FOUNDATION LESSON

WEEK 2 · LESSON 1

Decisions in Python

A precise business rule selects the instructions to run.

A RULE WITH A PRECISE BOUNDARY

MENTAL MODEL

A rule with a precise boundary

Fictional rule
Extra review applies when the amount is greater than 2000

2000 itself does not pass this condition

THE CONDITIONAL BLOCK

CODE WALKTHROUGH W02-L01-S03

```
01  amount = 2500
02  if amount > 2000:
03      print("review")
04  print("checked")
```

WHAT TO TRACK

The condition controls the indented line

PREDICT THE RESULT

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

THE TRUE PATH

CODE WALKTHROUGH W02-L01-S04

```
01  amount = 2500
02  if amount > 2000:
03      print("review")
04  print("checked")
```

WHAT TO TRACK

2500 > 2000 is True

Python enters the indented block

VALIDATE THE RESULT

```
review
checked
```

THE FALSE PATH

CODE WALKTHROUGH W02-L01-S05

```
01  amount = 1500
02  if amount > 2000:
03      print("review")
04  print("checked")
```

WHAT TO TRACK

Only the input changes

Predict what happens when amount is

PREDICT THE RESULT

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

EXECUTION AFTER THE BLOCK

CODE WALKTHROUGH W02-L01-S06

```
01  amount = 1500
02  if amount > 2000:
03      print("review")
04  print("checked")
```

WHAT TO TRACK

The review message is skipped

The checked message still runs

VALIDATE THE RESULT

checked

BOUNDARY TESTS

MENTAL MODEL

Boundary tests

Amount 1999: checked
Amount 2000: checked
Amount 2001: review, then checked

TWO ALTERNATIVE OUTCOMES

CODE WALKTHROUGH W02-L01-S08

```
01  balance = -25
02  if balance >= 0:
03      print("non-negative")
04  else:
05      print("negative")
06  print("classification complete")
```

WHAT TO TRACK

One if–else statement

One branch for True and another for

PREDICT THE RESULT

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

THE ELSE BRANCH

CODE WALKTHROUGH W02-L01-S09

```
01  balance = -25
02  if balance >= 0:
03      print("non-negative")
04  else:
05      print("negative")
06  print("classification complete")
```

WHAT TO TRACK

–25 >= 0 is False

The else block supplies the

VALIDATE THE RESULT

```
negative
classification complete
```

ZERO BELONGS TO THE NON-NEGATIVE PATH

CODE WALKTHROUGH W02-L01-S10

```
01  balance = 0
02  if balance >= 0:
03      print("non-negative")
04  else:
05      print("negative")
06  print("classification complete")
```

WHAT TO TRACK

The operator >= includes equality

Zero is a valid numeric input here

VALIDATE THE RESULT

```
non-negative
classification complete
```

A DELIBERATE INDENTATION ERROR

CODE WALKTHROUGH W02-L01-S11

```
01  amount = 2500
02  if amount > 2000:
03  print("review")
```

WHAT TO TRACK

The following block is intentionally invalid

VALIDATE THE RESULT

```
IndentationError: expected
an indented block after 'if'
statement on line 2
```

INDEPENDENT PREDICTION

CODE WALKTHROUGH W02-L01-S12

```
01  stock = 0
02  if stock > 0:
03      print("available")
04  else:
05      print("out of stock")
06  print("inventory checked")
```

WHAT TO TRACK

Predict every output line

Then change stock to 3 and run again

PREDICT THE RESULT

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

PRACTICE SOLUTION

LEARNER PRACTICE

Practice solution

stock = 0
  out of stock
  inventory checked

stock = 3
  available
  inventory checked

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.

A SHORT SELF-CHECK

MENTAL MODEL

A short self-check

A condition is False

Does Python skip the indented block,
the rest of the script, or both?

LESSON RECAP

Recap and next step

A condition selects a path
Indentation identifies its block
else supplies an alternative
Boundary inputs check the intended rule

UNDERSTAND APPLY VALIDATE EXPLAIN