



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

## Agentic Coding with Cursor: Agent Mode, the CLI, and Background Agents

### Class Duration

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

### Student Prerequisites

- Professional software development experience in at least one language
- Working knowledge of Git and GitHub (issues, pull requests, branches, reviews)
- Comfort with the command line; Cursor installed
- A Cursor plan that includes Agent and Background Agent features
- Familiarity with Cursor's core features is helpful (see *Cursor for Professional Developers* for the feature deep-dive)

### Target Audience

Software engineers, tech leads, and platform/DevEx teams who want to move beyond AI autocomplete and *delegate* engineering work to Cursor agents. Ideal for teams adopting agentic workflows across the IDE, the terminal, and the cloud—kicking off background agents from GitHub and Slack, driving plan-then-execute sessions locally, and standardizing how agents are supervised, reviewed, and governed. This is a practitioner course focused on agentic development with Cursor rather than a survey of every feature.

### Description

Cursor is now three agents in one ecosystem: the in-IDE Agent mode, the terminal-native Cursor CLI (`cursor-agent`), and cloud-based Background Agents. This course teaches developers how to put all three to work on real tasks—planning multi-step changes, editing across many files, running commands, and iterating until the job is done. You will learn to launch work from outside the editor by commenting `@cursor` on a GitHub PR or issue, from Slack and Linear, or from the web dashboard at [cursor.com/agents](https://cursor.com/agents), then track and steer the cloud agent as it works on a branch and opens a merge-ready pull request. We cover Plan Mode, grounding agents with



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

project Rules, Agent Skills, and MCP servers, the CLI's cloud handoff, running agents in parallel, and closing the loop with Bugbot's automated PR review. Throughout, we emphasize supervision, security, and cost so teams can adopt agentic coding responsibly.

## Learning Outcomes

- Explain Cursor's agentic surfaces—Agent mode, the Cursor CLI, and Background Agents—and choose the right one per task.
- Drive Agent mode for multi-step, multi-file tasks: reading, editing, running commands, and fixing errors.
- Use Plan Mode to research the codebase and produce a reviewable, editable plan before coding.
- Run the Cursor CLI (`cursor-agent`) for scripted tasks and hand work off to the cloud.
- Launch Background Agents from GitHub, Slack, Linear, the web dashboard, and the IDE.
- Understand how Background Agents run in a sandbox, work on a branch, and open pull requests.
- Write agent-ready issues with clear scope and acceptance criteria, and iterate through PR review.
- Ground agents in project context with Rules, Agent Skills, and MCP servers.
- Apply Bugbot's PR review and route findings back to an agent for fixes.
- Establish security, governance, and cost controls for agentic development at team scale.

## 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 installed with a plan that includes Agent and Background Agent features, the Cursor CLI (`cursor-agent`, Node.js 18+), a GitHub account with a repository and the Cursor GitHub app connected, optional Slack/Linear integrations, the GitHub CLI (`gh`), and Git.



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

## Training Topics

### Foundations of Agentic Coding

- What “agentic” means: plan, act, observe, iterate
- Cursor’s agentic surfaces: Agent mode (local), the Cursor CLI, and Background Agents (cloud)
- The supervision spectrum: assisted edits → autonomous tasks → hands-off background work
- Where agents excel and where to keep a human in the loop

### Agent Mode in the IDE

- Agent mode: autonomous reading, editing, running commands, and iterating on errors
- Multi-file edits and reviewing/accepting changes
- Checkpoints and reverting agent work
- Providing context with @-symbols (files, docs, web)

### The Cursor CLI as an Agent

- Running cursor-agent in the terminal (the same prompts and tools as the IDE agent)
- Shared MCP servers, Rules, and authentication with the desktop app
- Plan mode in the CLI (/plan, --mode=plan)
- Cloud handoff: escalating a CLI task to a background agent
- Scripted and scheduled (cron) usage

### Planning and Decomposition

- Plan Mode: research the codebase, ask clarifying questions, and generate an editable plan
- Reviewing and refining the plan before execution
- Turning a goal into agent-sized tasks
- Acceptance criteria the agent can verify against

### Grounding Agents in Context

- Project Rules (.cursor/rules) and user rules
- Agent Skills for repeatable workflows
- MCP servers (mcp.json) for internal tools, APIs, and data
- Docs indexing and @-context for grounding



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

### Launching Background Agents

- Web dashboard: [cursor.com/agents](https://cursor.com/agents) on any device
- IDE: the **Cloud** option in the agent input dropdown
- GitHub: comment @cursor on a PR or issue to kick off an agent
- Slack and Linear: @cursor to start work; plus the API

### How Background Agents Work

- Cloud-hosted, sandboxed execution environment
- Cloning the repo and working on a separate branch
- Pushing changes and opening merge-ready pull requests
- Required read-write repo privileges and environment configuration

### From Issue to Pull Request

- Writing agent-ready issues: scope, context, and constraints
- Reviewing the PR and the agent's artifacts
- Iterating with the agent through comments and re-prompts
- Knowing when to take over locally versus re-prompt

### Agentic Code Review with Bugbot

- Bugbot automated review on GitHub pull requests
- How Bugbot learns from reviewer feedback and promotes accurate rules
- Routing Bugbot findings to a background agent for fixes
- Combining human review with agent-generated follow-ups

### Remote and Multi-Agent Workflows

- Managing multiple background agents from the dashboard
- CLI cloud handoff and coordinating local + cloud agents
- Parallelizing independent tasks across agents
- Avoiding conflicts and merging parallel agent work

### Security, Governance, and Cost

- Repo access scope and read-write privileges
- Rules that constrain what agents may modify
- Privacy Mode and data-handling controls
- Model selection, usage, and cost controls



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

### Workshop

- Kick off a background agent from GitHub or Slack and ship its PR to merge
- CLI lab: Plan Mode → cloud handoff
- Bugbot review → fix agent
- Multi-agent parallel task exercise
- Q&A session