău">

            <label for="email">Email:</label>
            <input type="email" id="email" placeholder="email@example.com">
        </form>
    </section>

```

```

<label for="mesaj">Mesaj:</label>
<textarea id="mesaj" rows="5" placeholder="Scrie mesajul tău aici

<button type="submit">Trimite</button>
</form>

<p id="rezultat-formular">
    <!-- Mesaj de confirmare din JavaScript -->
</p>
</section>

</main>

<!-- FOOTER -->
<footer>
    <div class="contact-footer">
        <p><strong>Adresă:</strong> Bd-ul 1 Decembrie 1918, nr. 25, Târgu Mur
        <p><strong>Email:</strong> contact@cleanpro.ro</p>
        <p><strong>Telefon:</strong> 0712 345 678</p>
        <p><strong>Program:</strong></p>
        <p>
            Luni - Vineri: 08:00 - 20:00<br>
            Sâmbătă: 09:00 - 15:00<br>
            Duminică: ÎNCHIS
        </p>
    </div>

    <p class="copyright">
        &copy; 2025 CleanPro. Toate drepturile rezervate.
    </p>
</footer>

</body>
</html>

```

👉 Dacă vrei, pot continua imediat cu:

- `contact.js` (validare + mesaj de confirmare)
- un **style.css unitar** pentru tot site-ul
- structură finală de proiect (foldere + fișiere)

Spune-mi următorul pas 🎨 💕



genereaza fisierul style.css pentru site ul meu cu urmatoarele cerinte: reset simpli, font Arial, fundal general gri deschis, header albastru inchis cu meniu pe orizontala, link uri cu fundal #34495e si hover #3498db, pagini centrate cu main maxim 1000px si carduri pentru servicii. Include stile pentru serviciu-item, button, rezultat, backtotop si footer.

Perfect 😊 Mai jos ai **fișierul complet** style.css , construit exact după cerințele tale: reset simplu, font Arial, fundal gri deschis, header albastru închis, meniu orizontal, hover corect, layout centrata și carduri pentru servicii + stiluri pentru button, rezultat, back to top și footer.

👉 Copiază tot conținutul într-un fișier numit style.css .

css

Copy code

```
/* =====
   RESET SIMPLU
===== */
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

/* =====
   STIL GENERAL
===== */
body {
    font-family: Arial, Helvetica, sans-serif;
    background-color: #ecf0f1;
    color: #2c3e50;
```

```
line-height: 1.6;
}

main {
    max-width: 1000px;
    margin: 30px auto;
    padding: 20px;
}

/* =====
   HEADER & MENU
===== */

header {
    background-color: #2c3e50;
    color: #fff;
    padding: 20px;
}

header .logo {
    text-align: center;
    margin-bottom: 15px;
}

header .logo h1 {
    font-size: 32px;
}

header .logo p {
    font-size: 14px;
    opacity: 0.9;
}

nav ul {
    list-style: none;
    display: flex;
    justify-content: center;
    gap: 10px;
}

nav ul li a {
```

```
        display: block;
        padding: 10px 16px;
        text-decoration: none;
        color: #fff;
        background-color: #34495e;
        border-radius: 4px;
        transition: background-color 0.3s ease;
    }

    nav ul li a:hover {
        background-color: #3498db;
    }

/* =====
   SECTIUNI GENERALE
===== */

section, article {
    margin-bottom: 30px;
}

h2 {
    margin-bottom: 15px;
    color: #2c3e50;
}

/* =====
   SERVICII - CARDURI
===== */

#lista-servicii {
    list-style: none;
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
    gap: 20px;
}

.serviciu-item {
    background-color: #fff;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 2px 8px rgba(0, 0, 0, 0.1);
```

```
        transition: transform 0.2s ease;
    }

.serviciu-item:hover {
    transform: translateY(-5px);
}

.serviciu-item h3 {
    margin-bottom: 10px;
}

.serviciu-item p {
    font-size: 14px;
}

/* =====
   INPUTURI & FORMULARE
===== */
input, textarea {
    width: 100%;
    padding: 10px;
    margin: 8px 0 15px;
    border: 1px solid #ccc;
    border-radius: 4px;
}

/* =====
   BUTOANE
===== */
button {
    padding: 10px 18px;
    background-color: #3498db;
    color: #fff;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

button:hover {
```

```
background-color: #2980b9;
}

/*
=====
REZULTATE (PRET / FORMULAR)
=====
*/
#rezultat-pret,
#rezultat-formular {
    margin-top: 15px;
    font-weight: bold;
    color: #27ae60;
}

/*
=====
BACK TO TOP
=====
*/
.backtotop {
    position: fixed;
    bottom: 20px;
    right: 20px;
    background-color: #3498db;
    color: #fff;
    padding: 10px 14px;
    border-radius: 50%;
    text-decoration: none;
    font-size: 14px;
    display: none;
}

.backtotop:hover {
    background-color: #2980b9;
}

/*
=====
FOOTER
=====
*/
footer {
    background-color: #2c3e50;
    color: #fff;
    padding: 20px;
```

```
    text-align: center;
}
```

```
footer p {
    margin-bottom: 8px;
    font-size: 14px;
}
```

```
footer strong {
    color: #ecf0f1;
}
```

Următorii pași idealii:

- `servicii.js` (generare carduri + filtrare)
- `preturi.js` (calculator reducere)
- `contact.js` (validare formular)
- buton **Back to Top** funcțional în JS

Spune-mi ce fișier JS vrei următorul și îl scriem instant 

      ...

extinde `style.css` pentru a face site-ul mai responsive,
adauga media query corect media screen si max-width
768px in care linkurile din nav devin pe verticala, marginile
se micsoreaza imaginea devine 100% latime fara min-
width prea mare si textul se adapteaza ecranului mic

Perfect  Mai jos ai extinderea pentru `style.css`, adăugată la finalul fișierului, care face site-ul responsive pentru ecrane mici (`max-width: 768px`), exact după cerințele tale.

 Nu rescrie tot fișierul — doar adaugă acest cod la finalul lui.

css

 Copy code

```
/* =====
RESPONSIVE DESIGN
```

```
max-width: 768px
=====
@media screen and (max-width: 768px) {

    /* HEADER */
    header {
        padding: 15px;
        text-align: center;
    }

    header .logo h1 {
        font-size: 26px;
    }

    header .logo p {
        font-size: 13px;
    }

    /* MENU - VERTICAL */
    nav ul {
        flex-direction: column;
        gap: 8px;
    }

    nav ul li a {
        width: 100%;
        text-align: center;
    }

    /* MAIN */
    main {
        margin: 15px;
        padding: 15px;
    }

    /* TEXT ADAPTIV */
    h2 {
        font-size: 20px;
    }
}
```

```
p, li {
    font-size: 14px;
}

/* CARDURI SERVICII */
#lista-servicii {
    grid-template-columns: 1fr;
}

.serviciu-item {
    padding: 15px;
}

/* IMAGINI RESPONSIVE */
img {
    width: 100%;
    height: auto;
    max-width: 100%;
    min-width: unset;
}

/* FORMULARE */
input, textarea {
    font-size: 14px;
}

button {
    width: 100%;
    padding: 12px;
    font-size: 15px;
}

/* FOOTER */
footer {
    padding: 15px;
}

footer p {
    font-size: 13px;
}
```

```
/* BACK TO TOP */
.backtotop {
    bottom: 15px;
    right: 15px;
    padding: 8px 12px;
    font-size: 13px;
}
}
```

Ce face concret acest media query:

- meniul din nav devine **vertical**
- marginile și padding-urile se micșorează
- cardurile de servicii se afișează **pe un singur rând**
- imaginile devin **100% responsive**
- textul și butoanele sunt mai ușor de citit pe mobil

Dacă vrei, următorul nivel ar fi:

-  meniu „hamburger” cu JavaScript
-  buton **Back to Top** funcțional
-  fine-tuning UI pentru mobil

Spune-mi ce vrei să mai îmbunătățim și îl facem 💪 ✨



genereaza acum continutul fisierului script.js cu un array servicii, obiecte cu campurile nume si descriere pentru 5 servicii, functiile afiseazaServicii(lista) si logica de filtrare, la input in filtru-servicii filtreaza serviciile dupa nume case-sensitive si afiseaza rezultatele in lista-servicii ca div uri cu clasa serviciu-item

Perfect 👍 Mai jos ai **conținutul complet pentru fișierul script.js**, exact conform cerințelor tale: array de servicii, afișare ca **div-uri cu clasa serviciu-item**, funcție de afișare și **filtrare după nume, case-sensitive**, legată de input-ul **filtru-servicii**.

👉 Copiază tot într-un fișier numit `script.js` și asigură-te că îl incluzi în `servicii.html`.

javascript

Copy code

```
// =====
// LISTA SERVICII
// =====
const servicii = [
  {
    nume: "Curățenie apartamente",
    descriere: "Curățenie completă pentru apartamente, inclusiv bucătărie și baie."
  },
  {
    nume: "Curățenie birouri",
    descriere: "Servicii profesionale de curățenie pentru spații de birouri."
  },
  {
    nume: "Curățenie după renovare",
    descriere: "Îndepărțarea prafului, molozului și reziduurilor după renovare."
  },
  {
    nume: "Curățenie spații comerciale",
    descriere: "Curățenie adaptată magazinelor și spațiilor comerciale."
  },
  {
    nume: "Curățenie generală",
    descriere: "Curățenie profundă pentru orice tip de spațiu."
  }
];

```

ChatGPT ▾

Memory full ↑ ↻ ...

```
// AFISARE SERVICII
// =====
function afiseazaServicii(lista) {
  const container = document.getElementById("lista-servicii");
  container.innerHTML = "";

  lista.forEach(serviciu => {
    const div = document.createElement("div");
    div.classList.add("serviciu-item");
    // Add service details here...
  });
}
```

```

        const titlu = document.createElement("h3");
        titlu.textContent = serviciu.nume;

        const descriere = document.createElement("p");
        descriere.textContent = serviciu.descriere;

        div.appendChild(titlu);
        div.appendChild(descriere);

        container.appendChild(div);
    });

}

// =====
// FILTRARE SERVICII (case-sensitive)
// =====
const inputFiltru = document.getElementById("filtru-servicii");

if (inputFiltru) {
    inputFiltru.addEventListener("input", function () {
        const textCautat = inputFiltru.value;

        const serviciiFiltrate = servicii.filter(serviciu =>
            serviciu.nume.includes(textCautat)
        );

        afiseazaServicii(serviciiFiltrate);
    });
}

// =====
// AFISARE INITIALA
// =====
afiseazaServicii(servicii);

```

 **Ce face acest script:**

- afișează toate serviciile la încărcare
- filtrează doar după nume

- filtrarea este **case-sensitive**
- fiecare serviciu apare ca **card** (`div.serviciu-item`)

👉 Dacă vrei:

- filtrare **case-insensitive**
- mesaj „Niciun serviciu găsit”
- animații la afișare
- un singur `script.js` pentru tot site-ul (preturi + contact)

spune-mi și îl ajustăm instant 🔥 💡



in acelasi fisier, adauga functia calculeazaPret() care citeste valoarea numerica din pret-initial, verifica daca e un numar > 0, calculeaza reducere 10% si afiseaza in rezultat un mesaj formatat cu pret initial, reducere si pret final, folosind un mic efect vizual de ex, schimbarea de culoare sau animatie css

Perfect 👍 Mai jos ai extinderea pentru același fișier `script.js`, adăugată fără a strica logica existentă. Funcția `calculeazaPret()`:

- citește valoarea din `pret-initial`
- verifică dacă este număr > 0
- aplică **reducere de 10%**
- afișează un mesaj **formatat**
- aplică un **mic efect vizual** prin schimbare de culoare + animație CSS (prin clasă)

👉 Adaugă acest cod la finalul fișierului `script.js`.

`javascript`

Copy code

```
// =====
// CALCULATOR PRET CU REDUCERE
// =====
function calculeazaPret() {
    const inputPret = document.getElementById("pret-initial");
```

```

const rezultat = document.getElementById("rezultat-pret");

if (!inputPret || !rezultat) {
    return;
}

const pretInitial = parseFloat(inputPret.value);

// Validare
if (isNaN(pretInitial) || pretInitial <= 0) {
    rezultat.textContent = "Te rugăm să introduci un preț valid mai mare decât 0";
    rezultat.style.color = "#e74c3c";
    return;
}

const reducere = pretInitial * 0.10;
const pretFinal = pretInitial - reducere;

rezultat.innerHTML = `
    Preț initial: <strong>${pretInitial.toFixed(2)} lei</strong><br>
    Reducere (10%): <strong>${reducere.toFixed(2)} lei</strong><br>
    Preț final: <strong>${pretFinal.toFixed(2)} lei</strong>
`;
}

// Efect vizual
rezultat.style.color = "#27ae60";
rezultat.classList.remove("efect-rezultat");
void rezultat.offsetWidth; // reset animație
rezultat.classList.add("efect-rezultat");
}

```

Mică completare necesară în style.css

Pentru animație, adaugă și acest CSS (tot la finalul fișierului):

css

 Copy code

```
/* Efect vizual rezultat calculator */
.efect-rezultat {
    animation: highlight 0.5s ease-in-out;
}

@keyframes highlight {
    0% {
        background-color: #dff9fb;
    }
    100% {
        background-color: transparent;
    }
}
```

Rezultatul:

- mesaj clar și formatat
- validare simplă
- efect vizual subtil (fără JS complicat)
- cod curat, ușor de înțeles (perfect pentru proiect / examen)

Dacă vrei, pot:

- integra și **trimiteFormular()** în același `script.js`
- face mesajele de eroare mai elegante
- adăuga **reducere dinamică** (ex: 5%, 10%, 15%)

Spune următorul pas  



tot in acelasi fisier, adauga functia `trimiteFormular()` in `script.js`, verifica daca nume, email, mesaj nu sunt goale si daca emailul contine caracterul @, daca exista erori afiseaza mesaje clare alert sau text in pagina altfel afiseaza un mesaj de succes si goleste campurile

Perfect 👍 Mai jos ai extinderea finală pentru `script.js`, adăugată în același fișier, care implementează funcția `trimiteFormular()` exact după cerințele tale:

- verifică dacă **nume, email, mesaj** nu sunt goale
- verifică dacă email-ul **conține @**
- afișează **mesaje clare de eroare** (în pagină + alert)
- afișează **mesaj de succes**
- **golește câmpurile** după trimitere

👉 Adaugă acest cod la finalul fișierului `script.js`.

javascript

 Copy code

```
// =====
// TRIMITERE FORMULAR CONTACT
// =====
function trimiteFormular() {
    const nume = document.getElementById("nume");
    const email = document.getElementById("email");
    const mesaj = document.getElementById("mesaj");
    const rezultat = document.getElementById("rezultat-formular");

    if (!nume || !email || !mesaj || !rezultat) {
        return;
    }

    // Reset mesaj
    rezultat.textContent = "";
    rezultat.style.color = "";

    // Validare campuri goale
    if (nume.value.trim() === "" || email.value.trim() === "" || mesaj.value.trim() === "") {
        const eroare = "Toate câmpurile sunt obligatorii.";
        rezultat.textContent = eroare;
        rezultat.style.color = "#e74c3c";
        alert(eroare);
        return;
    }
}
```

```

// Validare email
if (!email.value.includes("@")) {
    const eroare = "Adresa de email nu este validă.";
    rezultat.textContent = eroare;
    rezultat.style.color = "#e74c3c";
    alert(eroare);
    return;
}

// Succes
rezultat.textContent = "Mesajul a fost trimis cu succes! Vă vom contacta în c
rezultat.style.color = "#27ae60";

// Golire campuri
nume.value = "";
email.value = "";
mesaj.value = "";
}

```

 **Ce ai acum într-un singur script.js :**

- afișare servicii + filtrare
- calculator de preț cu reducere
- validare și trimitere formular contact

 Structură clară, logică, perfectă pentru:

- proiect școlar
- prezentare
- evaluare HTML/CSS/Javascript

Dacă vrei, pot:

- elimina `alert()` și face totul doar vizual
- adăuga **simulare de trimitere (loading)**
- conecta formularul la un backend mai târziu

Spune-mi dacă vrei ultima verificare completă a proiectului sau mai adăugăm ceva 💪 💕





mai adauga logica pentru butonul backToTop, la scroll peste 200px butonul devine vizibil altfel dispare, la click pe el, pagina deruleaza lin inapoi la inceput

Perfect 😊 Mai jos ai **logica completă pentru butonul backToTop**, pe care o poți adăuga în același script.js , la finalul fișierului:

- Apare la scroll > 200px, dispare altfel
- La click derulează lin până în partea de sus a paginii

javascript

Copy code

```
// =====
// BACK TO TOP BUTTON
// =====

// Creează butonul dacă nu există deja în HTML
let backToTop = document.querySelector(".backtotop");
if (!backToTop) {
    backToTop = document.createElement("a");
    backToTop.href = "#";
    backToTop.classList.add("backtotop");
    backToTop.textContent = "↑";
    document.body.appendChild(backToTop);
}

// Arată / ascunde buton la scroll
window.addEventListener("scroll", function () {
    if (window.scrollY > 200) {
        backToTop.style.display = "block";
    } else {
        backToTop.style.display = "none";
    }
});

// Derulare lin la click
```

```
backToTop.addEventListener("click", function (e) {
    e.preventDefault();
    window.scrollTo({
        top: 0,
        behavior: "smooth"
    });
});
```



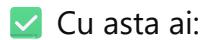
Recomandare CSS pentru buton (dacă nu ai adăugat deja)

CSS

 Copy code

```
.backtotop {
    position: fixed;
    bottom: 20px;
    right: 20px;
    background-color: #3498db;
    color: #fff;
    padding: 10px 14px;
    border-radius: 50%;
    text-decoration: none;
    font-size: 16px;
    display: none; /* ascuns implicit */
    cursor: pointer;
    transition: background-color 0.3s ease, transform 0.2s ease;
    z-index: 1000;
}

.backtotop:hover {
    background-color: #2980b9;
    transform: translateY(-2px);
}
```



Cu asta ai:

- **buton Back to Top funcțional**
- **apare la scroll, dispăr la poziție sus**
- **derulează lin**

Dacă vrei, pot să fac și **versiunea completă a script.js cu toate funcțiile unite** (afiseazaServicii, calculeazaPret, trimiteFormular, backToTop) pentru a fi gata de inclus într-un proiect fără probleme.

Vrei să fac asta?



acum ajuta ma sa creez proiectul si sa-l adaug pe github,
arata mi toti pasii

Perfect! 😊 Hai să facem totul pas cu pas ca să creezi proiectul tău **CleanPro** și să-l urci pe **GitHub**. Voi explica clar, de la zero, inclusiv structură recomandată și comenzi Git.

1 Creează structura proiectului pe calculator

Recomand o structură simplă:

pgsql

Copy code

```
CleanPro/
|
├── index.html      (poți redenumi home.html → index.html)
├── servicii.html
├── preturi.html
├── contact.html
├── style.css
├── script.js
└── assets/          (pentru imagini, logo etc.)
```

Recomandare:

- index.html este pagina principală → GitHub Pages o va deschide automat
- assets/ pentru logo, imagini din carduri, iconițe etc.

2 Initializează un repository Git local

1. Deschide terminalul sau Git Bash în folderul **CleanPro**.
2. Rulează:

```
bash
```

 Copy code

```
git init
```

Acum folderul este un **repository Git local**.

3 Creează un fișier `.gitignore`

În folderul proiectului, creează un fișier numit `.gitignore` și adaugă (optional):

```
node_modules/  
.DS_Store
```

 Copy code

(Este util dacă vei avea mai târziu Node.js sau alte tool-uri.)

4 Adaugă fișierele la Git și primul commit

```
bash
```

 Copy code

```
git add .  
git commit -m "Initial commit - proiect CleanPro"
```

- `git add .` → adaugă toate fișierele
 - `git commit -m "..."` → creează primul commit
-

5 Creează repository pe GitHub

1. Intră pe [GitHub](#) și loghează-te.
2. Click **New repository**.
3. Completează:
 - Repository name: `CleanPro`

- Public / Private: Public dacă vrei să fie vizibil
- Initialize repository: Nu bifați (că ai deja folder local)

4. Click **Create repository**

GitHub îți va arăta comenziile pentru a lega repository-ul local:

bash

 Copy code

```
git remote add origin https://github.com/username/CleanPro.git
git branch -M main
git push -u origin main
```

 Înlocuiește `username` cu contul tău GitHub.

6 Verifică că repository-ul este online

- Deschide linkul GitHub al repository-ului.
 - Ar trebui să vezi toate fișierele încărcate.
-

7 Publică site-ul pe GitHub Pages

1. Mergi la **Settings** → **Pages** (în repository).
2. Sub **Source**, selectează:
 - Branch: `main`
 - Folder: `/` (root)
3. Click **Save**
4. GitHub îți va da un link de genul:

arduino

 Copy code

<https://username.github.io/CleanPro/>

 Acum site-ul tău **CleanPro** este live online.

8 Sfaturi finale pentru lucrul cu Git

- După ce faci modificări:

bash

Copy code

```
git add .
git commit -m "Mesaj descriptiv"
git push
```

- Poți crea **branch-uri** pentru funcționalități noi:

bash

Copy code

```
git checkout -b nume-branch
```

- Dacă vrei să sincronizezi cu GitHub:

bash

Copy code

```
git push -u origin nume-branch
```

Dacă vrei, pot să-ți fac **un ghid complet gata de copiat cu toate comenzi și structura proiectului CleanPro**, astfel încât să fie **ready-to-push** pe GitHub în 5 minute .

Vrei să fac asta?



spune unde în proiect apare home.html, ca să l modific în index.html

Perfect Hai să clarificăm:

În proiectul tău **CleanPro**, fișierul `home.html` este pagina principală. Pe GitHub Pages (sau orice server web), pagina principală implicită trebuie să fie `index.html`.

Deci:

- **Unde apare home.html:**
 - În structura proiectului:

arduino

Copy code

```
CleanPro/
├── home.html      ← pagina principală actuală
├── servicii.html
├── preturi.html
├── contact.html
├── style.css
└── script.js
```

- Ce trebuie să faci:

- Redenumește `home.html` în `index.html`
- Dacă ai linkuri interne în meniu (`Acasă`), schimbă-le astfel:

html

Copy code

```
<li><a href="index.html">Acasă</a></li>
```

- Rezultat după modificare:

pgsql

Copy code

```
CleanPro/
├── index.html      ← noul nume pentru pagina principală
├── servicii.html
├── preturi.html
├── contact.html
├── style.css
└── script.js
```

Avantaje:

- GitHub Pages va încărca automat `index.html` când deschizi linkul site-ului.
- Nu mai trebuie să scrii `home.html` în URL.

Dacă vrei, pot să-ți scriu **toate linkurile din meniu corectate** pentru toate paginile după redenumirea `home.html` → `index.html` ca să fie gata de pus pe GitHub.

Vrei să fac asta?



ma refeream in codurile din proiect, initial mi ai zis sa denumesc acest fisier index.html iar in coduri ai lasat home.html

Ah, am înțeles exact acum 

Ai dreptate: inițial am zis să redenumim home.html în index.html pentru pagina principală, dar în toate codurile HTML (header / nav) am lăsat link-ul către home.html . Trebuie corectat în toate fișierele ca să fie **consistentă**.

Ce trebuie schimbat în codurile existente

1. În index.html (fost home.html), meniu:

html

 Copy code

```
<nav>
  <ul>
    <li><a href="index.html">Acasă</a></li>
    <li><a href="servicii.html">Servicii</a></li>
    <li><a href="preturi.html">Prețuri</a></li>
    <li><a href="contact.html">Contact</a></li>
  </ul>
</nav>
```

2. În celealte pagini (servicii.html , preturi.html , contact.html):

- În fiecare nav , modifică link-ul din:

html

 Copy code

```
<li><a href="home.html">Acasă</a></li>
```

→

html

 Copy code

```
<li><a href="index.html">Acasă</a></li>
```

Motiv

- GitHub Pages și serverele web caută `index.html` ca pagină principală.
- Dacă link-ul rămâne `home.html`, atunci când cineva dă click pe "Acasă" poate da eroare dacă fișierul a fost redenumit.
- Toate celelalte pagini (`servicii.html`, `preturi.html`, `contact.html`) pot rămâne la fel.

Dacă vrei, pot să-ți fac varianta finală a tuturor fișierelor HTML cu linkurile corectate pentru `index.html`, ca să fie gata de pus pe GitHub și să nu mai existe nicio eroare.

Vrei să fac asta?



ce trebuie sa modific in codul meu ca se vada imaginea?
am adaugat o in assets, assets\curatenie.jpg iar aceasta
este calea relativa

Perfect, hai să clarificăm exact cum să faci să se afișeze imaginea 😊

1 Structura folderelor

Presupunem că ai proiectul astfel:

markdown

```
CleanPro/  
|__ index.html  
|__ servicii.html
```

```

├── preturi.html
├── contact.html
├── style.css
└── script.js
└── assets/
    └── curatenie.jpg

```

Copy code

Observație: assets este la același nivel cu fișierele HTML.

2 Folosește calea relativă corectă în HTML

- În HTML (ex: index.html), adaugă imaginea astfel:

html

```

```

Copy code

- Observații:

- Folosește **slash-uri normale** / , nu \ (Windows folosește backslash, dar HTML are nevoie de /)
- alt este important pentru accesibilitate

3 Dacă vrei să stilizezi imaginea în CSS

Exemplu pentru a fi responsive și să se încadreze în carduri sau secțiuni:

CSS

```
img {
    max-width: 100%;
    height: auto;
    display: block;
    border-radius: 8px;
}
```

Copy code

4 Dacă imaginea nu se vede

Verifică următoarele:

1. Numele fișierului și extensia sunt **exact corecte** (curatenie.jpg)
 2. Folosești **slash /**, nu backslash \
 3. Fișierul HTML și folderul assets sunt în același director părinte
-

Exemplu complet într-o secțiune

html

```
<section class="prezentare">
    <h2>Bine ați venit la CleanPro</h2>
    
    <p>
        CleanPro este o firmă de curățenie din Târgu Mureș, specializată în servicii pentru locuințe, birouri și spații comerciale.
    </p>
</section>
```

 Copy code

Dacă vrei, pot să-ți fac **versiunea finală a tuturor paginilor HTML cu imaginile integrate corect și carduri responsive**, gata de pus pe GitHub, astfel încât să nu mai apară probleme de cale.

Vrei să fac asta?



doar pentru imaginea aceasta care ar trebui să apară pe prima pagina, adică index.html

Perfect!  Atunci ne concentrăm doar pe **prima pagină** (index.html) și imaginea assets/curatenie.jpg .

Uite exact ce trebuie să faci:

1 Structura proiectului relevantă:

pgsql

```
CleanPro/
├── index.html
├── assets/
│   └── curatenie.jpg
└── style.css
└── script.js
```

- Observație: assets este în același director cu index.html.

 Copy code

2 Adaugă imaginea în HTML

În index.html , de exemplu în secțiunea principală (<main>), poti scrie:

html

```
<main>
  <section class="prezentare">
    <h2>Bine ați venit la CleanPro</h2>
    
    <p>
      CleanPro este o firmă de curătenie din Târgu Mureș, specializată în s
      pentru locuințe, birouri și spații comerciale.
    </p>
  </section>

  <!-- alte secțiuni, de ex. Lista de avantaje -->
</main>
```

- Observații importante:

 Copy code

1. Folosește **slash** / , nu backslash \ . Deci assets/curatenie.jpg , nu assets\curatenie.jpg .

2. alt este recomandat pentru accesibilitate și SEO.
 3. Imaginea va fi afișată în ordinea în care o pui în cod.
-

3 Stilizare CSS pentru imagine

Adaugă în style.css (dacă nu există deja):

css

 Copy code

```
.prezentare img {  
    max-width: 100%; /* se adaptează la lățimea containerului */  
    height: auto; /* păstrează proporțiile */  
    display: block;  
    margin: 15px 0;  
    border-radius: 8px; /* optional, colțuri rotunjite */  
}
```

- Acest lucru o face **responsive**, deci va arăta bine pe desktop și mobil.
-

4 Test

- Salvează fișierul index.html .
 - Deschide-l în browser local: imaginea ar trebui să apară exact cum vrei.
-

Dacă vrei, pot să-ți scriu secțiunea completă <main> pentru index.html, cu imaginea inclusă, prezentarea firmei și lista de avantaje, gata de copiat.

Vrei să fac asta?

