A **Google** Antigravity   ≡

Tools > MCP

Antigravity Editor: MCP Integration

Antigravity supports the Model Context Protocol (MCP), a standard that allows the editor to securely connect to your local tools, databases, and external services. This integration provides the AI with real-time context beyond just the files open in your editor.

What is MCP?

MCP acts as a bridge between Antigravity and your broader development environment. Instead of manually pasting context (like database schemas or logs) into the editor, MCP allows Antigravity to fetch this information directly when needed.

Core Features

1. Context Resources

The AI can read data from connected MCP servers to inform its suggestions.

Example: When writing a SQL query, Antigravity can inspect your live Neon or Supabase schema to suggest correct table and column names.

Example: When debugging, the editor can pull in recent build logs from Netlify or Heroku.

2. Custom Tools

MCP enables Antigravity to execute specific, safe actions defined by your connected servers.

Example: "Create a Linear issue for this TODO."

Example: "Search Notion or GitHub for authentication patterns."

How to Connect

Connections are managed directly through the built-in MCP Store.

1. Access the Store: Open the MCP Store panel within the "..." dropdown at the top of the editor's side panel.

2. Browse & Install: Select any of the supported servers from the list and click Install.

3. Authenticate: Follow the on-screen prompts to securely link your

![Google Antigravity] Google Antigravity ☰

Tools > MCP

Connecting Custom MCP Servers

To connect to a custom MCP server:

1. Open the MCP store via the "..." dropdown at the top of the editor's agent panel.
2. Click on "Manage MCP Servers"
3. Click on "View raw config"
4. Modify the mcp_config.json with your custom MCP server configuration.

Supported Servers

The MCP Store currently features integrations for:

- Airweave
- Arize
- AlloyDB for PostgreSQL
- Atlassian
- BigQuery
- Cloud SQL for PostgreSQL
- Cloud SQL for MySQL
- Cloud SQL for SQL Server
- Dart
- Dataplex
- Figma Dev Mode MCP
- Firebase
- GitHub
- Harness
- Heroku
- Linear
- Locofy
- Looker
- MCP Toolbox for Databases

Tools > MCP

- Notion
- PayPal
- Perplexity Ask
- Pinecone
- Prisma
- Redis
- Sequential Thinking
- SonarQube
- Spanner
- Stripe
- Supabase

‹ Sandboxing Terminal Commands

Artifacts ›