

Google Drive MCP Setup Guide

This MCP connector lets Claude (or any MCP client) pull files and metadata from your Google Drive.

It uses OAuth 2.0 so each user authorizes with their own Google account.

1. Prerequisites

- Node.js ≥ 18 installed (`node -v` to check).
 - Claude Desktop installed (if you want to integrate into Claude).
 - A Google account with Drive access.
 - Access to this project folder.
-

2. Enable Google Drive API

1. Go to Google Cloud Console.
 2. Create a **new project** (or select an existing one).
 3. Navigate: **APIs & Services** → **Library** → enable **Google Drive API**.
-

3. Configure OAuth Consent

1. In the same project: **APIs & Services** → **OAuth consent screen**.
2. User type: **External**.
3. Fill basic info (App name, email, etc.).

4. Scopes:

- `https://www.googleapis.com/auth/drive.readonly`

5. Add your Google account as a **Test User**.

6. Save.

4. Create OAuth Credentials

1. Go to **APIs & Services** → **Credentials** → **Create Credentials** → **OAuth client ID**.
 2. Application type: **Desktop app**.
 3. Download the JSON file → rename it to `credentials.json` → put it in the `gdrive/` folder.
 4. Note the `client_id` and `client_secret` (you'll need them in `.env`).
-

5. Set Environment Variables

In the root of your project (`MCP_POC`), create or edit `.env`:

```
GOOGLE_CLIENT_ID=your_client_id_here
GOOGLE_CLIENT_SECRET=your_client_secret_here
GOOGLE_REDIRECT_URI=http://127.0.0.1:5555/callback
GOOGLE_SCOPES=https://www.googleapis.com/auth/drive.readonly
PORT=5555
```

6. Install Dependencies

From the project root:

```
npm install
```

7. Run the Google Drive MCP

```
npm run dev
```

The first time, it will:

- Print an **authorization URL**.
- You open it in your browser, sign in, and allow access.
- Google redirects to `http://127.0.0.1:5555/callback?code=...`
- The MCP exchanges the code for tokens and saves them (e.g. in `token.json`).

From then on, it will use the saved refresh token automatically.

8. Integrate with Claude Desktop

Edit this file (create if missing):

```
~/Library/Application Support/Claude/claude_desktop_config.json
```

Add this block under `"mcpServers"`:

```
"gdrive": {
  "command": "node",
  "args": [". /node_modules/.bin/tsx", "src/gdrive/server.ts"],
  "cwd": "/Users/YOURNAME/Desktop/MCP_POC",
  "env": {
    "GOOGLE_CLIENT_ID": "your_client_id_here",
    "GOOGLE_CLIENT_SECRET": "your_client_secret_here",
    "GOOGLE_REDIRECT_URI": "http://127.0.0.1:5555/callback",
    "GOOGLE_SCOPES": "https://www.googleapis.com/auth/drive.readonly",
    "PORT": "5555"
  }
}
```

```
}  
}
```

Restart Claude Desktop. Run `list tools` → you should see `gdrive.search` and `gdrive.get`.

9. Tools Exposed

- `gdrive.search({ q?: string, mimeTypes?: string[], max?: number })`
→ lists matching files with metadata.
 - `gdrive.get({ id: string })`
→ fetches file content (exported as text/csv where possible).
-

10. Troubleshooting

- **redirect_uri_mismatch** → Add `http://127.0.0.1:5555/callback` in Google Console under **Authorized redirect URIs**.
 - **403: user not in test group** → Add your email under “Test users” in the consent screen.
 - **App not verified** → Click *Advanced* → *Continue anyway* (since you’re the test user).
 - **No refresh token** → Revoke old tokens at <https://myaccount.google.com/permissions>, then re-run auth.
-

11. Security Note

- Never commit `token.json` or `.env` (keep them in `.gitignore`).
- Each user needs to authorize once with their own Google account.