

Real-world examples

Reading time: ~15 min · Prereqs: Lessons 01–06 · 3 interactive demos inside

Why concrete examples beat abstract advice

You've learned what AI is, how to access it, and how to prompt it well. Now the question everyone eventually asks is: *"But what do I actually use it for?"*

This lesson is the answer. Not a list of categories, but real, specific, copy-and-use examples across five domains: everyday life, work, creative, learning, and coding. By the end, you'll have a mental library of prompts you can reach for whenever you need them.

The common thread across all five categories is the same: **AI is a force multiplier for tasks you could do yourself, but would rather do faster — especially first drafts and "getting started."** Understanding this frame prevents both over-reliance (treating AI as an oracle) and under-use (never trying it because you're not sure if it "counts" as a valid use case).

How to use this page

Read through once to build the mental map. Then use the interactive Explorer below as a personal prompt library you can return to whenever you need a starting point for a new task. The Before/After gallery shows you what the difference between raw notes and polished AI output looks like in practice. The Pattern Quiz at the end tests whether you've internalized the categories.

Category 1 — Everyday life

The most underestimated use case. The tasks we do every day — writing emails, reading documents, making plans, understanding confusing things — consume real

time and mental energy. None of them require AI to be "smart" in any deep sense. They require drafting speed, patience, and the ability to explain things clearly. AI is excellent at all three.

Turning messy thoughts into clean emails

Most people write emails in three stages: first, they get the thoughts out in a messy, informal way; then they rewrite it into something readable; then they second-guess the tone and rewrite again. AI collapses this into one step. You dump your thoughts (in any order, in any style), describe who you're writing to and what you want them to feel, and ask for a draft. You read it, adjust anything that doesn't sound like you, and send.

This works for difficult emails especially well. Chasing an overdue invoice, responding to a complaint, telling a colleague something awkward — the hard part of these emails isn't the writing, it's navigating the tone. You can tell AI "be firm but not aggressive" or "be warm but not apologetic" and it will thread that needle better than most people do under pressure.

Reading without reading

Rental agreements, insurance policies, terms of service, long articles — everyone has a pile of documents they know they should read but haven't. Paste the text into an AI chat (or a big chunk of it) and ask: "What are the five things I need to know? Flag anything unusual or that I should be concerned about." You'll get a genuine summary in under a minute.

Important caveat: for anything legally or financially consequential, the AI summary gets you oriented, not off the hook. Use it to know *what to read carefully* in the original document, not as a replacement for reading it.

Explaining things at exactly the level you need

AI is uniquely patient about re-explaining things in different ways. You can ask for an explanation "like I'm 10 years old," then follow up with "now go deeper" or "explain the technical side." You can ask it to use a specific analogy ("explain this the way you'd explain it to someone who knows cooking but not physics"). This kind of adaptive explanation is something even excellent teachers struggle to do consistently under time pressure — AI does it every time, at no extra cost.

Category 2 — Work

Knowledge workers spend a large fraction of their day on writing, meetings, and communication. These activities are not the core of most jobs — they're the scaffolding around it. AI handles scaffolding well. Here are the most valuable work use cases.

From meeting notes to action items

Every meeting produces some notes. Often they're patchy, unformatted, and scattered across someone's laptop. Paste them into an AI and ask: "Extract the action items. For each, note who owns it and the deadline. If an owner or deadline isn't mentioned, flag it as unassigned." What comes back is a structured task list you can copy into your project management tool immediately.

You can also ask: "What decisions were made?" and "What questions were left open?" These three prompts together turn raw notes into a complete meeting record in about 90 seconds.

Drafting reports, proposals, and job descriptions

The blank page is the hardest part of any writing task. With AI, you never face it. Describe what you need (the document type, the audience, the tone, the key points to cover) and ask for a first draft. The draft will almost certainly need editing — it won't know your company's style, your specific numbers, or the political context of the proposal. But it gives you a structured starting point to react to, which is much faster than starting from scratch.

This is especially useful for document types you rarely write. If you have to write a job description for the first time, or a board-level summary, or a vendor RFP, you may not know the genre conventions. AI does. It will produce a structurally sound template that you fill in with real content.

Analysing data in plain English

You don't have to be a data analyst to ask intelligent questions about a dataset. Paste a table of numbers (copied from a spreadsheet) and ask: "What are the three biggest trends in this data? Which category is growing fastest? Is anything declining?" AI will interpret the numbers and surface patterns that would take a non-analyst much longer to find manually.

This works best with small-to-medium tables (up to a few hundred rows). For larger datasets, you'll need a code-based approach, but for everyday business data — sales figures, survey results, team metrics — conversational data analysis is remarkably effective.

Category 3 — Creative

AI's creative applications get misunderstood in two opposite ways: some people think it's too good ("it's replacing creatives") and others think it's useless ("the output has no soul"). The truth is it's neither. AI is an excellent *collaborator* for creative work, specifically for the phases of the process where you need quantity before quality.

Brainstorming at scale

Good brainstorming requires generating a lot of options without judging too early. Most people are bad at this alone — they generate three ideas, get stuck, and pick the least-bad one. With AI, you can generate 15, 20, or 30 options in a minute. Your job becomes curation, not generation.

The trick is to give constraints. Not "give me name ideas" but "give me 15 name ideas for a dog-walking app, short and memorable, playful tone, avoid anything with 'paw' or 'wag' since those are taken, mix of single words and two-word phrases." Tight constraints produce better brainstormers. If you like #4 and #11, say so and ask for 10 more in that direction.

Getting unstuck on writing

Every writer knows the feeling: you need a paragraph or a page, you know roughly what it should say, but you can't find the words. AI is genuinely useful here — not to write the thing for you, but to give you options to react to. Ask for three very different versions (lyrical, direct, anecdote-led) and see which one sparks something. You'll often end up discarding all three but knowing clearly what you actually want to write. That's the value: the contrast, not the output.

Editing and tone-shifting

You wrote something. It's okay, but it's too long, or too formal, or too casual. AI is a patient editor. "Make this punchier, cut it by 30%" or "rewrite this in a warmer, more conversational tone" or "make this suitable for a technical audience who already know

the domain" — these instructions produce reliable, fast results. You can chain edits: take the AI's version, ask for another pass, and so on until it's close to what you want.

Category 4 — Learning

This is perhaps the most powerful use case for anyone who is serious about learning a new skill or subject. AI removes the two biggest barriers to self-directed learning: access to a patient expert, and the ability to practice interactively.

The adaptive tutor

"You are a patient tutor helping me learn basic economics. I'm a complete beginner. Start with supply and demand — explain it simply, check that I understand, then move on when I say I'm ready." This prompt sets up an interaction that continues at your pace, adapts to what you already know, and never loses patience. You can stay on supply and demand for as long as you need before moving on. No textbook does this.

The key is to push back when something doesn't land: "I still don't understand the bit about equilibrium price — can you try a different analogy?" A human tutor might run out of ways to explain something. AI has essentially unlimited alternative explanations.

Active recall through quizzing

Reading is passive. Being tested is active. Decades of educational psychology research confirm that active recall — trying to retrieve information from memory — is one of the most effective ways to learn. AI makes it trivially easy to quiz yourself on anything. "Ask me 10 multiple-choice questions on the French Revolution. Give me one at a time. After I answer, tell me if I'm right and explain why. Keep track of my score."

This works for any subject: history, science, a programming language, a foreign language, or material from a course you're enrolled in. You can even paste in the content you want to be quizzed on: "Based on this chapter summary, create 8 short-answer questions for me to practice."

Getting feedback on your work

Writing an essay, drafting a business plan, reasoning through a problem — you can get structured feedback from AI on any of these. "Here is my essay draft. Give me honest feedback on the argument, the structure, and the clarity. Don't rewrite it — just tell me

where it's strong and where it's weak." This kind of feedback loop, which normally requires access to a mentor or writing group, is available any time for free.

Category 5 — Coding (even for near-beginners)

You do not have to be a developer to benefit from AI coding help. Some of the most valuable use cases are for people who know a little code — enough to understand what a script does and modify it, but not enough to write it from scratch. Here's what that looks like in practice.

Writing small scripts for repetitive tasks

The sweet spot for near-beginner coding with AI is small, well-defined tasks: rename all the files in a folder, combine several CSV files into one, send a formatted email from a spreadsheet, pull data from a website into a table. These tasks take an experienced developer about 20 minutes to write. They take a non-developer potentially never, because the blank page of a code file is impenetrable without a starting point.

With AI: describe the task in plain English, specify your operating system and any constraints, and ask it to explain each line. You get a working script and you understand what it does. If it doesn't work the first time (this happens), paste the error message back in and ask "what's wrong?"

Explaining code you didn't write

You inherited a script. Or you found something on the internet that almost does what you need. Or you wrote something yourself six months ago and can't remember how it works. Paste it into AI and ask: "Explain what this code does. Go line by line for anything complicated. Assume I know Python basics but not advanced concepts." What comes back is a plain-English walkthrough of every significant part of the code.

Debugging

Error messages are designed for compilers, not humans. They're often cryptic, referencing line numbers and jargon that means nothing without context. AI translates them. Paste the error message, paste the relevant code, and ask: "What's causing this error and how do I fix it?" You'll get both the fix and an explanation of *why* it was broken — which means you'll understand the concept for next time.

INTERACTIVE 1 OF 3 — USE-CASE EXPLORER

Browse real prompts by category. Click any "Copy this prompt" button to paste it directly into an AI chat.



Everyday



Work



Creative



Learning



Coding

Email from bullet points

PROMPT TYPE: DRAFT FROM NOTES

Turn my notes into a polite but firm email to my landlord. Keep it under 120 words. My notes: roof still leaking after 3 weeks, last repair didn't hold, want it fixed by Friday, don't want to sound aggressive but need to be clear.



Copy this prompt

Document summary

PROMPT TYPE: SUMMARISE LONG TEXT

Here is a rental contract. Give me the 5 most important things I need to know as a tenant, in plain English. Flag anything unusual or that I should be concerned about. [paste document]



Copy this prompt

Explain it simply

PROMPT TYPE: ADAPTIVE EXPLANATION

Explain compound interest like I'm 10 years old, using a simple analogy. Then give me a quick worked example with real numbers.



Copy this prompt

Trip planner

PROMPT TYPE: PLAN WITH CONSTRAINTS

Plan a 5-day trip to Lisbon for 2 adults in late October. Budget: around €1500 total excluding flights. We like food, walking, and history. We hate tourist traps. Give a day-by-day outline with 2-3 activities per day.



Copy this prompt

Meal plan for the week

PROMPT TYPE: PLANNING WITH CONSTRAINTS

Difficult conversation script

PROMPT TYPE: PREPARE FOR HARD TALK

Create a 5-day meal plan for 2 people. Constraints: one person is lactose intolerant, we have 30 minutes max on weeknights, budget is €60 for the week's groceries. Include a shopping list.

 Copy this prompt

Help me prepare for a difficult conversation. I need to tell a friend that their constant lateness is affecting our friendship. I want to be honest but not hurtful. Give me an opening line and 3 key points to make clearly.

 Copy this prompt

All prompts are generic starting points. Replace the bracketed placeholders with your real content before pasting into an AI chat.

Why the "before" matters as much as the "after"

One thing the prompts above share: they all start with something real that you already have. Meeting notes, raw thoughts, a block of code, a chapter of a textbook. This is intentional. AI doesn't create from nothing — it transforms what you give it. The better and more specific your input, the more useful the output.

The before/after gallery below shows exactly what this looks like in practice: messy raw input goes in, and polished, usable output comes out. More importantly, it shows the prompt that made the transformation happen.

INTERACTIVE 2 OF 3 — BEFORE / AFTER GALLERY

Three real transformations. Click "Show prompt used" to see exactly what was sent to the AI.

EXAMPLE 1 — MESSY NOTES → PROFESSIONAL EMAIL

BEFORE (RAW NOTES)

need to tell my manager i cant make the monday meeting. dentist appt i forgot to put in calendar. also need to ask if we can move it to tuesday afternoon or i can just get the notes. dont want to seem unprofessional. we

AFTER (POLISHED EMAIL)

Subject: Monday meeting — scheduling conflict

Hi Sarah,

I have a dentist appointment on Monday that conflicts with our 10 a.m.

have a big presentation thursday so this is kind of important timing.

meeting — I apologise for the late notice.

Would it be possible to move the meeting to Tuesday afternoon? If that doesn't work for the team, I'm happy to catch up from the notes instead. With the Thursday presentation coming up, I want to make sure I don't miss anything important.

Let me know what works best.

Best,
Alex

 Show prompt used

EXAMPLE 2 — RAW MEETING NOTES → ACTION LIST

BEFORE (RAW MEETING NOTES)

discussed new landing page - tom said he'd get designs by eow. budget still unclear. marketing wants analytics set up before launch. dev team said 2 weeks for build if designs come on time. need to confirm launch date - maybe 15th? sarah to check with legal about the disclaimer copy. follow up on the email list migration - that was supposed to be done last week.

 Show prompt used

AFTER (STRUCTURED ACTION LIST)

Action items

-  Tom — Landing page designs by end of week
-  Sarah — Check with legal on disclaimer copy (no deadline set)
-  Dev team — 2-week build once designs received
-  [Unassigned] — Confirm launch date (possibly 15th)
-  [Unassigned] — Follow up on email list migration (overdue)

Open questions: Budget not yet confirmed. Analytics setup needed before launch — who owns this?

EXAMPLE 3 — TECHNICAL EXPLANATION → PLAIN ENGLISH FOR CUSTOMERS

BEFORE (INTERNAL TECHNICAL DOC)

The new webhook integration enables real-time event-driven architecture by pushing POST requests to a configured endpoint whenever state transitions occur in the pipeline. Supports HMAC-SHA256 signature verification for security. Retry logic uses exponential backoff up to 5 attempts. Max payload size is 256 KB. Latency is typically under 200ms.

AFTER (CUSTOMER-FACING SUMMARY)

What's new: instant updates to your systems

You can now connect Basil to your own tools so they update automatically the moment something changes — no more manual exports or waiting for a daily sync.

It's fast (usually under a second), secure, and reliable. If your system is temporarily offline, Basil will keep retrying automatically for you.

Ask your developer to set it up — it takes about 15 minutes.

 Show prompt used

Notice that in every case, the raw input was imperfect — and that was fine. AI works with what you give it.

Recognising which category a task belongs to

After working through enough examples, you start to develop a mental reflex: you see a task and immediately know the shape of the prompt that will handle it. Before that reflex kicks in naturally, it helps to be deliberate about categorisation.

Each of the five categories has a characteristic "prompt shape":

- **Everyday life** tasks tend to involve transforming something messy into something polished, or explaining something to yourself. Shape: "take this [messy input] and give me [clean output]."
- **Work** tasks tend to involve structuring information, drafting documents, or extracting insights. Shape: "here is raw material, turn it into [structured output] for [audience]."

- **Creative** tasks tend to involve generating options to react to. Shape: "give me N options for [creative thing], I'll choose and iterate."
- **Learning** tasks tend to involve dialogue: explanation, testing, feedback. Shape: "explain [thing] to me at [level], then check my understanding."
- **Coding** tasks tend to involve writing, explaining, or fixing code with a clear success condition. Shape: "write/explain/fix [code thing], explain why."

INTERACTIVE 3 OF 3 — PATTERN RECOGNIZER

Five task descriptions. Click the category each one best belongs to. Track your score.

Question 1 of 5

"I have a Python script that imports data from a CSV, but it keeps crashing with a TypeError. I have no idea what's wrong."

Everyday

Work

Creative

Learning

Coding

The categories overlap at the edges — that's intentional. The goal is to develop a reflex, not to memorize rules.

One last thing: start with something real

Everything in this lesson becomes real skill the moment you use it on an actual task from your life. Not a practice exercise, not a hypothetical — a real email you've been avoiding, a document that's been sitting in your inbox, a piece of writing you're stuck on.

The thing that separates people who get a lot of value from AI and people who feel like it "never works for me" is almost always the same: the first group uses it on real tasks with real stakes. The second group tries it on abstract tests and gets abstract results.

Your task this week

Before you do the homework, try one real thing. Open Claude, ChatGPT, or any AI you have access to. Pick something from your actual life this week — an email, a plan, a question you've been meaning to research, something you need to write. Use a prompt from the Explorer above as a starting point. See what comes back. That experience will make the homework 10x more useful.