FOUNDATION LESSON

WEEK 3 · LESSON 1

Why functions matter: define once, call when needed

A named operation: define once, call when needed.

THE MENTAL MODEL

CODE WALKTHROUGH W03-L01-S02

```
01  def show_start_message():
02      print("Starting daily checks")
```

WHAT TO TRACK

Definition: write the checklist.
Call: carry it out.
Body: indented instructions.

PREDICT THE RESULT

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

MINIMAL EXAMPLE AND PREDICTION

CODE WALKTHROUGH W03-L01-S03

```
01  def show_start_message():
02      print("Starting daily checks")
03
04
05  print("Before call")
06  show_start_message()
07  print("After call")
```

WHAT TO TRACK

Trace execution from top to bottom.
The body runs at the call.

PREDICT THE RESULT

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

RESULT AND LINE-BY-LINE REASONING

CODE WALKTHROUGH W03-L01-S04

```
01  def show_start_message():
02      print("Starting daily checks")
03
04
05  print("Before call")
06  show_start_message()
07  print("After call")
```

WHAT TO TRACK

A definition does not print.
The call executes the body.

VALIDATE THE RESULT

```
Before call
Starting daily checks
After call
```

BUSINESS APPLICATION

CODE WALKTHROUGH W03-L01-S05

```
01  def show_invoice_checks():
02      amounts = [120, 2500, 0]
03
04      for amount in amounts:
05          if amount <= 0:
06              print(amount, "invalid")
07          elif amount >= 2000:
08              print(amount, "review")
09          else:
10              print(amount, "standard")
11
12
13  print("Run 1")
14  show_invoice_checks()
15
16  print("Run 2")
17  show_invoice_checks()
```

WHAT TO TRACK

Amount ≤ 0: invalid.
Amount ≥ 2000: review.
Other positive: standard.

PREDICT THE RESULT

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

PREDICT REPEATED CALLS

MENTAL MODEL

Predict repeated calls

Three amounts, two calls.
How many classification lines?

BUSINESS RESULT AND REASONING

CODE WALKTHROUGH W03-L01-S07

```
01  def show_invoice_checks():
02      amounts = [120, 2500, 0]
03
04      for amount in amounts:
05          if amount <= 0:
06              print(amount, "invalid")
07          elif amount >= 2000:
08              print(amount, "review")
09          else:
10              print(amount, "standard")
11
12
13  print("Run 1")
14  show_invoice_checks()
15
16  print("Run 2")
17  show_invoice_checks()
```

WHAT TO TRACK

Each call runs the same procedure.
Three records × two calls.

VALIDATE THE RESULT

```
Run 1
120 standard
2500 review
0 invalid
Run 2
120 standard
2500 review
0 invalid
```

DELIBERATE ERROR: CALLING TOO SOON

CODE WALKTHROUGH W03-L01-S08

```
01  show_end_message()
02
03
04  def show_end_message():
05      print("Checks complete")
```

WHAT TO TRACK

The call executes before the definition.
Predict the failure.

VALIDATE THE RESULT

```
NameError: name 'show_end_message' is not defined
```

REPAIR THE ORDER

CODE WALKTHROUGH W03-L01-S09

```
01  def show_end_message():
02      print("Checks complete")
03
04
05  show_end_message()
```

WHAT TO TRACK

Establish the function before calling it.

VALIDATE THE RESULT

```
Checks complete
```

LEARNER MODIFICATION

LEARNER PRACTICE

Learner modification

Use [999, 1000, 0].
Review threshold: 1000.
Keep one definition and call it once.

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 ENDING

Recap and next step

Definition establishes an operation.
The indented body runs on a call.
One definition supports repeated calls.

UNDERSTAND APPLY VALIDATE EXPLAIN