Topic 65 - Data Files: Appending Data

**What**

* This topic explains how to append data to a file without overwriting its existing content.

**Why**

* **Appending Data**: You may want to add new data to an existing file while keeping the old data intact (e.g., logs or records).

**How**

1. **Opening a File for Appending**  
   To append data to a file, you must open it in **append mode** ("a"):

python

Copy code

with open("greet.txt", "a") as f:

* + **"a"**: This mode opens the file for appending. If the file doesn’t exist, Python creates it.

1. **Writing to the File**  
   You can then append data to the file using the write() method. To add a new line before appending the text, use the newline character \n:

python

Copy code

with open("greet.txt", "a") as f:

f.write("\nHave a nice day!")

* + After this operation, the file content becomes:

bash

Copy code

Hello, World!

Have a nice day!

1. **Reading and Displaying the File**  
   After appending, you can read the file and see the appended data:

python

Copy code

with open("greet.txt") as f:

message = f.read()

print(message)

* + This will output:

bash

Copy code

Hello, World!

Have a nice day!

**Things to Remember**

* **"a" Mode**: The "a" mode allows you to append data to a file without removing its existing contents.
* **New Line Character (\n)**: The \n ensures that new data is placed on a new line. If you don’t include \n, the new data will be added at the end of the file without any separation.