

Cursor 101

An introductory walkthrough of Cursor's core features for new users.

Demo Flow

Follow the structured demo:

- [DEMO_FLOW.md](#) – Text-only guide
- [DEMO_FLOW.screenshots.md](#) – Full guide with screenshots

Features Covered

#	Feature	Description	Docs
1	Layout Walkthrough	Navigate the IDE layout and customize workspace	text screenshots
2	Tab + Quick Edit	AI autocomplete and surgical edits with Cmd+K	text screenshots
3	Agent Chat	The command center for agentic workflows	text screenshots

Getting Started

Start the Demo Server

Run the `/start-demo` slash command to: 1. Launch the Next.js development server 2. Open the blog app in the built-in browser 3. Set up the workspace for demo

Reset Between Sections

Use `/reset` to restore the workspace to a clean state: - Reverts any code changes - Ensures deterministic demo output - Recommended before each major section

Using Deep Links

Throughout the demo, you'll see clickable links like:

[Implement Dark Mode](cursor://anywhere.cursor-deeplink/prompt?text=...)

These **deep links** open a “Create chat with prompt” dialog with the prompt pre-filled:

1. Click the deep link in the documentation
2. Click “**Create Chat**” in the dialog
3. Select your model (recommend **composer-1** for speed)
4. Submit to run the agent

Deep links ensure: - Consistent prompts across demos - Reproducible results - Quick setup without typing

Quick Reference

Action	Shortcut/Command
Open Agent Chat	Cmd+L
Quick Edit	Cmd+K
Accept Tab suggestion	Tab
Chat History	Opt+Cmd+ '
Start demo server	/start-demo
Reset workspace	/reset
Add selection to chat	Select + Cmd+L

Topics Covered

1. Layout Walkthrough

- File Explorer, Editor, Terminal, Agent Panel
- Custom layouts and panel arrangement

2. Tab Autocomplete

- Full codebase indexing for understanding conventions
- Multi-line edits and auto-imports
- Cross-file jumps for coordinated edits
- Learns your preferences via accept/reject signals
- Suggestions based on recent changes and linter errors

3. Quick Edit (Cmd+K)

- Surgical code edits
- Natural language instructions
- Diff preview before accepting

4. Agent Chat

3.1 Agent Configuration

- Agent modes (Agent, Plan, Debug, Ask)
- Model selection and settings
- Adding context (@files, @folders, @docs, @terminal, @past chats)
- Image upload for visual context
- Voice input for faster iteration
- Execution mode (Local, Cloud, Worktree)
- Context window management

3.2 Working with the Agent

- Starting the server with terminal permissions
- Making code changes with review/undo
- Fixing bugs from terminal output
- Documentation integration (built-in + custom)
- Git diff context for code review
- Feature implementation demo (Dark Mode)

Next Steps

After completing Cursor 101:

- **Cursor 2.0 Demo** – Advanced features like Browser, Worktrees, Bugbot
- **Plan Mode** – Detailed implementation planning
- **Debug Mode** – Hypothesis-driven debugging