FOUNDATION LESSON

WEEK 3 · LESSON 8

Read and write portable text files

pathlib, UTF-8 and context managers

UNDERSTAND THE IDEA FIRST

MENTAL MODEL

Understand the idea first

A Path object represents a location; creating it does not create the file. The slash operator joins path components portably. Opening a text file translates bytes using an encoding. with introduces a context manager: when the block finishes, including through an exception, the file is closed. Closing is part of resource handling, not the same action as saving a Python variable.

WRITE AND READ A SHORT REPORT

CODE WALKTHROUGH W03-L08-S03

```
01  from pathlib import Path
02
03  path = Path("report.txt")
04  with path.open("w", encoding="utf-8") as handle:
05      handle.write("Sales: 300\n")
06  with path.open("r", encoding="utf-8") as handle:
07      text = handle.read()
08  print(text, end="")
```

WHAT TO TRACK

Write mode replaces existing content; use the isolated exercise folder.

PREDICT THE RESULT

Predict the output.
Explain each operation.
Pause before continuing.

WRITE AND READ A SHORT REPORT — RESULT

CODE WALKTHROUGH W03-L08-S04

```
01  from pathlib import Path
02
03  path = Path("report.txt")
04  with path.open("w", encoding="utf-8") as handle:
05      handle.write("Sales: 300\n")
06  with path.open("r", encoding="utf-8") as handle:
07      text = handle.read()
08  print(text, end="")
```

WHAT TO TRACK

Write mode replaces existing content; use the isolated exercise folder.

VALIDATE THE RESULT

```
Sales: 300
```

READ A SUPPLIED BUSINESS NOTE

CODE WALKTHROUGH W03-L08-S05

```
01  from pathlib import Path
02
03  path = Path("data") / "business_note.txt"
04  with path.open("r", encoding="utf-8") as handle:
05      lines = handle.read().splitlines()
06  print(lines[0])
07  print("Lines:", len(lines))
```

WHAT TO TRACK

Supplied UTF-8 note: café inventory; two lines.

PREDICT THE RESULT

Predict the output.
Explain each operation.
Pause before continuing.

READ A SUPPLIED BUSINESS NOTE — RESULT

CODE WALKTHROUGH W03-L08-S06

```
01  from pathlib import Path
02
03  path = Path("data") / "business_note.txt"
04  with path.open("r", encoding="utf-8") as handle:
05      lines = handle.read().splitlines()
06  print(lines[0])
07  print("Lines:", len(lines))
```

WHAT TO TRACK

Supplied UTF-8 note: café inventory; two lines.

VALIDATE THE RESULT

```
Café inventory
Lines: 2
```

A MISSING PATH IS A REAL FAILURE

CODE WALKTHROUGH W03-L08-S07

```
01  from pathlib import Path
02
03  path = Path("data") / "missing_note.txt"
04  with path.open("r", encoding="utf-8") as handle:
05      print(handle.read())
```

WHAT TO TRACK

Deliberate missing file; read mode does not create it.

PREDICT THE RESULT

Predict the output.
Explain each operation.
Pause before continuing.

A MISSING PATH IS A REAL FAILURE — RESULT

CODE WALKTHROUGH W03-L08-S08

```
01 from pathlib import Path
02
03 path = Path("data") / "missing_note.txt"
04 with path.open("r", encoding="utf-8") as handle:
05     print(handle.read())
```

WHAT TO TRACK

Deliberate missing file; read mode does not create it.

VALIDATE THE RESULT

```
FileNotFoundError: [Errno 2]
No such file or directory:
'data/missing_note.txt'
```

PAUSE AND APPLY

LEARNER PRACTICE

Pause and apply

Write a UTF-8 summary file in the exercise folder containing two lines: Accepted: 2 and Rejected: 1. Read it back with a context manager, print its text and check that splitlines contains two entries.

PREDICT · MODIFY · RUN

01 Restate the supplied rules.

02 Predict the result.

03 Modify and run the code.

04 Compare actual output.

05 Explain any difference.

RECAP AND NEXT STEP

Recap and next step

A path describes a location. Explicit UTF-8 controls text encoding, and with closes opened files reliably. Relative paths support sharing when the working folder is documented. Next, we read structured records with CSV and JSON instead of manually splitting text.

UNDERSTAND APPLY VALIDATE EXPLAIN