FOUNDATION LESSON

WEEK 3 · LESSON 11

Protect secrets and understand objects

Environment configuration, attributes and methods

UNDERSTAND THE IDEA FIRST

MENTAL MODEL

Understand the idea first

An object combines a value with operations and attributes associated with its type. An attribute is a named value accessed after a dot. A method is an operation accessed through an object and called with parentheses. An environment variable supplies configuration to a process. Reading it avoids embedding the value in source code, but environment variables are not automatically encrypted or safe to print.

REPORT CONFIGURATION WITHOUT REVEALING IT

CODE WALKTHROUGH W03-L11-S03

```
01  import os
02
03  key = os.environ.get("AISYNAPSE_DEMO_API_KEY")
04  if key:
05      print("Key configured")
06  else:
07      print("Offline mode: no key needed")
```

WHAT TO TRACK

Exercise runner supplies no key;
never print the secret value.

PREDICT THE RESULT

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

REPORT CONFIGURATION WITHOUT REVEALING IT — RESULT

CODE WALKTHROUGH W03-L11-S04

```
01  import os
02
03  key = os.environ.get("AISYNAPSE_DEMO_API_KEY")
04  if key:
05      print("Key configured")
06  else:
07      print("Offline mode: no key needed")
```

WHAT TO TRACK

Exercise runner supplies no key; never print the secret value.

VALIDATE THE RESULT

```
Offline mode: no key needed
```

ATTRIBUTES AND METHODS ON A PATH

CODE WALKTHROUGH W03-L11-S05

```
01  from pathlib import Path
02
03  p = Path("data") / "invoices.csv"
04  print(p.name)
05  print(p.suffix)
06  print(p.exists())
```

WHAT TO TRACK

name and suffix are attributes;
exists() is a method call.

PREDICT THE RESULT

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

ATTRIBUTES AND METHODS ON A PATH — RESULT

CODE WALKTHROUGH W03-L11-S06

```
01  from pathlib import Path
02
03  p = Path("data") / "invoices.csv"
04  print(p.name)
05  print(p.suffix)
06  print(p.exists())
```

WHAT TO TRACK

name and suffix are attributes;
exists() is a method call.

VALIDATE THE RESULT

```
invoices.csv
.csv
True
```

DO NOT CALL A DATA ATTRIBUTE

CODE WALKTHROUGH W03-L11-S07

```
01  from pathlib import Path
02
03  p = Path("data/invoices.csv")
04  print(p.name())
```

WHAT TO TRACK

Deliberate error: name contains a string, not a method.

PREDICT THE RESULT

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

DO NOT CALL A DATA ATTRIBUTE — RESULT

CODE WALKTHROUGH W03-L11-S08

```
01  from pathlib import Path
02
03  p = Path("data/invoices.csv")
04  print(p.name())
```

WHAT TO TRACK

Deliberate error: name contains a string, not a method.

VALIDATE THE RESULT

```
TypeError: 'str' object is not callable
```

PAUSE AND APPLY

LEARNER PRACTICE

Pause and apply

Create a Path for the supplied invoices.json file. Display its name and suffix, then call exists to check it. Separately check whether the demo key is configured, printing only a status. Do not supply or reveal any key.

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

Keep secrets outside shared source and never log their values. Attributes hold named values; methods are called operations. Existing Path and string objects already illustrate these ideas. Next, we define one small class and combine our offline ingestion skills.

UNDERSTAND APPLY VALIDATE EXPLAIN