
BUILD YOUR OWN · SETUP GUIDE

Roux

A meal planner that knows your family by heart.

You've been holding your household's food rules, preferences, and weekly schedule entirely inside your own head. Every single week.

This is the guide to stop doing that. Five prompts. One conversation with Claude Code Desktop. You answer questions — Claude builds your personal meal planning assistant, configured completely for your family.

-
- 01 Tell Claude about your family
 - 02 Claude writes your family.md
 - 03 Capture your pantry → inventory.md
 - 04 Set up your stores → deals template
 - 05 Claude becomes Roux → CLAUDE.md + commands

A free kit from
Melsplains
melaniedunn.work/melsplains/roux

BEFORE YOU START

Download **Claude Code Desktop** (free at claude.ai/download — Mac or Windows, no terminal needed). Create a folder called **Roux** anywhere on your computer and open it in Claude Code. Start a new chat. Then paste these prompts in order.

PROMPT 1 OF 5**Tell Claude about your family**

I'm setting up Roux, my personal AI meal planning assistant. Let's start with my family profile.

Ask me everything you need to know about my household so you can plan meals we'll actually eat. Cover:

- Who's in my family and their ages
- Dietary restrictions and hard allergies
- What each person loves and flat-out refuses to eat (be specific – "no cilantro" is more useful than "picky eater")
- My weekly grocery budget
- Which stores I shop at and which is primary
- When I usually shop each week
- Any weekly dinner traditions (taco night, pizza Friday, etc.)
- Nights when dinner needs to be fast or in the crockpot
- Cooking equipment I have (crockpot, Instant Pot, air fryer, grill, etc.)

Ask everything at once – I'll fill it all in.

Answer honestly and specifically. The more detail you give here, the better every future meal plan will be.

PROMPT 2 OF 5**Claude writes your family file**

Based on everything I just told you, write my family.md file and save it to this folder.

Include:

- Household overview (members, budget, stores)
- Hard rules table – the absolute non-negotiables
- A section per person: favorites, dislikes, meal patterns
- Dinner planning logic: protein rotation, easy night options, sides pairings
- Seasonal notes if relevant

Be thorough. This is the file Roux reads every time it plans a meal.

What happens: Claude creates family.md in your Roux folder. You don't have to write or edit anything manually.

PROMPT 3 OF 5

Capture your pantry

Now let's build my inventory so Roux knows what I already have before it plans or shops.

Ask me what's in my freezer, fridge, and pantry. As I answer:

- Note quantity or status where I give it (LOW, 1 pack, plenty)
- Anything I say I "always have" or "always keep stocked" goes in a Staples section – we won't track those weekly

When I'm done, write everything to inventory.md organized into: Freezer, Fridge, Pantry, Staples.

What happens: Claude asks questions, you answer, it writes inventory.md. This is how /shop knows what you already have and doesn't send you to buy things you have.

PROMPT 4 OF 5

Set up your stores — fill in the brackets before pasting

Let's set up my stores and create a weekly deals template.

My stores: *[e.g. Kroger, Meijer, Trader Joe's]*

Primary store: *[name]*

Their weekly deals reset on: *[day — usually Wednesday or Sunday]*

I usually shop on: *[day]*

Create stores/this_week.md as a template I'll fill in each week with sales.

Format: a simple table — Item | Sale Price | Store | Notes — with a couple example rows so I know how to use it.

This is the only prompt you customize before pasting. Fill in the teal text with your actual stores and schedule, then copy the whole thing.



One more prompt to go. Prompt 5 is the one that makes it all work together — it synthesizes everything you've told Claude into CLAUDE.md, the file that turns Claude into Roux.

PROMPT 5 OF 5

Claude becomes Roux

Now write my CLAUDE.md – the file that turns Claude into Roux, my personal meal planning assistant.

Using everything in family.md, inventory.md, and my store setup, define:

1. Roux's persona – efficient, organized, thinks like an executive assistant who knows food and family logistics. Makes me feel like I have help, not homework.
2. Core files table – what each file is and when to load it (only load what each command actually needs)
3. The commands: /plan, /shop, /deals, /inventory-add, /inventory-remove, /recipe-used [name], /week-used – write each as its own file in a .claude/commands/ folder (plan.md, shop.md...). That folder is what makes typing /plan actually work. In CLAUDE.md, just list them in a short table.
4. My family rules baked in as non-negotiable behaviors – not "see family.md," write them out so they're enforced every single time
5. /plan output format: lead with the complete meal list so I can scan the whole week in 5 seconds, then details below
6. Tone: efficient, no fluff, surface what matters

Save CLAUDE.md to this folder and the seven command files to .claude/commands/.

Using AnyList? Add before the final "Save" line: *"Include AnyList MCP integration via the mealplanner server. On /plan: pull my recipe library and check the meal plan calendar to avoid repeats. On /shop: push each dinner to the AnyList calendar and add ingredients to the correct store shopping list."*

You're live.

Start a new chat in Claude Code (Roux folder still open). Type your first command:

/plan

Tell Claude your week's schedule and weather. Watch it work.

YOUR WEEKLY RHYTHM

MON-TUE /deals Paste this week's store ad — Roux pulls what's worth buying	WED-THU /plan 7-dinner plan built around your family, pantry, and deals	SUNDAY /shop Grocery list organized by store section	AFTER SHOP /inventory-add Paste your receipt — inventory updates itself	FRI-SAT /week-used Clear used ingredients, start fresh next week
---	--	---	--	---

Be specific in Prompt 1.

"Picky eater" is useless. "Won't eat onions unless caramelized and not identifiable" is useful. Specificity is what makes the plans actually work for your family.

Run `/inventory-add` after every big shop.

If `inventory.md` gets stale, your grocery lists will include things you already have. One quick paste of your receipt keeps it accurate.

The files are just text — change them anytime.

New rule? Edit `family.md`. New store? Update `CLAUDE.md`. Roux adapts the moment you save a change.

The first week feels like magic.

The second week feels like routine. That's the goal — not a novelty, a system.

Melsplains

`melaniedunn.work/melsplains/roux`
`github.com/melsplains/roux`
Free forever · Your data stays local