

## Using SF-Github-Project-Helper

1. Create a folder to contain your project. This will contain the SF-Github-Project-Helper script as well as your actual project. Any test you see in commands that is enclosed in square brackets [] means to replace that with an appropriate value. Do not include the brackets in the command.

- a. Open a command prompt. In windows use the search box and type in 'cmd' (no quotes). Right click the entry that says cmd.exe and click 'Run as Administrator' (if possible).
- b. Navigate to where you would like to create your project folder by using the command

```
cd [folder name]
```

to move into sub directories and `cd ..` to move up in the directory tree. You may also use `cd \` to move to the root of drive.

- c. Use the command

```
md [folder name]
```

to create a new folder.

- d. Change directories into your newly created folder with

```
cd [folder name]
```

2. Download the utility by cloning the remote repository (repo). Use the command

```
3. git clone https://github.com/Kenji776/SF-Github-Project-Helper.git .
```

```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.19044.2364]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd \

C:\>md sf-project-demo

C:\>cd sf-project-demo

C:\sf-project-demo>git clone https://github.com/Kenji776/SF-Github-Project-Helper.git .
Cloning into '.'...
remote: Enumerating objects: 151, done.
remote: Counting objects: 100% (151/151), done.
remote: Compressing objects: 100% (102/102), done.
Receiving objects: 70% (106/151) 129 (delta 44), pack-reused 0 receiving objects: 69% (105/151)
Receiving objects: 100% (151/151), 88.13 KiB | 1.55 MiB/s, done.
Resolving deltas: 100% (62/62), done.

C:\sf-project-demo>node SF-Github-Project-Helper

```

4.

5. Open the config.json file and populate the attributes with the ones from your Salesforce and Github project. If you have not yet generated a Personal Access Token you will need to. You can create one at <https://github.com/settings/tokens/new>. If your Salesforce org is using mydomain, you must change the salesforceLoginUrl to [https://\[your domain here\].my.salesforce.com](https://[your domain here].my.salesforce.com). You can find this value in the Salesforce setup menu under 'my domain'. EX:

```

{
  "skipExistingChangeSets": true,
  "rootFolder": "packages",
  "changesetJSONFile": "changeSetNames.json",
  "downloadedPackagesFolder": "downloaded_packages",
  "salesforceUsername": "[redacted]@gmail.com",
  "githubRepoUrl": "https://github.com/Kenji776/Demo-SF-Repo.git",
  "gitUsername": "[redacted]",
  "projectName": "Demo_Project",
  "salesforceLoginURL": "https://d400000007rz2eae-dev-ed.my.salesforce.com",
  "githubPersonalAccessToken": "[redacted]",
  "autoCreatePullRequest": true
}

```

6.

7. Start the SF-Github-Project-Helper by either using the included .bat file (windows only) or using the command

a. `node SF-Github-Project-Helper`

8. Setup your project by selecting the Config Wizard

```
Please select option
0) Config Wizard (Import Settings from config file)
1) Connect GIT Repo
2) Display GIT Repo Information
3) Setup SFDX Project
4) Display SFDX Information
5) Push Changesets to GIT from config file
6) Push Changesets to GIT by entering names
7) Push Package.xml file contents to GIT
8) View Config File Information
9) Authorize Github CLI
10) Exit

Enter Selection: 0
Authorizing Github connection with personal access token: [REDACTED] 7bG
Cloning git repo into C:\sf-project-demo
Repo location set to: https://github.com/Kenji776/Demo-SF-Repo.git
Setting up Salesforce DX Project Demo_Project
target dir = C:\sf-project-demo

create Demo_Project\config\project-scratch-def.json
create Demo_Project\README.md
create Demo_Project\sfdx-project.json
create Demo_Project\manifest\package.xml
create Demo_Project\.husky\pre-commit
create Demo_Project\.vscode\extensions.json
create Demo_Project\.vscode\launch.json
create Demo_Project\.vscode\settings.json
create Demo_Project\force-app\main\default\lwc\eslinttrc.json
create Demo_Project\force-app\main\default\aura\eslinttrc.json
create Demo_Project\scripts\soql\account.soql
create Demo_Project\scripts\apex\hello.apex
create Demo_Project\.eslintignore
create Demo_Project\.forceignore
create Demo_Project\.gitignore
create Demo_Project\.prettierignore
create Demo_Project\.prettierrc
create Demo_Project\jest.config.js
create Demo_Project\package.json



















Authorizing Org Demo_Project. Wait for browser window to open and login...
Successfully authorized [REDACTED]@gmail.com with org ID 00D400000007Rz2EAE

Salesforce connected and git repo configured!
```

9.

10. You should now have a folder that looks like this. Ensure that the .git folder is there. It is a hidden folder so you will have to enable viewing of hidden files and folders to

confirm. The cloning of the repo can fail silently, if that happens the .git folder is not created and the linkage is not complete.

> sf-project-demo > Demo_Project		
Name	Date modified	Type
 .git	2/13/2023 6:21 PM	File folder
 .husky	2/13/2023 6:21 PM	File folder
 .vscode	2/13/2023 6:21 PM	File folder
 config	2/13/2023 6:21 PM	File folder
 force-app	2/13/2023 6:21 PM	File folder
 manifest	2/13/2023 6:21 PM	File folder
 scripts	2/13/2023 6:21 PM	File folder
 .eslintignore	2/13/2023 6:21 PM	ESLINTIGNORE File
 .forceignore	2/13/2023 6:21 PM	FORCEIGNORE File
 .gitignore	2/13/2023 6:21 PM	GITIGNORE File
 .prettierignore	2/13/2023 6:21 PM	PRETTIERIGNORE ...
 .prettierrc	2/13/2023 6:21 PM	PRETTIERRC File
 config.json	2/13/2023 6:21 PM	JSON File
 jest.config.js	2/13/2023 6:21 PM	JavaScript File
 log.txt	2/13/2023 6:21 PM	Text Document
 package.json	2/13/2023 6:21 PM	JSON File
 README.md	2/13/2023 6:21 PM	MD File
 sfdx-project.json	2/13/2023 6:21 PM	JSON File

a.