File Handling:

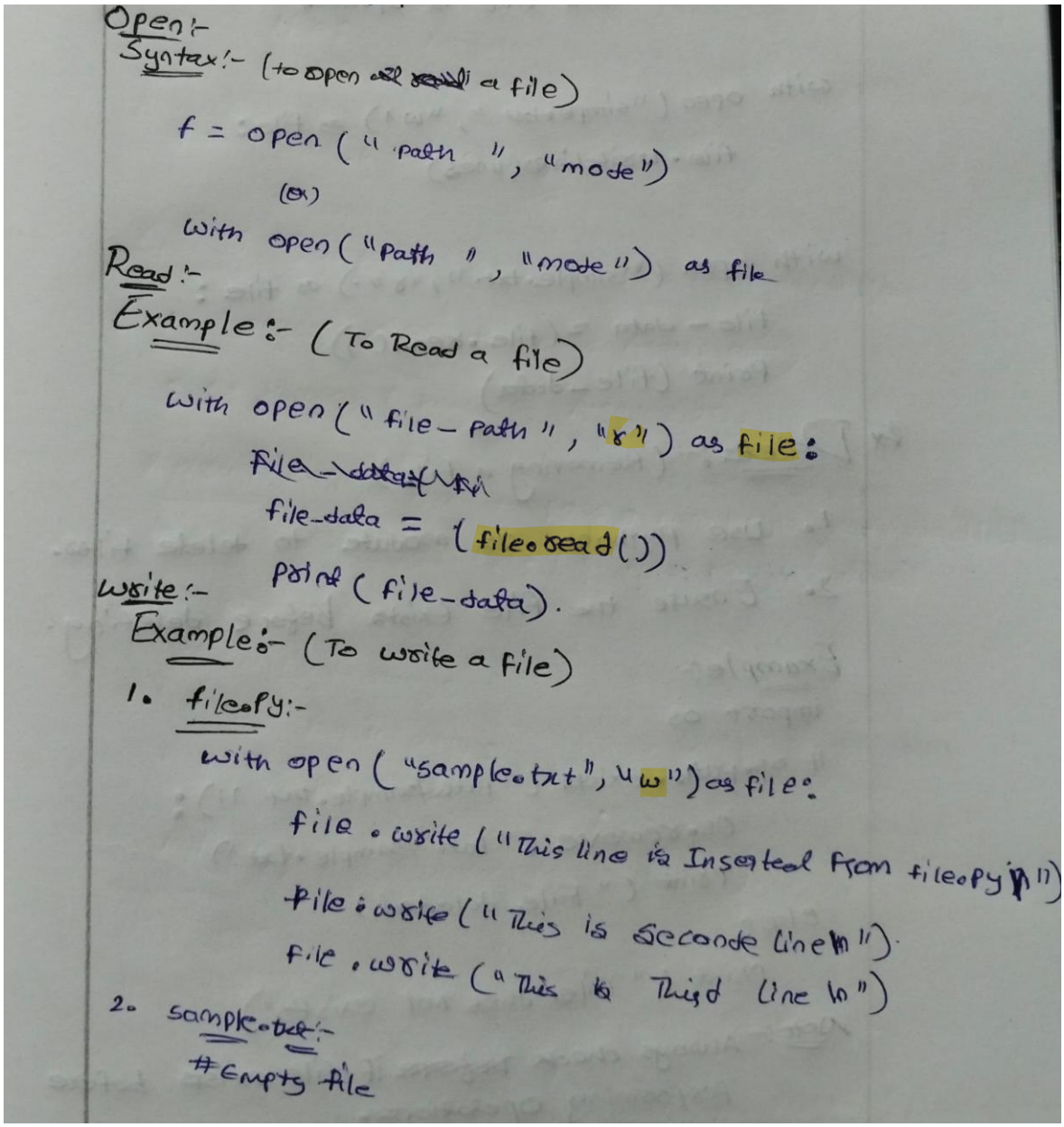
file handling and allows users to handle files i.e., to read and write files, along with many other file handling options, to operate on files.

Advantages of File Handling in Python: Versatility **,** Flexibility**,** User – friendly and Cross-platform

File Handling

The key function for working with files in Python is the open() function.
The open() function takes two parameters; *filename*, and *mode*.
There are four different methods (modes) for opening a file:

- "r" - Read - Default value. Opens a file for reading, error if the file does not exist
- "a" - Append - Opens a file for appending, creates the file if it does not exist
- "w" - Write - Opens a file for writing, creates the file if it does not exist
- "x" - Create - Creates the specified file, returns an error if the file exists



Read Lines :-

- readline() :- reads only one line
- readlines() :- reads all lines as a page

Update :-

Example :-

```
with open ("simple.txt", "r") as file:
    lines = file.readlines()
lines[1] = "This line is updated"

with open ("simple.txt", "w") as file:
    file.writelines(lines)

with open ("simple.txt", "r") as file:
    file_data = (file.read())
    Print (file_data)
```

Delete :- (Removing a file)

1. Use Python's os module to delete files.
2. Ensure the file exists before deleting.

Example :-

```
import os
if os.path.exists ("Path/sample.txt"):
    os.remove ("Path:/sample.txt")
    print ("File deleted")
else:
    print ("File does not exists")
```

Note :- Always check ~~before~~ if file exists before performing operations.

⇒ If we want to remove a particular line in a file just update the written line with empty string.

Example :-

```
with open ("path/simple.txt", "r") as file:
    lines = file.readlines()
Lines[i] = ""

with open ("path/simple.txt", "w") as file:
    file.writelines (lines)

with open ("path/simple.txt", "r") as file:
    file_data = (file.read())
    print (file_data)
```

File Paths :-

Relative path :- Refers to the file location relative to the script.

Ex :- 'subfolder/file.txt'

Absolute path :- Refers to the complete file location

Ex :- 'C:/Users/Name/Documents/file.txt'.

Example :- (Read function)

```
def read (f_name, m):
    with open (f_name, m) as file:
        temp = file.read()
        return temp

print (read ("sample.text", "r"))
```

TASK :- Create a file perform CURD operations.