
NRT AGENTIC AI · WEEK 5 · FRIDAY 16 OCTOBER 2026

Designing an agent for a task you already do

From your five homework answers to a workflow Claude
can run again

NRT Hot Topics: Agentic AI · Pattern design I · Jie Hou

**Scan & Share your
homework answers!
Find the Practice Demo**



Today's plan

THIS AFTERNOON

Intro to agent design

Student presentations

3 minutes each

Live demo: Bob's task

Prompt, test, verify in Claude Code

Work together · 2:30–4:30

Your own task, one step at a time

NRT Week 5 · Worked Demo

From a Running Example to a Skill and an Agent

Original Week 5 homework ("Automating a Task With AI"):

https://docs.google.com/document/d/1JNPICLO_eCSKT7jY-kVb1r3eA2L8DkCz/edit?usp=sharing

Research topic suggested by Dr. Robert Stein

Objective: Searching NASA ADS, downloading papers, and summarizing them with Claude Code

This document applies the Week 5 homework ("Automating a Task With AI") to one real task: finding and reading papers on **magneto-hydrodynamic simulations of sunspot formation (2016–2026)** in the NASA Astrophysics Data System (ADS, <https://ui.adsabs.harvard.edu>).

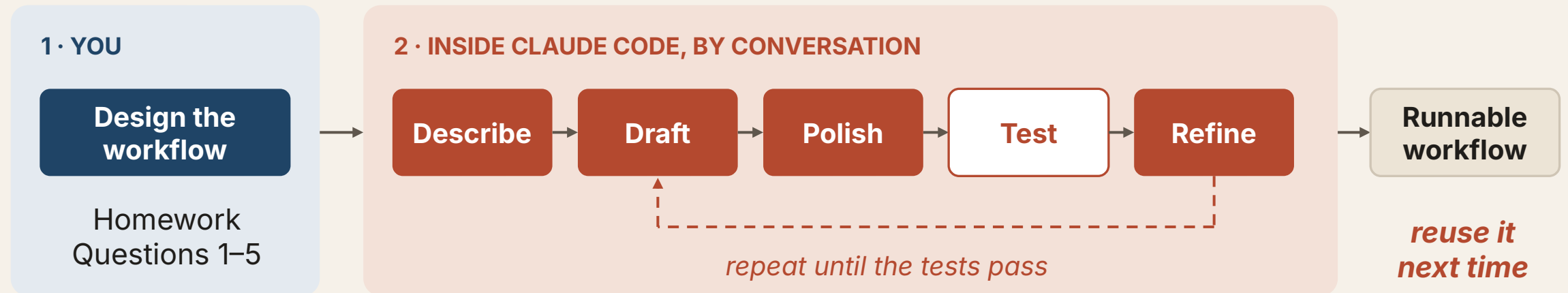
It answers the five homework questions for that task. Then it walks through the full progression we will demo in class: running prompts by hand, testing, verifying, packaging the work as a **skill**, wrapping the skill in a **subagent**, and finally installing it in **Claude Cowork** on the desktop.

<https://forms.gle/rNY8zydkMairSj5o9>

THE IDEA BEHIND THE HOMEWORK

Design the workflow first.

The agent is only one of the boxes.



Most steps in a real task belong to you, a script, or a check. Today we decide which box is which, and how you would know if a box got it wrong.

"This does not need an agent" is a perfectly good conclusion.

WORKED EXAMPLE · BOB'S TASK

Bob's task: find and read ten years of papers

BOB'S GOAL

An introduction for a journal paper on **MHD simulations of sunspot formation**, tracing earlier theories and simulations.

HOW HE DOES IT TODAY

Searches, downloads and reads by hand, or with Claude chat.

TODAY'S DEMO

The first half: **search NASA ADS for 2016–2026**, download the free PDFs, summarize the core papers.

NASA ADS: ui.adsabs.harvard.edu
free to search, with an API for scripts

Recommendations	Search examples
author: "Lee, Eve"	refereed: property:refereed
first author: first_author: "Mushotzky, Richard"	astronomy: collection:astronomy
abstract + title: abs: "dark energy"	exact search: -body: "intracluster medium"
year: year:2000	institution: inst:CfA
year range: year:2000-2005	author count: author_count:[1 TO 10]
full text: full: "super Earth"	record type: doctype:software
publication: bibstem:ApJ	newly ingested: entdate:[NOW-7DAYS TO NOW]
citations: citations(abstract:JWST)	eprint: property:"eprint_openaccess"

Writing the LaTeX introduction comes later, **built on these summaries**.

PATTERN 1 · WHO DOES EACH STEP

Every step goes to one of four owners

You

Steps that need your judgment: what the topic really is, what matters, what to keep.

IN BOB'S TASK

Choose the keywords.
Pick the core papers.

A script

Steps that give the same output for the same input, every time. **No AI needed.**

IN BOB'S TASK

Query the ADS API,
save the CSV,
download free PDFs.

Claude

Steps that need judgment over text:
**reading, summarizing,
drafting.**

IN BOB'S TASK

Summarize each paper,
with a page number for
every claim.

A checker

A separate, read-only check.
**An agent checking its
own work is not
independent.**

IN BOB'S TASK

Compare each claim in
a summary with the
PDF page.

Rule of thumb: give each step **the least machinery that does it reliably.**

PATTERN 2 · HOW MUCH TO BUILD

Package only what you will reuse

1

Prompt

One request, typed into Claude.

WORTH IT WHEN

You do it once, or you are still working out the steps.

2

Skill

A folder: SKILL.md with steps and checks, plus scripts. **Claude loads it when a request matches.**

WORTH IT WHEN

The same steps repeat with new inputs. Bob: a new topic every few weeks.

3

Subagent

A separate Claude with its own tools and context. **It hands back a short report.**

WORTH IT WHEN

Reading 20 PDFs would flood the main chat, or you need a separate check.

4

Share

Bundle the skill and agents as a plugin, or upload the skill to Cowork desktop.

WORTH IT WHEN

Someone else will run it, or **you want it without a terminal.**

Less to build, less to break

More to build, more to break

Climb one rung only when **the rung below stops being enough.**

PATTERN 3 · WHERE IT RUNS

Pick the room by what the agent must touch

Chat

IT CAN TOUCH

Its own sandbox. Nothing on your machine.

USE IT WHEN

One-off drafting and thinking, or a task that is really one good prompt.

Cowork

IT CAN TOUCH

Folders you grant it in the desktop app. No terminal.

USE IT WHEN

Many files and documents: tidy, summarize, reorganize.

Claude Code

IT CAN TOUCH

Your shell, scripts, environment variables and project folder.

USE IT WHEN

Repeatable pipelines that run code, call an API or need a secret.

BOB'S TASK

It needs an API token, scripts, and files passed between steps, so it runs in **Claude Code**. The reading and summarizing half also works in Cowork once the PDFs sit in a folder.

YOUR HOMEWORK, READ AS A DESIGN

The five homework questions are the design spec

HOMEWORK QUESTION		WHAT IT DECIDES
1	What do you do? →	The goal and what “done” means , so the agent knows when to stop.
2	How do you do it now? →	The steps of the pipeline , including the dull copy-and-paste ones.
3	Draw it →	The loops : where the time goes and where a check belongs.
4	In, out, who, where →	An owner and a room for every step : you, a script, Claude or a checker.
5	What could go wrong? →	The checks and the never-touch list that make the output trustworthy.

Claude Code drafts the workflow; **you check each decision.**

WORKED EXAMPLE · BOB'S PAPER SEARCH IN NASA ADS

Bob's task became eight steps with two loops



3–5 hours

by hand, every 2–4 weeks,
with a new topic each time

101 → 48 → 39

records, refereed papers and free
PDFs in the real run on 14 Oct

5 boxes, 0 loops

in Bob's first sketch. Answering Q3
and Q4 found the loops.

PATTERN 4 · HOW YOU KNOW IT WORKED

Every quiet failure needs a one-minute check

IF IT GETS THIS WRONG...

The query quietly misses key papers, e.g. a missing synonym like "MHD".

A "PDF" is really an error or cookie page saved as .pdf.

A summary invents or changes a number.

...YOU SPOT IT BY

Naming 3 papers you know should appear. **All 3 must be in the CSV.**

Checking that **every file starts with %PDF-**, then opening 3 at random.

Requiring **a page number for every number**, then looking up 2 of them.

NEVER TOUCH

The API token, the downloaded PDFs, and paywalls. A written "never" is only a request; **folder permissions and scripts enforce it.**

BEFORE THE DEMO · ONE-TIME SETUP

Claude reaches ADS through your API token

1 · GET A TOKEN

ADS → Account → Settings → **API Token**
→ Generate a new key.

2 · SAVE IT IN A FILE

In your project folder, make a file
ads_token.txt with the token as its only
line. Any text editor works.

3 · TELL CLAUDE THE RULE

*"My ADS token is in ads_token.txt.
Only scripts may read it. Never print
it or show it in the chat."*

If the token ever appears in the chat, generate a
new key.

NASA ADS: ui.adsabs.harvard.edu

API Token

No token currently exists

Generate a new key

This API token allows you to access ADS data programmatically. For instance, to fetch the first few bibcodes for the query "star", make the following request:

```
curl -H 'Authorization: Bearer [YOUR TOKEN HERE]' 'https://api.adsabs.harvard.edu/v1/search/query?q=star&fl=bibcode'
```

(If you've generated a token, you can copy-paste the preceding line directly into your terminal)

Documentation on how to use the API is available on the [ADS API Github repo](#).

Make sure to keep your API key secret to protect it from abuse.

If your key has been exposed publicly (say, by accidentally being committed to a Github repo) you can generate a new one by clicking on the button above.

ADS account settings: the API Token page

You do not need to figure out the next steps. **Claude will advise you!**

TODAY'S DEMO, STEP BY STEP

Prompt, test, verify, then package

MAKE IT WORK

1 Prompt by hand

Paste prompts into Claude Code, one step at a time.

2 Test

Run test cases. Include one that should fail.

3 Verify yourself

About ten minutes of your own checks.

MAKE IT REUSABLE

4 Skill

SKILL.md plus scripts, for the steps that passed.

5 Subagents

A worker, and a read-only checker.

6 Share

Upload to Cowork, or bundle a plugin.

Only steps that pass **both the tests and your own check** get packaged.

Bring your five answers and one real input file from your task.

YOUR TURN

Share your task, then build it with Claude

PRESENT YOUR TASK · 3 MINUTES

- 1 Your task**
and what “done” looks like
- 2 Your steps**
and where they loop back
- 3 The steps you keep**
and why they stay with you
- 4 One quiet failure**
and the one-minute check that catches it

Small and slightly boring is better than ambitious.

WHILE YOU LISTEN, ASK

Which step would you keep?

Where does their judgment matter?

Script, Claude, or nothing?

Does any step really need an agent?

How would they notice?

Is the check independent and quick?

Pick one: your own task or Bob's

OPTION A

Your own task

- 1 Open Claude Code in a new, empty project folder.
- 2 Paste your five homework answers.
- 3 Ask Claude to draft the workflow. Decide which steps stay with you.

OPTION B

Bob's paper search

- 1 Scan the QR code for the demo document.
- 2 Save your token in ads_token.txt (slide 11).
- 3 Paste the demo prompts one at a time, testing each.



PRACTICE DEMO AND HOMEWORK FORM

Either way: prompt by hand, test, verify, then package.

<https://forms.gle/rNY8zydkMairSj5o9>