**Challenge 3 – Brain Dump/Notes**

*More notes on physical notebook*

**Help adding to GitHub – original empty files**

To add a folder to your local Git repository, you can follow these steps using Git Bash:

1. **Navigate to Your Local Repository**: Open Git Bash and navigate to the directory of your local Git repository where you want to add the new folder. You can use the cd command to change directories.
2. **Create a New Folder**: Use the mkdir command to create a new folder. For example, to create a folder named NewFolder, you can run:
3. mkdir NewFolder
4. **Check the Status**: Use the git status command to see the status of your repository. It will show you the untracked new folder.
5. **Add the New Folder**: To stage the new folder for the commit, use the git add command. For example, to add the NewFolder, run:
6. git add NewFolder
7. **Commit the Changes**: After adding the new folder, commit the changes to your local repository with a descriptive message using the git commit command. For example:
8. git commit -m "Add NewFolder to the repository"
9. **Verify the Changes**: You can use git status again to verify that the new folder has been added and committed.

By following these steps, you can add a new folder to your local Git repository using Git Bash.

**Help remembering how to loop through column to find unique months**

Here's an example of how you can loop through a column in a CSV file containing dates in the format "10-Jan" to find the number of unique months:

import csv

# Open the CSV file

with open('data.csv', 'r') as file:

reader = csv.DictReader(file)

# Create a set to store unique months

unique\_months = set()

# Loop through the rows in the CSV file

for row in reader:

date = row['date\_column']

month = date.split("-")[1]

unique\_months.add(month)

# Count the number of unique months

num\_unique\_months = len(unique\_months)

print("Number of unique months:", num\_unique\_months)

In this code snippet, we use the csv module to read the CSV file, loop through the rows, extract the month from the specified column, and store unique months in a set. Finally, we calculate the number of unique months in the column. You can run this code in VSCode to get the desired output without using Pandas.

***Had issues with path***

1. **Verify the File Structure**: Ensure that the folder structure of your project matches the path you are trying to create. Check if the budget\_data.csv file is located in the correct directory relative to your Python script.