

FILE HANDLING

File handling involves working with files for reading, writing, and manipulating data in a program.

It includes operations like opening a file, reading its contents, writing data to it, and closing the file.

Proper error handling is important to handle issues that may occur during file handling operations.

File handling is a fundamental aspect of programming and essential for tasks like data storage and retrieval.

There are several methods to access files:

```
#'r'      - default mode for reading data
#'w'      - used to write data into a specific file
#'a'      - used for appending new data with old data
#'x'      - Creates a new file, if the file already exists it throws error
#'rt'     - r for read and t for text mode
#'rb'     - r for read and b for binary mode (eg: images)
#'a+'     - overwriting of new data with old data by checking file availability
```

OPEN AND READ FILES

```
# open a file and read in the same folder
f = open("main.py", "r")
print(f.read())

# readline() function is used to read first line in the file
f = open("main.py", "r")
print(f.readline())

# close the file when you read it using close function
f.close()

# Open and read a file in a different folder with 'OS' Module
# os.chdir("./Desktop")
# path = os.path.join(os.getcwd(), "test.py")
# f = open(path, "r")
# print(f.read())
```

OPEN AND WRITE IN A FILE

```
# open and write in a file using open() and os module
os.chdir("./Desktop")
path = os.path.join(os.getcwd(), "test.py")
file = open(path, "w")
file.write("this is for testing writing files")
file.close()
```

```
path = os.path.join(os.getcwd(), "test.py")
file = open(path, "r")
print(file.read())
file.close()
```