Cursor AI For Advanced Users
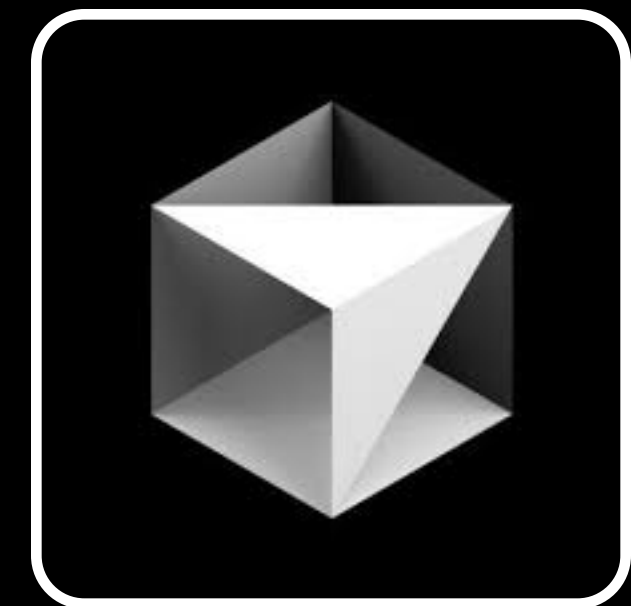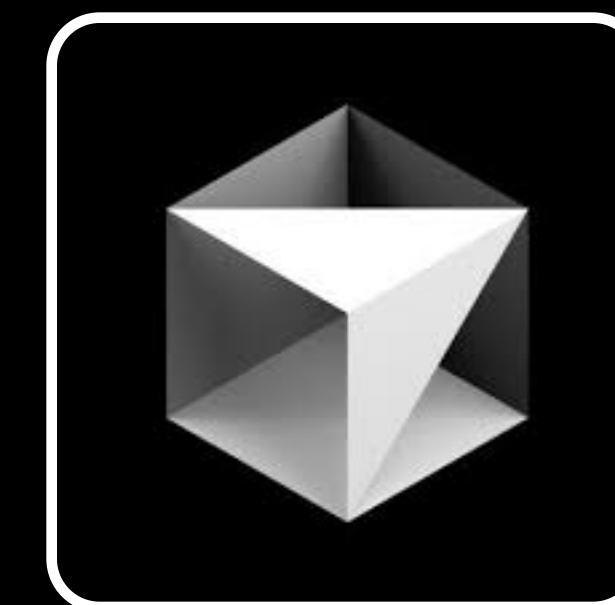# L-2

Codebase Indexing & Context

Overview

- Context is what makes AI useful - without it, even the smartest model guesses
- Cursor builds and uses project-wide understanding through indexing
- @-ref let us direct the AI's attention to the right files or symbols
- Together, they turn cursor from a "helper" into a true engineering partner
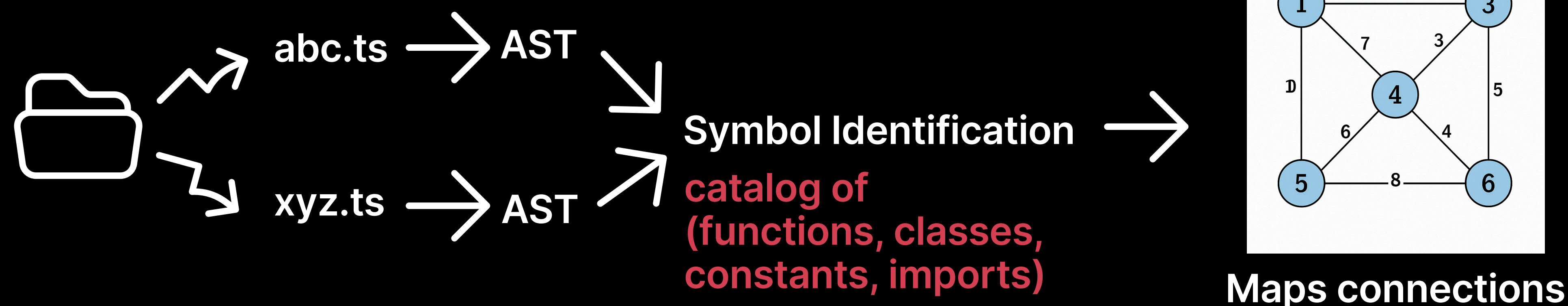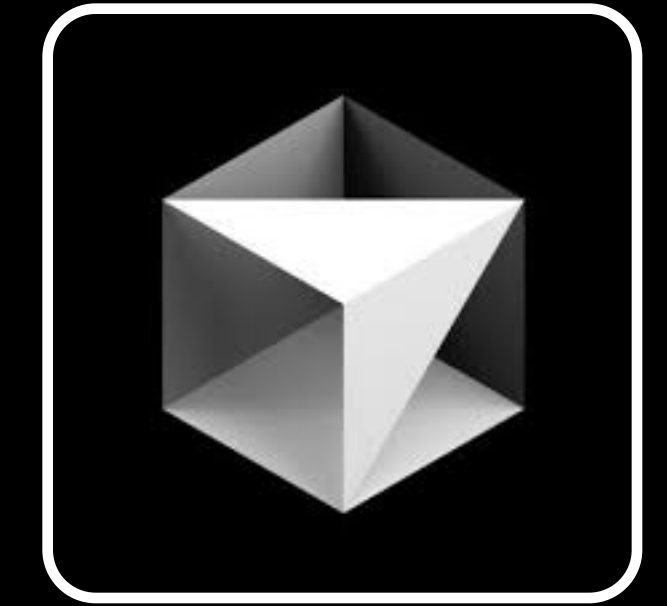
From guessing to knowing

+

Codebase Indexing

- When you open a **folder**, Cursor starts **scanning** your entire project automatically
- It doesn't just read file names - **it parses code structure (Abstract Syntax Tree) (AST)**
- Builds a **semantic map**: functions → classes → relationships → dependencies
- Knows who calls whom, where variables live, and how files connect
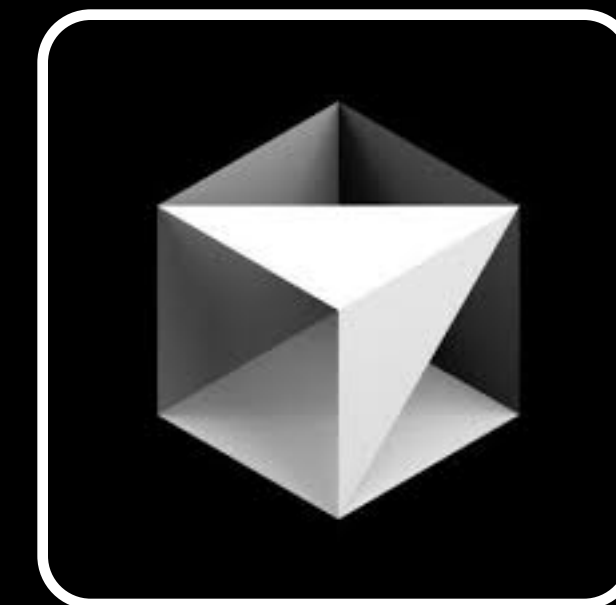
abc.ts → AST

xyz.ts → AST

Symbol Identification

catalog of (functions, classes, constants, imports)

Maps connections

Significance

abc.ts → AST

xyz.ts → AST

Symbol Identification

catalog of (functions, classes, constants, imports)
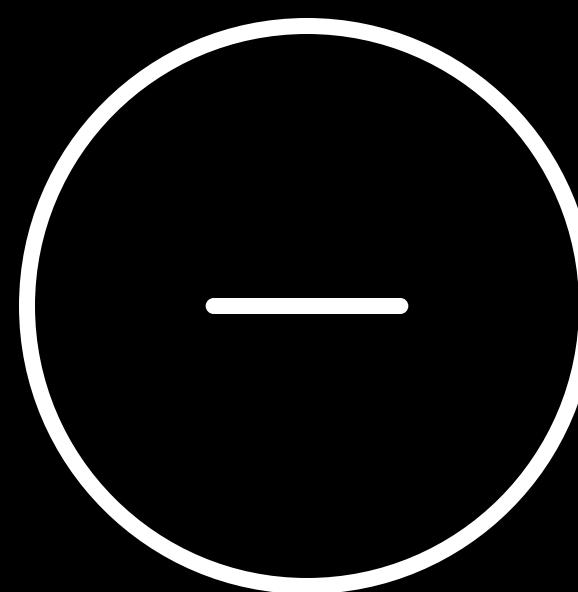
Maps connections

Embedding

- When you ask a question or give a command
  - converts your prompt into a vector and looks at these embeddings for the most semantically related files/snippets
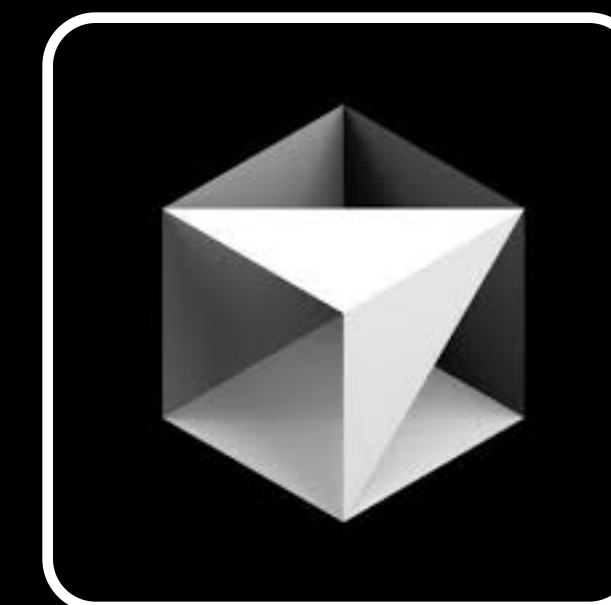 - uses those as context for the LLM

.cursorignore

- Not used for indexing
- Inherits .gitignore

@-ref (Hey AI, look here)

- Generate tests for a file
- Multiple files
- Function in a file
- Web