
ELLM

How to Make Code Changes with Reviewable Diffs

Learn how to make code changes using reviewable diffs, allowing you to approve, reject, or apply edits easily. The process ensures changes are clear and reversible, with automation options for trusted projects.

AUTHOR

Paul Flanders

PUBLISHED

10 Sep 2026

DOCUMENT

InfoHub — Knowledge, news, and updates from
EssingtonITS

How to Make Code Changes with Reviewable Diffs

Paul Flanders · 10 Sep 2026

Estimated time: 4 minutes

Difficulty: Beginner

Why you'd use this

Asking for a change is the core workflow: you describe what you want, the assistant reads the relevant files and proposes edits, and nothing touches disk until you approve it. Each proposed create, edit or delete arrives as a card with the diff, so you review exactly what will change. Once you trust the assistant on a project, a per-repository toggle lets approved edits land without a click per file.

Before you start

Permissions required:

- Write access to the workspace files.

You'll need:

- A clear description of the change and, ideally, the file or area it belongs to.
- A clean or committed working tree, so you can tell the assistant's changes apart from your own.

Steps

1. In the chat, describe the change, for example "Add input validation to the signup form so the email field rejects addresses without an @." Press **Enter**.
2. Watch the activity lines: the assistant reads the files involved before proposing anything.
3. Each proposed change appears as a card showing the file path and the diff. Existing files are changed with small search-and-replace edits rather than a full rewrite, so the diff is short.
4. Click **Apply** to write the change, **Copy** to put the new content on the clipboard, or **Reject** to discard it. Rejected changes are reported back to the assistant so it does not assume they landed.
5. After applying, the file is syntax-checked and the project's linter runs on it. Any problem is fed straight back and the assistant proposes a fix.
6. To open each changed file in the editor as it is applied, click ⚙️ and tick **Open changed files after applying**.
7. To skip the per-card click on a project you trust, click the ⏏️ **Auto** chip under the composer. Approved edits are then written as they arrive.

What you should see

A checkpoint line appears for each applied file, meaning a snapshot was taken so the change can be rolled back later. When the request finishes, a closing line summarises what actually changed on disk, taken from the record of applied edits rather than from the assistant's own description.

Troubleshooting

- **An "Edit miss" line appears:** the text the assistant tried to replace no longer matches the file, usually because it changed since it was read. The assistant re-reads

and retries automatically; if it keeps missing it is asked to resend the whole file.

- **A card is marked stale:** the file changed while the card was waiting. Ask for the change again so it is proposed against the current content.
- **A rewrite arrives as a card even with Auto on:** a full-file replacement that removes most of a file's code, drops exported symbols or matched only approximately is always held for review.
- **You want the changes gone:** see the guide on reverting a request or restoring a checkpoint.

Related guides

- How to Roll Back Changes with Revert and Checkpoints
- How to Run Commands from the Chat
- How to Use the Command Menu and Composer Toggles