

Codex Prompt Pack

Prompt patterns for safe Codex builds.

Use these prompt patterns when asking Codex to build pages, tools, imports, or safer WordPress assets.

1. Beginner /goal prompt

/goal Build [specific page/tool/workflow] for [audience]. Inspect the current project first. Reuse existing styles and patterns. Keep the first version narrow, useful, deterministic, and easy to test. Include metadata, FAQ, internal links, copy/download actions if relevant, and a summary of files changed.

2. WordPress tool prompt

/goal Build a WordPress-friendly [tool type] for [page/topic]. Prefer a shortcode, block, child-theme template, or Custom HTML-compatible widget depending on the existing setup. Scope CSS/JS to the academy route. Add labels, validation, copy/download actions, and no personal-data storage unless an approved provider hook exists.

3. Calculator prompt

/goal Build a beginner-safe calculator for [decision]. Inputs: [list]. Formula/logic: [plain-English rules]. Outputs: [number/range/explanation]. Include assumptions, test cases, empty-state validation, copy result, mobile layout, and a disclaimer if the topic is sensitive.

4. Landing page prompt

/goal Build a landing page for [offer]. Audience: [audience]. Promise: [outcome]. Sections: hero, problem, outcome, how it works, proof or example, FAQ, CTA. Reuse existing components and typography. Avoid fake proof, unsupported claims, and invented pricing.

5. Directory prompt

/goal Build a structured directory for [niche]. Data fields: name, category, best for, use case, caution, official URL, last reviewed. Include filters, empty states, editorial notes, current-detail disclaimers, and an update workflow.

6. Dashboard prompt

/goal Build a dashboard mockup for [workflow]. Use fake/sample data first. Include metric definitions, status cards, table, filters, empty states, mobile behavior, and notes about real-data permissions before production.

7. AI agent spec prompt

/goal Draft an AI agent spec for [task]. Include allowed inputs, allowed sources, forbidden actions, approval points, logs, escalation behavior, test cases with fake data, and a clear human handoff before external actions.

8. Review and QA prompt

/goal Review the current implementation for bugs, broken links, layout issues, accessibility gaps, privacy risks, missing metadata, and unsafe assumptions. Prioritize findings by severity, cite files/lines, and recommend scoped fixes.

9. Follow-up improvement prompt

/goal Improve the first version of [project] using feedback: [feedback]. Keep scope small, preserve working behavior, add one useful feature, rerun verification, and summarize what changed.

How to customize these prompts

Replace bracketed placeholders before running the prompt.

Name the exact page, file, route, shortcode, or tool where possible.

Include what must not change: navigation, footer, analytics, production data, parent theme, generated files, or unrelated pages.

Include the verification command when known; if unknown, ask Codex to inspect and identify it before editing.

WordPress/JNews adaptation prompt

/goal Adapt this Academy asset for Kingy AI WordPress/JNews. Inspect the real WordPress root or repo evidence first. Choose plugin shortcode, Custom HTML block, child-theme template, or REST import path based on the inspected setup. Keep assets scoped, avoid parent-theme edits, apply Yoast metadata, preserve rollback, and verify logged-in/logged-out staging views.

Article expansion prompt

/goal Improve this Build With AI Academy article for Kingy AI. Preserve the search intent, add a concrete Kingy AI example build, include one output sample to review, keep claims cautious, link to related Academy assets, and leave publish blockers for editor, Yoast, JNews mobile, and staging proof.

Tool QA prompt

/goal QA this AI-built tool. Test empty inputs, realistic inputs, edge cases, copy/download/reset/saved-plan behavior, mobile layout, keyboard path, localStorage privacy, links, metadata, and rollback. Report failures first with

file or URL references.

Prompt review checklist

Goal is specific.

Audience is named.

Inputs and outputs are concrete.

Safety limits are explicit.

Non-goals are explicit.

Verification is explicit.

Final report format is explicit.

Rollback expectation is explicit.

Kingy AI publishing reminders

Use fake or public-safe data until staging, privacy, and editor review are complete.

Keep the first version useful without accounts, payments, private databases, or autonomous external actions.

Connect the final page back to the Build With AI Academy hub, AI App Builder, Project Library, Templates, and Beginner Safety Rules.

If email capture is involved, consent must be unchecked by default, provider approval must be documented, and unsubscribe/privacy language must be visible.

Evidence to collect before publish

Staging URL:

Desktop screenshot:

Mobile screenshot:

QA owner:

Rollback owner:

Final PDF/media URL:

Yoast title/meta reviewed by:

Done means

A beginner can understand what to do next.

The output can be copied, saved, downloaded, or handed to Codex.

The risky parts are either removed, delayed, or protected by human approval.

The staging page has been tested logged in and logged out inside the real JNews environment.

Kingy AI - Build With AI Academy

Use with the AI App Builder for Beginners and Beginner Safety Rules.