Connecting Your Computer to AI

MGMT 675: Generative AI for Finance

Kerry Back

What is Claude Code?

Three Ways Claude Runs Code

Chat (Analysis)

- Sandboxed Python in the browser
- No local file access
- Must upload data manually
- State lost between chats
- Good for quick calculations

Cowork

- Runs in a remote VM
- Can install packages
- Files synced to/from VM
- VM is temporary
- Higher token cost

Code

- Runs on **your machine**
- Full local file access
- Persistent state
- Version control built in
- Most token-efficient

Code mode gives Claude direct access to your machine—no sandbox, no VM, no upload step.

Code Mode: Advantages

- **No sandboxing**—access databases, APIs, local files
- **Install anything**—pip install, npm, system packages
- **Large datasets**—no upload limits
- **Persistent state**—pick up where you left off
- **Version control**—code saved in your repo
- **Production path**—code is ready to deploy

Using Code Mode

Getting Started with Code Mode

Switch to the **Code** tab in Claude Desktop → choose a project folder.

How it differs from Cowork

- Runs code **directly on your machine** (no VM overhead)
- More token-efficient — fewer planning/orchestration tokens
- **Requires Python** (or R, Node, etc.) installed locally
- Coding-oriented interface, but works for data analysis

Example

"Read the file sp500_monthly.csv. Run a Fama–French regression for each of the 10 industry portfolios. Save the results to a table in results.xlsx and create a plot of the estimated betas with confidence intervals."

Setup Requirements

One-time setup

1. Install Claude Desktop (already done from Step 3)
2. Install Python: `python.org/downloads`
3. Recommended: install key packages

```
pip install pandas numpy matplotlib statsmodels openpyxl
```

Or let Claude install them for you — it will run `pip install` as needed.

Token comparison

Surface	Relative token cost	Why
Chat / Artifacts	Low	No code execution
Code	Medium	Direct execution, lean context

Comparison

Which Tool When?

Task	Chat	Artifacts	Cowork	Code
Explain WACC formula	✓			
Draft an investment memo	✓	✓		
Interactive DCF calculator		✓		
Analyze CSV, create Excel			✓	✓
Run Fama–French regressions			✓	✓
Organize 50 PDF 10-Ks			✓	
Fetch live data from APIs				✓

Rule of thumb: Start with Chat. If you need a visual, use Artifacts. If you need file I/O and code execution, use Code. Save Cowork for complex multi-file tasks.

Slash Commands and Subagents

Slash Commands

In the **Code tab**, type / to see available commands. These also work in the Claude Code CLI and VS Code extension. They are **not available** in the Chat or Cowork tabs.

Command	What It Does
`/clear`	Clear conversation history and start fresh
`/compact`	Compress the conversation to free up context window space
`/cost`	Show token usage for the current session
`/model`	Switch between models (Sonnet, Opus)
`/review`	Ask Claude to review your code for issues
`/init`	Create a `CLAUDE.md` project memory file
`/agents`	Create, browse, or run custom subagents

A slash command is just a prompt—it loads instructions into Claude's context. Other tools have equivalents: Copilot's `/fix`, Cursor's `/edit`, etc.

Subagents

A **subagent** is a specialized AI assistant with its own context window and system prompt. Create one with `/agents` in the Code tab.

What a Subagent Provides

- Dedicated context window (no distraction from other tasks)
- Custom system prompt with domain knowledge and standards
- Restricted tool set (e.g., read-only)
- Can run in parallel with other agents

When to Use One

- You repeat the same type of analysis often
- The task has specific standards Claude should always follow
- You want to delegate a subtask while Claude works on something else

Start by using Claude directly. When you notice you're repeating the same instructions, that's the signal to create a subagent.

AI Coding Tools: A Common Architecture

All AI coding tools share the same fundamental pattern: an **LLM** with **code execution** ability and **file access**. The differences are in UI, models, and pricing—not in the architecture.

	Interface	Model(s)	Runs Where
Claude Code	Terminal / Desktop / VS Code	Claude	Your machine
OpenAI Codex	Web / CLI	GPT / o3	Cloud sandbox
Cursor	Desktop editor	Multiple	Your machine
GitHub Copilot	VS Code / JetBrains	GPT / Claude	Your machine
Google Jules	Web	Gemini	Cloud VM

Learn the pattern once, and you can move between tools as they evolve.

Managing Usage

Managing Your Token Budget

All Claude products share the **same usage pool**. Usage resets every 5 hours.

Strategies for the Pro plan ($20/month)

1. **Use Chat for questions** — "Explain Jensen's alpha" costs very few tokens
2. **Use Artifacts for quick visuals** — paste data, get a chart
3. **Use Code for analysis assignments** — lean and efficient
4. **Reserve Cowork for heavy-lift tasks** — multi-file, complex output
5. **Clear context between tasks** — type `/clear` in Code sessions
6. **Bundle related work** — don't start a new session for every subtask

If you hit your limit: Wait for the 5-hour reset, or fall back to Chat + Artifacts (much lower token cost) to keep working.

Getting Started

Your Setup Checklist

1. **Create a Claude account** at `claude.ai`
2. **Subscribe to Pro** ($20/month) via the account settings
3. **Download Claude Desktop** from `claude.com/download`
   - Windows (x64) or Mac
4. **Install Python** from `python.org/downloads`
   - Check "Add Python to PATH" during installation
5. **Try each mode:**
   - Chat tab: ask a finance question
   - Chat tab: paste data and request an artifact
   - Code tab: point at a folder with a CSV, ask for analysis

Exercises

Exercise: Loan Portfolio Analysis

- Download loans.zip into your project folder.
- Extract the files. You will have a loan tape (CSV), a collateral appraisal report (PDF), and a policy exceptions memo (docx).
- In Claude Desktop, go to the Code tab and select your project folder.
- Ask Claude to load the loan tape, compute the weighted average interest rate and LTV ratio, and flag any loans that exceed the policy limits described in the memo.
- Ask it to cross-reference the collateral values from the appraisal report and produce a summary of loans where the appraised value has declined.

Download Data for Exercise

Exercise: Fetching Data with `curl`

Claude Code can run terminal commands. Ask it to fetch financial data using `curl` and then analyze the results.

- Get a free API key from Alpha Vantage and/or Financial Modeling Prep
- Ask Claude to use `curl` to fetch Apple's monthly stock prices from Alpha Vantage. Ask it to parse the JSON response and save the data to a CSV file.
- Ask Claude to use `curl` to fetch Apple's income statement from Financial Modeling Prep. Ask it to extract revenue and net income for the last 5 years and display them in a table.

Key point: These API calls **fail** in Chat and Cowork because the sandbox blocks internet access. Code mode runs on your machine, so it has full network access.

Exercise: Fetching Data with Python

- Ask Claude to write a Python script that uses the `requests` library to fetch Apple's quarterly income statement from Financial Modeling Prep. Ask it to compute revenue growth rates and create a bar chart.
- Ask Claude to write a Python script that uses the Alpha Vantage API to get daily prices for three stocks of your choice over the past year. Ask it to compute cumulative returns and plot them on a single chart.
- Ask Claude to combine data from both sources: fetch Apple's income statement from FMP and its stock price history from Alpha Vantage, then create a dual-axis chart showing revenue growth alongside stock returns.

Tip: Tell Claude your API key, or better yet, ask Claude to store it in an environment variable so it isn't hardcoded in scripts.

Exercise: Aggregating and Filtering Files

- Download aggregation.zip into your project folder and extract it.
- You will have several Excel workbooks, each containing a table with similar data. Some tables are missing columns and column names vary across the files.
- Ask Claude to read all the workbooks, reconcile the varying column names, combine everything into a single table (including all columns), and save the result as a new Excel file.
- Ask Claude to filter the combined table to rows matching criteria of your choice and produce a summary with descriptive statistics.

Download Data for Exercise