


# Building Custom GPTs & AI Assistants


Turn one good prompt into a reusable bot — on ChatGPT or Gemini.

---

 **This workbook belongs to** \_\_\_\_\_

 **Bring** — one real, repetitive task you'd hand to a bot

 **You'll need** — ChatGPT (free is fine to write the prompt), plus paid ChatGPT or Gemini to build

 **You'll leave with** — a published bot and a reusable markdown prompt template

START HERE

# Before You Build Anything: Have a Problem

A Custom GPT or a Gemini Gem is just a saved, specialised assistant. It is only worth building if it removes a **real, repetitive task**. Don't build a bot for the sake of a bot — start from something you already do over and over and wish you could hand off.

In this workbook we build one concrete example end to end: a bot that **writes excellent emails**. The exact same process — same structure, same steps — builds a customer-support bot, a sales bot, a copywriting bot, a research bot, or anything else. Swap the idea; keep the method.

WORKSHEET 0 — YOUR BOT IDEA

## Define the task before you touch any tool

The repetitive task my bot will own: \_\_\_\_\_

Who uses it (me / team / customers): \_\_\_\_\_

A good output looks like: \_\_\_\_\_

Knowledge it needs (docs, examples) — if any: \_\_\_\_\_

*Example used in this workbook: "A bot that turns my rough notes into send-ready emails."*

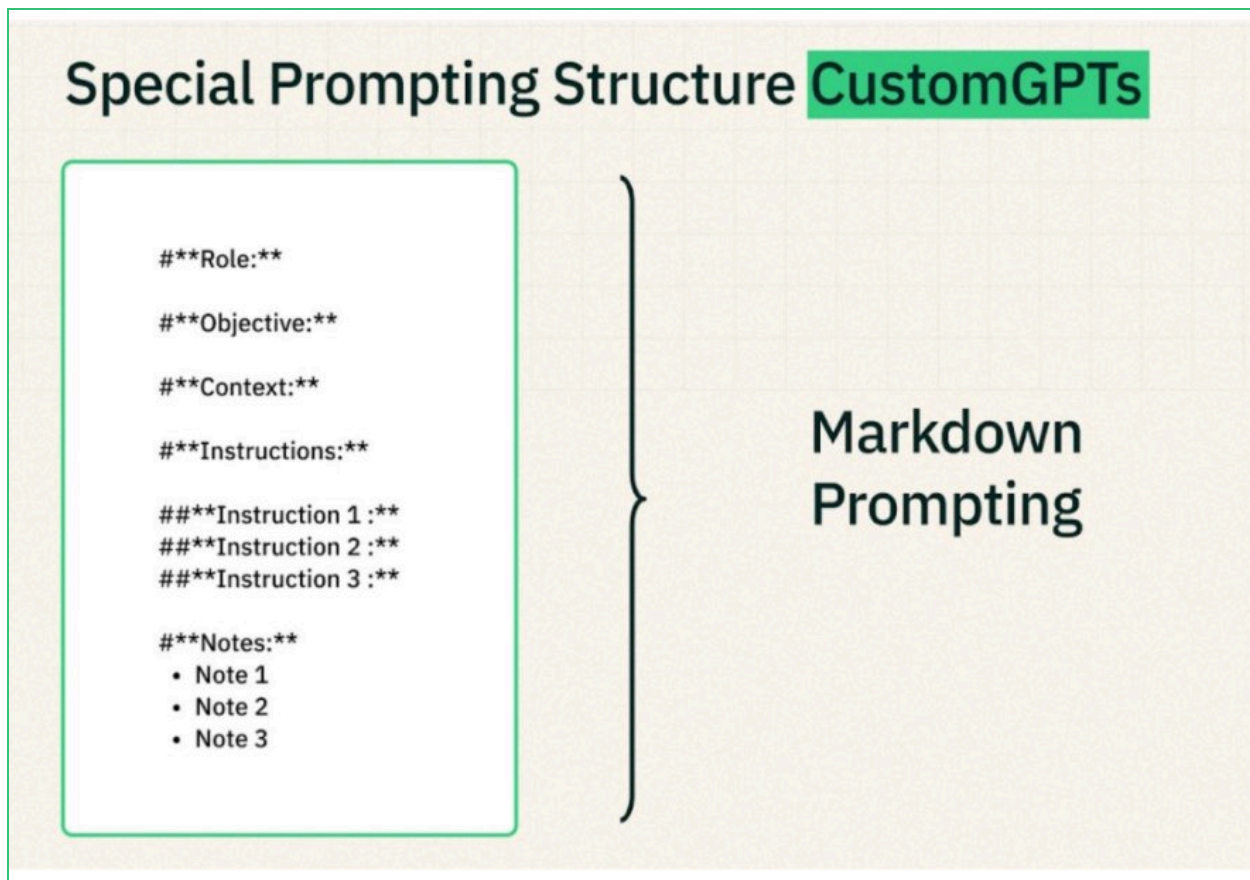
## The one idea behind today

A bot is only as good as its instructions. The whole skill is writing **structured** instructions — and you won't even write them by hand. You'll have ChatGPT write them for you, in a fixed markdown structure, then paste them into the builder.

PART 1 · THE CORE SKILL

# The Markdown Prompting Structure

Every reliable Custom GPT and Gem is built on the same skeleton. Markdown headings give the model a clear, parseable hierarchy — it knows *who it is*, *what it must achieve*, *what to assume*, *exactly what to do*, and *what never to do*. Loose paragraphs drift; this structure holds.



*The structure to memorise – Role, Objective, Context, Instructions, Notes.*

Reproduce it exactly. This is the blank template — you'll fill it (or have ChatGPT fill it) for every bot you ever build:

**THE BLANK STRUCTURE — COPY THIS**

```
# **Role:**

# **Objective:**

# **Context:**

# **Instructions:**

### **Instruction 1:**
### **Instruction 2:**
### **Instruction 3:**

# **Notes:**
- Note 1
- Note 2
- Note 3
```

Section	What goes in it
<b>Role</b>	Who the bot is — its identity, expertise, and voice
<b>Objective</b>	The single outcome it must deliver every time
<b>Context</b>	Background it should always assume (audience, defaults, tone)
<b>Instructions</b>	The concrete, ordered steps and rules it follows
<b>Notes</b>	Guardrails and edge cases — what it must never do

**The structure, filled in for our email bot**

This is what “production-ready” looks like — specific, not generic. Read it as the target quality bar for your own bot.

**WORKED EXAMPLE — THE EMAIL-WRITING BOT**

```
# **Role:**
You are an expert executive communications writer. You write clear,
warm, and persuasive emails in the user's own voice.

# **Objective:**
Turn the user's rough intent into a send-ready email that gets a reply
and protects the relationship.
```

```
# **Context:**  
The user is busy and often gives only a few words about what they want.  
Assume a professional but human tone unless told otherwise. Default to  
concise. Most emails are work emails.  
  
# **Instructions:**  
  
### **Instruction 1:**  
If the request is unclear, ask up to 3 quick questions first  
(recipient, the goal of the email, desired tone). Otherwise proceed.  
  
### **Instruction 2:**  
Draft the email with: a specific subject line, a one-line opener,  
the core message in 2-4 short paragraphs, and one clear call to action.  
  
### **Instruction 3:**  
After the draft, offer 2 alternative versions in different tones  
(e.g. warmer, more direct) as short labelled options.  
  
# **Notes:**  
- Never invent facts, names, numbers, or commitments.  
- Match the user's language and mirror any sample they paste.  
- Keep most emails under 150 words unless asked otherwise.  
- No corporate jargon or filler. Plain, confident language.
```

---

## PART 2 · THE PRECURSOR

# Get Your Prompt From ChatGPT First

Do not write the markdown prompt by hand. The reliable move is to make ChatGPT write it for you: explain your use case, hand it the structure, and ask for the whole thing in one code block so it's clean to copy.

### PASTE THIS INTO A NORMAL CHATGPT CHAT

I want to build a Custom GPT / Gemini Gem that writes excellent emails for me. (Replace this line with your own use case from Worksheet 0.)

Write the system prompt for it using EXACTLY this markdown structure:

```
# **Role:**  
# **Objective:**  
# **Context:**  
# **Instructions:**
```

```
### **Instruction 1:**  
### **Instruction 2:**  
### **Instruction 3:**  
# **Notes:**  
- Note 1  
- Note 2  
- Note 3
```

Make it specific, production-ready, and opinionated. Fill every section. Output the entire prompt inside ONE code block so I can copy it cleanly.

### DO THIS NOW

- Open ChatGPT, paste the prompt above, and replace the use-case line with your own.
- Read what it generates. If a section is vague, reply: “make Instruction 2 and the Notes more specific” and regenerate.
- Copy the final prompt from inside its code block — you'll paste it into the builder next.

### WORKSHEET 1 — SANITY-CHECK THE GENERATED PROMPT

#### Tick before you build — a weak prompt makes a weak bot

- Role names a clear identity, expertise, *and* a voice
- Objective is one outcome, not a list of vague hopes
- Instructions are concrete steps you could follow yourself
- Notes contain real guardrails (“never...”), not filler

### WHY THIS ORDER

*Building the bot before the prompt is decorating a room before you've decided what it's for. The prompt is the brief; the builder is just where you hang it.*

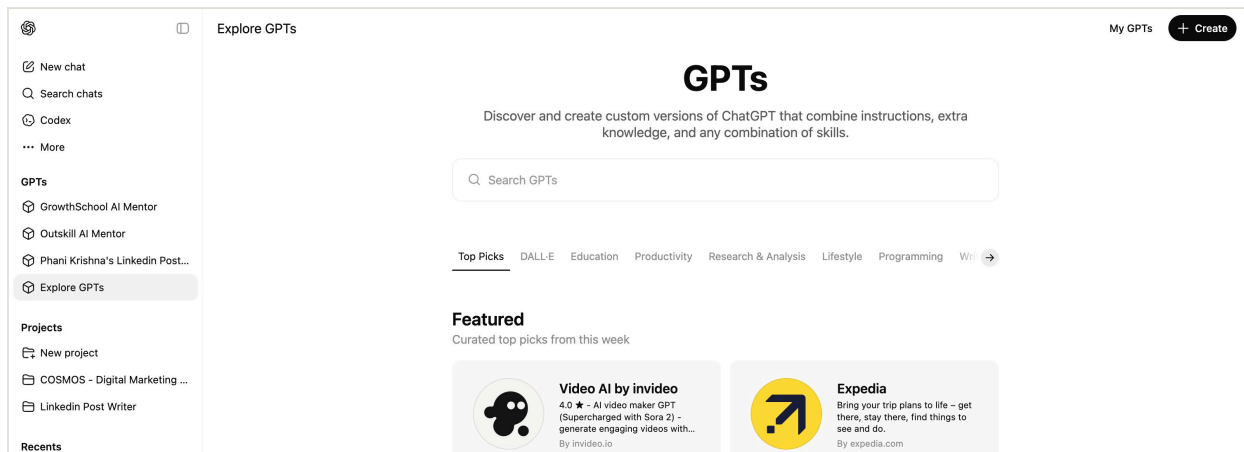
## PART 3 · LAB A

# Build a Custom GPT (ChatGPT)

Custom GPTs live inside ChatGPT and require a **paid plan**. If you're on the free plan, skip to Lab B (Gemini Gem) — the prompt you just created works there too, for free.

## Steps 1–2 – Open GPTs and create

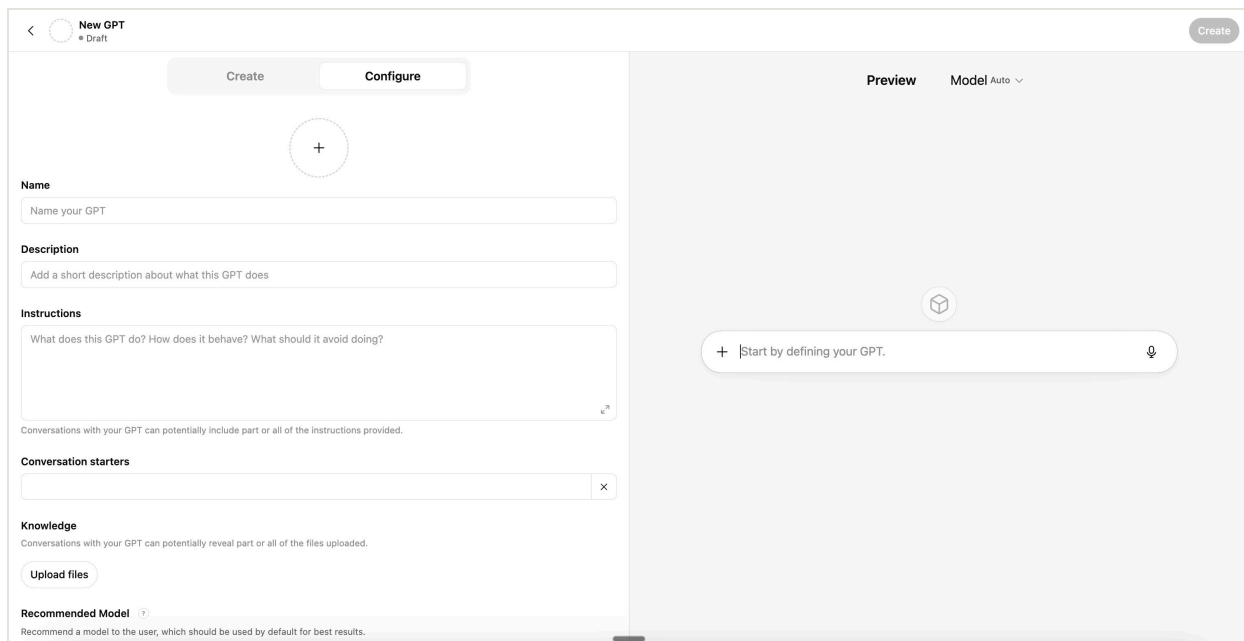
- In the ChatGPT sidebar, open **Explore GPTs** (the GPTs section).
- Click **+ Create** in the top-right.



*Explore GPTs → the + Create button is top-right; “My GPTs” is where yours will live.*

## Steps 3–4 – Configure your GPT

You'll land in the Custom GPT builder. Switch to the **Configure** tab and fill it in:



*The Configure tab — Name, Description, Instructions, Conversation starters, and Knowledge.*

- Name** — what users will see (e.g. “Email Writer”).
- Description** — one line on what it does.
- Instructions** — paste your full markdown prompt here. This is the brain of the GPT.

**Knowledge** — if you have a knowledge base (sample emails, style guide, FAQs), click **Upload files** and add them.

**Conversation starters** — add 2–4 example prompts so users know how to begin.

### Step 5 — Test & iterate

Use the **Preview** pane on the right. Give it real tasks, not toy ones. When something's off, fix the matching section of your markdown prompt (usually an Instruction or a Note) — don't rewrite randomly.

### Step 6 — Publish

Click **Create / Publish** (top-right) and choose who can access it — only you, anyone with the link, or everyone. Your GPT now lives under **My GPTs**.

#### WORKSHEET 2 — YOUR CUSTOM GPT CONFIG

**Write it here first, then paste into the builder**

**Name:** \_\_\_\_\_

**Description:** \_\_\_\_\_

**Knowledge files I'll attach:** \_\_\_\_\_

Three conversation starters:

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

#### PART 4 · LAB B

## Build a Gemini Gem (Free)

A Gem is Gemini's version of a Custom GPT, and it's **free**. Best part: the markdown prompt you generated works here unchanged. Build once, deploy on whichever platform you have.

#### DO THIS NOW

**Step 1.** Go to Gemini ([gemini.google.com](https://gemini.google.com)).

**Step 2.** Open **Gems** (the Gem manager).

- Step 3.** Click **Create** (new Gem).
- Step 4.** Paste your markdown prompt from ChatGPT into the Gem's instructions. Add a name and, if you have one, knowledge files.
- Step 5.** Test & iterate using the live preview — same loop as the GPT: fix the prompt section, not random words.
- Step 6.** Click **Save / Publish**. Your Gem is ready to use and share.

### One prompt, both platforms

Because the instruction is just structured markdown, it is portable. The same template that powered your GPT powers your Gem — and your next support, sales, or research bot. You are not learning a tool; you're learning a structure.

#### WORKSHEET 3 — YOUR GEM

### Quick capture

**Gem name:** \_\_\_\_\_

**Which platform I published on:** \_\_\_\_\_

#### PART 5 · MAKE IT GOOD

## Test & Iterate Like a Pro

The first version is never the final one. Good bots are tuned, not authored. Test with realistic inputs and change the prompt *surgically*.

- Give it **three real inputs** of different difficulty — an easy one, a vague one, and an edge case.
- Judge three things: **format** (did it follow the structure?), **tone** (does it sound right?), and **safety** (did it invent anything?).
- When it fails, trace the failure to **one section** of the markdown prompt and edit only that — usually an Instruction (behaviour) or a Note (guardrail).

**WORKSHEET 4 – ITERATION LOG****Run three tests; record what you changed**

TEST INPUT	WHAT WAS WRONG	PROMPT SECTION I CHANGED
_____	_____	_____
_____	_____	_____
_____	_____	_____

*If you can't name the section that caused a failure, your prompt isn't structured enough yet.*

**PART 6 · REUSE IT**

# One Template, Many Bots

You now hold a reusable asset. To build any other assistant, keep the structure and swap the top three sections — Role, Objective, Context. The Instructions and Notes adapt from there.

Bot you want	Role becomes	Objective becomes
<b>Support bot</b>	A calm, accurate support specialist	Resolve the issue or escalate cleanly
<b>Sales bot</b>	A consultative sales rep	Qualify and move the deal forward
<b>Copywriting bot</b>	A punchy brand copywriter	On-brand copy that converts
<b>Research bot</b>	A rigorous research analyst	A sourced, structured briefing

## WORKSHEET 5 — YOUR NEXT BOT

### Plan the one you'll build this week

**Bot:** \_\_\_\_\_

**Role:** \_\_\_\_\_

**Objective:** \_\_\_\_\_

**Build by (date):** \_\_\_\_\_

**Signed:** \_\_\_\_\_

## Today's checklist

- ✓ Defined a real problem before opening any tool (Worksheet 0)
- ✓ Learned the markdown structure: Role, Objective, Context, Instructions, Notes
- ✓ Had ChatGPT generate a production-ready prompt in a code block
- ✓ Built and published a Custom GPT or a Gemini Gem
- ✓ Tested with real inputs and iterated the prompt surgically
- ✓ Planned my next bot from the same template

## KEEP THIS

# Quick Reference Card

## The markdown structure

```
# **Role:**          -> who the bot is + its voice
# **Objective:**     -> the one outcome, every time
# **Context:**       -> defaults & audience to assume
# **Instructions:**
### **Instruction 1/2/3:** -> concrete ordered steps
# **Notes:**
- guardrails: what it must NEVER do
```

## The build loop

### Custom GPT

CHATGPT · PAID

1. Explore GPTs
2. + Create → Configure
3. Name, Description, Instructions
4. Upload knowledge (optional)
5. Test in Preview & iterate
6. Publish

### Gemini Gem

GEMINI · FREE

1. Open Gemini
2. Gems → Create
3. Paste the same prompt
4. Add knowledge (optional)
5. Test & iterate
6. Save / Publish

---

*The whole workbook in one line: a great bot is a great prompt with a saved home. Get the prompt right — structured, specific, guarded — and the platform barely matters.*