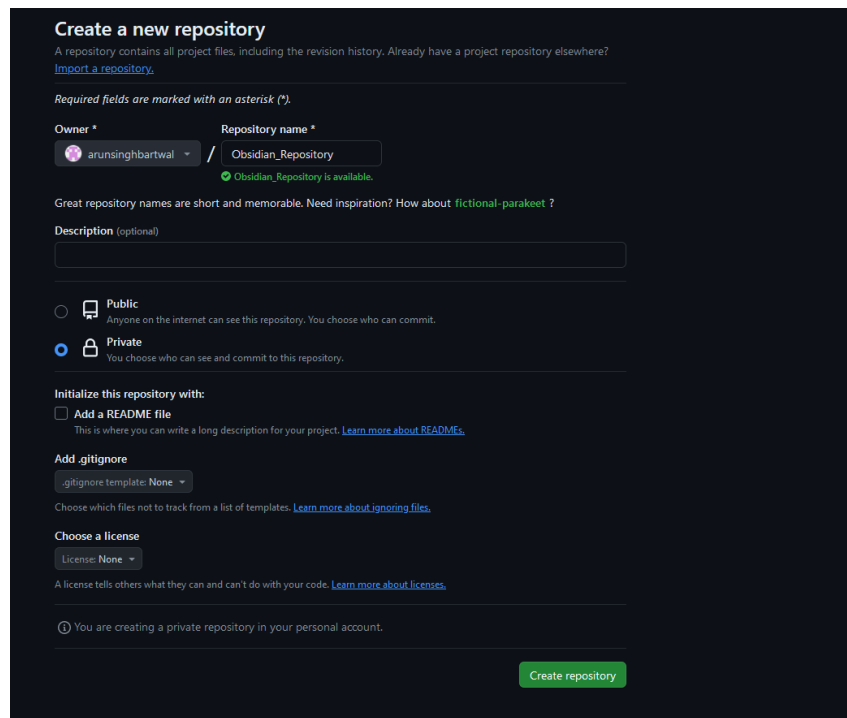


Configuring Obsidian on Your Windows PC

1. Create a Private Repository on GitHub:

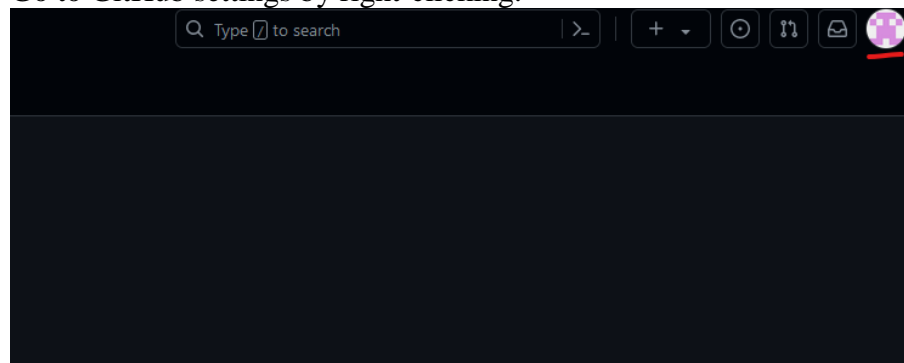
- Log in to your GitHub account and create a private repository.



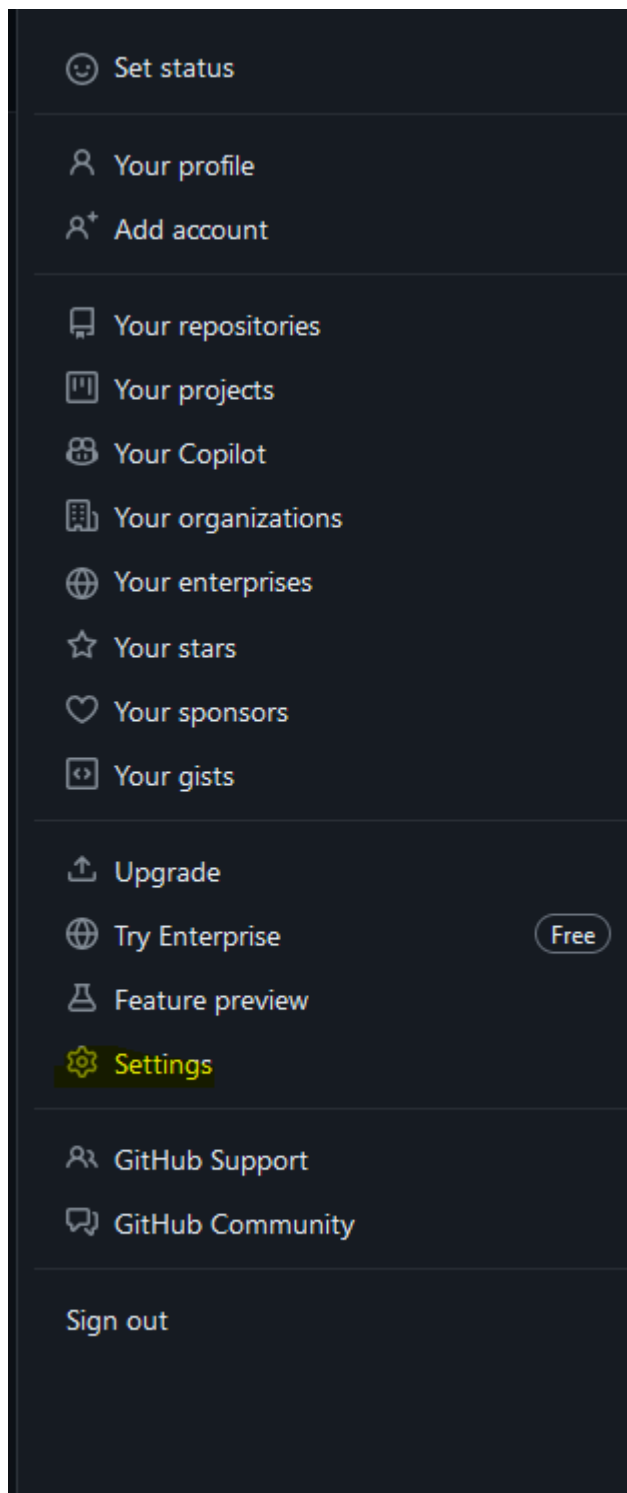
The screenshot shows the GitHub 'Create a new repository' page. At the top, it says 'Create a new repository' and 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. Below this, it states 'Required fields are marked with an asterisk (*)'. The 'Owner' field is set to 'arunsinghbartwal' and the 'Repository name' field is 'Obsidian_Repository', with a green checkmark indicating 'Obsidian_Repository is available.'. There is a note: 'Great repository names are short and memorable. Need inspiration? How about fictional-parakeet?'. The 'Description' field is optional and empty. Under 'Visibility', the 'Private' option is selected with a blue circle, while 'Public' is unselected. Below this, it says 'Initialize this repository with:' and 'Add a README file' is checked, with a note: 'This is where you can write a long description for your project. [Learn more about READMEs.](#)'. The 'Add .gitignore' section shows '.gitignore template: None' selected. The 'Choose a license' section shows 'License: None' selected. At the bottom, there is a note: 'You are creating a private repository in your personal account.' and a green 'Create repository' button.

2. Generate an Access Token:


- Go to GitHub settings by right-clicking.





- Select the settings option.





- Navigate to Developer Settings at the bottom.

 **Public profile**



 Account


 Appearance


 Accessibility


 Notifications


Access


 Billing and plans 


 Emails



 Password and authentication

 Sessions


 SSH and GPG keys


 Organizations


 Enterprises


 Moderation 


Code, planning, and automation


 Repositories

 Codespaces


 Packages

 Copilot


 Pages


 Saved replies

Security


 Code security and analysis

Integrations


 Applications

 Scheduled reminders

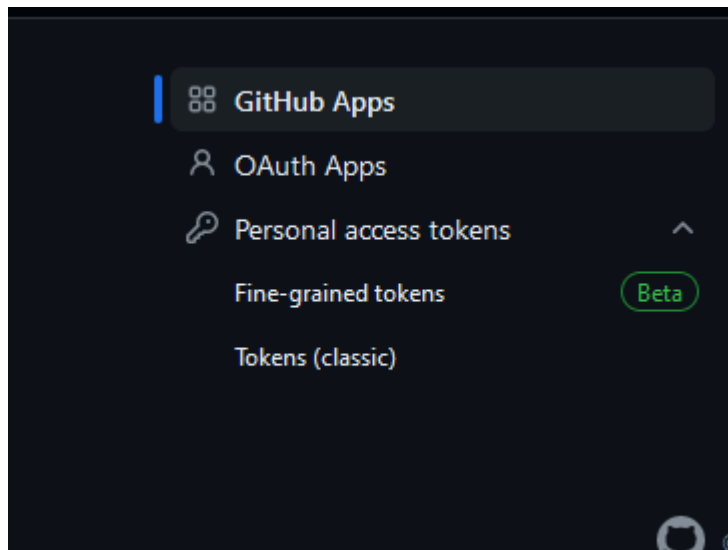
Archives

 Security log

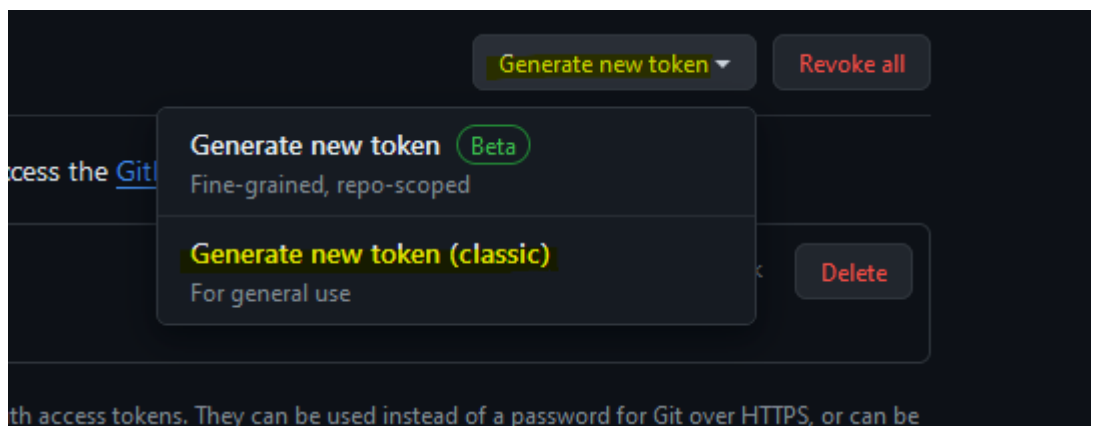
 Sponsorship log

 **Developer settings**

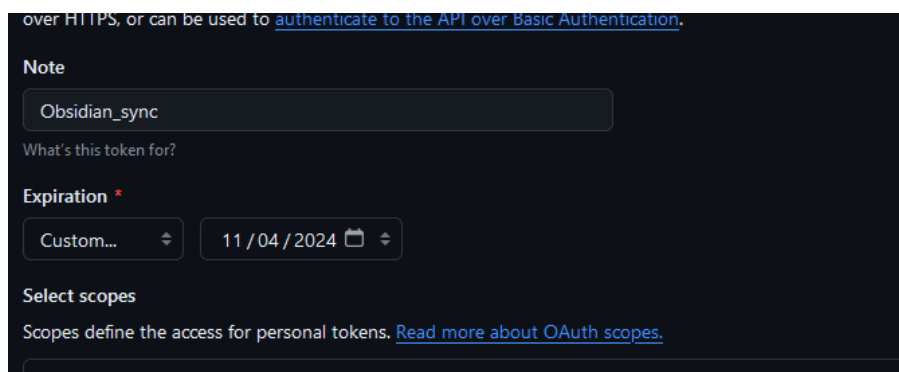
- Go to Personal Access Tokens → Tokens Classic.



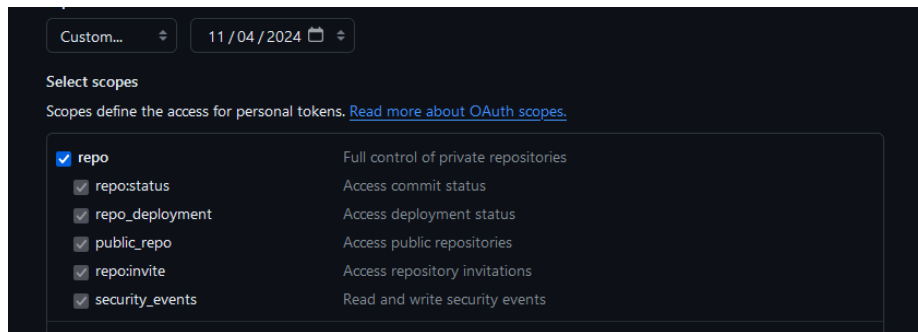
- Click on Generate New Token → Generate new Token (Classic).



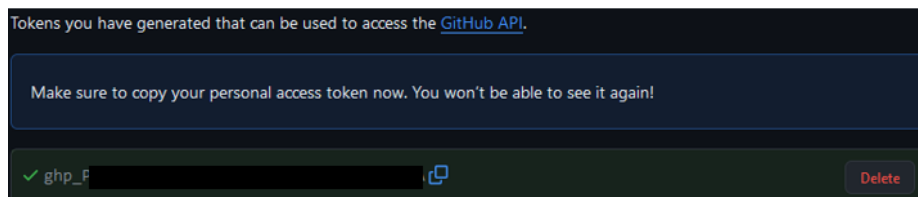
- Give it a name and select an expiration date (it's recommended to use a token for a month at max).



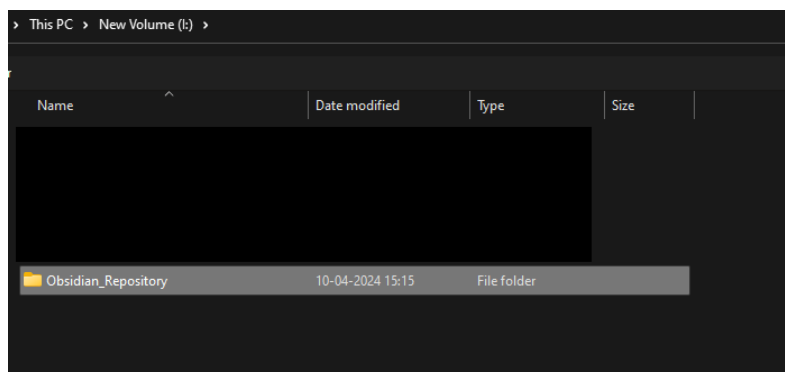
- Grant it rights to all repo options (you can also generate and configure fine-grain tokens to restrict user access further).



- Copy the generated token and keep it handy for plugin authentication on mobile devices.



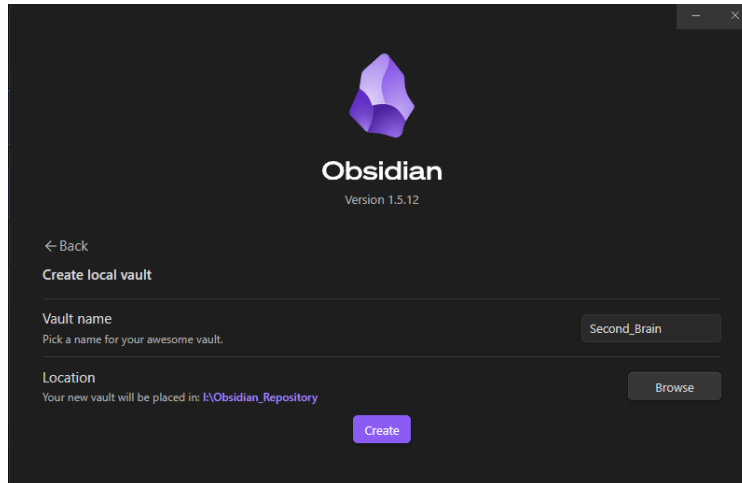
- 3. Install Obsidian and Git Bash:**
 - Download the files from the internet & install them
- 4. Create a New Vault in Obsidian:**
 - Name it the same as your GitHub Repository (e.g., Obsidian_Repository).



- Click on Select Folder

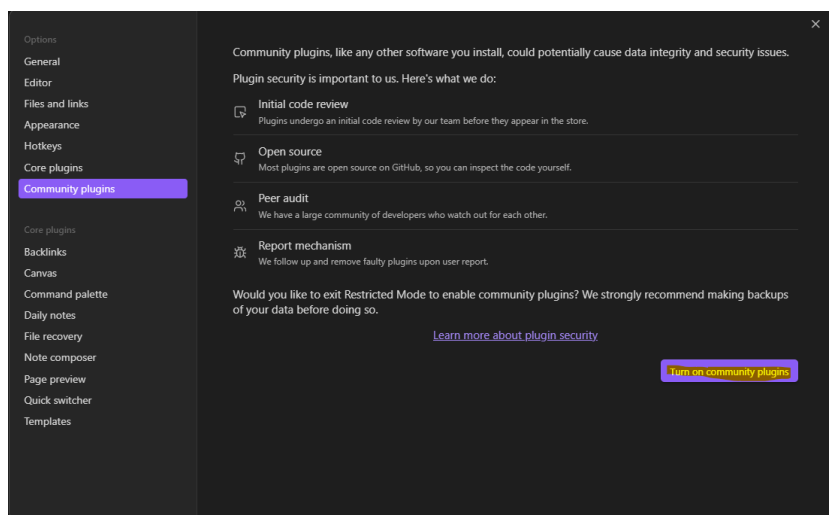


- Give it a name and select create.

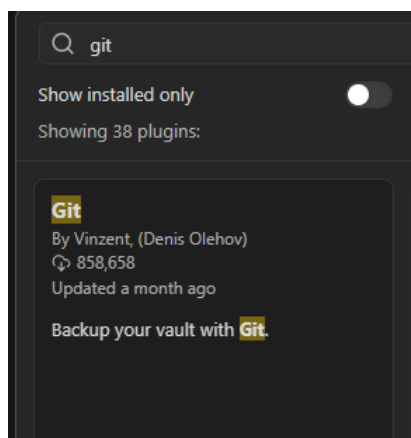


5. Enable Community Plugins:

- Go to Settings and enable Community Plugins.



- Install the Git Plugin by browsing the community plugins.

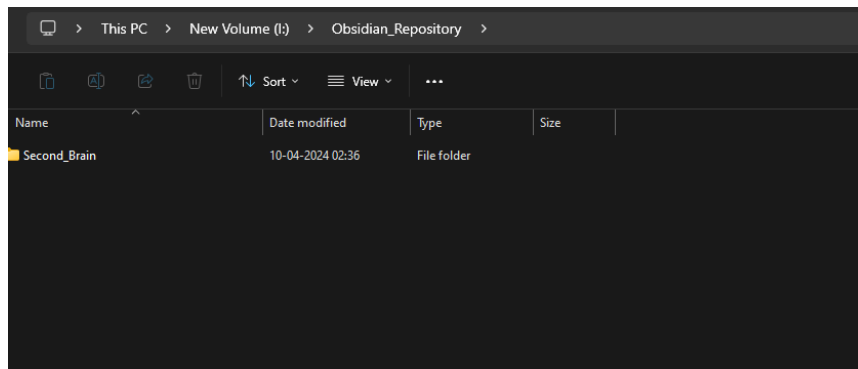


6. Configuration of Git Plugin:

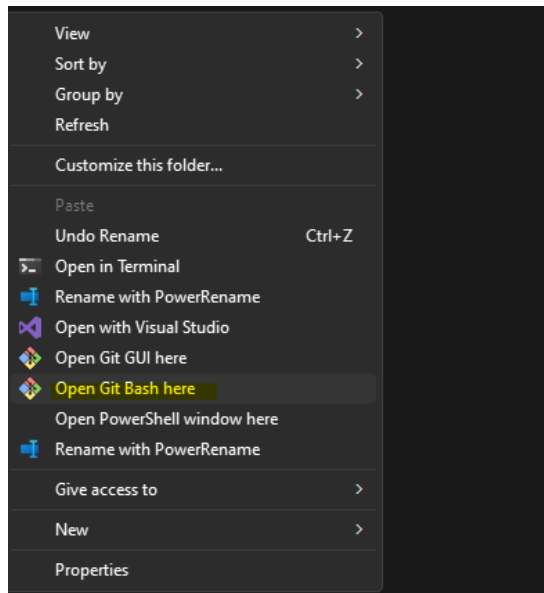
- Enable the Git plugin.



- Browse the folder where you have created the vault using Windows Explorer or CMD.

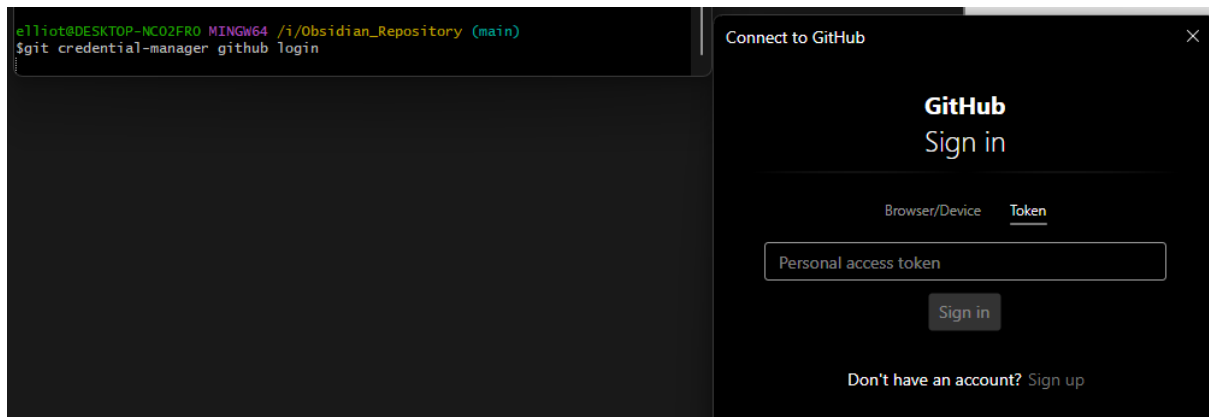


- If you have Git Desktop installed, you can use it as well.
- Use ctrl+shift+right-click to access more options on the folder and open Git Bash or CMD.



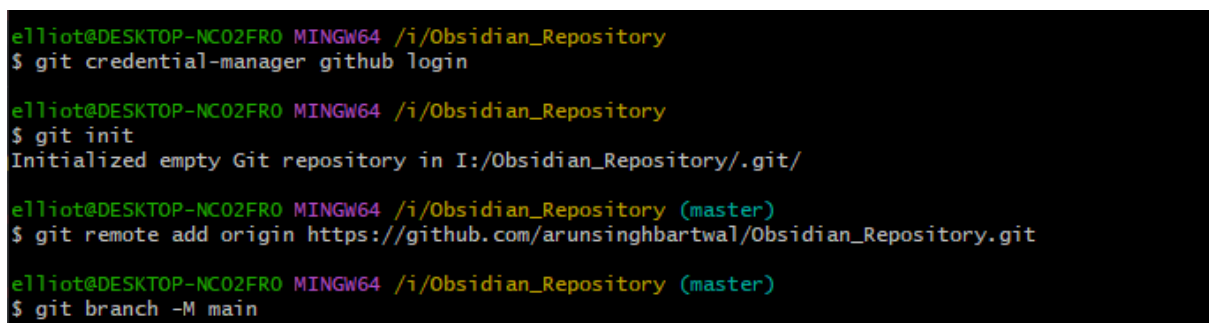
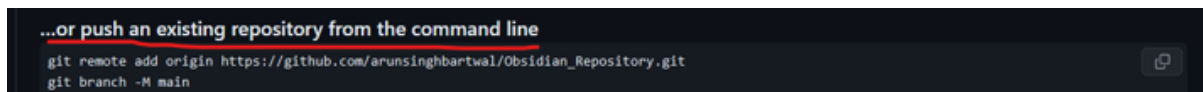
7. Run the Following Commands:

- ```
bash
```
- `git credential-manager github login`
- ```
# Sign in using the token generated in Step 2
```



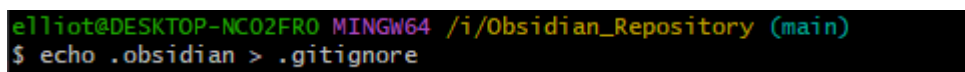
git init

Run the first 2 commands to add your remote repository

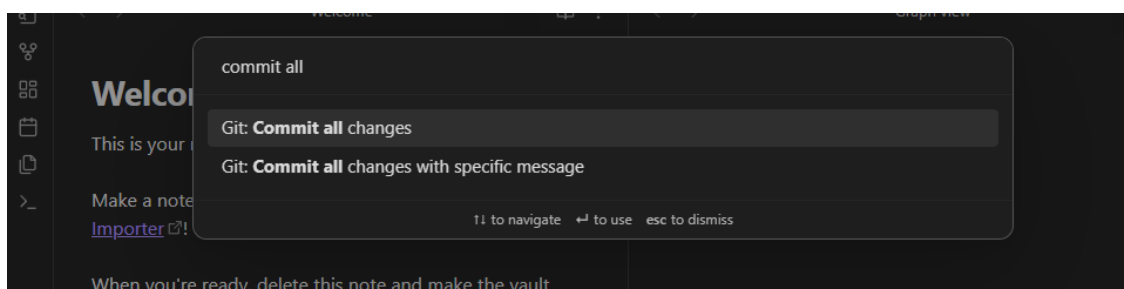


As we are syncing on multiple platforms hence we have to not sync our .obsidian folder as it may break the format on other devices (recommended if using cross platform devices like Android, IOS, Linux, Windows)

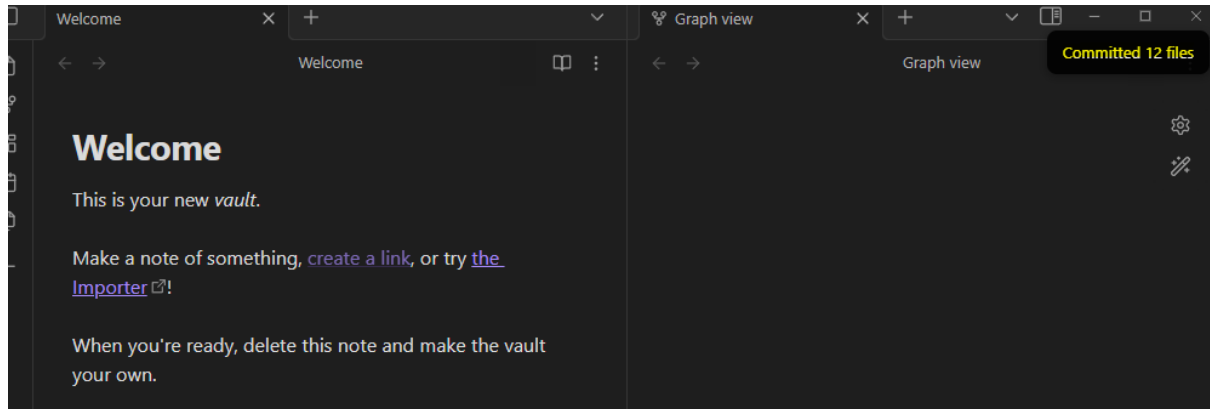
For that we will be creating a .gitignore file using the git bash
Run the below command
echo .obsidian > .gitignore



Commit by searching it on obsidian command palette.



Wait for this notification



Add your Author Name and email in the Git Plugin (if you have set up the email and author within git variables that can work)

Author name for commit

Author email for commit

Now run the last command from the github

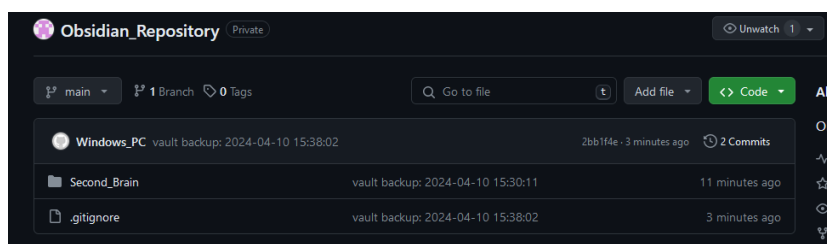
```
..or push an existing repository from the command line

git remote add origin https://github.com/arunsinghbartwal/Obsidian_Repository.git
git branch -M main
git push -u origin main
```

```
elliott@DESKTOP-NC02FRO MINGW64 /i/Obsidian_Repository (main)
$ git push -u origin main
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 32 threads
Compressing objects: 100% (15/15), done.
Writing objects: 100% (21/21), 287.51 KiB | 7.77 MiB/s, done.
Total 21 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/arunsinghbartwal/Obsidian_Repository.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

• Verify Pushed Files:

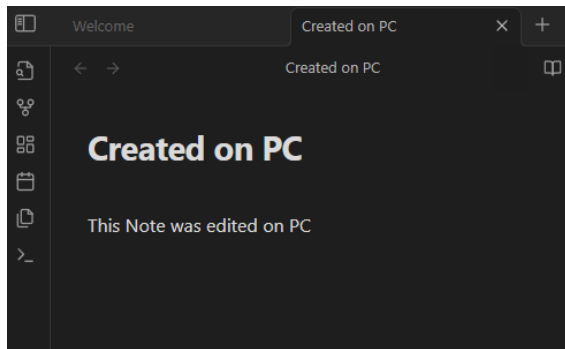
- Make sure .obsidian folder is not synced.



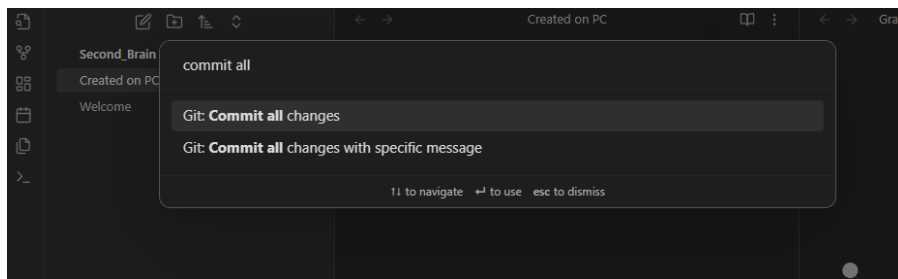
Restart Obsidian on PC to synchronize in case it is missing the new set variables (recommended)

- Syncing first doc

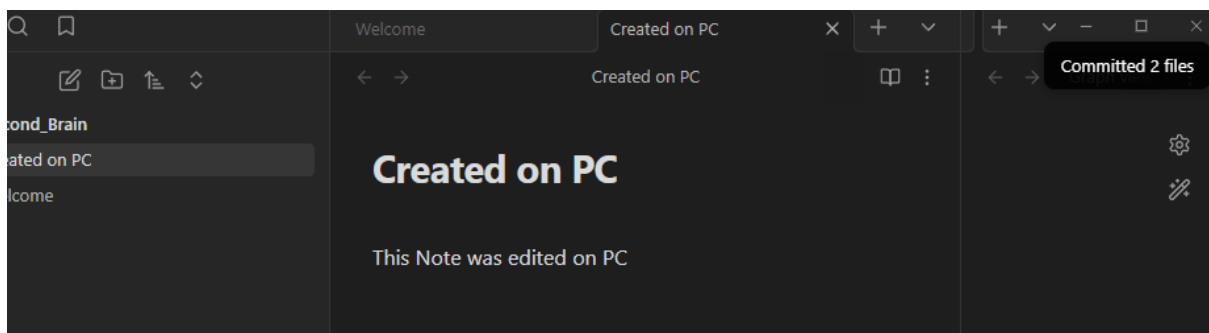
Create a New Note



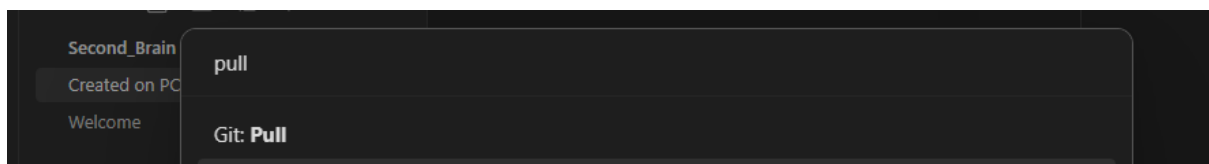
Commit the changes



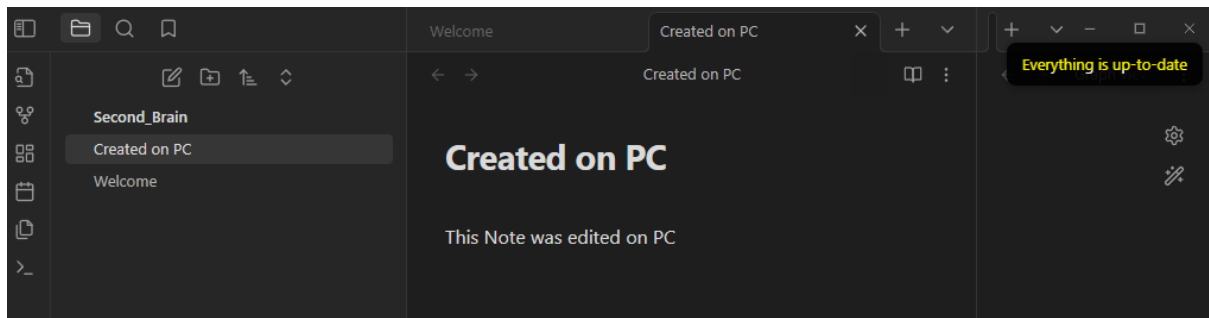
Wait for commit notification



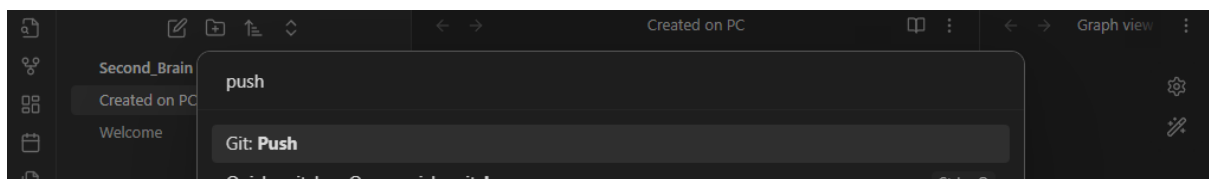
Pull the repository before pushing



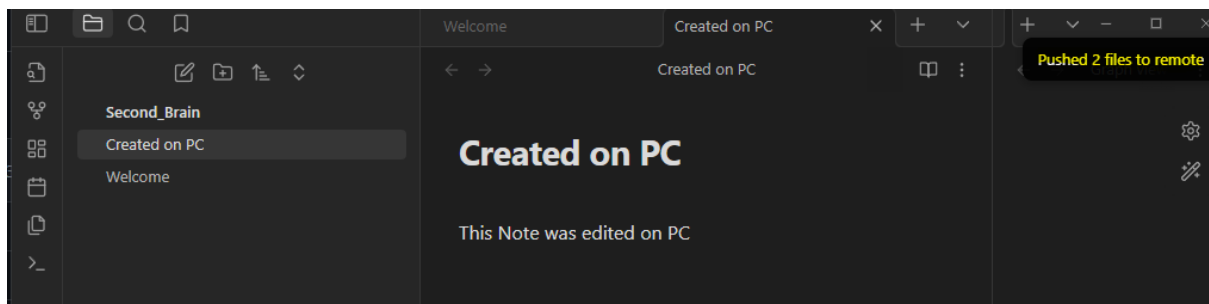
Wait for the notification



Push the changes



Wait for the notification



- **Check Repository Online:**
 - Refresh the folder on the web to check for the new files.

Windows_PC vault backup: 2024-04-10 15:45:35		d2de926 · 2 minutes ago	History
Name	Last commit message	Last commit	
..			
.obsidian	vault backup: 2024-04-10 15:45:35	2 minutes	
Created on PC.md	vault backup: 2024-04-10 15:45:35	2 minutes	
Welcome.md	vault backup: 2024-04-10 15:30:11	18 minutes	

You can now enable Autocommit and autopull of the Plugin

Now we can start syncing other Devices

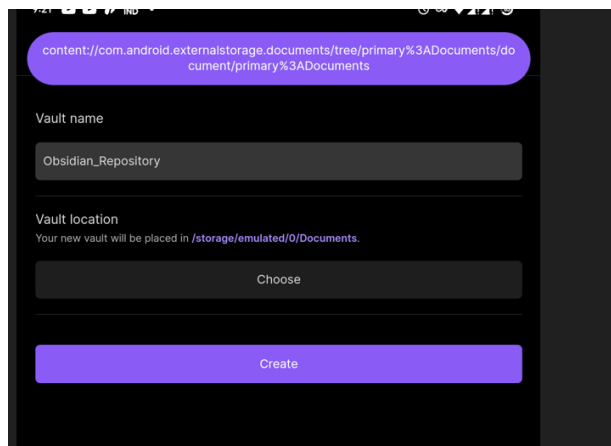
Android

1. Install the Application:

- Install the application on your Android device.

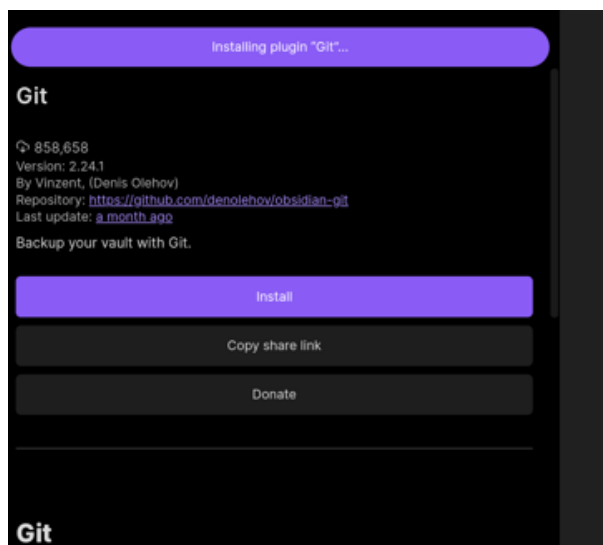
2. Create a Vault:

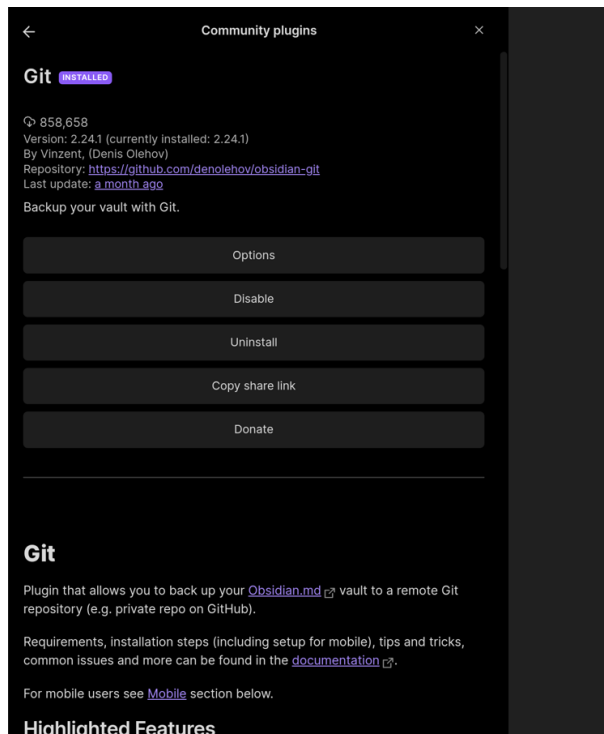
- Create a Vault on your Android and select the desired folder.



3. Enable Community Plugins:

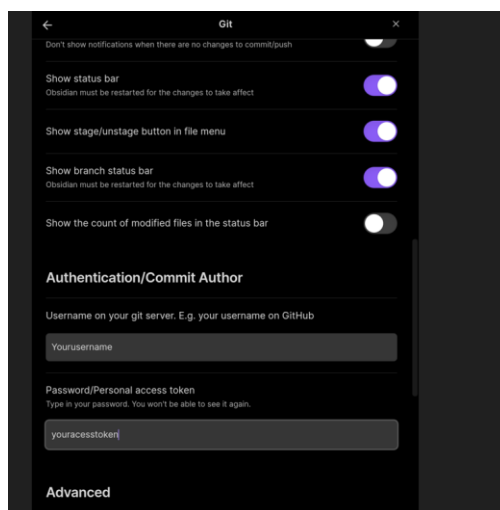
- Enable community plugins and search for "Git", then proceed to install it.





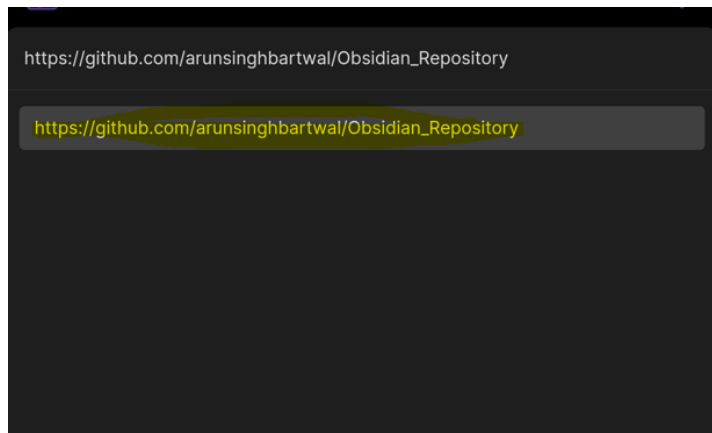
4. Configure Git:

- Fill in your Git username and access token in the respective fields under Options.

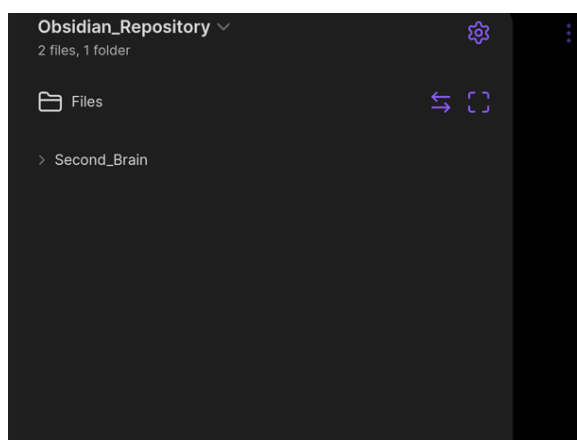
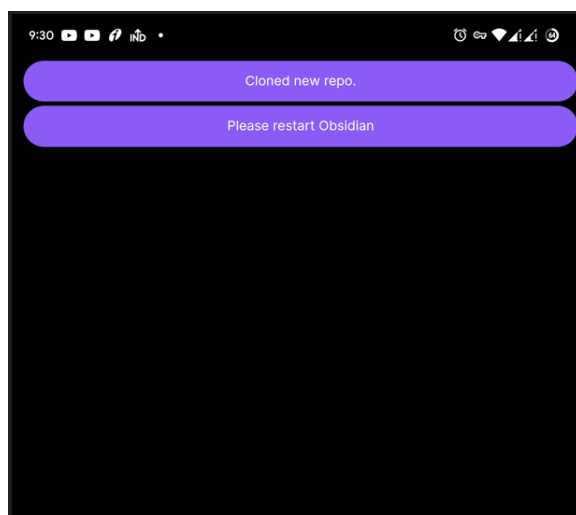


5. Clone Repository:

- Open the command palette by swiping from top to bottom or using the options on the bottom right.
- Search and select "Clone an Existing Repo".
- Enter the path of your repository.



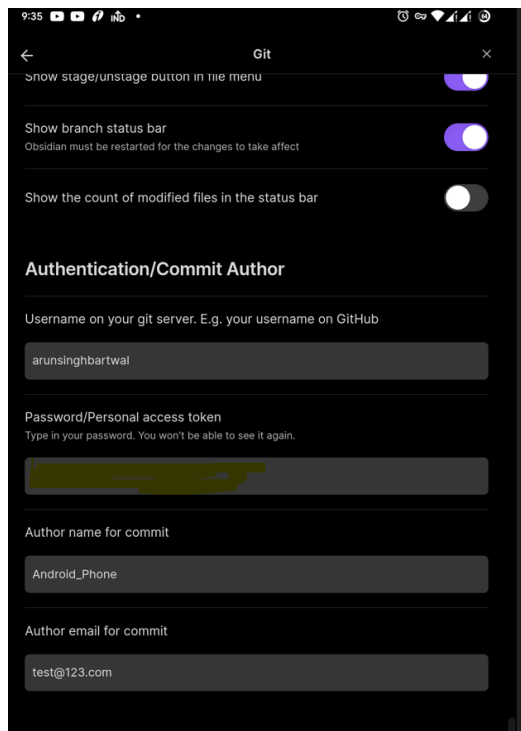
- Select the option "Vault Root".
- If the .obsidian directory hasn't been synced, select "NO", otherwise select "Yes".
- Click on the grey bar without inputting anything.
- Once you receive the notification to restart Obsidian, fully close the app and reopen it. Your repository files should now be cloned.



6. Configure Git Plugin for Autosync:

- Go to your Git plugin options.
- Scroll down after the Password field to find the Author Name and Email.

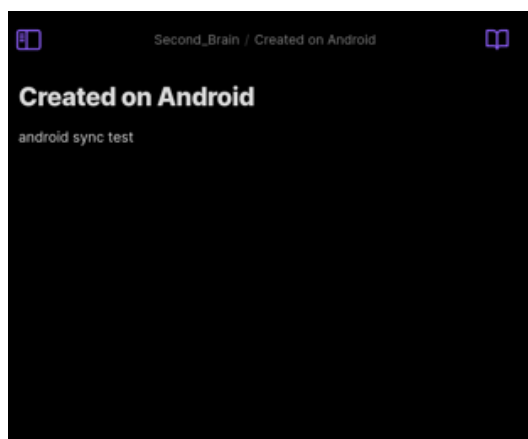
- **Note- Please overlook the blank Password/Auth token field as it is designed to not show the information**



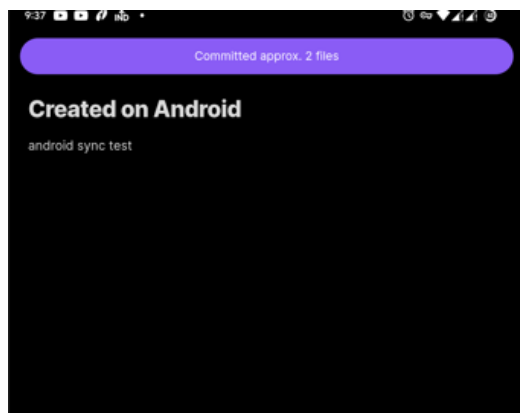
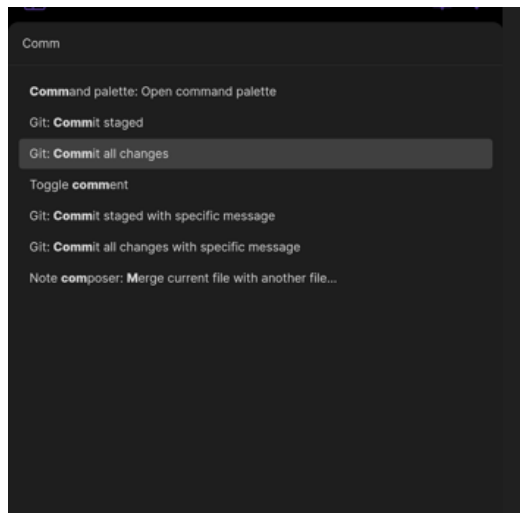
- Enter your device name (for clarity) and email address.
- Go back.

7. Test Sync:

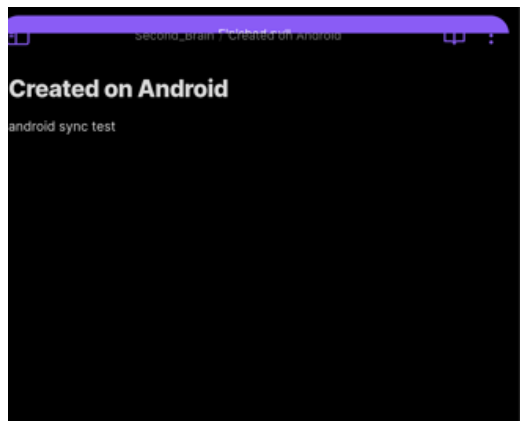
- Make changes in the vault to test synchronization.
- Create a note or folder.



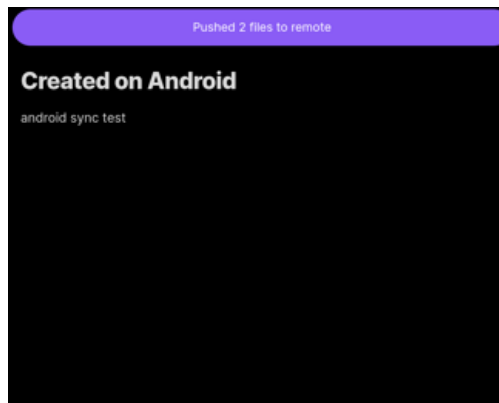
- Open the command palette and search for "Commit".
- Click on "Commit All Changes".



- Search and select in command palette of Obsidian "Git: Pull".



- Once the notification appears, search and select in command palette of Obsidian "Git: Push".



- Once the notification appears, verify the repository on GitHub.

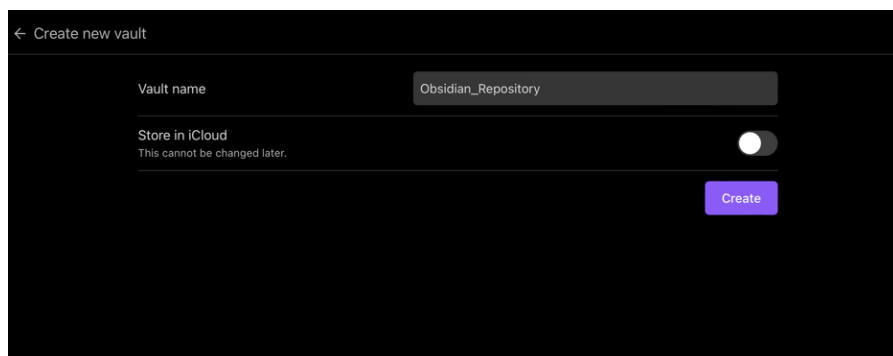
Name	Last commit message	Last commit date
..		
.obsidian	vault backup: 2024-04-10 15:45:35	23 minutes ago
Created on Android.md	vault backup: 2024-04-10 21:37:45	now
Created on PC.md	vault backup: 2024-04-10 21:37:45	now
Welcome.md	vault backup: 2024-04-10 15:30:11	39 minutes ago

- Now we can enable Git Plugin AutoSync.

iOS Device

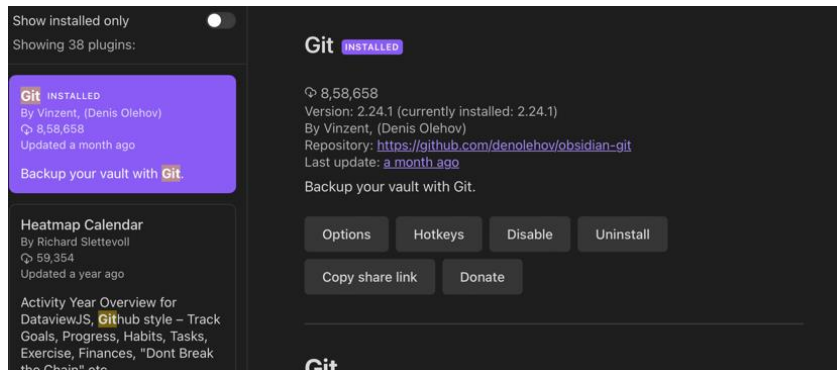
The steps are mostly the same for iOS:

1. **Install Obsidian:**
 - Install Obsidian from the App Store.
2. **Create Vault:**
 - Open Obsidian and create a new vault named "Obsidian_Repository" (or your preferred name).



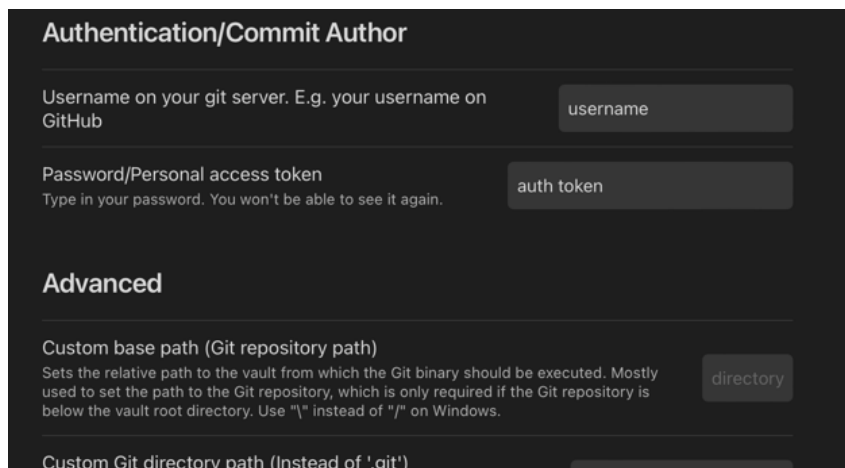
3. Enable Community Plugins and Git Plugin:

- Enable community plugins and install/enable the Git plugin.



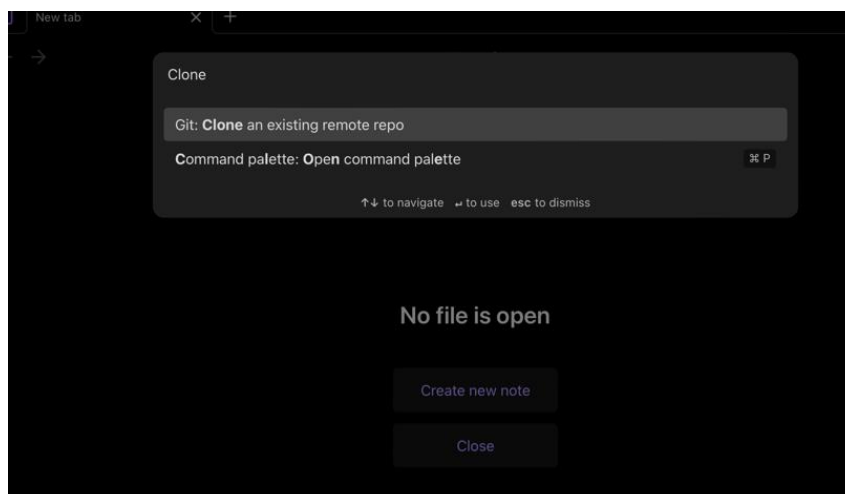
4. Configure Git:

- Go to Options and input your username and token.

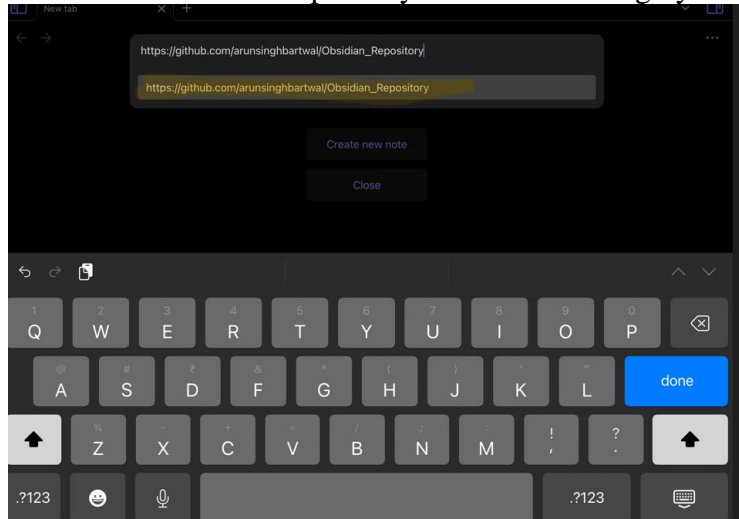


5. Clone Repository:

- Open the command palette and search for "Clone an Existing Remote Repo".



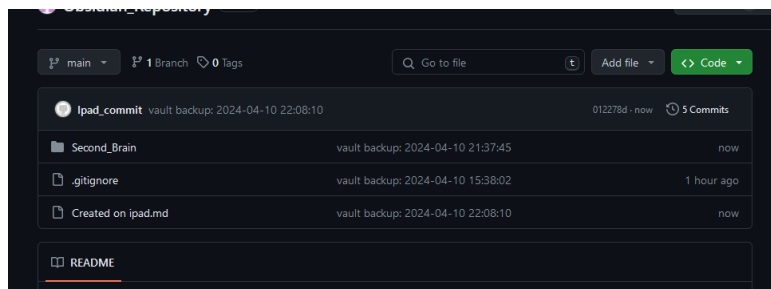
- Enter the URL of the repository and click on the greyed-out option.



- Follow the same steps as mentioned for Android for further configuration and setup.

6. Test Sync:

- Perform the same steps as mentioned for Android to test synchronization.



- Now we can enable Git Plugin AutoSync.