

AIDD 30 Days Challenge Task: 6

Connecting GitHub MCP Server with Google Gemini CLI.

I'm writing the complete process as well so anyone new to it will be able to do the same.

Step 1:

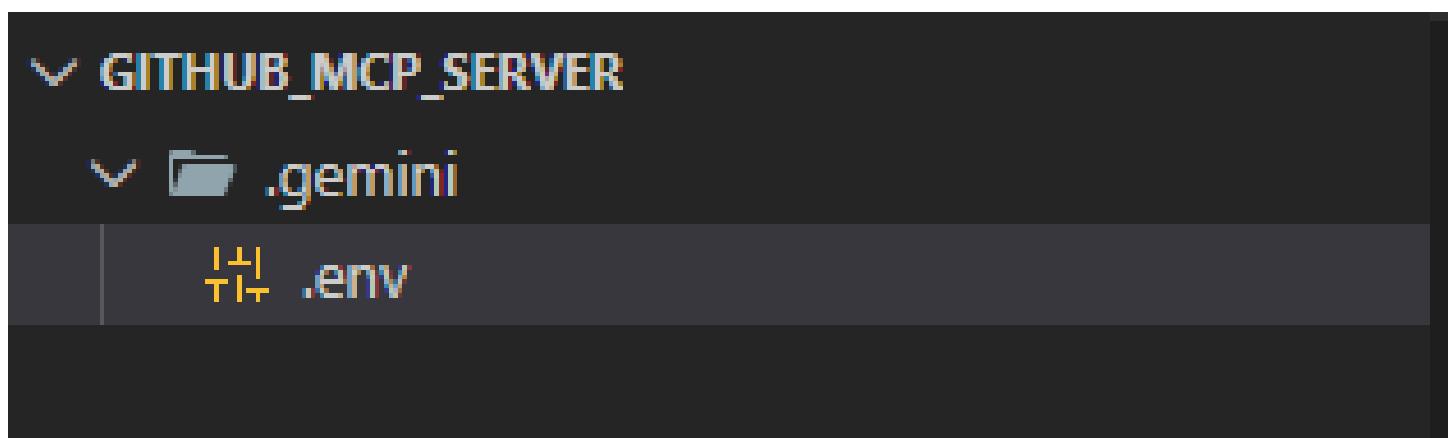
Generation of PAT (Personal Access Token).

To generate it visit <https://github.com/settings/personal-access-tokens/new> and your token. Copy it and save it.

Step 2:

Store your token in .env:

Create a file **.env** in **.gemini** folder to store your PAT.



Now, add your personal access token in it like this:

```
Welcome .env .gemini > .env
1 GITHUB_MCP_PAT=github_pat
```

Step 3:

Configuring Gemini CLI with GitHub MCP Server.

Create a **settings.json** in **.gemini** file and paste the following code.

```
{
  "mcpServers": {
```

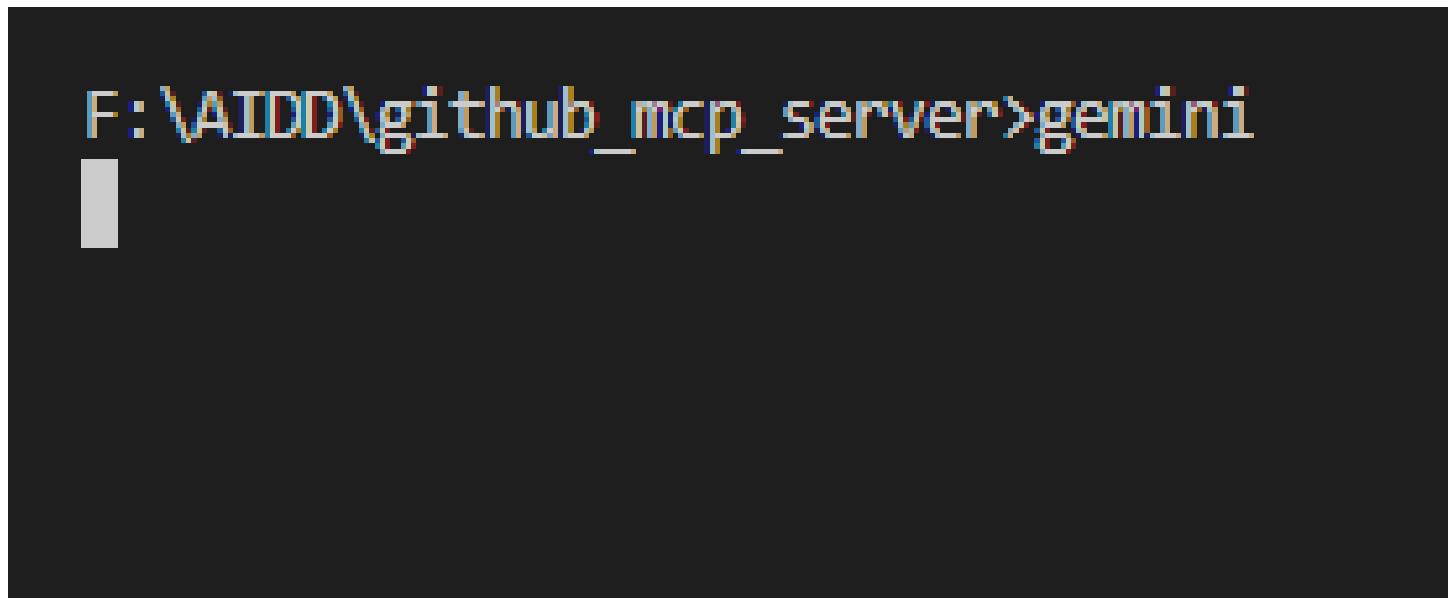
```
"github": {  
    "httpUrl": "https://api.githubcopilot.com/mcp/",  
    "headers": {  
        "Authorization": "Bearer $GITHUB_MCP_PAT"  
    }  
}  
}  
}
```

This will connect the GitHub MCP server with your Gemini CLI.

Step 4:

Start Gemini CLI.

Run gemini in you cmd.



The screenshot shows a dark-themed Windows Command Prompt window. The title bar reads 'cmd'. The command line shows the path 'F:\AIDD\github_mcp_server>gemini' followed by a cursor. The rest of the screen is blank, indicating no output has been displayed yet.

Step 5:

Verifying the connection.

Run /mcp list to check weather the MCP Server is connected or not.

Expected Result:

```
> /mcp list
```

Configured MCP servers:

● **github** - Ready (40 tools, 2 prompts)

Tools:

- `add_comment_to_pending_review`
- `add_issue_comment`
- `assign_copilot_to_issue`
- `create_branch`
- `create_or_update_file`
- `create_pull_request`
- `create_repository`
- `delete_file`

Step: 6:

Testing the Server.

Ask your CLI “List my GitHub Repositories.”

It will gather all your Repo names and using tools from GitHub MCP Server.

```
➔ Here are your GitHub repositories:  
- PYTHON-PROJECTS  
- codecommons  
- project-calculator  
- responsive-grid  
- AVION--E-COMMERCE-  
- Growth-Mindset-Challenge-Web-App-with-Streamlit  
- Secure-Data-Encryption-System-Using-Streamlit  
- mutahir-atm  
- mutahir-bmi-calculator  
- number-guessing-game  
- Static-Resume  
- responsive-figma-design  
- Marketplace-Technical-Foundation--AVION  
- project1-calculator  
- Kamrantessori.com-Clone  
- connect-sindh  
- simple-calculator02  
- simple-grid  
- kokab-cosmetics  
- portfolio  
- AVION-ADMIN-PANEL  
- openrouter-chatbot  
- UNIT-CONVERTER-WITH-LLM-INTEGRATION  
- GIAIC_Q4_AIDD_30_Days_Challenge  
- HTML-CSS-BOOK  
- Personal-Portfolio-Tailwind-CSS  
- PERSONAL-LIBRARY-MANAGER  
- Resume  
- figma-template-assignment  
- PASSWORD-STRENGTH-METER
```

Using: 1 GEMINI.md file | 1 MCP server

By the end of this Assignment:

In this assignment we learned how to connect GitHub MCP Server with our Gemini CLI. From Generating Personal Access Token PAT to have all our repo names.