

yimatt/resources · Vercel

# Getting started with Vercel

by yimatt.com · 2 min · updated 2026-10-28 · beginner

## What you'll do

In this guide you will set up a clean, reproducible Vercel workflow from the official documentation, verify your local configuration, and run a minimal working check so you can build with confidence.

## Prerequisites

- Access to the official documentation at [Vercel \(https://vercel.com\)](https://vercel.com).
- A terminal and a modern code editor such as [VS Code](#) or [Neovim](#).
- Basic familiarity with command-line navigation and version control with [Git](#).

## Step-by-step setup

1. Verify the official source and release channel. Open [Vercel Documentation](#) and confirm you are reading the current stable release notes rather than outdated third-party blog posts.
2. Initialize a minimal workspace. Create a dedicated directory for testing Vercel in isolation before integrating it into a larger repository:

code

```
mkdir -p ~/projects/vercel-starter
cd ~/projects/vercel-starter
git init
```

**TIP**

Keep a small `README.md` in your starter folder recording the exact version and flags you used so future projects can copy a known-good baseline.

3. Configure strict defaults early. Create your baseline configuration file and commit it before adding optional plugins or third-party dependencies:

code

```
{
  "resource": "vercel",
  "type": "platform",
  "strict": true,
  "verified": "2026-10-28"
}
```

4. Run a smoke check. Execute the minimal build, query, or verification step described in [Vercel Documentation](#) and confirm zero warnings in your terminal output.

**NOTE**

Per-PR preview URLs make design and QA reviews effortless. Just watch the non-commercial restriction on the Hobby tier and set spend limits on Pro.

## Common mistakes

**COMMON MISTAKE**

Copy-pasting unpinned snippets from older tutorials. Always check the version selector in [Vercel Documentation](#) first. Major releases frequently simplify configuration and deprecate legacy flags.

**WARNING**

Avoid installing five companion libraries or extensions on day one. Run Vercel with its built-in defaults for a full day before adding abstractions.

## Next steps

- Bookmark [Vercel Documentation](#) for daily API lookups.
- Return to the [Vercel resource page](#) to watch the curated explainers or explore paired tools.

A note from yimatt.com

Field notes from using Vercel in production projects.

---

© 2026 yimatt.com · Source:

<https://resources.yimatt.com/r/vercel/guides/getting-started/>