

Guiding Copilot with context

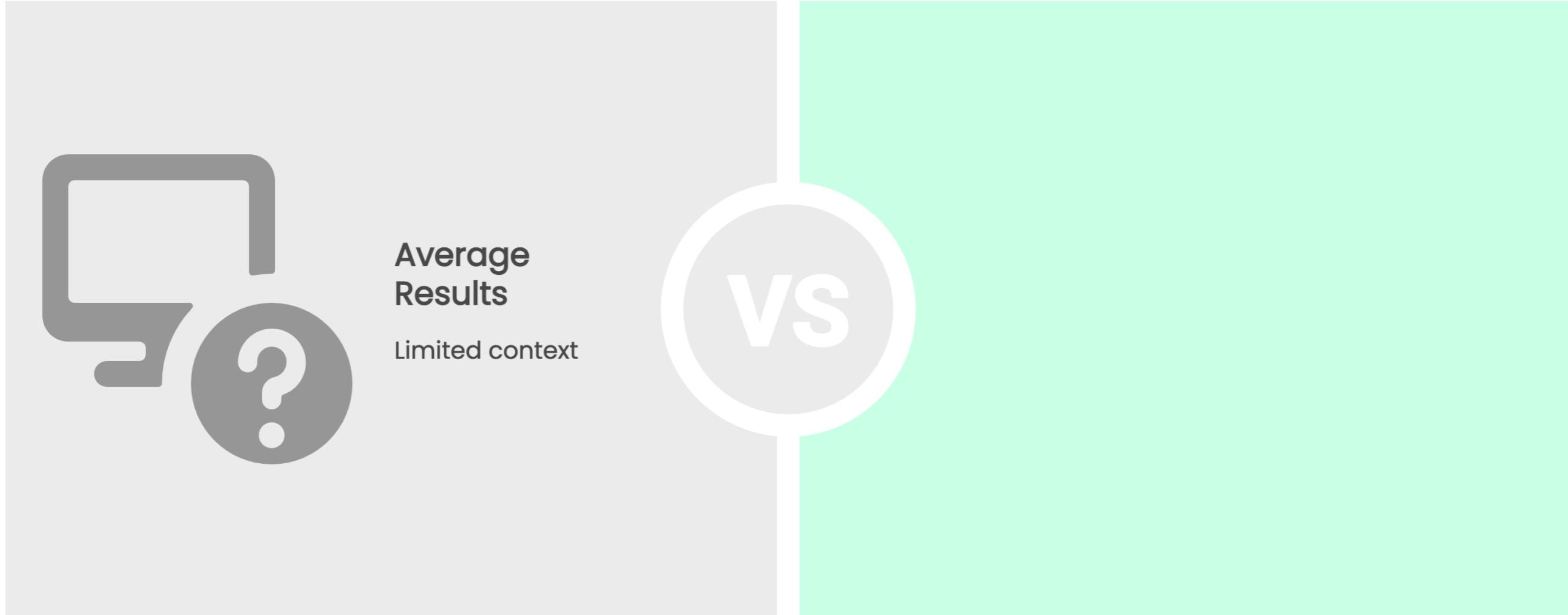
SOFTWARE DEVELOPMENT WITH GITHUB COPILOT



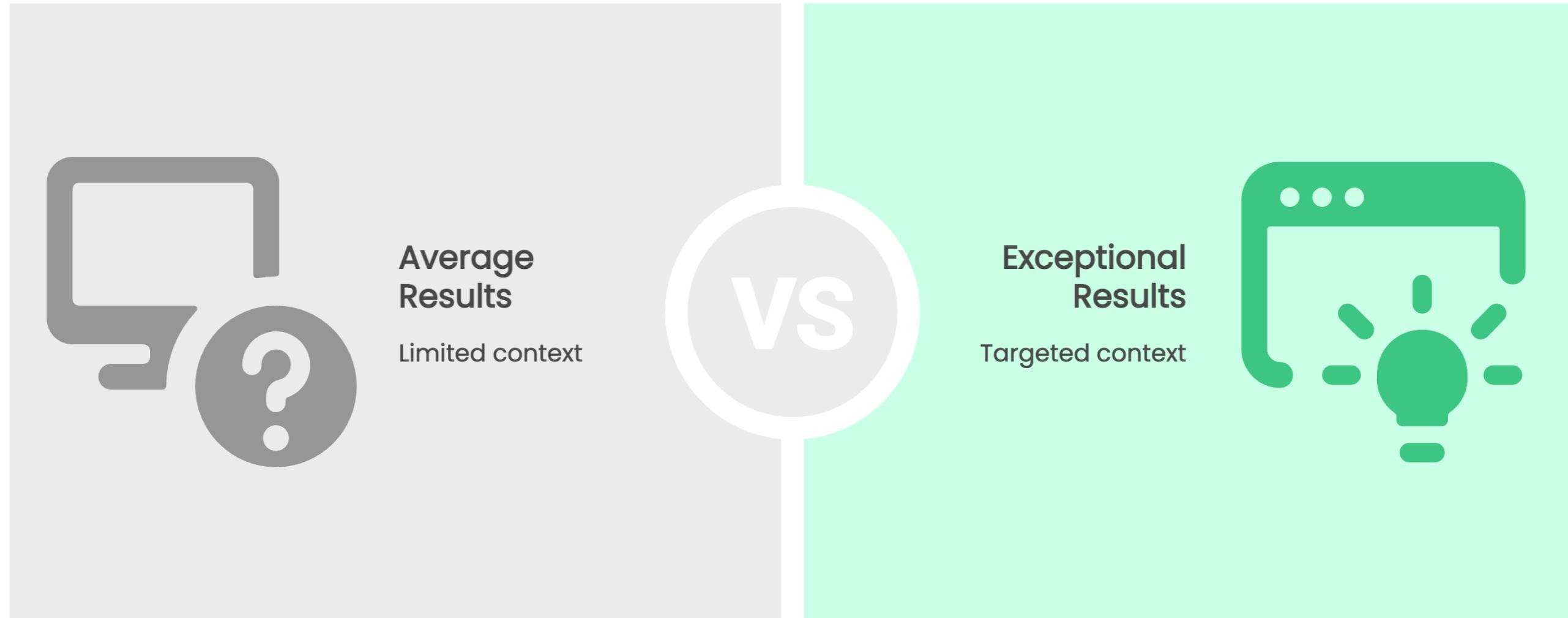
Thalia Barrera

AI Engineering Curriculum Manager,
DataCamp

The importance of context



The importance of context



The `#codebase` variable

`#codebase`: Performs semantic code search across an entire project

Example prompt:

How is user authentication handled? `#codebase`

Result:

- Scans entire codebase
- Includes relevant logic, function calls, or config settings in its answer

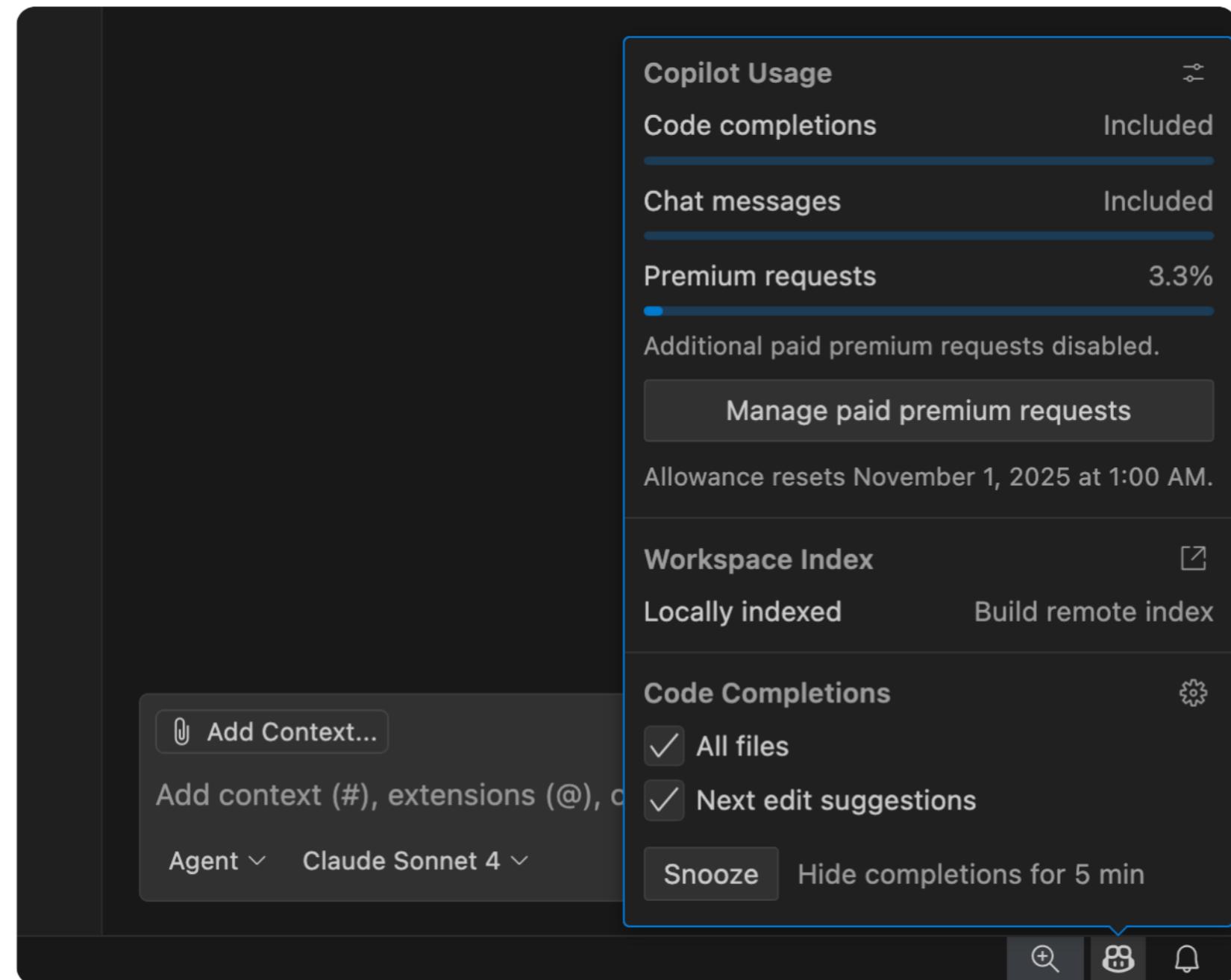
How Copilot understands your codebase



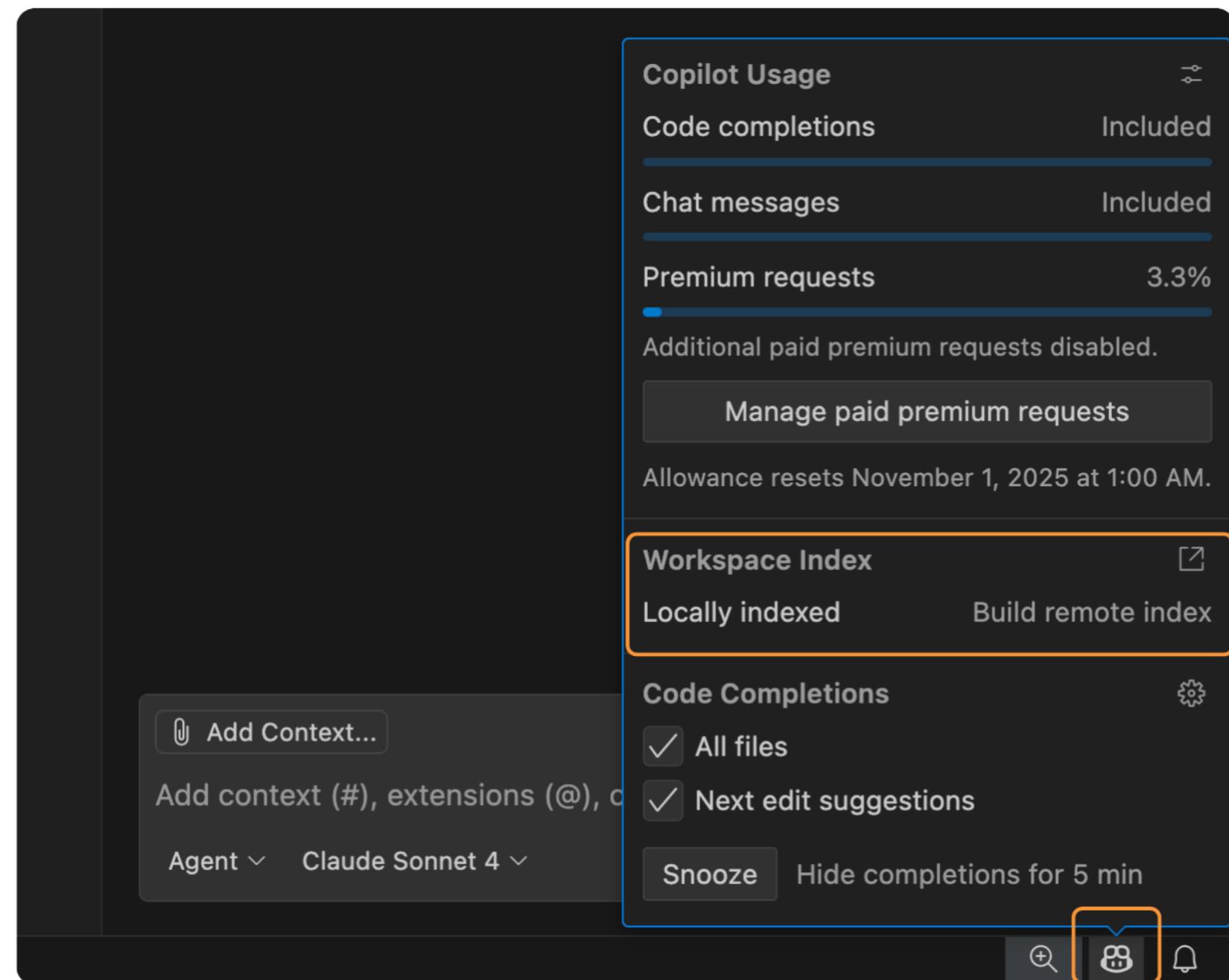
Index types:

- **Local index:** Stored on your machine
- **Remote index:** For repositories hosted on GitHub

Checking your index type



Checking your index type



The `#changes` variable

`#changes`: References most recent source control changes

Example prompt:

Will any of these changes break the login flow? #changes

Use when:

- Evaluating code modifications

The `#selection` variable

`#selection`: References highlighted blocks of code

Example prompt:

Can you refactor this? #selection

Use when:

- Making quick edits
- Creating targeted prompts

The #fetch variable

- Documentation [\[\]](#), API reference [\[\]](#), blog post [\[\]](#)

#fetch: Pulls the contents of a web page

Example prompt:

Summarize this documentation for me #fetch example.com/docs

Result:

- [\[\]](#) Fetches a web page and uses it for context

The #files variable

#files: Attaches specific files to a prompt

Example prompt:

Can you check this file for error handling issues? #files:utils/data_loader.py

Result:

- Analyzes the contents of the referenced file
- Includes it as part of the prompt context

Other chat variables and tools

Variable/tool	Description
#search	Enable searching for files in the current workspace.
#testFailure	Get unit test failure information. Useful when running and diagnosing tests.
#problems	Add workspace issues and problems from the Problems panel as context. Useful while fixing code or debugging.
#terminalLastCommand	Get the last run terminal command and its output.
#githubRepo	Perform a code search in a GitHub repo.

¹ https://code.visualstudio.com/docs/copilot/reference/copilot-vscode-features#_chat-tools

Building a progressive chat interaction

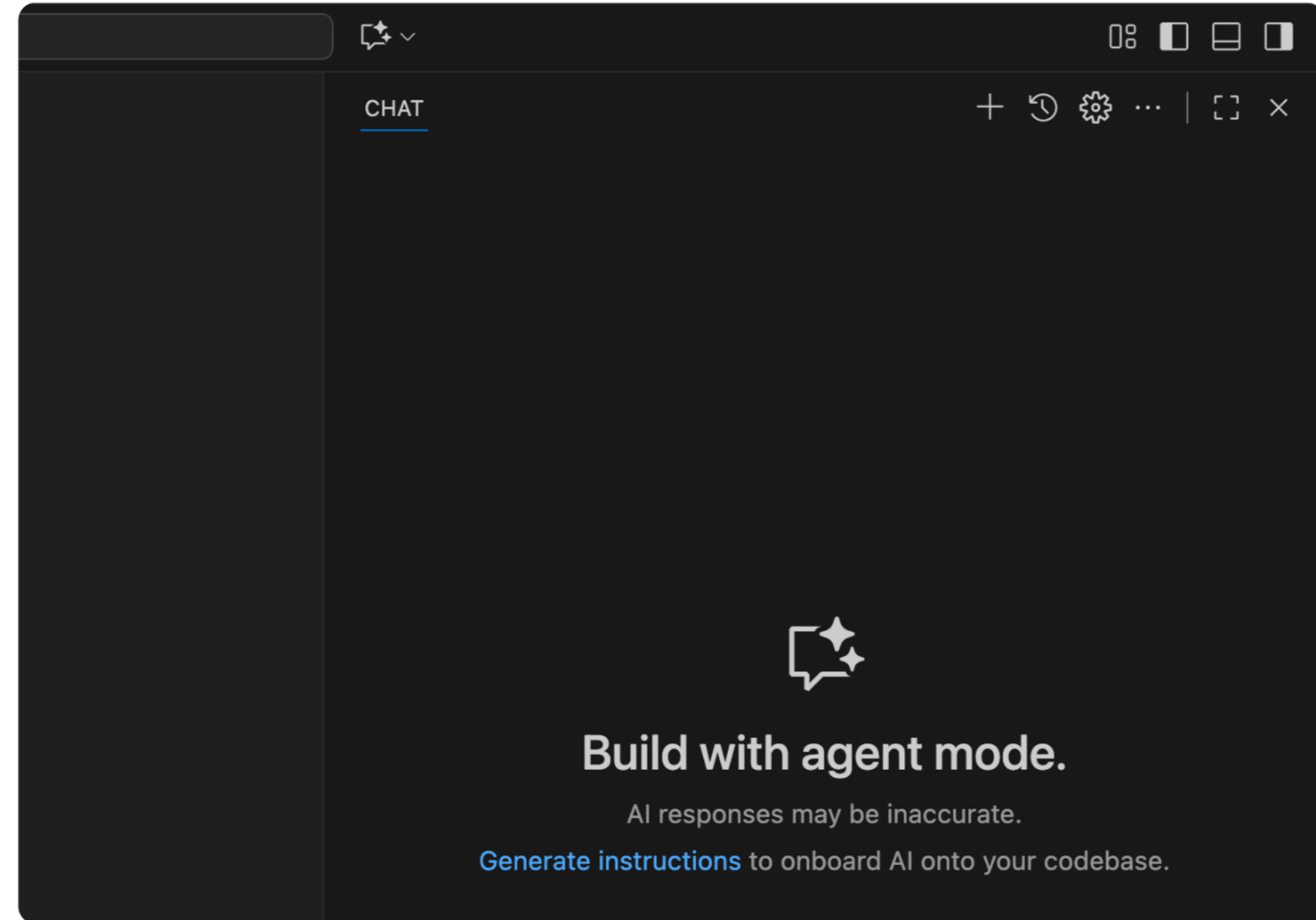
Single, complex prompt:

Add PostgreSQL integration with authentication, persistence, sessions, error handling, and connection pooling

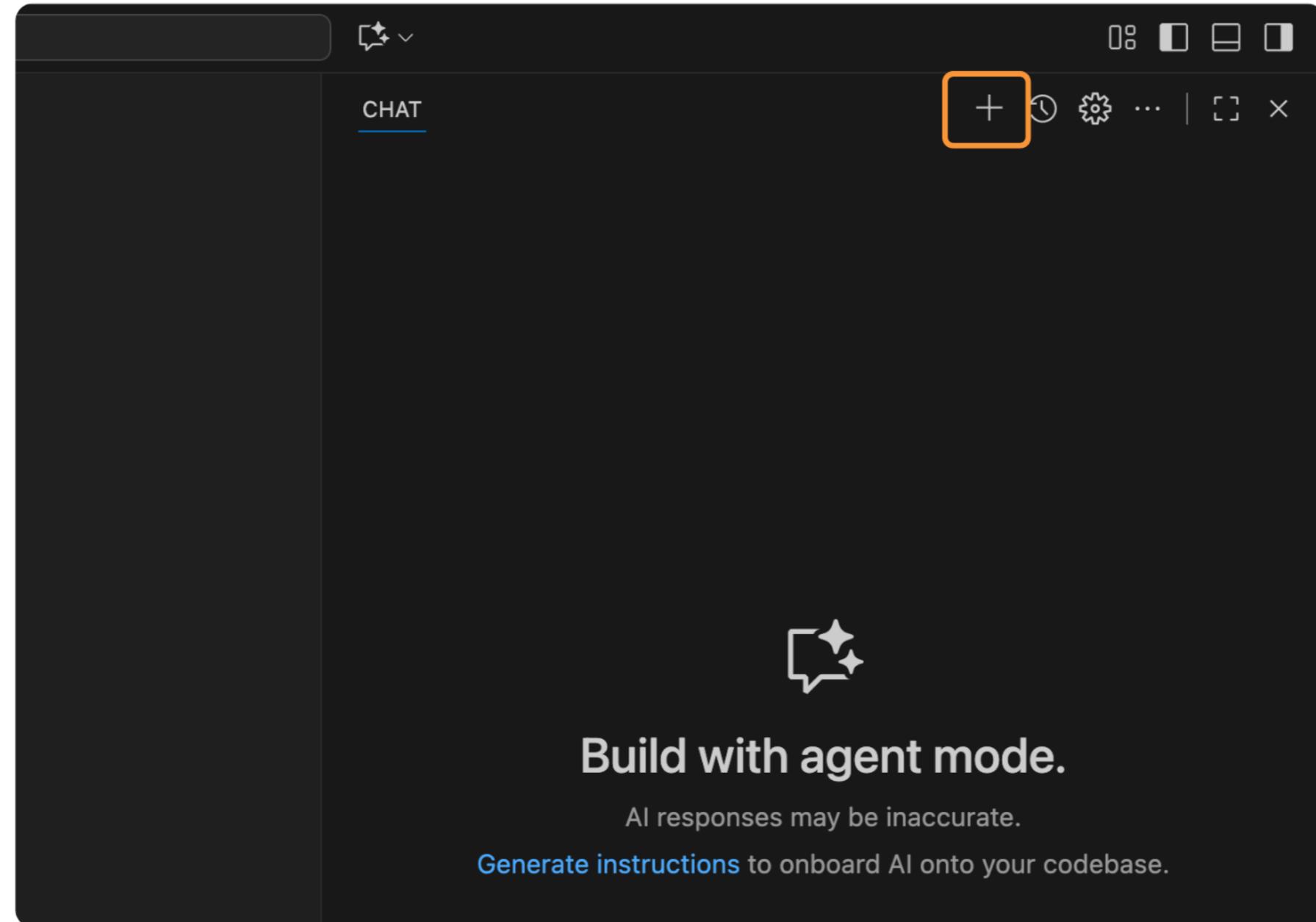
Progressive complexity flow:

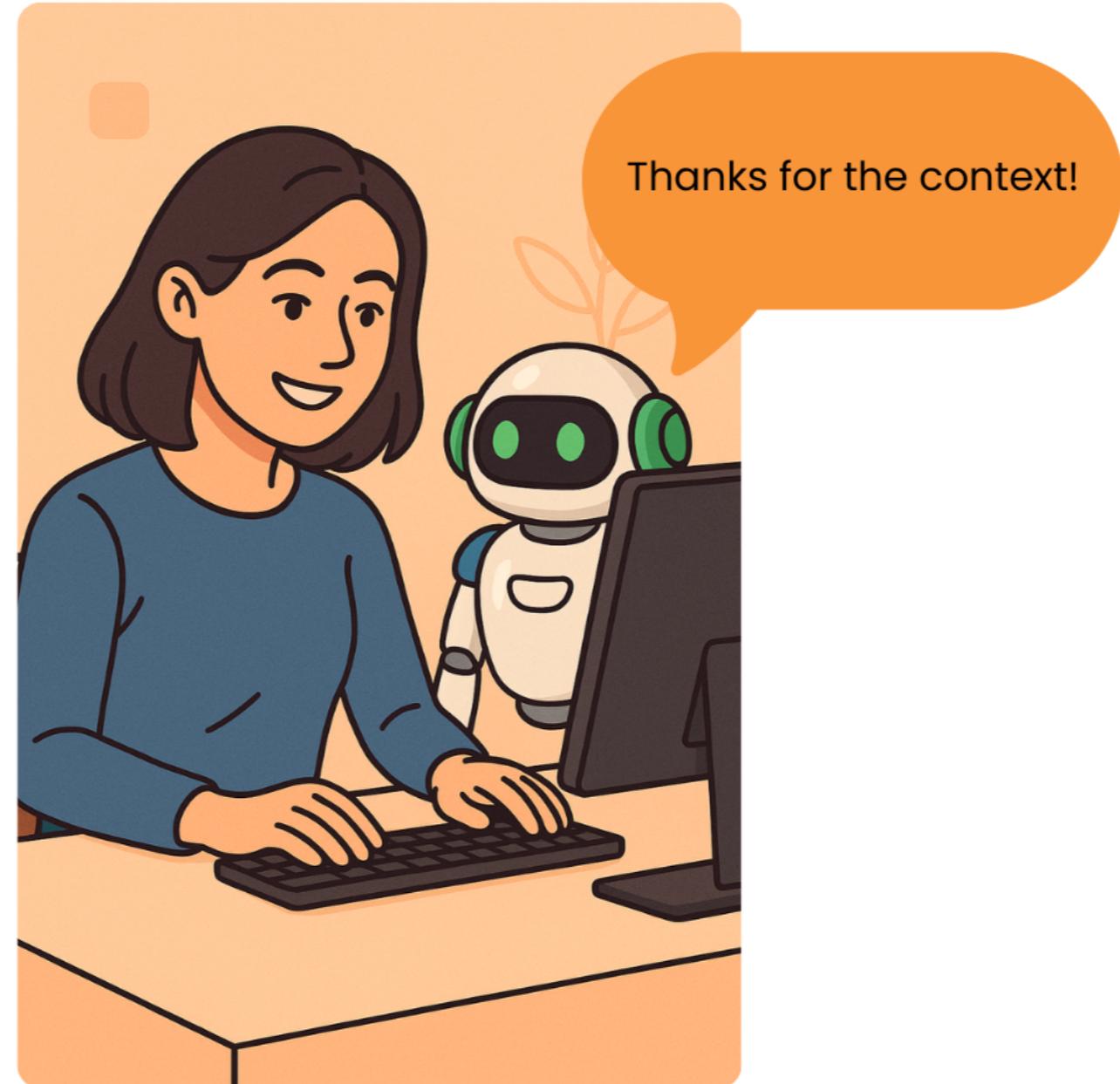
1. *What's the best approach for adding database persistence? #codebase*
2. *Implement the database connection using the suggested approach*
3. *Help me with the schema design for the Task and the User classes*
4. *Now add session management and basic error handling*
5. ...

Starting a new chat to clear context



Starting a new chat to clear context





Let's practice!

SOFTWARE DEVELOPMENT WITH GITHUB COPILOT

Adding domain expertise

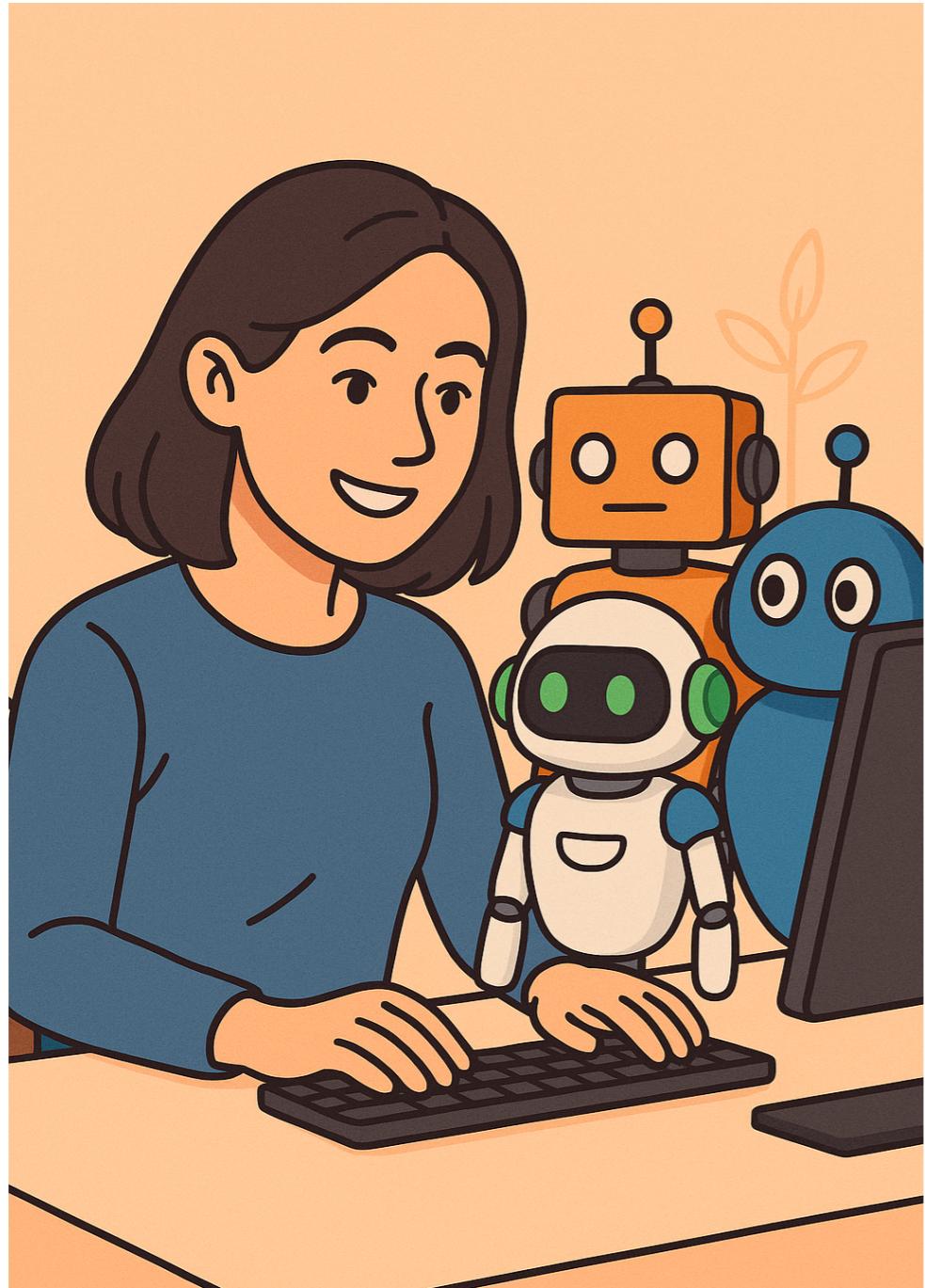
SOFTWARE DEVELOPMENT WITH GITHUB COPILOT



Thalia Barrera

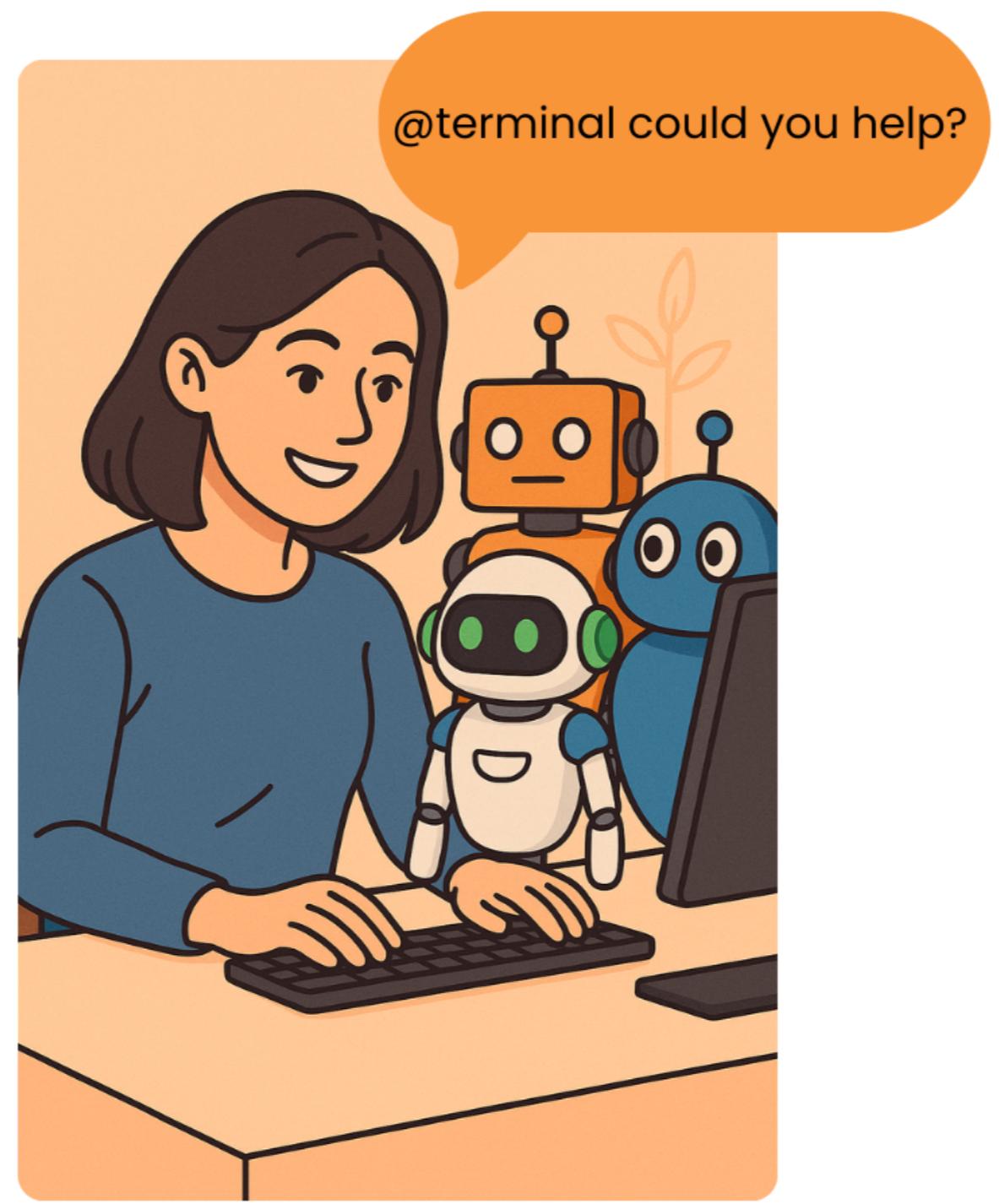
AI Engineering Curriculum Manager,
DataCamp

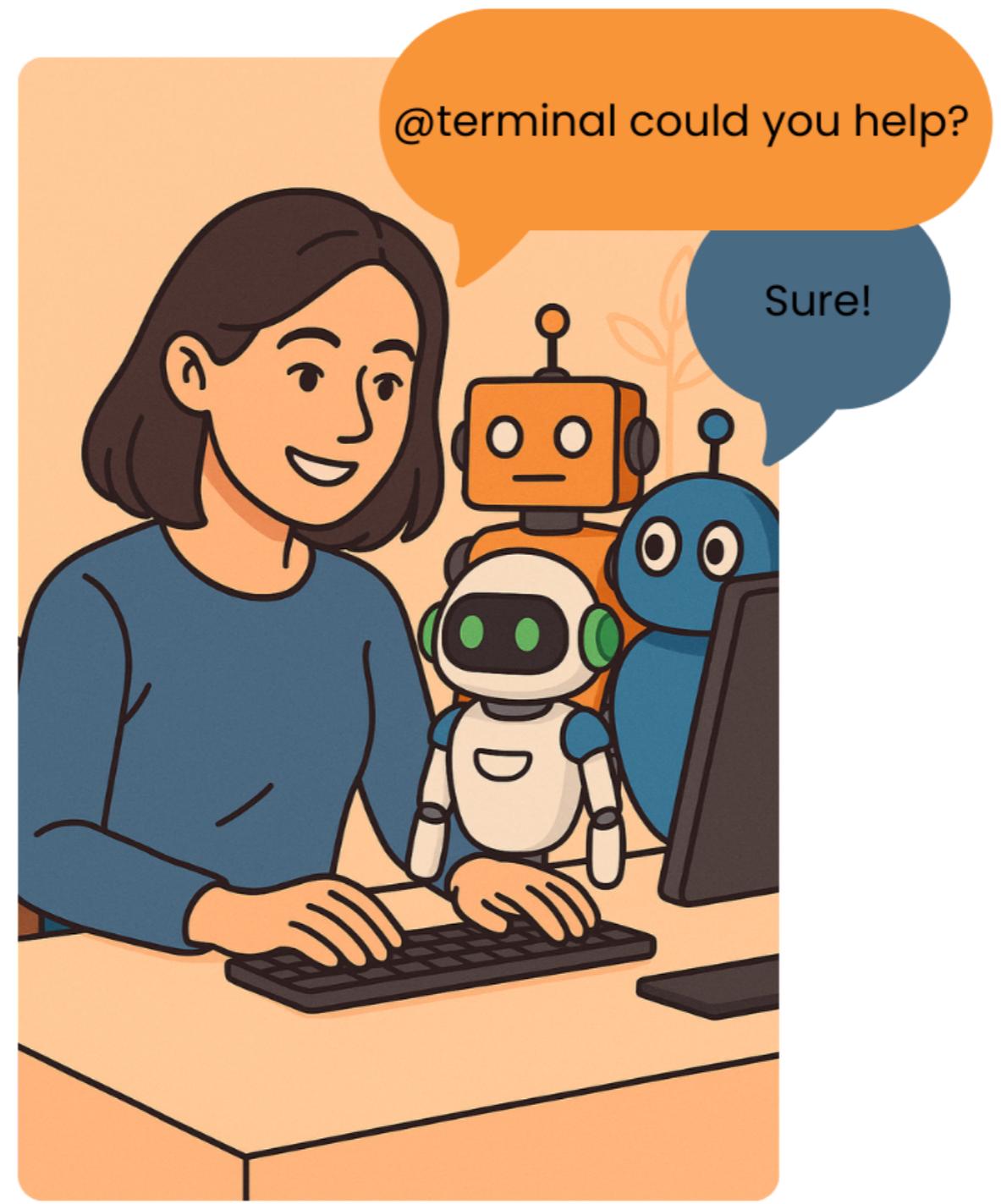
What are chat participants?



Chat participants:

- Built-in AI experts
- Tag them using @ in the chat





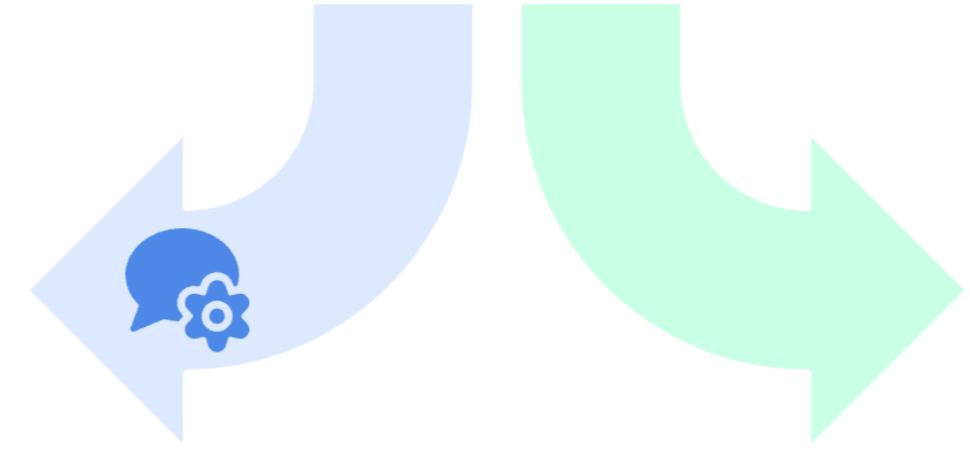
Participants vs. chat variables



Participants vs. chat variables

Chat Variables and Tools

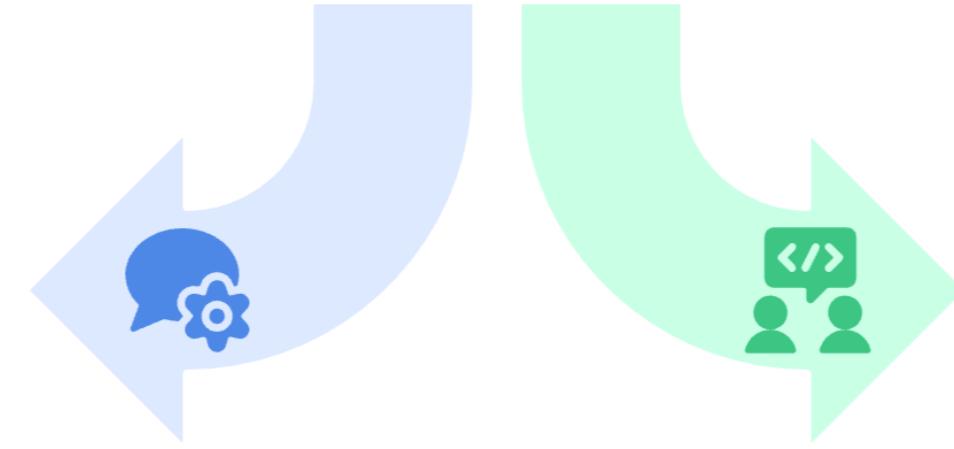
Guide focus on specific elements



Participants vs. chat variables

Chat Variables and Tools

Guide focus on specific elements



Chat Participants

Bring in expertise

Using variables and participants together

Example:

- @workspace how do these changes affect routing? #changes

Use case:

- Ask a workspace-aware expert to review recent edits

Using variables and participants together

Example:

- `@terminal` explain the error from the command `#terminalLastCommand`

Use case:

- Diagnosing a failed command with context

CHAT

+ ⏴ ⚙️ ⋮ | [] ×

@terminal explain the error from the command
#terminalLastCommand

terminalLastCommand

 Terminal

- > Used 1 reference
- ✓ Got last terminal command

The error occurred because you used `pip install requirements.txt` instead of the correct flag. The `-r` flag is needed to install packages from a requirements file:

`pip install -r requirements.txt`

The error message indicates that pip tried to install a package literally named "requirements.txt" rather than reading the file contents to install the listed dependencies.

⟳ ⏵ ⌛ ⌂

Let's practice!

SOFTWARE DEVELOPMENT WITH GITHUB COPILOT

Working efficiently with Copilot

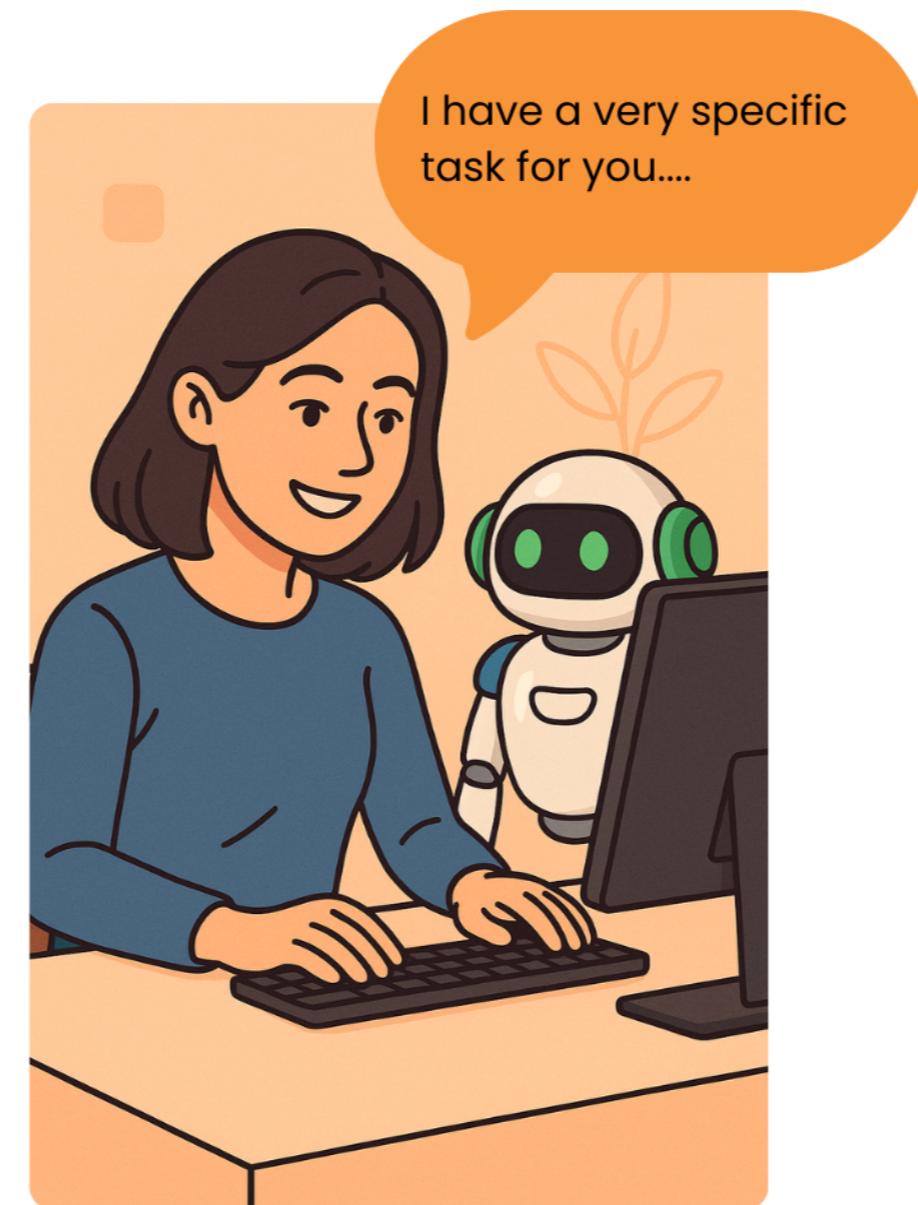
SOFTWARE DEVELOPMENT WITH GITHUB COPILOT



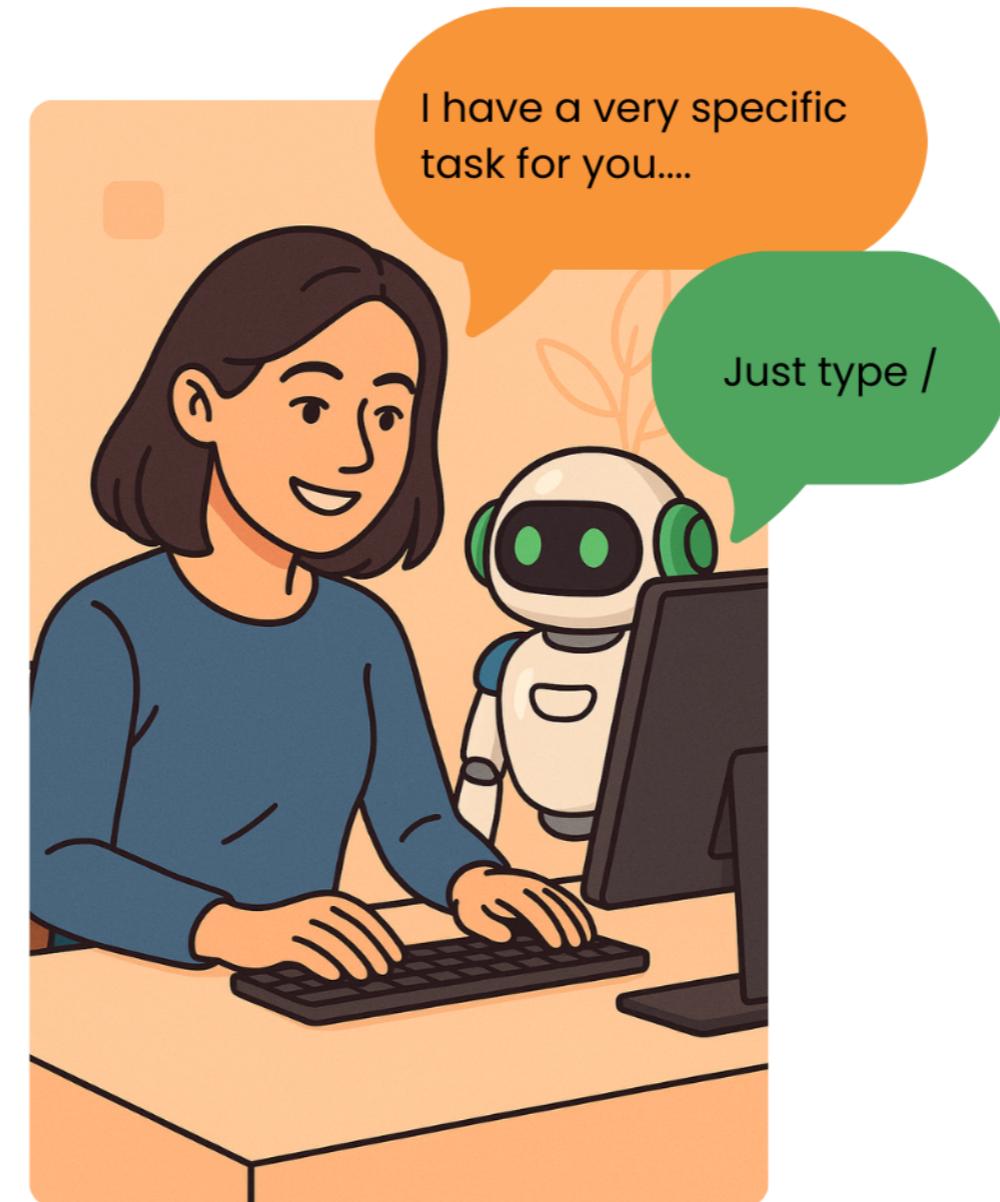
Thalia Barrera

AI Engineering Curriculum Manager,
DataCamp

Slash commands: predefined actions



Slash commands: predefined actions

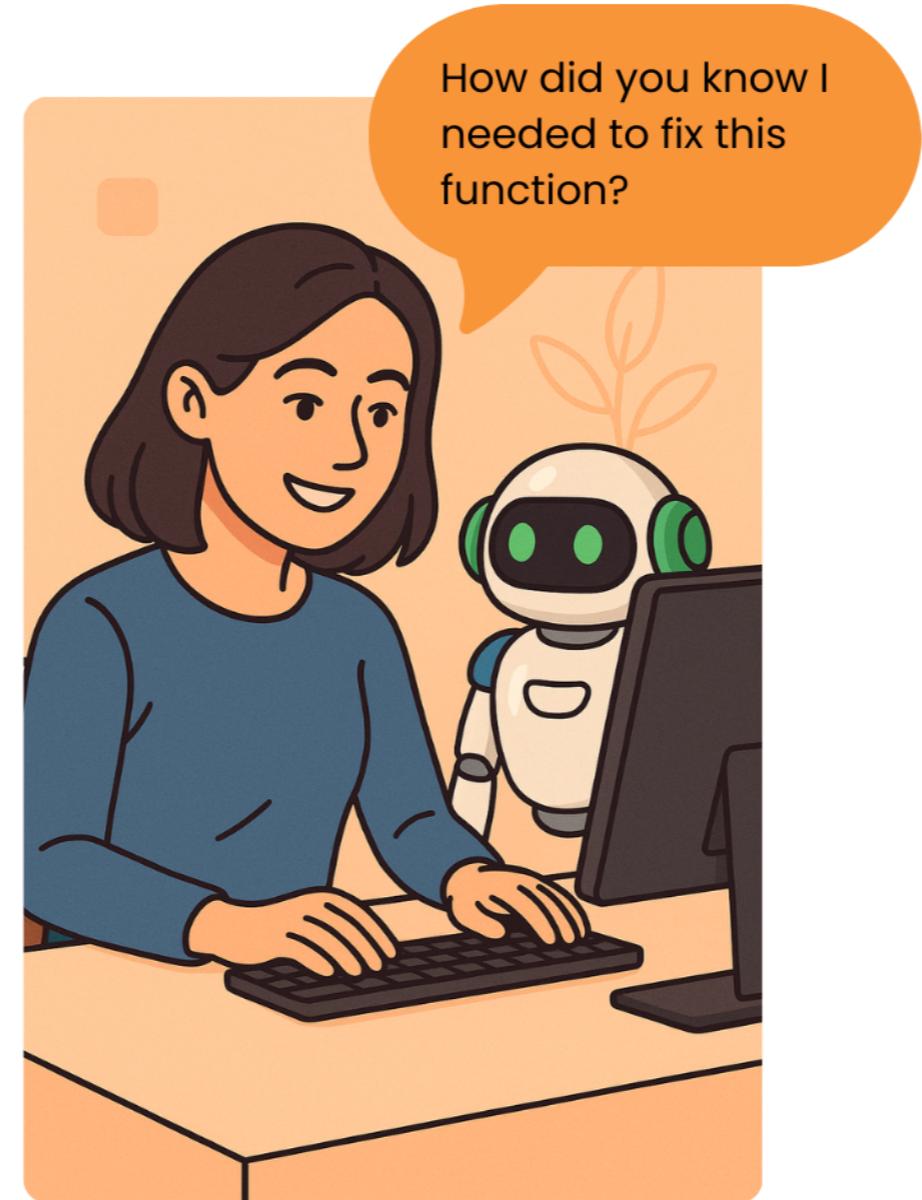


Other slash command examples

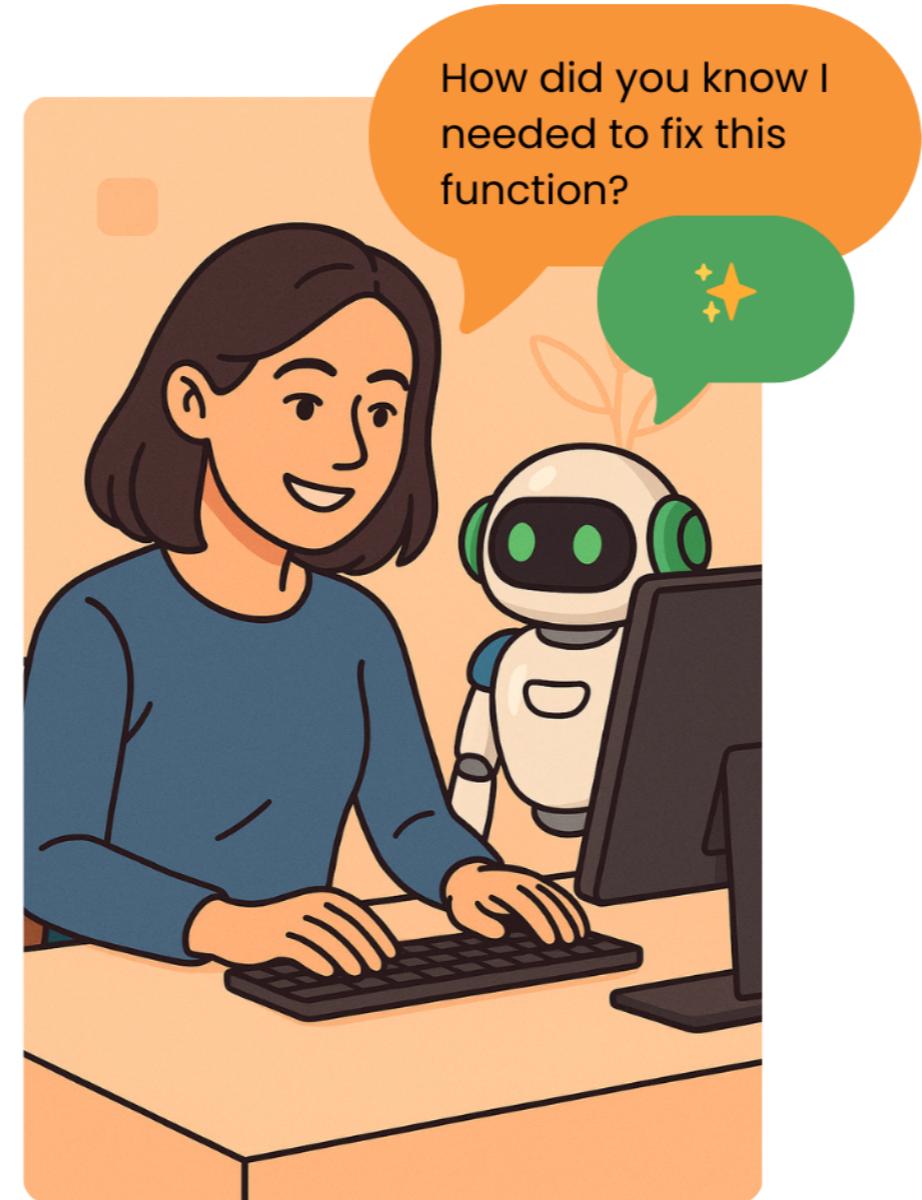
Slash command	Description
/tests	Generate tests for all or only the selected methods and functions in the editor.
/new	Scaffold a new VS Code workspace or file.
/clear	Start a new chat session in the Chat view.

¹ https://code.visualstudio.com/docs/copilot/reference/copilot-vscode-features#_slash-commands

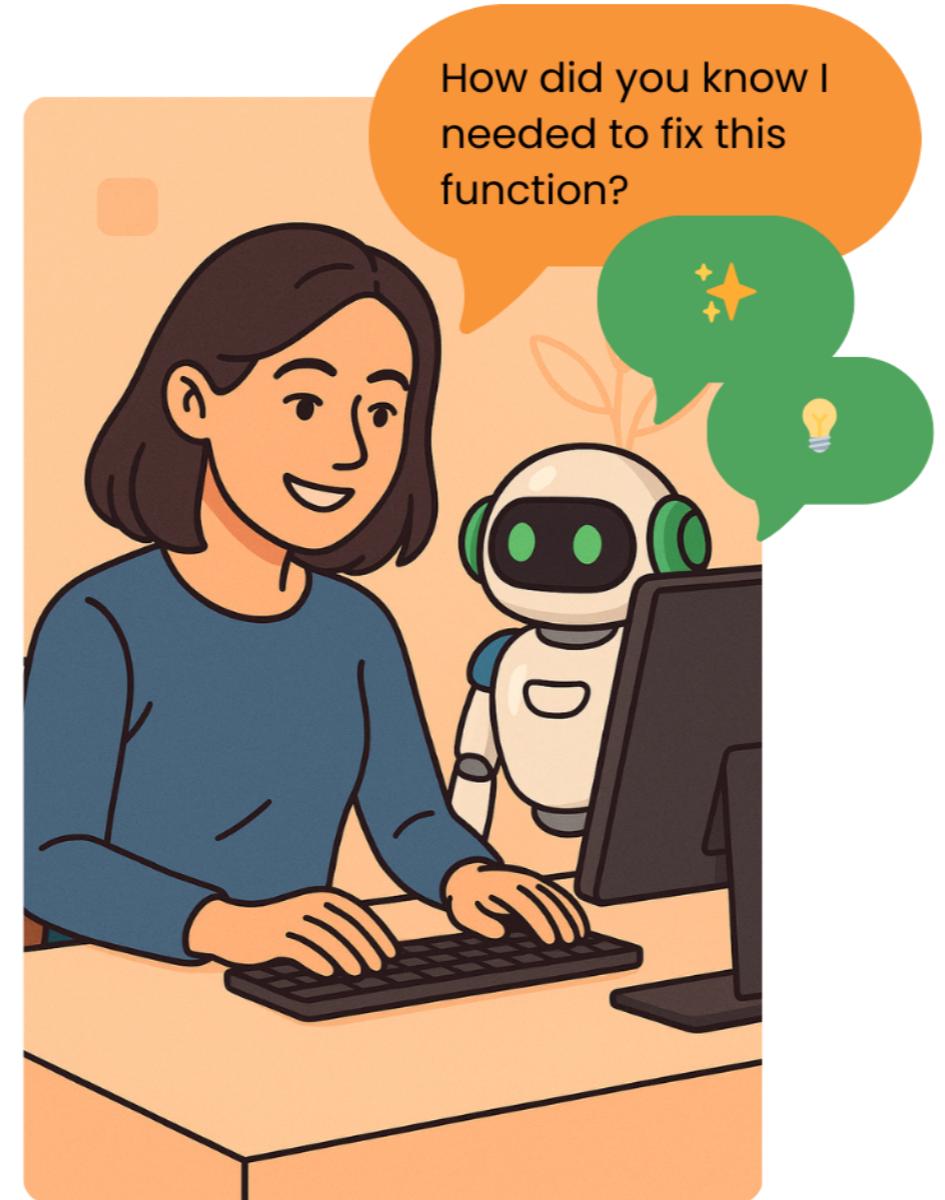
Smart actions: AI-powered suggestions



Smart actions: AI-powered suggestions



Smart actions: AI-powered suggestions







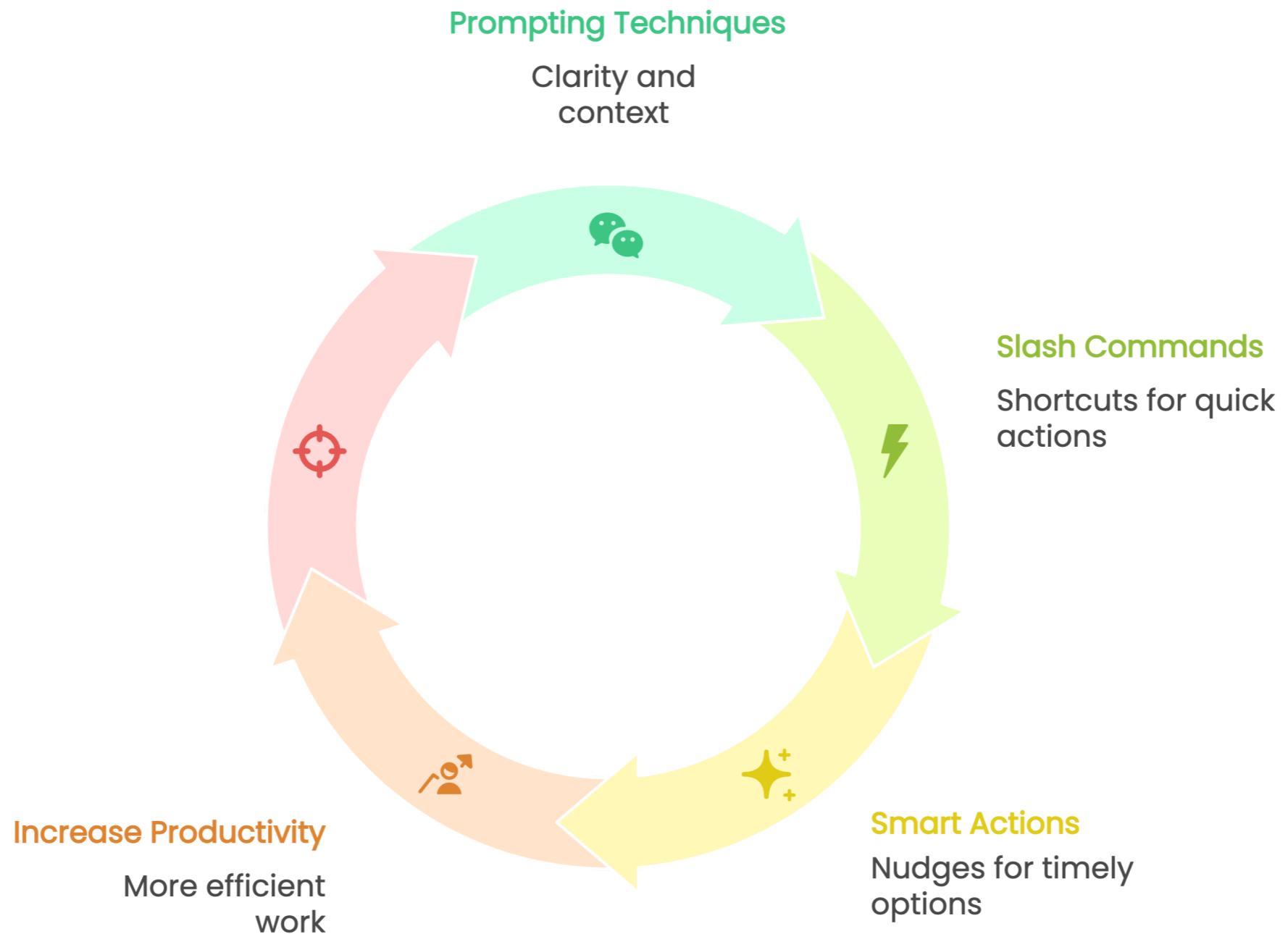
Prompting Techniques

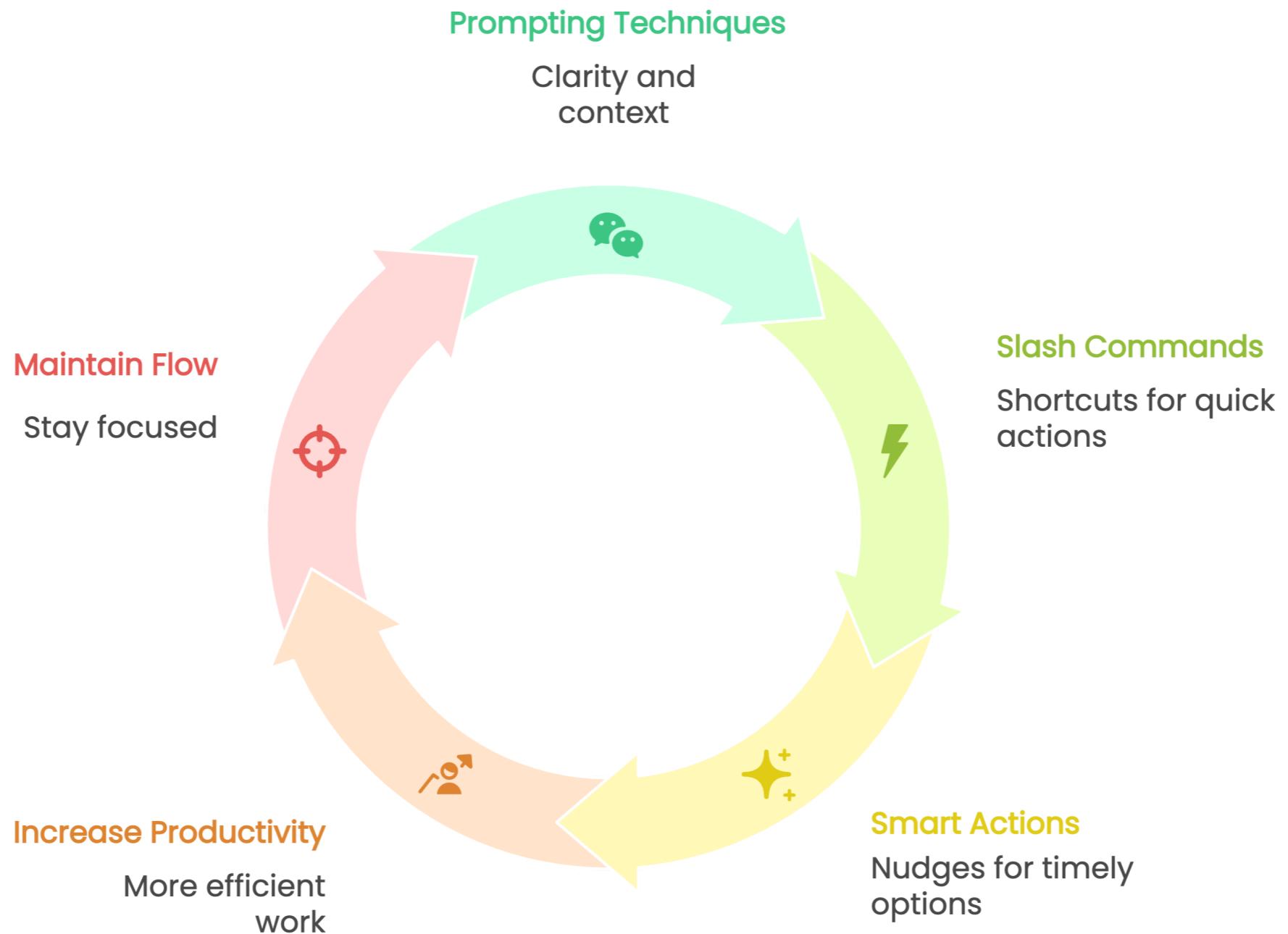
Clarity and
context

Slash Commands

Shortcuts for quick
actions







Let's practice!

SOFTWARE DEVELOPMENT WITH GITHUB COPILOT