**Smart To-Do (tags, filters, drag-reorder)**

Build a SINGLE-FILE app in one HTML document (inline CSS + inline JS, no external libraries) called “Smart To-Do”.

Requirements:

- Add tasks with title, optional description, due date, and tags (comma-separated).

- Filter by tag and search by text; show counts (All, Active, Done).

- Drag-and-drop to reorder; click to mark done/undo.

- Persist all data in localStorage. Include “Export/Import JSON”.

- Responsive layout, accessible (labels, keyboard focus), and clean minimalist styling.

- Comment the code well and keep everything in one HTML file only.

**2) Daily Habit Tracker (streaks + calendar view)**

Create a SINGLE HTML file (inline CSS + JS only; no external libs) named “Habit Tracker”.

Features:

- Add habits with color and icon (use emoji), track completion per day on a mini calendar.

- Show current streak and best streak per habit.

- Week and month summaries with completion percentages.

- Edit/Archive habits; localStorage persistence and “Reset all” control.

- Mobile-friendly grid and keyboard navigation for toggling days.

- Clear, modern UI with comments throughout. One file only.

**3) Personal Expense Tracker (categories + charts)**

Generate ONE self-contained HTML file (inline CSS and JS) “Expense Tracker”.

- Add expenses: date, category, amount, note.

- Category manager (add/rename/delete categories).

- Summary cards: total this month, average per day, top 3 categories.

- Simple bar chart and pie chart using <canvas> (vanilla JS, no libraries).

- Filter by date range and category; export CSV; import CSV (append or overwrite).

- localStorage persistence; responsive and accessible.

- Clean UI and well-commented code. No external files.

**4) Flashcards Studio (CSV import + spaced review)**

Write a SINGLE-PAGE HTML app (all CSS/JS inline) named “Flashcards Studio”.

- Create decks; add cards (front/back). Bulk import from CSV (front,back).

- Study mode with spaced repetition (SM-2 simplified or Leitner-like).

- Keyboard shortcuts: space=flip, 1..4=grade recall quality.

- Deck stats (due today, new, mature, ease factor).

- localStorage persistence; export/import deck JSON.

- Focus on accessibility and responsive layout. One file only, no libraries.

**5) Pomodoro + Tasks (auto cycles + stats)**

Produce a SINGLE HTML FILE app (inline CSS + JS) “Pomodoro Focus”.

- Timers: 25/5 default with adjustable durations; auto-start next session toggle.

- Task list with estimates (# of pomodoros), check/complete.

- Session log and daily/weekly stats (completed pomodoros, focus time).

- Desktop notifications and optional ticking sound via Web Audio API (muteable).

- localStorage save; clean, accessible UI with keyboard controls.

- No external libraries or assets; everything self-contained and commented.

**6) Unit & Currency Converter (offline presets)**

Create a SINGLE-FILE HTML app (inline CSS/JS) “Quick Converter”.

- Units: length, mass, temperature, volume, speed. Provide sensible presets.

- Currency: include a static demo table with a few hardcoded rates (e.g., USD, EUR, GBP, JPY) and let users edit rates manually (no network calls).

- Bidirectional conversion with instant update and copy-to-clipboard.

- Keyboard-first UX; responsive two-column layout; accessible labels.

- Persist last-used category and custom rates in localStorage.

- No external libs; add clear comments.

**7) Markdown Notes with Live Preview (autosave + export)**

Build a SINGLE-PAGE HTML app (inline CSS + JS) “Markdown Notes”.

- Split view: editor (textarea) + live preview (support headings, bold/italic, lists, links, code blocks).

- Toolbar buttons for common markdown inserts.

- Auto-save drafts to localStorage; list of notes (create/rename/delete).

- Export note as .md file and as .html (download).

- Minimal, elegant styles; accessible and responsive; no external libraries.

- Heavily comment the code. All in one file.

**8) Kanban Board (columns + WIP limits + JSON import/export)**

Generate a SINGLE HTML FILE (inline CSS + JS) “Mini Kanban”.

- 3 default columns: Backlog, In Progress (WIP limit editable), Done.

- Add cards with title, description, tags; drag-and-drop between columns.

- Filter by tag/search; column counts; simple aging indicator (color if >X days).

- localStorage persistence; export/import board as JSON.

- Mobile-friendly drag; accessible controls; clean, modern UI.

- No external libraries; comment code clearly.

**9) Quiz Builder (question bank + timed quizzes)**

Create a ONE-FILE HTML app (all CSS/JS inline) “Quiz Builder”.

- Build a question bank: MCQ and True/False with correct answer and explanation.

- Create quizzes by selecting questions, set a timer, shuffle options.

- Take quiz: progress bar, per-question feedback, final score + review with explanations.

- Import/export question bank as JSON; localStorage persistence.

- Keyboard navigation; responsive layout; clear styles; no external libs.

- Include thorough code comments.

**10) Image Toolbox (compress/resize/crop in browser)**

Output a SINGLE HTML document (inline CSS + JS) titled “Image Toolbox”.

- Load image via file input or drag-and-drop (no network).

- Tools: resize by width/height, crop selection, and compress (quality slider).

- Show before/after preview and estimated size; download result as JPG/PNG.

- Work entirely client-side using canvas; no external libraries or fonts.

- Accessible controls, responsive layout, helpful inline tips.

- Commented, readable code in one file only.

**Bonus: A generic template you can tweak for anything**

You are to output ONE SINGLE HTML FILE that runs standalone with all CSS and JavaScript inline (no external libraries, fonts, images, or CDN). App name: “{{APP NAME}}”.

Core features:

- {{FEATURE 1}}

- {{FEATURE 2}}

- {{FEATURE 3}}

Quality bar:

- Fully responsive; keyboard accessible; ARIA labels where helpful.

- Persist state with localStorage; include Export/Import JSON if relevant.

- Clean, modern styling; clear component structure; semantic HTML.

- Vanilla JS only; comment the code generously.

Deliverable: a single <html> file with everything embedded.