Custom Prompts

Custom prompts let you turn Markdown files into reusable prompts that you can invoke as slash commands in both the Codex CLI and the Codex IDE extension.

Custom prompts require explicit invocation and live in your local Codex home directory (for example, `~/.codex`), so they're not shared through your repository. If you want to share a prompt (or want Codex to implicitly invoke it), use skills.

1. Create the prompts directory:

   ```
   mkdir -p ~/.codex/prompts
   ```

2. Create `~/.codex/prompts/draftpr.md` with reusable guidance:

   ```
   ---
   description: Prep a branch, commit, and open a draft PR
   argument-hint: [FILES=<paths>] [PR_TITLE="<title>"]
   ---

   Create a branch named `dev/<feature_name>` for this work.
   If files are specified, stage them first: $FILES.
   Commit the staged changes with a clear message.
   Open a draft PR on the same branch. Use $PR_TITLE when supplied;
           otherwise write a concise summary yourself.
   ```

3. Restart Codex so it loads the new prompt (restart your CLI session, and reload the IDE extension if you are using it).

Expected: Typing `/prompts:draftpr` in the slash command menu shows your custom command with the description from the front matter and hints that files and a PR title are optional.

Add metadata and arguments

Codex reads prompt metadata and resolves placeholders the next time the session starts.

- **Description:** Shown under the command name in the popup. Set it in YAML front matter as `description:`.

- **Argument hint:** Document expected parameters with `argument-hint: KEY=<value>`.
- **Positional placeholders:** `$1` through `$9` expand from space-separated arguments you provide after the command. `$ARGUMENTS` includes them all.
- **Named placeholders:** Use uppercase names like `$FILE` or `$TICKET_ID` and supply values as `KEY=value`. Quote values with spaces (for example, `FOCUS="loading state"`).
- **Literal dollar signs:** Write `$$` to emit a single `$` in the expanded prompt.

After editing prompt files, restart Codex or open a new chat so the updates load. Codex ignores non-Markdown files in the prompts directory.

Invoke and manage custom commands

1. In Codex (CLI or IDE extension), type `/` to open the slash command menu.

2. Enter `prompts:` or the prompt name, for example `/prompts:draftpr`.

3. Supply required arguments:

   ```
   /prompts:draftpr FILES="src/pages/index.astro src/lib/api.ts"
   PR_TITLE="Add hero animation"
   ```

4. Press Enter to send the expanded instructions (skip either argument when you don’t need it).

Expected: Codex expands the content of `draftpr.md`, replacing placeholders with the arguments you supplied, then sends the result as a message.

Manage prompts by editing or deleting files under `~/.codex/prompts/`. Codex scans only the top-level Markdown files in that folder, so place each custom prompt directly under `~/.codex/prompts/` rather than in subdirectories.