

Chapter 6 - 14 Io Files

Theory:

This chapter covers the topic of 14 Io Files.

Code Example:

```
# ■ Python File I/O (Input/Output) - Basic to Advanced Guide
# -----
# 1. What is File I/O in Python?
# -----
# File I/O allows Python to read from and write to files on disk.
# The built-in open() function is used for this purpose.
# Step 1: Create and write to the file
with open("example.txt", "w") as file:
    file.write("Hello, this is a new example file.\n")
    file.write("This is the second line.\n")
# -----
# 2. Opening Files
# -----
file = open("example.txt", "r") # Open file in read mode
# ■ File Modes:
# "r" - Read (default), error if file doesn't exist
# "w" - Write, creates file or overwrites
# "a" - Append, creates file or adds to end
# "x" - Create, error if file exists
# "b" - Binary mode (e.g., "rb", "wb")
# "t" - Text mode (default)
# -----
# 3. Reading Files
# -----
# Read entire file
with open("example.txt", "r") as file:
    content = file.read()
print("Entire File:\n", content)
# Read line by line
with open("example.txt", "r") as file:
    print("Reading line by line:")
    for line in file:
        print(line.strip())
# Read into a list
with open("example.txt", "r") as file:
    lines = file.readlines()
print("List of Lines:", lines)
# -----
# 4. Writing to Files
# -----
# Overwrite existing content
with open("example.txt", "w") as file:
    file.write("Hello, this is a new file.\n")
    file.write("Another line.\n")
# Append to a file
with open("example.txt", "a") as file:
    file.write("Appended line here.\n")
# -----
```

```

# 5. Closing Files Manually
# -----
file = open("example.txt", "r")
# Do something with the file...
file.close() # Always close to free up resources
# Best practice: use "with open(...)" to auto-close
# -----

# 6. Using Try-Except for Safety
# -----
try:
    with open("notfound.txt", "r") as file:
        content = file.read()
        print(content)
except FileNotFoundError:
    print("File does not exist.")
# -----

# 7. Working with Binary Files (Advanced)
# -----
# Copying an image file
with open("image.png", "rb") as source:
    data = source.read()
with open("copy.png", "wb") as target:
    target.write(data)
# -----

# 8. File Methods Summary
# -----
# read() -> Reads entire file
# readline() -> Reads a single line
# readlines() -> Reads all lines into a list
# write(text) -> Writes a string
# writelines(lst)-> Writes a list of strings
# seek(offset) -> Moves file cursor to a position
# tell() -> Returns current file cursor position
# close() -> Closes the file manually
# -----

# 9. File Handling Best Practices
# -----
# ■ Always use with open(...) as to auto-close
# ■ Always check if file exists for reading
# ■ Use try-except to handle file errors
# ■ Avoid hardcoded paths, use os.path.join()
# -----

# 10. BONUS - Reading/Writing with Paths
# -----
import os
folder = "myfolder"
os.makedirs(folder, exist_ok=True)
path = os.path.join(folder, "myfile.txt")
with open(path, "w") as f:
    f.write("Organized file writing!")
# -----

# 11. Project-Level Practice Code
# -----

```

```

def write_sample():
with open("data.txt", "w") as f:
f.write("Line 1\nLine 2\nLine 3\n")
def read_sample():
with open("data.txt", "r") as f:
print("File Content:\n", f.read())
def append_sample():
with open("data.txt", "a") as f:
f.write("Appended Line\n")
# ■ Run all functions in order
write_sample()
read_sample()
append_sample()
read_sample()
#-----
#MY code
#-----
#CREATE A FILE
open("example.txt", "w").close()
#CREATE AND WRITE
with open("HELLO.txt", "w") as f:
f.write("Hello world!")
# OPEN FILE
f = open("file.txt", "r")
data = f.read()
print(data)
f.close()
#WRITE IN FILE
st = "HEY MILAN YOU ARE AMAZING"
f = open("myfile.txt" , "w")
f.write(st)
f.close()
# read file
f = open("myfile.txt")
lines = f.readlines()
line = f.read()
print(lines,type(lines))
print(line,type(line))
f.close()
# read single line
f = open("myfile.txt")
lines = f.readline()
line = f.read()
print(lines,type(lines))
print(line,type(line))
f.close()
#loop in file
f = open("myfile.txt")
line = f.readline()
while (line != ""): #here ye loop jabtak chalega jab tak line "" empty na hojaye
#end of line denote by ("" )
print(line)
line = f.readline()

```

```
f.close()  
# with statement  
with open("HELLO.txt", "w") as f:  
    f.write("Hello world!")  
# close automatically file with "with" statement
```