S.P Group of Institute

Programming and Problem Solving Through Python Language Notes

Unit 7: File Processing

01. Concept of Files

File computer में data को स्थायी रूप से store करने का माध्यम है। Program बंद होने के बाद भी file में रखा data सुरक्षित रहता है।

File के प्रकार

Text file में readable characters और lines store होती हैं। Binary file में data bytes के रूप में store होता है, जैसे image या audio।

File processing का सामान्य flow

open → process (read / write) → close
पहले file खोलें, फिर data पढ़ें या लिखें, और अंत में file बंद करें।

02. File opening और modes

Python में file खोलने के लिए open() function का उपयोग होता है। यह एक file object return करता है।

```
file_object = open("filename", "mode")
```

मुख्य modes

Mode	अर्थ
r	read; file पहले से होनी चाहिए
w	write; पुराना data मिट सकता है
a	append; अंत में data जोड़ता है
x	नया file बनाता है
b / t / +	binary / text / read और write दोनों

Default mode r होता है और default type text होता है।

03. Closing a file

File पर काम पूरा होने के बाद close() method से उसे बंद करना चाहिए। इससे system resources release होते हैं और लिखे हुए data के save होने में मदद मिलती है।

```
f = open("data.txt", "r")
# file processing
f.close()
```

Recommended तरीका: with open()

```
with open("data.txt", "r") as f:
    data = f.read()
# block पूरा होने पर file स्वतः close
```

बंद file पर read() या write() करने पर error आ सकता है।

04. Reading from a file

File से data पढ़ने के लिए read(), readline() और readlines() methods उपयोग किए जाते हैं।

तीनों methods का अंतर

Method	क्या करता है
read()	पूरी file पढ़ता है; read(n) पहले n characters पढ़ता है
readline()	एक बार में एक line पढ़ता है
readlines()	सभी lines को list के रूप में लौटाता है

```
with open("notes.txt", "r") as f:
    all_data = f.read()
    # या: one_line = f.readline()
    # या: lines = f.readlines()
```

readlines() से मिली list की हर item सामान्यत: एक line होती है और उसमें newline भी हो सकता है।

05. Writing onto a file

File में text लिखने के लिए write() और writelines() methods उपयोग होते हैं।

```
with open("notes.txt", "w") as f:
    f.write("Python\n")
    f.write("File Processing")
```

write() एक string लिखता है। नई line के लिए string में "\n" स्वयं लिखना पड़ता है।

```
with open("list.txt", "w") as f:
    f.writelines(["one\n", "two\n", "three\n"])
```

w mode पुराने content को overwrite कर सकता है, जबकि a mode file के अंत में नया content जोड़ता है।

06. File functions: open(), close(), read()

open()

open() file को दिए गए mode में खोलता है और file object देता है।

```
f = open("data.txt", "r")
```

close()

close() खुले हुए file object को बंद करता है।

```
f.close()
```

read()

read() पूरी file का text लौटाता है। read(n) केवल पहले n characters पढ़ता है।

```
data = f.read()
first_10 = f.read(10)
```

07. File functions: readline(), readlines(), write(), writelines()

Function	कार्य
readline()	अगली एक line पढ़ता है
readlines()	सभी lines की list देता है
write()	एक string लिखता है
writelines()	strings की sequence लिखता है; newline खुद नहीं जोड़ता

छोटा उदाहरण

```
with open("demo.txt", "w") as f:
    f.writelines(["A\n", "B\n"])

with open("demo.txt", "r") as f:
    first = f.readline()
    rest = f.readlines()
```

write() लिखे characters की संख्या return कर सकता है। writelines() को सही नई lines के लिए strings में "\n" दें।

08. tell() और seek()

File pointer वह cursor है जो बताता है कि अगला read या write किस स्थान से होगा।

tell()

tell() current file pointer position बताता है।

```
position = f.tell()
```

seek(offset)

seek(offset) file pointer को दिए गए position पर ले जाता है। seek(0) pointer को file की शुरुआत पर ले जाता है।

```
with open("data.txt", "r") as f:
    first = f.read(5)
    print(f.tell())
    f.seek(0)
    again = f.read(5)
```

read() या write() के बाद pointer की position बदल जाती है; जरूरत पड़ने पर seek() का उपयोग करें।

09. Command Line Arguments

Command Line Arguments वे values हैं जो program run करते समय command के साथ दी जाती हैं। Python में इन्हें sys.argv list से पढ़ा जाता है।

```
# command
python add.py 10 20

# add.py
import sys
a = int(sys.argv[1])
b = int(sys.argv[2])
print("Sum =", a + b)
```

sys.argv[0] program का नाम होता है। sys.argv[1] और आगे के arguments strings होते हैं, इसलिए calculation के लिए int() या float() लगाएँ।

Output

```
Sum = 30
```

10. Quick revision और practice

एक नज़र में पूरा flow

open → read / write → tell / seek → close

File processing में सही mode चुनना, newline का ध्यान रखना और काम पूरा होने पर file close करना महत्वपूर्ण है। with open() सुरक्षित और सुविधाजनक तरीका है।

Practice questions

प्रश्न	संकेत
r और a में अंतर क्या है?	r पढ़ता है; a अंत में जोड़ता है
readline() और readlines() में अंतर?	एक line बनाम सभी lines की list
tell() का उपयोग क्यों?	pointer की current position जानने के लिए
10 और 20 कैसे दें?	python add.py 10 20

याद रखें: File खोलें, सही operation करें, pointer समझें, और file बंद करें।