FOUNDATION LESSON

WEEK 3 · LESSON 2

Parameters, arguments and safe defaults

Inputs, binding and defaults without hidden shared state.

EXPLAIN BINDING BEFORE SYNTAX

MENTAL MODEL

Explain binding before syntax

Parameter: a name in the definition.
Argument: a value supplied by a call.
Binding associates the two.

MINIMAL EXAMPLE AND PREDICTION

CODE WALKTHROUGH W03-L02-S03

```
01  def show_stock(item, quantity):
02      print(item, quantity)
03
04
05  show_stock("Pens", 12)
06  show_stock(quantity=4, item="Folders")
```

WHAT TO TRACK

Positional arguments bind in order.
Keyword arguments bind by name.

PREDICT THE RESULT

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

RESULT AND BINDING TRACE

CODE WALKTHROUGH W03-L02-S04

```
01  def show_stock(item, quantity):
02      print(item, quantity)
03
04
05  show_stock("Pens", 12)
06  show_stock(quantity=4, item="Folders")
```

WHAT TO TRACK

Keyword order does not change the body.

VALIDATE THE RESULT

```
Pens 12
Folders 4
```

BUSINESS APPLICATION WITH A DEFAULT

CODE WALKTHROUGH W03-L02-S05

```
01  def show_delivery_total(order_total, delivery_fee=5):
02      print("Total:", order_total + delivery_fee)
03
04
05  show_delivery_total(100)
06  show_delivery_total(100, 0)
07  show_delivery_total(delivery_fee=8, order_total=100)
```

WHAT TO TRACK

Total = amount + fee.
Default fee: 5.
Explicit 0: free delivery.

PREDICT THE RESULT

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

DEFAULT RESULT AND REASONING

CODE WALKTHROUGH W03-L02-S06

```
01  def show_delivery_total(order_total, delivery_fee=5):
02      print("Total:", order_total + delivery_fee)
03
04
05  show_delivery_total(100)
06  show_delivery_total(100, 0)
07  show_delivery_total(delivery_fee=8, order_total=100)
```

WHAT TO TRACK

Omitted, zero and eight are different inputs.

VALIDATE THE RESULT

```
Total: 105
Total: 100
Total: 108
```

DELIBERATE DESIGN MISTAKE: MUTABLE DEFAULT

CODE WALKTHROUGH W03-L02-S07

```
01  def record_review(invoice_id, reviewed=[]):
02      reviewed.append(invoice_id)
03      print(reviewed)
04
05
06  record_review("INV-01")
07  record_review("INV-02")
```

WHAT TO TRACK

Intention: each omitted list is fresh.
Does the default implement it?

PREDICT THE RESULT

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

THE RETAINED-STATE RESULT

CODE WALKTHROUGH W03-L02-S08

```
01  def record_review(invoice_id, reviewed=[]):
02      reviewed.append(invoice_id)
03      print(reviewed)
04
05
06  record_review("INV-01")
07  record_review("INV-02")
```

WHAT TO TRACK

The same default list retains mutations.

VALIDATE THE RESULT

```
['INV-01']
['INV-01', 'INV-02']
```

REPAIR WITH NONE

CODE WALKTHROUGH W03-L02-S09

```
01  def record_review(invoice_id, reviewed=None):
02      if reviewed is None:
03          reviewed = []
04
05      reviewed.append(invoice_id)
06      print(reviewed)
07
08
09  record_review("INV-01")
10  record_review("INV-02")
```

WHAT TO TRACK

None marks an omitted list.
Create a fresh list inside the call.

VALIDATE THE RESULT

```
['INV-01']
['INV-02']
```

GUIDED MODIFICATION

LEARNER PRACTICE

Guided modification

Add record_review("INV-03").
Predict its list after the None repair.

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

Parameters name inputs; arguments supply them.
Defaults are evaluated at definition time.
A mutable default can retain state.

UNDERSTAND APPLY VALIDATE EXPLAIN