GitHub Copilot: VS Code vs IntelliJ — Capability Comparison

Feature / Capability	VS Code (GitHub Copilot)	IntelliJ IDEA (GitHub Copilot)
Inline Code Completion / Suggestions	Very strong: suggestions appear as you type, multi-line completions, block-level completions, next edit suggestions.	Fully supported with similar quality. Multi-line and block completions work well for Java/Kotlin.
Copilot Chat (Natural Language Interface)	Fully supported: side-panel chat, inline chat, quick chat (⌘I), terminal chat.	Fully supported: dedicated tool window, inline chat, context-aware suggestions, code explanation.
Multi-file / Workspace Context & File Attachments	Excellent: `#file`, `#codebase`, `#selection`, `@workspace` agent, drag-and-drop file attachments.	Good support: file attachments in chat, project-wide context. `@workspace` equivalent through project indexing.
Model Picker / Multiple LLM Models	Full support: GPT-4o, GPT-4 Turbo, Claude 3.5 Sonnet, Gemini models (based on subscription).	Supported: model switching available in chat settings. Same models available as VS Code.
Slash Commands	Full support: `/fix`, `/tests`, `/explain`, `/doc`, `/simplify`, `/optimize`, custom slash commands.	Full support: `/fix`, `/tests`, `/explain`, `/doc` available in chat. Feature parity achieved.
Copilot Edits (Multi-file Editing)	Available: Agent-based multi-file editing with edit/accept/reject workflow.	Preview/Beta: Multi-file editing capabilities being rolled out.
Code Review (Pre-PR / Self-Review)	Supported: Review suggestions, security scanning, code quality feedback.	Supported: Integrated code review in IDE with similar capabilities.
Commit Message Generation	Supported: AI-generated commit messages based on staged changes, customizable with instructions.	Supported: Commit message suggestions in VCS commit dialog.

GitHub Copilot: VS Code vs IntelliJ — Capability Comparison

Feature / Capability	VS Code (GitHub Copilot)	IntelliJ IDEA (GitHub Copilot)
Keyboard Shortcuts / UX	Mature UX with extensive customization. Tab to accept, Esc to dismiss, ⌘I for inline chat.	Good integration with IntelliJ keymap system. Slightly different shortcuts but fully customizable.
Feature Rollout Speed	Features typically arrive first in VS Code (1-4 weeks ahead).	Features follow VS Code release. Gap has narrowed significantly.
Enterprise / Organization Controls	Full enterprise features: content exclusions, audit logs, policy controls, SSO.	Same enterprise features via GitHub Copilot Business/ Enterprise subscription.
Refactoring / IDE-native Awareness	Good contextual suggestions. Lighter IDE means less deep refactoring awareness.	Stronger for JVM projects due to deep project model, type inference, and refactoring API integration.
Custom Instructions	Supported: `.github/copilot-instructions.md`, workspace settings, user preferences.	Supported: Same custom instructions file format, IDE-level settings.

Note: IntelliJ Copilot has achieved near feature parity with VS Code. The main differences are now UX preferences and rollout timing rather than missing features. Always check the latest plugin version for current capabilities.