# AF-C1 · Four-choice setup picker

*Claude study notes · Module 1 · Make these four choices before your first prompt. Names on screen change often: match them to what your app shows.*

## 1 · Where?

| Use… | When… |
|---|---|
| **Chat** | A one-off question. Nothing to keep. |
| **Project** | Work you repeat with the same files and rules (templates, policies, a report you keep asking about). |
| **Desktop app** | You work with Claude all day, next to your other apps. |
| **Mobile app** | You're away from your desk. Quick questions only, nothing that needs careful checking. |
| **Claude Code** | Code, or many files in a folder on your computer. |

**Rule:** once and no files → chat · again and again → Project · code or a folder → Claude Code.

## 2 · Features?

| Switch on… | When… | Watch out |
|---|---|---|
| **File upload** | The answer is in your document. | The whole file goes, hidden sheets too (Module 6). |
| **Web search** | The answer must be current. | Open the sources and find the line (Module 3). |
| **Artifacts** | You want a reusable document, table or small tool. | Check it before you share it. |
| **Connectors** | The data lives in your mail, drive or calendar. | Claude reads live data: least access (Module 6). |
| **Extended thinking** | A hard, multi-step problem. | Slower. Not needed for quick tasks. |

**Rule:** on only for what the task needs.

## 3 · Model?

| Model | For | Rule |
|---|---|---|
| **Fast** | Bulk and simple tasks: short replies, sorting, quick rewrites | Go **down** for quick, repeated work. |
| **Balanced** | Most daily work | **Start here.** |
| **Deepest** | Hard reasoning, long or complex documents | Go **up** when the task is hard. |

**Test:** same task on two models. A tie → keep the faster, cheaper one.

## 4 · Context?

- □ Claude can see: this chat, files uploaded here, the Project's files and instructions, and what web search or connectors bring in. Not your screen, other files, or what's in your head.
- □ Upload the source instead of describing it.
- □ Put standing rules in a Project's instructions, not only at the top of a long chat.
- □ Signs a chat has gone on too long: it **repeats**, **ignores a rule**, or **mixes up names**.
- □ Fix: ask for a summary (who you are, the rules, what you've done) → new chat → paste the summary.

## Quick picker

| Task | Where | Features | Model | Context |
|---|---|---|---|---|
| | | | | |

*Our own checklist, written for this course. Practise it with exercise AF-E1.*

# AF-C2 · Six-part prompt card

*Claude study notes · Module 2 · Use it before any prompt that matters.*

| # | Part | Ask yourself | Example (HR warning letter) |
|---|---|---|---|
| 1 | **Role** | Who is writing, or what job is Claude doing? | "You are the HR officer at Demo Traders Pvt. Ltd." |
| 2 | **Goal** | What exactly do I want at the end? One sentence. | "A first written warning to Ram Sample for repeated late arrival." |
| 3 | **Context** | What facts does Claude not know? Upload or paste the source. | The attendance record, the policy clauses, the letter date, the reply deadline. |
| 4 | **Constraints** | What are the rules? Tone, length, must-include, must-not. Use numbers. | "Polite, no threats. Quote dates exactly. Under 200 words." |
| 5 | **Format** | What should it look like? Layout, sections, exact columns. | "Our letter layout. [Placeholders] for anything not given." |
| 6 | **Example** | Can I show one? An old letter, one finished row, a sample paragraph. | "Match the style of the attached older letter. Don't copy its facts." |

## Which parts matter most

| Task type | Strongest parts | Why |
|---|---|---|
| **Writing** (letters, notices, posts) | Role · Constraints · Example | The voice and the rules decide whether you can send it. |
| **Replies** (customers, staff) | Context · Constraints | What happened, what the policy allows, and what you're offering. |
| **Data** (tables from documents) | Format · Example | Exact columns and one finished row, or Claude picks its own. |
| **Analysis** (questions about data) | Goal · Context | The exact question and the data to answer it from. |

## Habits

- **Close the gaps that matter.** Every gap gets a guess.
- **Numbers beat adjectives.** "Under 200 words", not "concise". "By Tue 1 Dec", not "soon".
- **Rules come from the source.** Copy business rules from the policy, not from memory.
- **Facts you can work out, work out yourself.** Deadlines, totals and working days: use a calendar or a spreadsheet.
- **Say what to do when something is missing:** "write [ASK] instead of guessing", "write UNCLEAR".
- **Big task? Outline → draft one part → ask what's missing.**
- **Fix the prompt, not the answer.** Then save the prompt for next time.
- **Still check the output** with the five questions (AF-C3).

*Our own checklist, written for this course. Practise it with exercise AF-E2.*

# AF-C3 · Five-question output review

*Claude study notes · Module 3 · Run this on any Claude output before you use it.*

| # | Question | Check | One-line prompt to help |
|---|---|---|---|
| 1 | **Right?** | Recompute every number that matters outside the chat (calculator, spreadsheet, a colleague). Check names, dates and amounts against the source. If it matters, ask again in a fresh chat: two different answers mean the source is unclear. | "Show your working for each total." |
| 2 | **Missing?** | Compare with the source. Allergies, doses, dates, amounts, deadlines and conditions must never be dropped. | "List everything in the source that is not in your answer." |
| 3 | **Source?** | Every important claim should point to a line you can open. No line means it isn't a fact from your file. With web search, open the links and find the line yourself. | "For each point, quote the exact line from the file. If it isn't there, say 'not in the file'." |
| 4 | **Stakes?** | Who gets hurt if it's wrong? See the ladder below. Watch for loaded words about people (for example "unexplained", "concerns", "not a culture fit"). | "Rewrite this neutrally, using only facts stated in the source." |
| 5 | **Reader?** | Who reads it, and what must they do? Action first, plain words, right length. After a rewrite, re-check question 1: rewrites move dates and numbers. | "Rewrite for [reader]. Put what they must do first. Keep every date and amount the same." |

## Stakes ladder

| Level | Examples | Review |
|---|---|---|
| Low | Notes to yourself, first drafts, brainstorms | A quick look |
| Middle | Internal emails, meeting summaries | Read it fully |
| **High** | **Money · Health · Legal and contracts · Anything public · Decisions about people** (hiring, pay, grades, discipline) | **A person reviews every word before it is used. AI never makes the decision.** |

## Pick the format

- Comparing options or dates → **table**
- Asking someone to decide → **one-page memo**, decision at the top
- Presenting to a room → **slides**, one idea each
- Say the format in your prompt.

## Decide

| Stamp | When |
|---|---|
| **SHIP** | Passes all five questions. High-stakes work has had its human review. |
| **FIX** | A small, clear problem you can correct yourself and then check against the source. |
| **REJECT** | The problem is at the root: an invented source or fact, a safety error, the wrong task, or so many errors you can't trust the rest. Start again with a better source or prompt. |

*Our own checklist, written for this course. Practise it with exercise AF-E3.*

# AF-C4 · Delegation map worksheet

*Claude study notes · Module 4 · Use it on one task you do every week.*

## 1 · Pick: score the task (1–5 each, 5 = good fit)

| Question | 1 means | 5 means | Score |
|---|---|---|---|
| How often? | Once a year | Every day or every week, in volume | |
| How long? | A few minutes a week | Hours a week | |
| Mostly reading or writing? | Physical work, phone calls, judgement | Reading, comparing, drafting | |
| Mistakes caught early? | A mistake reaches a customer, a payment or the public | A person checks before anything is used | |
| Data safe to share? | Names with IDs, health details, passwords, salaries | No personal or confidential data | |
| **Total** (out of 25) | | | |

**Rule of thumb:** 18 or more, try it. A 1 or 2 on either of the last two rows: fix that first (remove the personal data, add a check step), or don't use AI for it.

## 2 · Map: list every step as it happens today

Write the steps before you add any AI. Then put each one in a column.

| Column | Put a step here when... |
|---|---|
| **Claude does** (a person spot-checks) | It is reading, copying, sorting or comparing, and a later step catches mistakes. |
| **Claude drafts, person approves** | It leaves the team, or someone will rely on it: emails, notices, reports, translations. |
| **Person only** | It moves money, decides something about a person, needs a signature, a relationship or a source only people have, or uses data you can't share. |

| # | Step | Column | Why |
|---|---|---|---|
| 1 | | | |
| 2 | | | |
| 3 | | | |
| 4 | | | |
| 5 | | | |
| 6 | | | |

**Never in the Claude columns:** approving or making payments, hiring, pay, grades or discipline decisions, signing, and anything you would be fired for pasting (see Module 6).

## 3 · Test: on past work, not next week

- □ Take 5–10 items you already finished, so you know the right answer.
- □ Time it by hand first. Then time the AI version **including your own checking**.
- □ For each item: **right**, **flagged**, **missed** or **false alarm**.
- □ Fix one or two steps (a missing check in the prompt, or sums moved to a spreadsheet). Run the same items again.
- □ Write down each change and what it did.

## 4 · Explain: one page for your manager

- □ Measured time before and after, on the same items.
- □ What it caught **and what it missed**.
- □ What Claude does **not** do.
- □ Who still reviews, and when.
- □ Next step: a short trial with a date to decide.
- □ No number you didn't measure. No "dramatically", no "revolutionary".

*Our own worksheet, written for this course. Practise it with exercise AF-E4.*

# AF-C5 · Project setup and monthly clean-up

*Claude study notes · Module 5 · Use it when you set up a Project for a recurring task, and once a month after that.*

## Set up (once)

| # | Step | Check |
|---|---|---|
| 1 | **One job** | The Project does one kind of task. HR letters, sales quotes and meeting notes are three Projects, not one. |
| 2 | **Name and description** | The name says the job and who it's for. The description says what it is for **and what it is not for** (for example "Not for payroll or personal records"). |
| 3 | **Owner and sharing** | One named owner changes the instructions. If your plan can share Projects, decide who can use it and who can edit it. |
| 4 | **Instructions in sections** | Built in an editor first, then pasted. Short lines under headings. |
| 5 | **Knowledge files** | 2–5 files about *how you work*: templates, policies, tone guide, one good example. Only the current version of each. |
| 6 | **Never-upload check** | No salary sheets, ID numbers, health notes, passwords or files about one particular person. Upload how you work, not who you work with. |
| 7 | **Three test requests** | One per kind of output, each with a short pass list. One of them leaves out a must-have detail on purpose. |
| 8 | **Run the tests** | All three pass, in fresh chats inside the Project. If one fails, fix the instruction or the file, not the answer, then re-run all three. |

## Instruction block template

```
[Project name] · v1.0 · [YYYY-MM] · Owner: [name, role]

WHO WE ARE          2 lines: the organisation and what you help with.
WHO READS IT        The reader, their language level, where they read it.
WHAT YOU DO         Draft only. A person checks, edits and approves. You never decide.
TONE                Name the tone guide. Specific limits, e.g. "under 200 words".
ALWAYS              Short rules, one per line. "Name the rule or clause you use."
NEVER               "Never invent a date, number, rule or reason." Plus your red lines.
FORMAT              Exactly what comes back, and nothing before or after it.
IF SOMETHING IS MISSING
  Must-haves:       List them for each kind of output.
  Missing must-have → do not write; ask me in one short list.
  Anything else     → leave a [placeholder].
```

**Specific, not vague:** "under 200 words", not "be concise" · "no threats, one offer of help", not "be polite" · "a table with these columns", not "well formatted".

## Monthly clean-up (10 minutes)

| | Step | Done |
|---|---|---|
| 1 | Remove outdated or duplicate files from knowledge. | ☐ |
| 2 | Re-read the instructions. Is every rule still true? | ☐ |
| 3 | Re-run the three test requests in fresh chats. All pass? | ☐ |
| 4 | Update the version line: number · date · what changed. | ☐ |

**Connectors** (live access to mail, drives and other apps): add one only when the task needs live data, give it the least access that works, and remove it when you are done. See Module 6.

*Our own checklist, written for this course. Practise it with exercise AF-E5.*

# AF-C6 · Before-you-paste data check

*Claude study notes · Module 6 · Run it before every paste or upload into any AI tool.*

## 1 · Is any of these in it? Then it doesn't go in.

| Never paste | Examples |
|---|---|
| **Personal IDs** next to a name | Citizenship, passport, PAN, Aadhaar, driving licence, staff or student ID numbers |
| **Health data** tied to a person | Diagnosis, test results, medicines, hospital numbers, with a name, phone or date of birth |
| **Credentials** | Passwords, one-time codes (OTPs), API keys, config files with secrets |
| **Confidential business data** | Unreleased results, contracts under NDA, salary sheets, customer lists |

Exception: only if your organisation has approved a specific tool for that exact data, in writing. If you're not sure, the answer is no.

**Pasted a real password or key by mistake?** Change it now, and tell whoever owns it. Deleting the chat doesn't undo it.

## 2 · Everything else: clean it first

- □ Work on a **copy**, never the original.
- □ **Remove** what the task doesn't need (addresses, phone numbers, ID and record numbers).
- □ **Replace** names with placeholders: [PATIENT], [STAFF 1], [CUSTOMER].
- □ **Shift dates** by the same amount (Day 0, Day 7…), or keep only the month or year.
- □ Keep the **key** (placeholder → real value) on your computer. It never goes in the chat.
- □ After Claude answers, put real values back **by hand** and re-check every date and number.

## 3 · Check the setup

- □ I'm using a tool my organisation approves, on the right account (work, not personal).
- □ I have read my plan's data and privacy settings once, on the settings page itself.
- □ Only the features this task needs are on:

| Feature | What leaves or what it reaches | Check |
|---|---|---|
| Web search | Your question goes out; pages can carry hidden text | No private details in the question. Read before you act. |
| Connectors | Your mail, drive or calendar | Least access. Remove it when you're done. |
| File upload | The whole file, including hidden sheets and comments | Open the file and check every sheet first. |
| Code execution | Runs code on your files | Only files you would upload anyway. |
| Skills (add-ons) | New instructions or tools | Only from sources you trust. |

**Prompt injection:** text in a page or file that tries to give the AI orders ("ignore your instructions…"). AI can read it for you; a person decides what to do.

## 4 · Five-rule AI policy

1. Approved tools only.
2. No IDs, health data, passwords or confidential files.
3. A person checks before anything goes out.
4. Say when AI helped.
5. Report mistakes; don't hide them.

## The law, in one line each (not legal advice)

- **Nepal**, Individual Privacy Act 2075: use personal information only with consent and for the purpose it was collected for.
- **India**, DPDP Act 2023: you need a lawful reason (usually consent) to use digital personal data, and you must keep it safe.
- **Health** (HIPAA-style thinking): treat health data tied to a person as protected; share the minimum; de-identify first.

Read the official texts (linked on the course page) for anything that matters.

*Our own checklist, written for this course. Practise it with exercise AF-E6.*

# AF-C7 · Five-step fix sequence

*Claude study notes · Module 7 · Use it when an output is weak, before you reword anything.*

| # | Step | Ask yourself | Common causes |
|---|---|---|---|
| 1 | **Gap** | What exactly is wrong, in one line? What is right and must stay? | "The date is wrong; everything else is right." If you can't write one line, look again. |
| 2 | **Input** | What did Claude actually receive? Read your own request and files as if you were Claude. | A detail never given (a date, a name, a number) · relative words ("next Friday") · a paste that was cut off · the wrong file |
| 3 | **Instruction** | Did anything say what to do in this case? | No reader or tone named · no format · no "if it's missing, ask me" · a vague rule ("be concise") · a rule buried in a long block |
| 4 | **Setup** | Right place, right files, right model, right size of job? | Asked outside the Project that holds the file · an old or duplicate file · a very long chat (start fresh with a summary) · too big for one answer (split it and ask for a count) |
| 5 | **One change** | Change one thing, re-run in a fresh chat, compare side by side. | Change the cause you found, not everything at once. If it works, you know why. |

**Most fixes are at step 2 or 3.** Check the model last, not first.

## Feedback that steers

Instead of "that's wrong, try again":

1. **What's wrong:** "Too stiff for a regular customer."
2. **What good looks like:** "Warmer and shorter."
3. **One example:** one line in the tone or format you want.
4. **What must not change:** "Keep both dates exactly."

## Make a recurring task faster

| Habit | How |
|---|---|
| Save the fix | Put the working rule in the Project instructions or a saved prompt, not in your head. |
| Keep 5 test cases | Real requests, each with a "pass if". Include every one that has failed before. Keep the sheet **next to** the Project, not as a knowledge file. |
| Re-test every change | Run all five in fresh chats after each change. |
| Keep a change log | Version · date · what changed · why · test results. |

## Row-count check (any table or list)

"How many rows are in the file, and how many are in your table?" If they differ, split the job into parts and ask for the count and a subtotal for each.

*Our own checklist, written for this course. Practise it with exercise AF-E7.*

# AF-C8 · Seven skills on one page + 30-day plan

*Claude study notes · Module 8 · The whole course on one page.*

## The seven skills

| Pin | Skill | The habit | Checklist |
|---|---|---|---|
| 1 | **Set up** | Before the first prompt, choose where (chat, Project, desktop, Claude Code), which features, which model, and what context Claude gets. | AF-C1 |
| 2 | **Ask** | Role, goal, context, constraints, format, example. Fix the prompt, not the answer. | AF-C2 |
| 3 | **Check** | Right? Missing? Source? Stakes? Reader? Then ship, fix or reject. | AF-C3 |
| 4 | **Delegate** | Map the steps: Claude does · Claude drafts, a person approves · person only. Test on last week's work. | AF-C4 |
| 5 | **Set it up once** | A Project for each recurring job: short instructions, files about how you work, three tests, a monthly clean-up. | AF-C5 |
| 6 | **Stay safe** | No IDs, health data tied to a person, passwords or confidential files. Redact first. Least access for connectors. | AF-C6 |
| 7 | **Fix** | Gap → Input → Instruction → Setup → One change. Keep a test set and a change log. | AF-C7 |

## The four Ds, in our words

Anthropic publishes a free AI Fluency course built on four Ds. This is how the seven skills sit under them; the wording is ours.

| D | In one line | Skills |
|---|---|---|
| **Delegation** | Deciding what to hand to AI, what to keep, and where to do it. | 1 Set up · 4 Delegate |
| **Description** | Telling AI clearly what you want, once or as a standing setup. | 2 Ask · 5 Set it up once |
| **Discernment** | Judging what comes back, and finding the cause when it's weak. | 3 Check · 7 Fix |
| **Diligence** | Staying responsible for what you share and what you send. | 6 Stay safe |

Free course: AI Fluency: Framework & Foundations (Anthropic Academy).

## 30-day plan: one D a week, one real task each week

| Week | Focus | Do this | Done |
|---|---|---|---|
| 1 · Days 1–7 | **Delegation** (AF-C1, AF-C4) | Pick one weekly task. Score it, map its steps into the three columns, and choose the setup. | ☐ |
| 2 · Days 8–14 | **Description** (AF-C2, AF-C5) | Write the six-part prompt for that task. Turn it into a Project with three test requests. | ☐ |
| 3 · Days 15–21 | **Discernment** (AF-C3, AF-C7) | Run the five-question check on every output this week. Fix one failure with the five steps and log it. | ☐ |
| 4 · Days 22–28 | **Diligence** (AF-C6) | Run the before-you-paste check on everything you share. Remove anything from the Project that is about one person. | ☐ |
| Days 29–30 | **Review** | Do the 20 practice scenarios (AF-E8). Re-read your change log. Write down minutes saved and errors caught, measured, not guessed. | ☐ |

*Our own checklist, written for this course. Practise it with exercise AF-E8.*