AI-Powered Coding and Analytics with Claude

A Practical Guide to Using Claude Desktop, Claude Code, and MCP Integrations for Academic and Professional Work

Chenhao Zhou

Department of Supply Chain Management
Rutgers Business School

Spring 2026

This guideline is part of a series designed to support students' professional development beyond the classroom, fostering self-directed learning, technical fluency, and career readiness.

Contents

1 Introduction: Why AI-Assisted Coding Matters

Artificial intelligence is transforming how professionals write code, analyze data, and manage daily workflows. Tools like Anthropic's Claude are not replacing human judgment; they are amplifying it. A business student who can effectively leverage AI-assisted coding tools will be dramatically more productive in analytics, reporting, and technical problem-solving than one who cannot. This guide introduces you to two primary ways of working with Claude: the Claude Desktop application (a graphical chat interface) and Claude Code (a command-line tool that operates directly in your terminal).

The key insight is that AI tools are most powerful when used by someone who understands the domain. A well-crafted prompt from a student who understands supply chain analytics will produce dramatically better results than a vague prompt from someone without that context. Your business education is what makes these tools truly valuable in your hands.

2 Getting Started with Claude Desktop

2.1 What is Claude Desktop?

Claude Desktop is Anthropic's official graphical application available for macOS and Windows. It provides a conversational interface where you can ask questions, upload documents, analyze data, generate code, draft reports, and much more. The application supports file uploads (PDFs, images, CSVs, and code files), making it ideal for tasks that involve working with existing materials.

2.2 Installation and Setup

Visit https://claude.ai and create a free account, or download the Claude Desktop application from https://claude.ai/download. The free tier provides access to Claude Sonnet, which is sufficient for most student tasks. The Pro subscription ($20/month) provides access to Claude Opus, extended thinking, higher usage limits, and additional features including Claude Code access.

2.3 Core Features

Claude Desktop supports several powerful features beyond simple conversation. You can upload files directly into the chat for analysis: drag a CSV file and ask Claude to perform statistical analysis, upload a PDF and ask for a summary, or share code and ask for debugging help. The application also supports **Projects**, which allow you to organize conversations and uploaded files around a specific topic or assignment. Claude can generate **artifacts** such as code files, documents, interactive visualizations, and web pages directly within the conversation.

Pro Tip

Use Claude's Projects feature to organize your work by course or assignment. Upload your syllabus and relevant readings at the start of the semester, and Claude will have that context available throughout your conversations.

3 Getting Started with Claude Code (CLI)

3.1 What is Claude Code?

Claude Code is a command-line interface (CLI) tool that brings Claude directly into your terminal. Unlike the Desktop app, Claude Code operates in the context of your local file system and can read, write, and execute code on your machine. It is designed for developers and analysts who prefer working in the terminal and want Claude to interact directly with their codebases and data files. Claude Code can understand your entire project structure, run scripts, execute git commands, and perform multi-step tasks autonomously.

3.2 Installation

Claude Code requires Node.js (version 18 or higher) and a Claude Pro or Team subscription. Install it globally using npm:

```
npm install -g @anthropic-ai/claude-code
```

After installation, navigate to any project directory and type `claude` to start an interactive session. Claude Code will automatically detect your project structure and provide context-aware assistance. On first launch, you will be prompted to authenticate with your Anthropic account. Full documentation is available at https://code.claude.com.

3.3 Key Commands and Navigation

Once inside a Claude Code session, you interact with Claude using natural language. Simply type what you need, and Claude will determine whether to read files, write code, run commands, or provide explanations. Several slash commands provide additional control:

Command	Description
`/help`	Display available commands and keyboard shortcuts
`/model`	Switch between Claude models (Haiku, Sonnet, Opus)
`/compact`	Summarize conversation to free context window space
`/config`	View and modify configuration settings
`/rewind`	Revert to a previous checkpoint in the session
`/clear`	Clear the current conversation history
`Ctrl+R`	Search through previous prompts in the session
`Esc` (2x)	Rewind to the last checkpoint

Table 1: Essential Claude Code commands.

3.4 Model Selection

Claude Code supports multiple models with different cost-performance tradeoffs. **Haiku** is the fastest and most affordable model, suitable for simple queries, file searches, and quick tasks. **Sonnet** is the default workhorse for daily development, handling most coding, analysis, and writing tasks effectively. **Opus** is the most capable model, reserved for complex reasoning tasks such as architectural decisions, intricate debugging, and deep analytical work. Use `/model` to switch between them during a session.

3.5 Checkpoints and Safety

Claude Code automatically saves checkpoints before making changes to your files. If a modification produces undesirable results, you can instantly revert by pressing `Esc` twice or using the `/rewind` command. This safety mechanism allows you to experiment confidently, knowing you can always return to a prior state.

4 Effective Prompting Techniques

The quality of Claude's output depends heavily on the quality of your prompt. Effective prompting is a skill that improves with practice. The following principles will help you get consistently strong results.

4.1 Be Specific About Context and Expectations

Rather than asking "help me with my data analysis," provide Claude with the specific context: what data you are working with, what question you are trying to answer, what tools or languages you prefer, and what format you want the output in. For example:

> "I have a CSV file with monthly sales data for 50 stores over 3 years. Using Python and pandas, calculate the year-over-year growth rate for each store and create a heatmap visualization showing seasonal patterns."

4.2 Break Complex Tasks into Steps

For multi-part projects, provide Claude with a step-by-step plan rather than a single monolithic request. This approach produces better results because Claude can focus its reasoning on one step at a time, and you can verify each step before proceeding. You can also ask Claude to first outline its approach before implementing, which gives you an opportunity to redirect if needed.

4.3 Provide Examples and Constraints

When you have a specific output format in mind, show Claude an example. If you want a table in a particular style, paste a sample. If you want code that follows certain conventions, state those conventions explicitly. Similarly, state constraints clearly: "Do not use any external libraries beyond pandas and matplotlib" or "The report should be no longer than two pages."

4.4 Iterate and Refine

AI-assisted work is inherently iterative. Your first prompt may not produce the perfect result, and that is expected. Review Claude's output, identify what needs improvement, and provide specific feedback. Phrases like "The code works but the visualization needs a legend and axis labels" are far more effective than "make it better."

Golden Rule

The more precisely you define success, the more likely Claude is to achieve it. Ambiguous prompts produce ambiguous results.

5 Key Capabilities and Use Cases

5.1 Coding and Debugging

Claude can write, explain, debug, and refactor code in virtually any programming language. For business students, the most relevant applications include Python scripts for data analysis, R code for statistical modeling, SQL queries for database management, VBA macros for Excel automation, and Stata `.do` files for econometric analysis. When debugging, paste the error message along with the relevant code, and Claude will identify the issue and suggest a fix.

5.2 Data Analysis and Visualization

Upload a dataset (CSV, Excel, or describe its structure) and ask Claude to perform exploratory data analysis, compute summary statistics, run regressions, create visualizations, or build predictive models. Claude can generate publication-quality charts using matplotlib, seaborn, or plotly, and can format output tables to meet academic journal standards.

5.3 Writing and Editing

Claude excels at drafting reports, executive summaries, literature reviews, and professional correspondence. It can adapt its writing style to match academic conventions, business memo formats, or casual communication. For editing tasks, ask Claude to review your draft for clarity, grammar, logical flow, or to suggest structural improvements.

5.4 Research and Learning

Use Claude as a study partner. Ask it to explain complex concepts, generate practice problems, quiz you on course material, or walk through the derivation of a formula step by step. Claude can also help you understand academic papers by summarizing their methodology and findings in accessible language.

6 MCP Servers and Integrations

The Model Context Protocol (MCP) is a standardized protocol that extends Claude's capabilities by connecting it to external tools and services. Think of MCP servers as plugins that give Claude access to additional resources such as databases, APIs, file systems, and web services. The MCP ecosystem has grown rapidly, with over 300 integrations available as of early 2026.

6.1 Notable MCP Integrations for Students

Integration	Capability
GitHub	Manage repositories, review pull requests, automate git workflows
Google Drive	Search, read, and organize files in your Google Drive
Gmail	Draft, search, and manage emails programmatically
Google Calendar	Create and manage calendar events from within Claude
Slack	Send messages and manage channels for team projects
Scholar Gateway	Search peer-reviewed academic literature
Plotly MCP	Generate interactive data visualizations
PostgreSQL / SQLite	Query and manage databases directly

Table 2: Selected MCP integrations relevant to business students.

6.2 Adding an MCP Server

To add an MCP server to your Claude Code environment, use the `claude mcp add` command:

```
claude mcp add github-server - npx @modelcontextprotocol/server-github
```

Claude Code will then have access to the tools provided by that server in all subsequent sessions within the configured scope. You can run `claude mcp add` without arguments to launch a step-by-step interactive setup interface.

7 Practical Workflows for Students

7.1 Homework and Assignment Assistance

When working on quantitative assignments, use Claude to *check your work* rather than to do it for you. Solve the problem first, then ask Claude to verify your approach and identify any errors. This workflow builds your skills while leveraging AI as a quality-assurance mechanism. For coding assignments, write your initial solution and then ask Claude to review it for edge cases, efficiency improvements, and coding best practices.

7.2 Group Project Coordination

Claude can serve as a project management assistant for group work. Use it to draft project timelines, create task breakdowns, generate meeting agendas, and consolidate individual contributions into a cohesive final deliverable. When team members have written different sections of a report, Claude can harmonize the writing style and ensure consistent formatting throughout.

7.3 Career Preparation

Claude can help you prepare cover letters tailored to specific job postings, practice behavioral interview questions with follow-up probing, analyze job descriptions to identify the key skills you should emphasize, and draft networking emails. Upload a job posting and your resume together, and ask Claude to identify gaps and suggest how to address them in your cover letter.

8 Ethical Considerations and Academic Integrity

AI tools are powerful, but they come with important responsibilities. Always adhere to your institution's academic integrity policies regarding AI use. As a general principle, you should be able to explain and defend every element of the work you submit. If Claude generated code or analysis for you, make sure you understand every line. If you used Claude to help draft text, make sure the ideas and arguments are genuinely yours.

Transparency is essential. When your instructor permits AI use, disclose it. When your instructor prohibits it, respect that boundary. The goal of AI-assisted work is not to circumvent learning but to accelerate it. Used responsibly, these tools make you a more capable professional. Used irresponsibly, they undermine the very competencies you are trying to build.

Academic Integrity Principle

AI should amplify your understanding, not replace it. If you cannot explain how the output was generated and why it is correct, you have not used the tool effectively.