**File operation**

File operation File operations in Python refer to performing actions on files, such as **reading, writing, appending, deleting**, and **closing**.

**File modes:-**when working with files,you specify the mode in which the file should opened

The common modes are :-

* **r:- read**
* **w:-create a new file or over writes if it exists)**
* **a:-append (adds content to the end of a file )**
* **b:-binary mode**
* **x:-exclusive creation (fails if already exists)**

**Basic file operations:**

* **Opening a file**
* **Reading a file**
* **Writing a file**
* **Appending a file**
* **Closing a file**
* **Checking file existence**

1. **opening a file:-**use the open() function to open a file .the syntax is

* **File=open (“file name.txt” , “mode”)**

1. **reading files:-**you can read content of the file using several methods :
2. **Read method :- read entire content of the file**

**file=open (“example.txt”, “r”)**

**Content = file.read()**

**print(content)**

**file.close()**

1. **Readline method:-**reads only one line from the file

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

**line = file.readline()**

**print("First line:", line)**

**file.close()**

1. **Readlines method:-read all lines in a file and returns them as a list**

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

**lines = file.readlines()**

**print("All lines:", lines)**

**file.close()**

1. **writing to files :-**write a content to a file using write() method

**# Open file in write mode (overwrites existing content)**

**file = open("output.txt", "w")**

**file.write("Hello, World!\n")**

**file.write("Python is great.\n")**

**file.close()**

1. **apending to a file :-**use append mode ‘a ’to add a content to the end of a file .

**file = open("output.txt", "a")**

**file.write("This is an appended line.\n")**

**file.close()**

This mode ensures new content is added to the end of the file without deleting existing data.

1. **closing a file:**it is essential to close a file after performing operations to free system resources

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

**content = file.read()**

**print(content)**

**file.close()**

1. **checking file existence:-**check file whether a file exists before performing operations,

**import os**

**if os.path.exists("example.txt"):**

**print("The file exists.")**

**else:**

**print("The file does not exist.")**

Advantages of file handling In python

* Versatility
* Flexibility
* User friendly
* Cross-platform

Dis advantages of file handling In python

* Error prone (can be prone to error like code issues)
* Security risk(post security issues
* User friendly(file handling will be complex when large files handling)
* Cross platform (it can be slower