

DAY TWO

02 /10

Working in Claude.ai — *the workspace, not just the chat box*

Most people only ever use the message box. The learners you teach today will discover the rest of the workspace — Projects, Artifacts, memory, files — the features that turn Claude from a clever chat into a place you do real work.

BY THE END OF DAY 2, YOU WILL BE ABLE TO TEACH OTHERS TO —

- Navigate the conversation surface fluently
- Recognise and iterate on Artifacts
- Bring documents and images into a chat
- Set up a Project with instructions & knowledge
- Control memory, preferences & styles
- Choose the right surface for a given task

01 Why today matters

ORIENTATION

Yesterday’s learners can hold a good conversation with Claude. Today they learn that the conversation is only one room in the building. Claude.ai is a workspace: it remembers context you choose to keep, produces editable documents and apps beside the chat, accepts your files, and adapts to how you like to work. Skipping these features is the difference between using ten percent of the tool and most of it.

The throughline for today is “the right surface for the task.” A quick question belongs in a plain chat. A month-long body of work belongs in a Project. A document you’ll revise belongs in an Artifact. By the end of the day your learners should reach for the right one without thinking.

PREREQUISITES

Day 1’s mental model, plus a signed-in account. Interfaces evolve — teach the concepts (a project is shared context; an artifact is editable output) so the lesson survives a UI refresh. Where a menu has moved, the idea still maps.

02 The conversation surface

CORE CONCEPT

Start where everyone already is — the chat — and reveal the controls most people never touch. Each one removes a common frustration.

CONTROL	WHAT IT DOES	TEACH LEARNERS TO REACH FOR IT WHEN...
Model picker	Switches between Opus, Sonnet, Haiku.	...a task is unusually hard (Opus) or they want speed (Haiku).
Web search	Lets Claude look up current information.	...the question is about now — prices, news, recent releases.
Attach / upload	Adds documents, images, data to the message.	...the source material exists as a file, not text to paste.
Edit a message	Rewrites an earlier turn and re-runs from there.	...they misspoke — fix the prompt instead of piling on corrections.
Regenerate	Produces a fresh answer to the same prompt.	...the answer was fine but they want a different angle.

THE HIGHEST-LEVERAGE HABIT

Teach editing the prompt over arguing with the output. When a conversation goes sideways, the fastest recovery is almost always to go back, fix the original instruction, and re-run — not to send five follow-up corrections that clutter the context.

03 Projects — a room with shared context

CORE CONCEPT

A Project is a workspace that bundles related conversations together and gives them a shared foundation. Instead of re-explaining your situation at the start of every chat, you set the context once at the Project level and every conversation inside it inherits that context.

WHAT A PROJECT HOLDS

Instructions, knowledge, and its own chats.

A Project carries custom instructions (how Claude should behave for this work — role, tone, rules), project knowledge (documents and reference material every chat can draw on), and a set of conversations kept together. Its memory and history are scoped to the Project — separate from your other work.

3.1 When a Project earns its keep

- ✓ Ongoing work — a client account, a course you're writing, a product you maintain.
- ✓ Repeated context — anything where you'd otherwise paste the same background again and again.
- ✓ A consistent voice or set of rules — "always answer as our brand," "follow this style guide."
- ✓ Shared reference material — a handbook, a spec, a dataset the conversations keep returning to.

A one-off question does not need a Project. Teach the test: "Will I come back to this more than once?" If yes, make a Project.

KEY IDEA TO INSTALL

A Project is context you set once instead of re-typing forever. The instructions shape behaviour; the knowledge feeds every chat; the conversations stay organised together. It is the single biggest upgrade most people are missing.

04 Artifacts — output you can work on

CORE CONCEPT

When Claude produces something substantial and self-contained — a document, a piece of code, a working app, a structured report — it can place it in an Artifact: a panel beside the conversation. The chat stays for discussion; the Artifact holds the deliverable.

CLAUDE TENDS TO USE AN ARTIFACT FOR

- ✓ Documents and reports you'll keep or edit
- ✓ Code files and components
- ✓ Working interactive apps and visualisations
- ✓ Anything long enough that it shouldn't clutter the chat

WHY IT MATTERS FOR YOUR LEARNERS

- The deliverable is editable and iterable — "make the intro shorter" updates the Artifact in place.
- It is separable from the conversation — easy to copy out or keep.
- It makes Claude a building tool, not just an answering tool — the seed of Day 9.

"The chat is where you think. The Artifact is what you keep."

Teach the loop: ask → Claude drafts into an Artifact → you give plain-language feedback → the Artifact updates → repeat until it's right. This is the same collaboration loop from Day 1, now producing a durable thing.

05 Memory, preferences & styles

CORE CONCEPT

Day 1 established that conversations start fresh by default. Today's learners meet the features that deliberately carry things forward — and, just as importantly, how to control them.

5.1 Memory

Claude can build memory from your past conversations so it can be informed by your shared history rather than starting cold every time. Memory can be reviewed and edited, and it is kept separate per Project. You can ask Claude directly to remember or forget specific things.

5.2 Preferences & styles

Preferences capture standing instructions — formatting habits, tone, things to always or never do. Styles shape how Claude writes, so output matches a particular voice. Set these once and they apply across conversations, which keeps individual prompts short.

5.3 Incognito & control

For a conversation you don't want influencing memory or kept in history, an Incognito chat keeps it separate. Teach control as a feature, not a footnote: learners should know how to see what Claude remembers and how to change it.

TEACH CONSENT AND CLARITY

Memory is genuinely useful, but learners should understand what persists and where. Make a habit of showing the room how to review memory and how to start an Incognito chat — informed users trust the tool more, not less.

06

Bringing your material in

CORE CONCEPT

Claude works best on your material, not generic examples. Learners should be comfortable handing it real inputs.

- **Documents** — drop in a report, contract, or notes and ask Claude to summarise, restructure, critique, or extract from it.
- **Images** — Claude can see and reason about images: a diagram, a screenshot, a photo of a whiteboard, a chart.
- **Data & spreadsheets** — hand over tabular data for analysis, cleaning, or charting.
- **Project knowledge** — for material you'll reuse, put it in the Project once rather than re-uploading.

COACH THIS PHRASE

"Don't describe the document to Claude — give Claude the document." Most beginners paraphrase their source material and lose detail. Uploading the real thing is faster and far more accurate.

LAB 02 ~25 MIN

Build a real Project, end to end

Each learner builds a Project for a genuine, recurring piece of their own work — not a toy. By the end they will have set context once and produced a real deliverable as an Artifact.

- 1. Pick the work.** Each learner names something they return to often — a newsletter, a class, a client, a product area.
- 2. Create the Project** and write custom instructions: who Claude is for this work, the tone, any rules. Three or four sentences is plenty to start.
- 3. Add knowledge.** Upload one real reference document — a style guide, a brief, a spec — so every chat can draw on it.
- 4. Run a task** inside the Project that should produce an Artifact — e.g. “draft this week’s edition” or “write the onboarding checklist.” Confirm it appears in the side panel.
- 5. Iterate** on the Artifact with two rounds of plain-language feedback, then start a second chat in the same Project and notice it already knows the context.

a Project that would genuinely save the learner time next week, with at least one Artifact they’d actually use.

TEACHING NOTES

How to teach Day 2 well

OPEN WITH THIS

Ask: “How many of you re-explain your situation at the start of every chat?” Most hands go up. That shared pain is the doorway to Projects — you’re not selling a feature, you’re solving something they already feel.

PACE & EMPHASIS

Projects and Artifacts are the heart of the day — give them the most time and the Lab. Memory and files are quick, important context. Don’t get lost in menu locations; teach the concepts, since the UI shifts.

DISCUSSION PROMPTS

· What’s a piece of context you re-type constantly that should live in a Project?
· Which of your outputs would benefit from being an editable Artifact rather than a chat message?
· What would you put in custom instructions for your most common work?

COMMON MISCONCEPTIONS TO PRE-EMPT

“A Project is just a folder of chats.”

It’s more — shared instructions and knowledge that feed every chat. The organisation is a side benefit.

“Artifacts are just long messages.”

They’re editable, separable deliverables. Show the in-place update — that’s the “aha.”

“Memory means it records everything.”

It’s reviewable, editable, Project-scoped, and Incognito exists. Teach the controls.

IF YOU ONLY HAVE 30 MINUTES Teach “the right surface for the task,” build one Project together live (instructions + one document + one Artifact), and skip the deep dive on styles.

Day 2 Cheat Sheet

Project	A workspace with shared custom instructions, knowledge, and its own chats & memory — context set once.
Custom instructions	Project-level rules for how Claude should behave — role, tone, do's and don'ts.
Project knowledge	Reference documents every chat in the Project can draw on.
Artifact	Substantial, self-contained output shown beside the chat — editable and iterable in place.
Memory	Detail carried across past conversations; reviewable, editable, Project-scoped.
Preferences	Standing instructions — formatting and behaviour — applied across chats.
Styles	Reusable controls for how Claude writes, to match a voice.
Incognito	A chat kept out of history and memory.
Edit-and-rerun	Fix the original prompt instead of stacking corrections.

Check for understanding

Five questions. Learners should be able to answer all five before Day 3.

1. Name the three things a Project bundles together, and give the one-question test for whether to make one.
2. What is an Artifact, and what can you do with it that you can't do with an ordinary chat message?
3. A conversation has gone off the rails after a vague first prompt. What's the fastest recovery, and why?
4. Where does memory live relative to Projects, and how would you start a chat that doesn't affect it?
5. A learner keeps describing their report instead of uploading it. What's the coaching phrase, and why does it matter?

Answer notes — 1) Custom instructions, project knowledge, and its own chats/memory; test: "will I come back to this more than once?" 2) A substantial editable deliverable beside the chat; you can iterate on it in place and separate it from the conversation. 3) Edit the original prompt and re-run — it fixes the root cause without cluttering context. 4) Memory is scoped per Project; use an Incognito chat to keep a conversation out of history and memory. 5) "Give Claude the document, don't describe it" — paraphrasing loses detail and accuracy.

Day 2 in five lines

- Claude.ai is a workspace, not just a chat box — match the surface to the task.
- The conversation surface has controls most people miss; editing-and-rerunning beats arguing with output.
- A Project sets context once — custom instructions, shared knowledge, and organised chats.
- Artifacts turn substantial output into editable, separable deliverables you iterate on in place.
- Memory, preferences and styles carry your way of working forward — and you stay in control of all of it.

TOMORROW — DAY 3 → **The Art of the Prompt — getting consistently great results**

NOTES & DISCLAIMER

Independent resource. The Field Guide for Humans is an independent, unofficial educational resource produced and published by Kirevra Press, an imprint of Kyvara Pty Ltd (ACN 697 072 049). It is not affiliated with, endorsed by, sponsored by, or officially connected to Anthropic, PBC (“Anthropic”), the maker of Claude.

Trademarks. “Anthropic,” “Claude,” “Claude Code,” and related names and marks are trademarks of Anthropic, PBC, used here for identification and descriptive purposes only. Their use does not imply any endorsement by or affiliation with Anthropic.

Accuracy & currency. The Guide describes third-party products that change frequently. All product details — features, model names, capabilities, commands, interfaces, and pricing — are believed accurate as of the 2026 edition but are provided “as is,” without warranty of any kind, and are subject to change without notice. Always confirm current information against Anthropic’s official documentation before relying on it.

No professional advice; no guaranteed results. The Guide is provided for general educational and informational purposes only. It does not constitute professional, legal, financial, or technical advice. No particular outcome, result, or level of proficiency is promised or guaranteed; results depend on the individual.

Copyright. The Field Guide for Humans — its text, design, structure, and original illustrations — is © 2026 Kyvara Pty Ltd. All rights reserved.