

AI coding prompts you can copy and paste

These prompts cover the everyday work of writing software with AI: debugging from a stack trace, reviewing a pull request, writing tests before the code, planning a feature, building small tools, learning a new framework and writing docs. The chat prompts work in Claude, ChatGPT and Gemini. The agent prompts are written for Claude Code and Cursor, which can read your project, run commands and edit files.

Two habits make every prompt here work better: paste the real code, error or diff instead of describing it, and say what "done" looks like. Remove passwords, API keys and customer data before you paste anything, and always run and read the code the AI gives you.

How to use: copy a prompt into ChatGPT, Gemini or Claude and replace every [BRACKETED] part with your own details. Each prompt also has its own page with a copy button, linked under its title. 76 prompts in this pack.

1 Write a CLAUDE.md file for your project

Claude Code · Cursor · usepromptup.com/prompts/text-code-claude-md-file/

Help me write a CLAUDE.md file for this project, so an AI coding agent works the way my team does.

Project: [WHAT IT DOES]

Stack: [e.g. Next.js, TypeScript, Prisma, Postgres]

Commands: [HOW TO INSTALL, RUN, TEST, LINT, BUILD]

Rules we follow: [e.g. no default exports, tests next to files, never edit generated code]

Keep it short and specific: project overview in two lines, key commands, folder map, code style rules, testing rules, and things the agent must never do. Only include rules that are not obvious from the code. Use bullet points, not paragraphs.

Tip: Short beats long. If a rule is ignored, make it more specific instead of adding more text.

2 Give an AI coding agent a clear task brief

Claude Code · Cursor · usepromptup.com/prompts/text-code-agent-task-brief/

Goal: [WHAT SHOULD BE TRUE WHEN DONE, e.g. users can export their invoices as CSV]

Context: [WHERE IN THE CODE, e.g. see src/invoices/ and the existing PDF export]

Constraints: [e.g. no new dependencies, keep the API backward compatible]

Done means:

- [e.g. new endpoint with tests]
- [e.g. button on the invoices page]
- [e.g. all tests and lint pass]

First explore the relevant code and tell me your plan in a few bullet points. Wait for my OK. Then implement it in small steps, run the tests after each step, and finish with a summary of what changed and anything you were unsure about.

Tip: A clear "done means" list is the single biggest improvement for agent results. It tells the agent when to stop.

3 Find the root cause from a stack trace

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-stack-trace-root-cause/

Act as a senior [LANGUAGE, e.g. Python / Django] developer helping me debug.

Error and full stack trace:

[PASTE THE STACK TRACE]

The code it points to:

[PASTE THE RELEVANT FILE OR FUNCTION]

What I expected to happen: [EXPECTED, e.g. the order saves and returns 201]

Walk the trace from the bottom up and tell me which line actually causes the failure, not just where it crashes. Give the most likely root cause, one or two other possible causes, a minimal fix as a code diff, and one check I can run to confirm the fix worked.

Tip: Paste the whole trace, not only the last line. The real cause is often a few frames above the crash.

4 Review a pull request diff before you merge

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-pr-diff-review/

Act as a careful senior reviewer on my team.

What this PR is meant to do: [GOAL, e.g. add password reset by email]

Diff:

[PASTE git diff OR THE CHANGED FILES]

Review only what changed. Look for bugs, missed edge cases, security problems, breaking changes, missing tests and confusing names. List findings as blocking or non-blocking, each with the file and line, why it matters, and a suggested fix. If the PR does something it did not mean to do, call it out. Skip style nitpicks a linter would catch.

Tip: In Claude Code you can just say "review my staged changes" and it reads the diff itself.

5 Plan a feature before writing any code

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-plan-before-coding/

Act as a tech lead. Before we write code, help me plan.

Feature: [WHAT AND WHY]

Current system: [SHORT DESCRIPTION OR RELEVANT CODE]

Constraints: [e.g. must ship in 1 week, no new services, must work offline]

Give me: a short summary of the approach, the files and parts that change, the data model changes, risks and unknowns, and a step-by-step build order where each step can be tested on its own. Offer one simpler alternative and say when it would be the better choice. Do not write code yet.

Tip: In Claude Code, use plan mode for this. Agree on the plan, then say "go ahead with step 1".

6 Review code written by an AI before you trust it

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-review-ai-written-code/

Act as a sceptical senior reviewer. The code below was written by an AI assistant.

What it is supposed to do: [GOAL]

Code:

[PASTE]

Check especially for the mistakes AI code often has: invented functions or library options that do not exist, outdated APIs, missing error handling, security gaps (injection, secrets, weak validation), code that handles the happy path only, and tests that do not really test anything. For each problem, give the line, why it is wrong, and the fix. End with what I should test by hand.

Tip: Run the code and read it. AI code often looks right and fails on the first unusual input.

7 Fix a bug that works locally but fails in production

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-works-locally-fails-prod/

Act as a senior engineer who has debugged many "works on my machine" problems.

What fails in production: [SYMPTOM, e.g. image upload returns 500 on the live site]

Local setup: [e.g. Node 22, SQLite, .env file]

Production setup: [e.g. Node 20 on Vercel, Postgres, env vars in dashboard]

Logs or error: [PASTE]

Code involved: [PASTE]

List the differences between the two environments that could cause this, ranked by likelihood (env vars, versions, file paths, permissions, timezones, build step, caching). For the top three, tell me exactly how to check it and what the fix would be.

Tip: The more precisely you describe both environments, the faster it finds the gap. Versions and env vars are the usual suspects.

8 Fix a flaky test that passes and fails at random

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-flaky-test-fix/

Act as a test reliability expert. This test fails about [HOW OFTEN, e.g. 1 in 5 runs] with no code change.

Test code:
[PASTE THE TEST]

Code under test:
[PASTE]

Failure output:
[PASTE A FAILING RUN]

Find the source of the flakiness: timing and async waits, shared state between tests, test order, random data, dates and timezones, network calls or leftover files. Explain the cause in plain words, then rewrite the test so it is deterministic. Do not fix it by adding sleeps or retries.

Tip: If you can, run the test 20 times in a loop and paste two different failures. Patterns between them point to the cause.

9 Speed up a slow API endpoint or function

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-slow-endpoint-speedup/

Act as a performance engineer. This [ENDPOINT OR FUNCTION, e.g. GET /orders] takes [TIME, e.g. 4 seconds] and should take under [TARGET, e.g. 300 ms].

Code:
[PASTE]

Database queries or ORM calls it makes: [PASTE OR DESCRIBE]
Data size: [e.g. 200k orders, 5k users]

Find the biggest slowdowns first (N+1 queries, missing indexes, loops inside loops, repeated work, large payloads). For each, estimate how much time it costs, show the fix, and tell me how to measure before and after. Change only what gives a real gain.

Tip: Paste real timing numbers or a profiler output if you have them. Guesses about performance are often wrong.

10 Track down a memory leak

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-memory-leak-hunt/

Act as an expert in [RUNTIME, e.g. Node.js / Python / Java] memory issues.

Symptom: [e.g. memory grows from 200 MB to 2 GB over 6 hours, then the app restarts]

What the app does: [SHORT DESCRIPTION]

Suspicious code:

[PASTE: caches, event listeners, timers, global lists]

List the most likely leak sources in this code, explain why each one keeps memory alive, and show the fix. Then give me a step-by-step plan to confirm the leak with a heap snapshot or profiler, including exactly what to compare between two snapshots.

Tip: Common leaks: listeners that are never removed, caches with no size limit, and timers that keep running after a request ends.

11 Resolve a Git merge conflict safely

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-merge-conflict-resolve/

Act as a senior developer helping me resolve a Git merge conflict.

What my branch does: [e.g. adds discount codes at checkout]

What the other branch does: [e.g. refactors checkout into smaller functions]

Conflicted file with markers:

[PASTE THE FILE WITH <<<<<< ===== >>>>>> MARKERS]

Explain what each side changed and why they clash. Then give me the merged version that keeps both intentions, and point out any line where you had to choose one side. Finish with the exact git commands to mark it resolved and a quick test to confirm nothing broke.

Tip: Describe what each branch was trying to do. The AI resolves conflicts much better when it knows the intent, not just the lines.

12 Fix a CORS error between frontend and API

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-cors-error-fix/

Act as a web developer who understands CORS well.

Frontend: [e.g. React app on http://localhost:5173]

API: [e.g. Express on http://localhost:3000, later api.mysite.com]

Exact browser error:

[PASTE THE CONSOLE ERROR]

The request: [METHOD, HEADERS, e.g. POST with JSON and a cookie]

Server code that sets headers (if any): [PASTE]

Explain in plain words why the browser blocks this request, including whether it triggers a preflight. Give the correct server-side fix for development and for production. Do not suggest allowing every origin with credentials, and do not suggest browser extensions.

Tip: CORS is fixed on the server, not in the frontend. Paste the exact error text, because each message points to a different missing header.

13 Check a UI component for accessibility

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-accessibility-review/

Act as an accessibility specialist who knows WCAG 2.2 AA.

Component: [WHAT IT IS, e.g. a signup modal with a form]

Code:

[PASTE HTML / JSX AND CSS]

Check keyboard use, focus order and focus trap, labels and names for screen readers, colour contrast, error messages, alt text and motion. For every issue, give the WCAG rule, who it affects, and the fixed code. Order them by impact on real users, and tell me how to test each fix with only a keyboard and a free screen reader.

Tip: Afterwards, try the component using only the Tab and Enter keys. It is the fastest real test you can do.

14 Review a database schema before it goes live

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-database-schema-review/

Act as a database architect. Review this schema for [DATABASE, e.g. PostgreSQL].

What the app does: [SHORT DESCRIPTION, e.g. a booking app for clinics]

Schema:

[PASTE CREATE TABLE STATEMENTS OR ORM MODELS]

Main queries it must handle: [e.g. find free slots for a doctor on a date]

Expected size: [e.g. 50 clinics, 2 million bookings a year]

Check keys, relations, data types, nullability, indexes, naming, and anything that will hurt at scale or make migrations painful. Give each issue a severity, and show the corrected SQL. Suggest indexes for the main queries.

Tip: List your real queries. A schema can only be judged against the questions it has to answer.

15 Find dead code and safe things to delete

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-dead-code-cleanup/

Act as a senior engineer tidying up a codebase.

Language and framework: [e.g. TypeScript, Next.js]

Code:

[PASTE THE FILES OR MODULE]

Find unused functions, variables, imports, exports, feature flags that are always on or off, commented-out blocks and duplicate logic. For each, say how confident you are that it is safe to remove and what could still use it (dynamic imports, reflection, other services, templates). Group the result into safe to delete now, check first, and keep.

Tip: Remove things in small commits and run the tests after each one, so a mistake is easy to undo.

16 Improve error handling in your code

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-error-handling-review/

Act as a senior [LANGUAGE] engineer focused on reliability.

Code:

[PASTE]

This code runs in [CONTEXT, e.g. a payment webhook handler]. Find places where errors are swallowed, caught too broadly, logged without context, or shown to users with internal details. Also find places with no error handling that need it (network calls, file access, parsing, database writes). Rewrite the code with clear error handling, useful log messages and safe user-facing messages. Explain each change in one line.

Tip: Tell it where the code runs. A CLI script, a web API and a background job each need different error handling.

17 Make confusing code easy to read

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-code-readability-rewrite/

Act as a senior developer who writes very readable code.

Code:

[PASTE]

Rewrite it so a new team member can understand it quickly, without changing what it does. Use clear names, smaller functions, early returns instead of deep nesting, and remove clever tricks. Keep the same inputs, outputs and side effects. Show the new code, then list each change with a short reason, and point out anything whose behaviour you were unsure about.

Tip: Run your tests before and after. If there are no tests, ask for a few first so you know the rewrite did not change behaviour.

18 Write the tests first for a new feature

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-test-first-feature/

Act as a developer who practises test-driven development.

Feature: [WHAT IT SHOULD DO, e.g. apply a coupon code to a cart]

Rules:

[LIST THE RULES, e.g. one coupon per cart, expired codes rejected]

Language and test framework: [e.g. TypeScript with Vitest]

Write the failing tests first, one per rule, plus edge cases. Use names that read like sentences. Do not write the implementation yet. After the tests, list any rule that was unclear and the assumption you made, so I can confirm before we write the code.

Tip: Check the tests match what you want before asking for the code. Then ask: "now write the simplest code that makes these pass".

19 List the edge cases your code might miss

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-edge-case-finder/

Act as a QA engineer who is very good at breaking things.

Function or feature:

[PASTE THE CODE OR DESCRIBE IT]

List every edge case you can think of: empty and missing values, very large inputs, special characters and emoji, dates across timezones and leap years, duplicates, concurrency, slow or failed network calls, and permission problems. For each, say what the code would do now (from reading it) and what it should do. Mark the ones most likely to cause a real bug.

Tip: Turn the top three into tests straight away. An edge case on a list does not protect anything.

20 Write an end-to-end test for a user journey

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-playwright-e2e-test/

Act as a QA automation engineer using [TOOL, e.g. Playwright with TypeScript].

User journey to test: [STEPS, e.g. sign up, verify email, create first project]

App URL: [e.g. <http://localhost:3000>]

Useful selectors or page code: [PASTE OR DESCRIBE]

Write a reliable end-to-end test. Use role and label selectors, not CSS classes. Wait for real conditions, never fixed sleeps. Set up and clean up test data. Add clear step names so a failure is easy to read, and explain how to run it headed and in CI.

Tip: Selectors like `getByRole` and `getByLabel` survive design changes. CSS class selectors break often.

21 Mock an external API in your tests

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-mock-external-api/

Act as a senior [LANGUAGE] developer.

My code calls this external service: [e.g. Razorpay payments API]

Code that makes the call:

[PASTE]

Test framework: [e.g. Jest / pytest]

Show me how to test this code without calling the real service. Mock the success case, an error response, a timeout and a malformed response. Keep the mock at the network or client boundary, not deep inside my code. Explain what the mocks can miss and suggest one contract or sandbox test to cover that gap.

Tip: Mocks prove your code handles responses. They cannot prove the real API still behaves the same, so keep one real check somewhere.

22 Turn a fixed bug into a regression test

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-bug-to-regression-test/

Act as a careful engineer.

Bug that happened: [WHAT WENT WRONG, e.g. total showed ₹0 when cart had one free item]

How it was fixed: [PASTE THE FIX OR DIFF]

Test framework: [e.g. pytest]

Write a regression test that fails on the old code and passes on the fixed code. Name it after the behaviour, not the ticket number. Keep it small and focused on this bug, and add a one-line comment linking the test to the bug so nobody deletes it by mistake.

Tip: To be sure the test works, undo the fix for a moment and check the test fails.

23 Build a REST API endpoint with validation

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-rest-api-endpoint/

Act as a senior backend developer using [STACK, e.g. Node.js with Express and Zod].

Endpoint: [METHOD AND PATH, e.g. POST /api/bookings]

What it does: [e.g. creates a booking if the slot is free]

Input fields and rules: [LIST]

Data store: [e.g. Postgres with Prisma]

Write the route with input validation, clear status codes, error responses in one consistent shape, and no internal details leaked to the client. Handle the case where two requests arrive at once. Include two example requests with curl and two tests.

Tip: Ask for the error format you already use in your app, so every endpoint answers in the same shape.

24 Build a React component from a description

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-react-component-from-spec/

Act as a senior frontend developer using [STACK, e.g. React with TypeScript and Tailwind].

Component: [WHAT IT IS, e.g. a pricing table with monthly and yearly toggle]

Props and data: [DESCRIBE]

Behaviour: [e.g. yearly shows 2 months free, highlights the middle plan]

Style: [e.g. clean, rounded cards, works on mobile]

Write one self-contained component with typed props, accessible markup, keyboard support and a mobile layout. Keep state minimal. Add a small usage example and list any assumption you made.

Tip: Paste an existing component from your project and say "match this style" for code that fits in.

25 Build a small command-line tool

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-cli-tool-builder/

Act as a developer who writes friendly command-line tools in [LANGUAGE, e.g. Python].

The tool should: [WHAT IT DOES, e.g. rename photos by the date they were taken]

Inputs and options: [e.g. folder path, --dry-run, --pattern]

Write the complete tool with helpful --help text, a dry-run mode, clear error messages and a non-zero exit code on failure. Never delete or overwrite files without a flag that says so. Use only the standard library unless a package really helps. Show three example commands and their output.

Tip: Always try a new file-changing tool with --dry-run on a copy of your folder first.

26 Build a landing page in plain HTML and CSS

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-landing-page-html/

Act as a frontend developer and designer.

Product: [WHAT IT IS, e.g. a tiffin delivery service in Pune]

Sections: [e.g. hero, how it works, menu, pricing, FAQ, contact]

Style: [e.g. warm colours, rounded, friendly, lots of white space]

Brand colour: [e.g. #F97316]

Write one HTML file with embedded CSS and no frameworks. Make it responsive, fast and accessible, with real-looking placeholder copy I can edit. Use semantic tags, good contrast, a sticky header and a clear call-to-action button. Add comments marking where to change text and images.

Tip: Open the file in your browser, then ask for one change at a time: "make the hero shorter", "add a testimonials section".

27 Build a simple Chrome extension

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-chrome-extension-builder/

Act as a developer experienced with Chrome extensions (Manifest V3).

The extension should: [WHAT IT DOES, e.g. show word count of selected text]

Where it works: [e.g. on every page, or only on docs.google.com]

Give me every file (manifest.json, scripts, popup if needed) with the smallest set of permissions possible, and explain why each permission is needed. Include step-by-step instructions to load it unpacked in Chrome and test it. Do not collect or send any browsing data.

Tip: Fewer permissions means fewer scary warnings when you install it. Ask the AI to justify each one.

28 Clean a messy spreadsheet with Python

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-pandas-data-cleaning/

Act as a data analyst who writes clean pandas code.

My file: [e.g. sales.csv, about 20,000 rows]

Columns and a few sample rows:

[PASTE HEADER + 5 ROWS]

Problems I see: [e.g. mixed date formats, ₹ symbols in amounts, duplicate customers]

What I want at the end: [e.g. monthly revenue per city]

Write a script that loads the file, fixes each problem step by step with a comment, reports how many rows each step changed, and saves a clean file. Never overwrite the original. Then produce the summary I asked for.

Tip: Paste real sample rows, with personal details removed. Messy data is always messier than the description.

29 Automate Google Sheets with Apps Script

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-google-sheets-apps-script/

Act as a Google Apps Script expert.

My sheet: [DESCRIBE THE TABS AND COLUMNS]

What I want automated: [e.g. every morning, email me rows where due date is today]

Write the full script with comments, a menu item to run it by hand, and a time-driven trigger if it should run on its own. Handle empty rows and bad dates. Tell me exactly where to paste the code, what permissions Google will ask for and why, and how to test it on a copy of the sheet first.

Tip: Test on a copy of your sheet. Scripts can change many rows at once, and undo does not always catch it.

30 Build a simple Telegram or Discord bot

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-telegram-bot-builder/

Act as a developer who builds chat bots in [LANGUAGE, e.g. Python].

Platform: [Telegram or Discord]

The bot should: [e.g. reply with today's gold rate when someone types /gold]

Commands: [LIST]

Write the complete bot with a well-maintained, popular library, the token read from an environment variable (never hard-coded), clear replies for unknown commands, and basic rate limiting. Explain how to create the bot token, run it locally, and keep it running on a free or cheap host.

Tip: Never paste your real bot token into a chat or commit it to Git. Keep it in a .env file.

31 Turn a user story into working code

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-feature-from-user-story/

Act as a senior full-stack developer on my project.

User story: As a [WHO], I want to [WHAT], so that [WHY].

Acceptance criteria:

[LIST]

Stack: [e.g. Next.js, Postgres, Prisma]

Relevant existing code:

[PASTE OR DESCRIBE THE FILES]

First, restate the feature in two lines and list questions or assumptions. Then list every file you would add or change. Then write the code, following the patterns in the existing code. Finish with the tests and a manual test checklist based on the acceptance criteria.

Tip: Answer its questions before it writes code. Five minutes of clarifying saves an hour of rework.

32 Add secure login to a web app

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-auth-login-flow/

Act as a security-minded backend developer.

Stack: [e.g. Express + Postgres, or Django]

Login type: [e.g. email and password with email verification]

Explain whether I should use a trusted library or service instead of building this myself, and recommend one for my stack. Then show the setup: password hashing, sessions or tokens, cookie settings, rate limiting on login, password reset with expiring links, and logout. Point out the five most common security mistakes in login code and how this setup avoids them.

Tip: For real users, a well-known auth library or service is almost always safer than hand-written login code.

33 Understand a new codebase fast

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-codebase-onboarding-map/

Act as a senior engineer giving me a tour of a codebase I just joined.

Project: [WHAT IT DOES]

Folder structure:

[PASTE THE OUTPUT OF `tree` OR `ls -R`, trimmed]

Key files:

[PASTE `package.json` / `requirements.txt` AND THE MAIN ENTRY FILE]

Explain the architecture in plain words: the main parts, how a request or action flows through them, where data is stored, and where config lives. Then give me a reading order of the ten most important files, and three small starter tasks that would teach me the codebase.

Tip: In Claude Code or Cursor, skip the pasting: ask it to explore the repo and write this tour for you.

34 Learn a programming concept with analogies

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-concept-with-analogy/

Act as a patient programming teacher.

Concept: [e.g. `async/await`, closures, database indexes, Docker]

My level: [e.g. I know basic Python loops and functions]

Explain it in four steps: a real-life analogy, a plain definition, a tiny code example with each line explained, and a common mistake beginners make with it. Then give me two small exercises, and wait for my answers before showing solutions.

Tip: Say what you already know. The explanation fits much better when it builds on something familiar.

35 Practise coding interview problems with hints

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-dsa-hint-coach/

Act as a coding interview coach. I want to learn, not copy answers.

Problem: [PASTE THE PROBLEM, e.g. from LeetCode]

Language: [e.g. Java]

My attempt so far: [PASTE CODE OR MY IDEA]

Do not give me the solution. Give one hint at a time, starting small, and wait for my reply before the next hint. When I have a working answer, review its time and space complexity, name the pattern it uses (e.g. two pointers, sliding window), and suggest one similar problem to practise.

Tip: Try for 15 minutes before asking for the first hint. The struggle is where the learning happens.

36 Learn a new framework using one you know

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-learn-framework-from-known/

Act as a senior developer who knows both [FRAMEWORK I KNOW, e.g. React] and [NEW FRAMEWORK, e.g. SvelteKit].

Teach me the new one by comparing it with the one I know. Cover project structure, components, state, routing, data loading, forms and deployment. For each, show a side-by-side code example and point out where my old habits will trip me up. Finish with a small project I can build in one weekend to practise.

Tip: Build the weekend project for real. Reading comparisons helps, but typing the code is what makes it stick.

37 Understand a long SQL query step by step

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-explain-sql-query/

Act as a SQL teacher.

Query:

[PASTE THE SQL]

Database: [e.g. PostgreSQL]

Tables involved (optional): [PASTE SCHEMAS]

Explain the query in the order the database actually runs it (FROM, JOIN, WHERE, GROUP BY, HAVING, SELECT, ORDER BY), not the order it is written. Describe what the data looks like after each step with a tiny example table. Then point out any bug risk, like duplicate rows from a join or NULLs being dropped.

Tip: Ask for the tiny example tables. Seeing the rows change at each step is what makes SQL click.

38 Choose a tech stack for your project

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-choose-tech-stack/

Act as a pragmatic CTO advising a small team.

Project: [WHAT YOU ARE BUILDING]

Team: [e.g. 2 developers who know JavaScript and a little Python]

Needs: [e.g. mobile-friendly web app, payments, 5k users in year one]

Budget: [e.g. under ₹5,000 a month for hosting]

Recommend a stack for frontend, backend, database, auth, hosting and payments. For each, give one alternative and the reason to pick it. Prefer boring, well-documented tools my team already knows. Flag anything that will be hard to change later.

Tip: The best stack is usually the one your team already knows. Say honestly what everyone is comfortable with.

39 Break a project into small tickets

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-break-into-tickets/

Act as an experienced engineering manager.

Project or feature: [DESCRIBE]

Team: [e.g. 1 frontend, 1 backend developer]

Deadline: [e.g. 3 weeks]

Break the work into tickets that each take less than a day. For every ticket give a title, a two-line description, acceptance criteria and dependencies. Put them in a sensible order, mark what can happen in parallel, and highlight the riskiest ticket so we can start it first. Output as a table I can paste into Jira, Linear or Trello.

Tip: Start with the riskiest ticket. If something is going to blow up the estimate, it is better to know in week one.

40 Write a pull request description reviewers love

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-pr-description-writer/

Act as a senior developer writing a pull request description.

What the PR does: [SHORT SUMMARY]

Why: [THE PROBLEM OR TICKET]

Diff or list of changes:

[PASTE]

How I tested it: [e.g. unit tests, manual checks]

Write a description with: a two-line summary, why the change is needed, a list of the main changes, how to test it, screenshots needed (if UI), and risks or things reviewers should look at closely. Keep it scannable and short.

Tip: Tell reviewers where to look first. A good "please check this part" saves everyone time.

41 Write an architecture decision record

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-architecture-decision-record/

Act as a staff engineer documenting a technical decision.

Decision: [e.g. move background jobs from cron to a queue]

Context: [WHY WE HAD TO DECIDE]

Options we considered: [LIST]

What we chose and why: [SHORT]

Write an ADR with title, status, context, decision, options considered with pros and cons, consequences (good and bad), and when we should revisit it. Keep it under one page, neutral in tone, and understandable to someone joining the team next year.

Tip: The "consequences" section is the most useful one later. Be honest about the downsides you accepted.

42 Write Cursor project rules for your codebase

Claude Code · Cursor · usepromptup.com/prompts/text-code-cursor-rules-file/

Help me write project rules for Cursor (files in `.cursor/rules`) for this codebase.

Stack: [e.g. React Native, Expo, TypeScript]

Conventions: [e.g. functional components, Zustand for state, styles in a separate file]

Things the AI keeps getting wrong: [e.g. uses old API, adds new libraries]

Split the rules into a few small files by topic, and say which files or folders each one should apply to. Each rule should be one clear instruction with a short reason. Add a rule telling the AI to ask before adding a new dependency.

Tip: Add a rule every time the AI makes the same mistake twice. Rules work best when they come from real problems.

43 Let an AI agent fix failing tests properly

Claude Code · Cursor · usepromptup.com/prompts/text-code-fix-failing-tests-agent/

Run [TEST COMMAND, e.g. `npm test`] and fix the failing tests.

Rules:

- Find out why each test fails before changing anything.
- If the code is wrong, fix the code. If the test is outdated, explain why before changing it.
- Never delete a test, skip it, or loosen an assertion just to make it pass.
- Keep changes small and run the tests again after each fix.

When everything passes, give me a short list of each failure, its cause and the fix, and point out anything that looks like a real bug in the product.

Tip: The "never delete or skip a test" line matters. Without it, agents sometimes make tests pass the easy way.

44 Do a large refactor in small safe steps

Claude Code · Cursor · usepromptup.com/prompts/text-code-safe-large-refactor/

Refactor goal: [e.g. replace our custom fetch wrapper with the official SDK across the app]

Scope: [FOLDERS OR FILES]

Must not change: [e.g. public API, database schema, UI behaviour]

First, find every place this touches and show me the list with a plan split into small steps, each one leaving the app working. Then do one step at a time: make the change, run [TEST AND LINT COMMANDS], and stop if anything fails. Do not mix in unrelated clean-ups. At the end, list anything you left for later and why.

Tip: Commit after each step. If step 7 goes wrong, you only lose step 7.

45 Fix a layout bug from a screenshot

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-ui-bug-from-screenshot/

Act as a senior frontend developer. I am attaching a screenshot of a layout bug.

What looks wrong: [e.g. the card buttons overflow on mobile, text overlaps the image]

Where it happens: [e.g. Safari on iPhone, screens under 400px]

Component code and CSS:

[PASTE]

Describe what you see in the screenshot first, so I know you read it correctly. Then explain which CSS rule causes the problem, give the fixed code, and tell me which other screen sizes I should check so the fix does not break something else.

Tip: Attach a screenshot of the broken view and, if you can, one of how it should look. Two pictures beat a long description.

46 Write a Dockerfile for your app

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-dockerfile-for-app/

Act as a DevOps engineer.

App: [e.g. a FastAPI app with a requirements.txt]

How it runs locally: [COMMANDS]

Needs: [e.g. Postgres, Redis, env vars for API keys]

Write a production-ready Dockerfile with a small base image, a multi-stage build if it helps, a non-root user, cached dependency layers and a health check. Add a docker-compose.yml for local development with the services it needs, and a .dockerignore. Explain each important line briefly, and keep secrets out of the image.

Tip: Check the image size with `docker images`. If it is over 1 GB, ask the AI to make it smaller.

47 Check your code for leaked API keys and secrets

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-secrets-leak-check/

Act as a security engineer.

Code or config files:

[PASTE, with any real keys replaced by XXXX]

Where the code lives: [e.g. public GitHub repo, frontend bundle, mobile app]

Find every place where a secret could leak: hard-coded keys, tokens in frontend code, keys in logs or error messages, .env files that could be committed, and secrets in URLs. For each, explain the risk and the fix. Then give me a checklist for what to do if a key has already been pushed to Git, in the right order.

Tip: If a real key was ever pushed, rotate it first. Deleting it from Git history is not enough, because it may already be copied.

48 Add logging that helps you debug later

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-useful-logging/

Act as a senior engineer who has debugged many production incidents.

Code:

[PASTE]

It runs as: [e.g. an API on a server, a nightly job, a serverless function]

Logging tool: [e.g. console, Python logging, pino, or none yet]

Add logging that would let me answer "what happened?" during an incident: key steps, IDs to follow one request, timings for slow calls and clear error context. Use levels properly (debug, info, warn, error). Never log passwords, tokens or personal data. Explain what each log line would tell me.

Tip: Add a request or job ID to every log line. It lets you follow one user's journey through thousands of lines.

49 Create TypeScript types from a JSON response

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-json-to-types/

Act as a TypeScript expert.

API response examples:

[PASTE 2 OR 3 REAL JSON RESPONSES]

Where it comes from: [e.g. a weather API]

Write TypeScript types that match the data, marking fields as optional or nullable where the examples differ. Use clear names and union types for fixed values. Then write a runtime validator with [LIBRARY, e.g. Zod] so bad data is caught at the edge of my app, and show how to use both in a fetch call.

Tip: Paste more than one response, including an error one. A single example hides which fields are optional.

50 Upgrade a library to a new major version

Claude · ChatGPT · Claude Code · usepromptup.com/prompts/text-code-upgrade-major-version/

Act as a senior developer planning an upgrade.

Library: [e.g. React Router] from version [OLD] to [NEW]

How we use it: [PASTE THE MAIN FILES OR DESCRIBE]

Tests we have: [e.g. unit tests, no end-to-end tests]

List the breaking changes that affect code like mine, with before and after examples. Give me an upgrade order that keeps the app working, including any codemods or compatibility modes. Tell me what to test by hand afterwards. If you are not sure about a detail of the new version, say so and point me to the library's own migration guide.

Tip: Always check the library's own migration guide too. AI knowledge of very new versions can be out of date.

51 Add an AI feature to your app with an LLM API

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-code-ai-feature-llm-api/

Act as a developer who has shipped features using LLM APIs.

Feature: [e.g. summarise customer reviews into pros and cons]

Stack: [e.g. Next.js server route]

API: [e.g. the Claude or OpenAI API]

Write the server-side code that calls the API with a clear system prompt, handles errors, timeouts and rate limits, and returns structured JSON I can render. Keep the API key on the server. Add a simple way to limit cost (input size, caching, per-user limits), and give me five test inputs, including tricky ones, to check the output quality.

Tip: Keep the API key on the server, never in frontend code. Anyone can read a key that ships to the browser.

52 Senior engineer code review & debugging

Any chatbot · usepromptup.com/prompts/text-code-debug/

Act as a Staff Principal Software Engineer.

Review and debug this [LANGUAGE / FRAMEWORK, e.g. React TypeScript] code:

...

[PASTE YOUR CODE HERE]

...

1. Identify the root cause of [THE BUG / ERROR MESSAGE].
2. Provide the corrected drop-in code snippet.
3. Explain what went wrong and how to avoid it in production.

Tip: Always paste the exact terminal error stack trace for instant pinpoint diagnosis.

53 Debug an error message step by step

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-debug-error-message/

I get this error:

[PASTE ERROR OR STACK TRACE]

Language and framework: [LANGUAGE AND VERSION]

The code that triggers it:

[PASTE CODE]

What I already tried: [ATTEMPTS]

Explain what the error means in plain English, list the three most likely causes in order, show how to test each quickly and provide the corrected code for the most likely fix.

Tip: Include the exact error text and the smallest code snippet that reproduces it.

54 Get a code review like a senior engineer

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-code-review/

Act as a senior [LANGUAGE] engineer doing a code review.

Context: [WHAT THIS CODE DOES]

Code: [PASTE]

Review for bugs, security issues, performance, readability and tests. Group findings as must fix, should fix and nice to have. For each, show the problem line and a corrected snippet. Be direct and skip praise.

Tip: Tell the AI your team's style guide or framework version for better advice.

55 Write unit tests for your code

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-unit-tests/

Act as a senior engineer. Write thorough unit tests for this [LANGUAGE / FRAMEWORK, e.g. Python with pytest] code:

[PASTE YOUR CODE]

Cover normal cases, edge cases (empty, null, very large, and invalid input), and error handling. Use clear test names that describe the behaviour. Point out any bug you find while writing the tests, and explain how to run them.

Tip: Run the tests yourself — if one fails, paste the output back and ask why.

56 Explain code line by line

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-explain-code/

Explain this code to a [BEGINNER / INTERMEDIATE] developer.

Language: [LANGUAGE]

Code: [PASTE CODE]

Give: what it does in one sentence, line-by-line explanation of the tricky parts, the flow of data, edge cases it does not handle and one way to improve it. Use plain words.

Tip: Ask "what would break if the input were empty?" for a quick review.

57 Refactor messy code safely

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-refactor-clean-code/

Refactor this code to be cleaner without changing what it does.

Language: [LANGUAGE]

Code: [PASTE]

Rename unclear variables, split long functions, remove duplication and add short comments only where the reason is not obvious. Show the new code, then list every change and confirm behaviour is unchanged. Do not add new features.

Tip: Run your tests before and after. Refactoring without tests is a gamble.

58 Security review checklist for your code

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-security-review-checklist/

Review this code for security problems:

[PASTE CODE]

Context: [WHAT IT DOES AND WHO USES IT]

Check for injection, broken authentication, insecure secrets, unsafe input handling, missing authorisation checks, sensitive data exposure and unsafe dependencies. For each issue explain the risk in plain English, rate severity and show a fix.

Tip: Run it against code that handles logins, uploads and payments first; that is where the risk is.

59 Write an SQL query from plain English

Claude · ChatGPT · Gemini · usepromptup.com/prompts/text-sql-from-english/

Act as a database expert. Write an SQL query for [DATABASE, e.g. PostgreSQL].

My tables:

[PASTE TABLE NAMES AND COLUMNS]

What I need: [QUESTION, e.g. the top 10 customers by total spend in the last 90 days, with their order count].

Return the query, explain each part in one line, and mention any index that would make it faster.

Tip: Test on a copy of your data first, especially for UPDATE or DELETE queries.

60 Speed up a slow SQL query

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-sql-optimise/

Help me optimise this slow SQL query.

Database: [POSTGRES / MYSQL / SQL SERVER]

Query: [PASTE]

Table sizes: [ROW COUNTS]

Existing indexes: [LIST]

Explain why it is slow, suggest indexes and a rewritten query, and tell me what to look for in EXPLAIN output. Warn me about any change that could alter the results.

Tip: Paste the EXPLAIN ANALYZE output if you can. It removes the guesswork.

61 Design a REST API for your app

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-api-design-helper/

Help me design a REST API.

App: [WHAT IT DOES]

Main resources: [e.g. users, orders, products]

Auth: [JWT / SESSION / NONE]

List endpoints (method, path, purpose), request and response JSON examples for 3 of them, error format, pagination approach and status codes to use. Follow REST conventions and point out anything I may have missed.

Tip: Ask for an OpenAPI (YAML) version once the design looks right.

62 Design a database schema

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-database-schema-design/

Design a database schema for [APPLICATION].

Database: [POSTGRES / MYSQL / OTHER]

Main entities and rules: [DESCRIBE]

Most common queries: [QUERIES]

Expected size: [SCALE]

Provide the tables with columns, types, primary and foreign keys, indexes, constraints and a short explanation of key design choices. Point out any trade-offs and future changes to consider.

Tip: Describe your top five queries up front so the schema is built for how you will use it.

63 Write a GitHub Actions workflow

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-github-actions-workflow/

Write a GitHub Actions workflow for a [LANGUAGE] project.

It should: [RUN TESTS / LINT / BUILD / DEPLOY] on [EVENT, e.g. pull requests to main].

Deployment target: [TARGET]

Caching needed: [DEPENDENCIES]

Provide the YAML with comments, explain each step, list the secrets I must add and suggest one improvement to make the pipeline faster.

Tip: Pin action versions and keep secrets in repository settings, never in the workflow file.

64 Plan how to find a performance bottleneck

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-performance-profiling-plan/

My [APP OR PAGE] is slow when [SITUATION].

Stack: [STACK]

What I measured: [MEASUREMENTS]

Scale: [USERS OR DATA SIZE]

Give me a step-by-step plan to find the bottleneck: what to measure, which tools to use, likely culprits in order, quick wins and how to confirm each fix helped.

Tip: Measure before you change anything, or you will never know whether it helped.

65 Plan a safe technology migration

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-migration-plan-code/

We want to migrate from [OLD TECHNOLOGY] to [NEW TECHNOLOGY].

System size: [SIZE]

Team: [TEAM SIZE AND SKILLS]

Downtime tolerance: [TOLERANCE]

Create a phased migration plan with an assessment step, pilot, incremental rollout, testing, rollback plan and success criteria. List the top risks and how to reduce each, and estimate effort in ranges.

Tip: Migrate one small, low-risk part first to learn what will go wrong.

66 Write a technical spec

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-technical-spec-writer/

Write a technical specification for [FEATURE OR SYSTEM].

Problem: [PROBLEM]

Requirements: [REQUIREMENTS]

Constraints: [CONSTRAINTS]

Include goals and non-goals, proposed design, data model, API changes, alternatives considered, security and privacy notes, testing plan, rollout plan and open questions.

Tip: Have a colleague from another team read it; unclear parts show up immediately.

67 Write clear git commit messages

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-git-commit-messages/

Write git commit messages for these changes.

Changes: [PASTE DIFF OR DESCRIBE]

Convention: [CONVENTIONAL COMMITS / PLAIN]

Give a summary line under 72 characters in the imperative mood, then a short body explaining why the change was needed. Suggest splitting into multiple commits if the change does more than one thing.

Tip: Explain why in the body. The diff already shows what changed.

68 Write a README for your project

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-readme-generator/

Write a README.md for my project.

Name: [NAME]

What it does: [DESCRIPTION]

Tech stack: [TECH]

How to install and run: [STEPS]

Main features: [LIST]

Include a short pitch, features, quick start, usage example, configuration, contributing and licence sections. Clear and scannable, with code blocks for commands.

Tip: Add a screenshot or GIF. It gets more stars than a longer README.

69 Write friendly release notes

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-release-notes-writer/

Write release notes for version [VERSION] of [PRODUCT] from these engineering changes:

[PASTE CHANGE LIST OR COMMITS]

Audience: [USERS / DEVELOPERS]

Tone: [FRIENDLY / FORMAL]

Group into "New", "Improved" and "Fixed", write each item in a benefit-first sentence and add a short headline. Leave out internal details users do not care about.

Tip: Lead with what users can now do, not the technical change behind it.

70 Write API documentation from code

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-api-docs-from-code/

Write API documentation for these endpoints:

[PASTE ROUTES, HANDLERS OR OPENAPI SNIPPET]

For each endpoint include purpose, method and path, parameters, request body, example request, example response, error codes and authentication requirements. Organise them into sections and start with a short "getting started" guide.

Tip: Generate a working curl example for each endpoint and test it before publishing.

71 Add helpful comments and docstrings to code

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-write-code-comments/

Add clear comments and docstrings to this code:

[PASTE CODE]

Explain why decisions were made, not what each line does. Document parameters, return values, side effects and exceptions. Flag anything confusing or risky without changing the logic, and list any names you would rename.

Tip: Ask it to explain why, not what; comments that repeat the code are noise.

72 Write a regex from a plain description

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-regex-builder/

Write a regular expression.

Language or tool: [e.g. JavaScript, Python, grep]

It must match: [EXAMPLES THAT SHOULD MATCH]

It must not match: [EXAMPLES THAT SHOULD NOT]

Task: [FIND / VALIDATE / REPLACE]

Give the regex, a part-by-part explanation, a test table of my examples with expected results and any edge cases to watch out for.

Tip: Always test with your real data. Regex answers from AI can be subtly wrong.

73 Convert code from one language to another

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-convert-code-language/

Convert this code from [SOURCE LANGUAGE] to [TARGET LANGUAGE]:

[PASTE CODE]

Use idiomatic patterns in the target language rather than a literal translation, keep the same behaviour, and list any differences in libraries, types, error handling or performance I should test. Add brief comments where the approach changed.

Tip: Ask for a list of behaviour differences so you know what to test after converting.

74 Python script to automate a task

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-python-script-helper/

Write a Python script to automate this task.

Task: [e.g. rename files by date, merge CSVs, send daily email]

Input: [FILE TYPE / LOCATION]

Output: [WHAT I WANT]

My skill level: [BEGINNER]

OS: [WINDOWS / MAC]

Keep the code simple and commented. Explain how to install what is needed, how to run it, and add a dry-run mode so it does not change anything on the first try.

Tip: Test on copies of your files first.

75 Get a beginner roadmap for learning to code

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-learn-programming-plan/

Create a learning plan for me to learn [LANGUAGE / SKILL].

Goal: [e.g. get a junior job, build an app]

Experience: [NONE / SOME]

Hours per week: [HOURS]

Give an 8-week plan with topics per week, one small project per week, free resources and a daily 20-minute practice routine. Tell me what not to waste time on.

Tip: Build small things from week one. Tutorials alone will not make you a developer.

76 Practise a system design interview

ChatGPT · Gemini · Claude · usepromptup.com/prompts/text-system-design-interview/

Act as a system design interviewer at a top tech company.

Level: [JUNIOR / MID / SENIOR]

Problem: [e.g. design a URL shortener]

Run the interview step by step. Start by asking for requirements, then let me propose the design. Push back with questions on scale, data storage, failure and trade-offs. At the end give feedback on structure, depth and what I missed.

Tip: Talk through your trade-offs. Interviewers care more about reasoning than the final diagram.

More free prompts at usepromptup.com/ai-coding-prompts/ · You may copy and adapt these prompts for your own use.