

Cursor ↔ JetBrains

A translation handout. *Cursor concepts, mapped to their JetBrains equivalents. Workshop content is Cursor-first — this is to keep you oriented.*

The mapping — pink = Cursor. Bold = JetBrains. Where the row says "no equivalent," it's genuinely absent, not just named differently.

Cursor	JetBrains	Notes
Tab autocomplete	Full Line Code Completion + AI inline completion	Same idea: predictive inline suggestions. JB's ships with IDE; AI Assistant extends it with cloud-model suggestions.
Ask mode	AI Assistant chat (read-only)	Side panel chat for Q&A about your code. No edits unless you trigger an AI Action from it.
Agent mode	Junie	JetBrains' agent is a separate product from AI Assistant. Reads, edits, runs terminal, iterates. Closer to Cursor's Agent than AI Assistant chat is.
Plan mode	Junie — plan-first mode	Junie can present a plan for your review before executing. Less of a distinct "mode" — it's a setting on how it runs.
Inline Edit (Cmd+K)	AI Actions → "AI Edit" on selection	Select code → Alt+Enter → AI Actions → "Edit with AI". Same surgical spirit, different keystroke path.
Debug mode	"Explain Error" / "Fix with AI" AI Actions	Fires from the Problems tool window or inline error. Less of a dedicated mode than in Cursor.
@file / @folder	@file in AI Assistant chat + "Attach files"	Same @ affordance in chat; JB also supports drag-and-drop of files and dirs into the chat context.
.cursor/rules/ (.mdc)	Custom project instructions (Settings → AI Assistant → Project Prompts) +	File format and activation rules differ. No exact mdc equivalent. Use AGENTS.md instead for portability.
AGENTS.md	AGENTS.md (supported by Junie)	Good news — Junie picks up AGENTS.md at the repo root. One file, two tools. This is the most portable surface.
MCP servers	Native MCP client	JetBrains supports MCP in AI Assistant and Junie. Same config pattern as Cursor (though UI to add servers differs).
Mode picker (Shift+Tab)	Tool picker + AI Assistant dropdown	Cursor has 6 distinct modes; JB has fewer surfaces. Use AI Actions for surgical edits, chat for Q&A, Junie for agentic work.

Two JetBrains products, one IDE

Easy to confuse them — they're bundled, but separate.

AI Assistant chat + AI Actions + inline completion. The side-panel Q&A and selection-based edits live here. Most like Cursor's Ask + Cmd+K.

Junie the agent. Multi-file reads/writes, terminal commands, iterates against tests. Most like Cursor's Agent. Reads AGENTS.md.

Gotchas when translating

Things the workshop assumes that don't carry over cleanly.

Spend model is different. No Auto-router discount, no in-house Composer 2. JetBrains AI is subscription+quota; the model picker is the whole control. Max Mode shows up as a "long context" toggle on some models.

Rules activation is simpler. No .mdc frontmatter, no glob-scoped rules, no @-mention manual rules. Use AGENTS.md for portable guidance; keep project prompts short.

Hooks + checkpoints diverge. The stop-hook TDD loop is Cursor-specific — use Junie's plan + retry budget instead. For rollback, JB's Local History is richer than Cursor checkpoints.