I currently have a local folder structure of this form:

A screen shot of a computer

AI-generated content may be incorrect.

I am going to send it to an S3 bucket that exists (partha-program-css436) and also to an S3 bucket that does not exist and is created (pmalladi-created-css436):

A computer screen with text on it

AI-generated content may be incorrect.

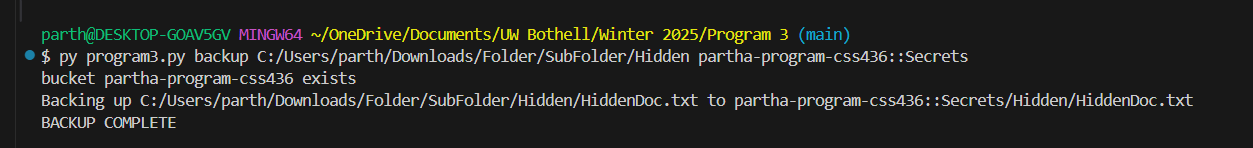
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

I can also create a new folder (Secrets) in AWS and backup the local sub folder called “Hidden” to it:



A screenshot of a computer

AI-generated content may be incorrect.

I am going to restore the contents from the S3 bucket “partha-program-css436” to my downloads folder on my local computer:

A screen shot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

I can also specify certain folders from AWS and can create local folders to store the bucket contents into using the terminal:

A black screen with colorful text

AI-generated content may be incorrect.

A close-up of a date

AI-generated content may be incorrect.

**Architecture and Design**

This program allows local file backup to AWS S3 and restoration from S3 to local storage using the boto3 library. I used a modular approach with three main functions: createBucket(), which checks if an S3 bucket exists and creates one if needed; backup(), which recursively uploads files from a local directory to a specified S3 bucket path while preserving the directory structure; and restore(), which downloads files from S3 back to a local directory, ensuring the folder hierarchy remains intact. The script operates via command-line arguments, allowing users to specify whether they want to back up or restore files, along with the necessary paths. It also ensures compatibility across different operating systems by handling file paths appropriately, converting Windows-style backslashes to Unix-style slashes for S3 compatibility. There is basic error handling in order to handle edge cases and errors but if execution fails, the user should simply retry the command based on the error they receive.