Home.ts  
import { Component } from '@angular/core';

import { CommonModule } from '@angular/common';

import { FormsModule } from '@angular/forms';

@Component({

selector: 'app-home',

standalone: true,

imports: [CommonModule, FormsModule],

templateUrl: './home.html',

styleUrl: './home.css'

})

export class Home {

prompt: string = '';

isSubmitting = false;

// Array to store submitted prompts

submittedPrompts: string[] = [];

submitPrompt(): void {

const text = this.prompt.trim();

if (!text || this.isSubmitting) return;

this.isSubmitting = true;

// Simulate async submission

setTimeout(() => {

console.log('Submitted prompt:', text);

this.submittedPrompts.push(text); // Add submitted prompt to array

this.prompt = ''; // clear input

this.isSubmitting = false;

}, 600);

}

clearPrompt(): void {

if (this.isSubmitting) return;

this.prompt = '';

}

get isSubmitDisabled(): boolean {

return !((this.prompt ?? '').trim()) || this.isSubmitting;

}

}

Html.  
<div class="home-center dark-theme" style="--header-h: 0px;">

<section class="prompt-card" role="region" aria-labelledby="promptTitle">

<h2 class="title" id="promptTitle">Ask anything</h2>

<label class="sr-only" for="promptInput">Your prompt</label>

<textarea

id="promptInput"

class="prompt-input"

[(ngModel)]="prompt"

placeholder="Type your prompt here…"

(keydown.control.enter)="submitPrompt()"

(keydown.meta.enter)="submitPrompt()"

></textarea>

<div class="actions">

<button class="clear-btn" type="button" (click)="clearPrompt()" [disabled]="!prompt.length || isSubmitting">

Clear

</button>

<button class="primary-btn" type="button" (click)="submitPrompt()" [disabled]="isSubmitDisabled">

{{ isSubmitting ? 'Submitting…' : 'Submit' }}

</button>

</div>

<!-- Submitted prompts displayed below -->

<div class="submitted-prompts" \*ngIf="submittedPrompts.length">

<h3 class="submitted-title">Submitted Prompts:</h3>

<div class="prompt-card-small" \*ngFor="let p of submittedPrompts">

{{ p }}

</div>

</div>

</section>

</div>

Css  
/\* Ensures the viewport fills fully and has no default margins \*/

html, body {

  height: 100%;

  margin: 0;

}

/\* Centering wrapper + dark page background \*/

.home-center.dark-theme {

  /\* If you have a fixed header, set --header-h inline or in CSS \*/

  min-height: calc(100svh - var(--header-h, 0px));

  display: grid;

  place-items: center;

  padding: clamp(16px, 3vw, 48px);

  background:

    radial-gradient(80rem 50rem at 10% 0%, #152043 0%, transparent 55%),

    radial-gradient(70rem 40rem at 90% 100%, #0d3b66 0%, transparent 55%),

    linear-gradient(180deg, var(--bg) 0%, var(--bg) 100%);

  color: var(--fg);

  /\* Tell the browser we’re in dark mode (affects form controls, scrollbars) \*/

  color-scheme: dark;

}

/\* Prompt card \*/

.prompt-card {

  width: min(480px, 92vw);

  background: var(--card);

  backdrop-filter: blur(10px);

  -webkit-backdrop-filter: blur(10px);

  border: 1px solid var(--border);

  border-radius: 16px;

  padding: clamp(16px, 2.5vw, 28px);

  box-shadow:

    0 20px 60px rgba(0, 0, 0, 0.35),

    inset 0 1px 0 rgba(255, 255, 255, 0.04);

  animation: floatIn 560ms ease-out both;

  /\* Safety: if a parent uses align-items: stretch, keep it compact \*/

  max-width: 100%;

  justify-self: center;

  align-self: center;

}

/\* Title \*/

.title {

  margin: 0 0 12px;

  font-weight: 700;

  letter-spacing: 0.2px;

  font-size: clamp(1.1rem, 1rem + 0.8vw, 1.5rem);

  color: #25369b;

}

/\* DARK textarea \*/

.prompt-input {

  width: 100%;

  max-width: 100%;        /\* Prevents textarea from overflowing card \*/

  box-sizing: border-box;  /\* Includes padding in width \*/

  min-height: clamp(130px, 24vh, 280px);

  background: rgba(141, 148, 179, 0.86);         /\* dark slate \*/

  color: var(--fg);

  border: 1.5px solid rgba(255, 255, 255, 0.16);

  border-radius: 14px;

  padding: 14px 16px;

  resize: vertical;

  outline: none;

  font: 500 1rem/1.6 ui-sans-serif, system-ui, -apple-system, Segoe UI,

    Roboto, Noto Sans, Ubuntu, Cantarell, Helvetica Neue, Arial, "Apple Color Emoji", "Segoe UI Emoji";

  box-shadow: inset 0 1px 0 rgba(154, 138, 138, 0.06);

  transition: border-color 160ms ease, box-shadow 160ms ease, background 160ms ease;

}

.prompt-input::placeholder {

  color: #d6dadf; /\* slate-400-ish \*/

}

.prompt-input:focus {

  border-color: var(--primary);

  box-shadow:

    0 0 0 3px var(--ring),

    inset 0 1px 0 rgba(255, 255, 255, 0.06);

  background: rgba(25, 38, 95, 0.92);

}

/\* Buttons row \*/

.actions {

  display: flex;

  gap: 12px;

  justify-content: flex-end;

  margin-top: 12px;

}

/\* Buttons \*/

.primary-btn,

.clear-btn {

  appearance: none;

  border: 1px solid transparent;

  border-radius: 12px;

  padding: 10px 16px;

  font-weight: 600;

  cursor: pointer;

  transition: transform 120ms ease, box-shadow 160ms ease, background 160ms ease, border-color 160ms ease;

}

.primary-btn {

  color: #093333;

  background: linear-gradient(180deg, var(--primary) 0%, var(--primary-strong) 100%);

  box-shadow:

    0 8px 20px rgba(99, 102, 241, 0.35),

    inset 0 1px 0 rgba(255, 255, 255, 0.25);

}

.primary-btn:hover { transform: translateY(-1px); }

.primary-btn:active { transform: translateY(0); }

.primary-btn:disabled { opacity: 0.55; cursor: not-allowed; box-shadow: none; }

/\* Ghost/clear button \*/

.clear-btn {

  background: transparent;

  color: var(--fg);

  border-color: var(--border);

}

.clear-btn:hover {

  border-color: rgba(255, 255, 255, 0.32);

  background: rgba(255, 255, 255, 0.06);

}

/\* Hint + kbd \*/

.hint {

  margin-top: 10px;

  color: var(--muted);

  font-size: 0.85rem;

}

kbd {

  background: rgba(255, 255, 255, 0.18);

  border: 1px solid rgba(255, 255, 255, 0.25);

  border-bottom-width: 2px;

  padding: 2px 6px;

  border-radius: 6px;

  font-family: ui-monospace, SFMono-Regular, Menlo, Monaco, Consolas, "Liberation Mono", "Courier New", monospace;

  font-size: 0.85em;

}

/\* Screen reader helper \*/

.sr-only {

  position: absolute;

  width: 1px;

  height: 1px;

  padding: 0;

  margin: -1px;

  overflow: hidden;

  clip: rect(0, 0, 1px, 1px);

  white-space: nowrap;

  border: 0;

}

/\* Animation \*/

@keyframes floatIn {

  from { opacity: 0; transform: translateY(8px) scale(0.99); }

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

}

/\* Submitted prompts section \*/

.submitted-prompts {

  margin-top: 16px;

}

.submitted-title {

  margin-bottom: 8px;

  font-weight: 600;

  font-size: 1rem;

  color: #25369b; /\* same as your title color \*/

}

.prompt-card-small {

  background: rgba(141, 148, 179, 0.2); /\* subtle shade, matches dark-theme style \*/

  padding: 10px 14px;

  border-radius: 12px;

  margin-bottom: 8px;

  font-size: 0.95rem;

  color: var(--fg); /\* same text color as original \*/

  word-wrap: break-word;

  box-shadow: inset 0 1px 2px rgba(0,0,0,0.1);

}

/\* IMPORTANT: If you had a light-mode block elsewhere, this keeps us dark \*/

@media (prefers-color-scheme: light) {

  .home-center.dark-theme {

    /\* Stay dark even if system is light \*/

    background:

      radial-gradient(80rem 50rem at 10% 0%, #152043 0%, transparent 55%),

      radial-gradient(70rem 40rem at 90% 100%, #0d3b66 0%, transparent 55%),

      linear-gradient(180deg, var(--bg) 0%, var(--bg) 100%);

    color: var(--fg);

  }

  .home-center.dark-theme .prompt-card {

    background: var(--card);

    border-color: var(--border);

    box-shadow:

      0 20px 60px rgba(0, 0, 0, 0.35),

      inset 0 1px 0 rgba(255, 255, 255, 0.04);

  }

  .home-center.dark-theme .clear-btn { color: var(--fg); border-color: var(--border); }

  .home-center.dark-theme .hint { color: var(--muted); }

}