**Module 6: File Handling**

File handling allows you to **create, read, write, and delete** files using built-in Python functions.

**Create example.txt file**

Go to VS Studio🡪click on new file🡪Give name like example3.txt🡪click enter

A text file created into the folder

**Opening a File**

file = open("example.txt", "mode")

**File handling Modes**

| **Mode** | **Description** |
| --- | --- |
| 'r' | Read (default) |
| 'w' | Write (creates or overwrites) |
| 'a' | Append (adds to end) |
| 'x' | Create (fails if exists) |
| 'b' | Binary mode (rb, wb, etc.) |
| 't' | Text mode (default) |
| '+' | Read and Write (r+, w+) |

**Reading Files**

file = open("example.txt", "r")

content = file.read() # Read entire file

line1 = file.readline() # Read first line

lines = file.readlines() # Read all lines as list

file.close()

1st

file = open('D:\ACTE\Python-Tulsidash\example.txt' , 'r')

content = file.read()

print(content)

file.close

2nd

file = open('D:\ACTE\Python-Tulsidash\example.txt' , 'r')

content = file.readline()

print(content)

file.close

3rd

import pandas as pd

df = pd.read\_excel('D:\ACTE\Python-Tulsidash\Company Data.xlsx')

print(df)

**Writing & Appending to Files**

**Write to File(it’s overwrite the value)**

file = open("example.txt", "w")

file.write("Hello, World!")

file.close()

Note: w mode will **overwrite** the file if it exists.

1st

file = open('D:\ACTE\Python-Tulsidash\example2.txt' , 'w')

file.write("Namaste!, app kise ho ji")

**Append to File (it’s add the value)**

file = open("example.txt", "a")

file.write("\nThis is a new line.")

file.close()

1st

file = open('D:\ACTE\Python-Tulsidash\example2.txt' , 'a')

file.write("Namaste!, app kise ho ji")

2nd

file = open('D:\ACTE\Python-Tulsidash\example2.txt' , 'a')

file.write("\nNamaste!, app kise ho ji")

**The with statement**

Automatically handles file closing:

with open("example.txt", "r") as file:

data = file.read()

print(data)

**File Object Methods**

| Method | Description |
| --- | --- |

|  |  |
| --- | --- |
| read()------------------------------------------ | Reads entire file |

|  |  |
| --- | --- |
| readline()-------------------------------------- | Reads a single line |

|  |  |
| --- | --- |
| readlines()------------------------------------- | Reads all lines into a list |

|  |  |
| --- | --- |
| write(text)--------------------------------------- | Writes text to file |

|  |  |
| --- | --- |
| writelines(list) | ----------------------------Writes list of lines |

|  |  |
| --- | --- |
| seek(n) | -------------------------------------Moves cursor to position n |

|  |  |
| --- | --- |
| tell() | -----------------------------------------Returns current file position |

|  |  |
| --- | --- |
| close() | -----------------------Closes the file |
| **Check if File Exists**  import os  if os.path.exists("example.txt"):  print("File exists")  else:  print("File does not exist")  import os  if os.path.exists("D:\ACTE\Python-Tulsidash\example3.txt"):      print("File exists")  else:      print("File does not exist")  **Delete a File**  import os  os.remove("example.txt")  **Handling File Exceptions**  Handling file exceptions in Python is important to avoid crashes when something goes wrong during file operations, such as when a file doesn't exist or there's a permission issue.  Basic Example: Handling File Not Found  try:  file = open("data.txt", "r")  content = file.read()  print(content)  file.close()  except FileNotFoundError:  print("The file was not found.")  Using with Statement and Exception Handling  try:  with open("example.txt", "r") as f:  print(f.read())  except FileNotFoundError:  print("File not found.")  except PermissionError:  print("Permission denied.") |  |