



To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Cursor for Professional Developers

Class Duration

14 hours of live training delivered over 2-3 days to accommodate your scheduling needs.

Student Prerequisites

- Professional software development experience
- Cursor installed (Pro or Business plan recommended for full feature access)

Target Audience

Software engineers and tech leads who are adopting Cursor as a primary development environment and want to use its most powerful features—Composer 2.0, Background Agents, Cloud Agents, multi-file refactoring, rules files, and context management—at a professional level, not just for basic autocomplete. Especially relevant for teams evaluating Cursor 3.0's multi-repo workspace and self-hosted cloud agents.

Description

Cursor is built from the ground up as an AI-first code editor, and the 3.0 release turned it into a unified workspace for human-and-agent software development—shipping Cursor's in-house frontier coding model, Composer 2 (built on Moonshot's Kimi K2.5 with continued pretraining and reinforcement learning, 200+ tok/s, scoring 61.3 on CursorBench). This course covers the professional-grade features that distinguish it from add-on assistants. We cover Composer 2 for multi-file generation and refactoring, agent mode for longer-horizon autonomous tasks, the new Background Agents that work in parallel locally while you keep typing, Cloud Agents (and self-hosted cloud agents) for off-machine task execution, Design Mode for visual UI iteration, the multi-repo workspace, rules files (`.cursorrules` and project-scoped rules) for standardizing AI behavior, and advanced context management using `@codebase`, `@docs`, `@web`, and custom context sources. Labs use realistic multi-file scenarios and include team-configuration best practices.



To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Learning Outcomes

- Navigate Cursor 3.0's feature set: chat, inline edit, Composer 2, agent mode, Design Mode, and the multi-repo workspace.
- Use Composer 2 (Cursor's in-house frontier coding model, built on Moonshot's Kimi K2.5) effectively for multi-file generation, refactoring, and scaffolding tasks.
- Configure agent mode for autonomous multi-step coding tasks with appropriate guardrails.
- Run Background Agents locally and Cloud Agents (Cursor-hosted or self-hosted) for off-thread, multi-tasking development.
- Hand off work seamlessly between local and cloud agents and back to the IDE.
- Use Design Mode for visual UI iteration on real tasks.
- Write and maintain project-level and user-level rules files to standardize AI behavior.
- Manage context explicitly: @codebase, @file, @docs, @web, and pinned context.
- Integrate Cursor into team workflows with shared configuration and version-controlled rules.
- Apply privacy and security best practices for codebases with sensitive data, including self-hosted cloud agent deployment.

Training Materials

Comprehensive courseware is distributed online at the start of class. All students receive a downloadable MP4 recording of the training.

Software Requirements

Cursor (Pro or Business plan), Git, and the student's preferred language runtime for labs.

Training Topics

Cursor Architecture and Philosophy

- Cursor vs. VS Code Copilot: key design differences
- Cursor 3.0's unified human-and-agent workspace
- Model configuration: choosing Claude, GPT, and Gemini backends
- Running the OpenAI Codex IDE extension inside Cursor as a complementary surface



To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

- Privacy modes and codebase indexing controls

Chat and Inline Edit

- Inline code generation and edit shortcuts
- Chat with @ context references
- Apply vs. preview edit workflows

Composer 2: Multi-File Generation

- Composer 2 as Cursor's in-house frontier coding model (built on Moonshot's Kimi K2.5 with continued pretraining + RL; 200+ tok/s with custom GPU kernels)
- Opening and driving Composer sessions
- Scoping Composer to a feature, module, or refactor
- Reviewing and accepting multi-file diffs
- Iterating with follow-up prompts
- Auto mode: when Composer 2 picks the model for you

Agent Mode

- Enabling and configuring agent mode
- Autonomous task execution and step-by-step review
- Interruption, rollback, and scope limits
- Combining agent mode with tests for validation

Background Agents and Cloud Agents

- Background Agents: parallel local agent sessions while you keep coding
- Cloud Agents: isolated VMs with full development environments
- Self-hosted cloud agents for sensitive codebases
- Multi-repo workspace and seamless local/cloud handoff

Design Mode

- Design Mode: iterate on UI visually with the agent driving rendered changes
- Pairing Design Mode with rules files for consistent styling
- When Design Mode replaces hand-coded UI scaffolding

Rules Files and Project Configuration

- User-level rules for personal preferences
- Project-level `.cursorrules`: format, conventions, and constraints



To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

- Version-controlling rules alongside code
- Combining rules with system prompts
- Pairing Cursor rules with a GitHub Spec Kit constitution and spec-driven workflow (covered in depth in the dedicated Spec Kit course)

Context Management

- @codebase semantic search and relevance tuning
- @docs for third-party library documentation
- @web for up-to-date reference material
- Pinning critical context files

Team Workflows and Security

- Shared rules configuration across a team
- Sensitive file exclusion and .cursorignore
- Self-hosted cloud agent deployment patterns
- Audit trail and team policy considerations

Workshop

- Multi-file Composer lab
- Background and Cloud Agent task lab
- Rules file authoring exercise
- Q&A session