FOUNDATION LESSON

WEEK 4 · LESSON 3

Select slices and protect the original

Basic slices share numeric storage; copy creates independent numeric data

THE IDEA BEHIND THE OPERATION

MENTAL MODEL

The idea behind the operation

Basic slices share numeric storage; copy creates independent numeric data

POSITIONS AND WINDOWS

CODE WALKTHROUGH W04-L03-S03

```
01  import numpy as np
02  units = np.array([10, 20, 30, 40, 50])
03  print("First:", units[0])
04  print("Last:", units[-1])
05  print("Middle:", units[1:4])
06  print("Step:", units[::2])
```

WHAT TO TRACK

Indices start at zero; slice stop is excluded.

PREDICT THE RESULT

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

POSITIONS AND WINDOWS — RESULT

CODE WALKTHROUGH W04-L03-S04

```
01  import numpy as np
02  units = np.array([10, 20, 30, 40, 50])
03  print("First:", units[0])
04  print("Last:", units[-1])
05  print("Middle:", units[1:4])
06  print("Step:", units[::2])
```

WHAT TO TRACK

Indices start at zero; slice stop is excluded.

VALIDATE THE RESULT

```
First: 10
Last: 50
Middle: [20 30 40]
Step: [10 30 50]
```

AN EDITABLE DRAFT NEEDS A COPY

CODE WALKTHROUGH W04-L03-S05

```
01  import numpy as np
02  source = np.array([100, 200, 300, 400])
03  view = source[1:3]
04  view[0] = 999
05  draft = source[1:3].copy()
06  draft[0] = 50
07  print("Source:", source)
08  print("Draft:", draft)
```

WHAT TO TRACK

Compare shared data with independent numeric storage.

PREDICT THE RESULT

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

AN EDITABLE DRAFT NEEDS A COPY — RESULT

CODE WALKTHROUGH W04-L03-S06

```
01  import numpy as np
02  source = np.array([100, 200, 300, 400])
03  view = source[1:3]
04  view[0] = 999
05  draft = source[1:3].copy()
06  draft[0] = 50
07  print("Source:", source)
08  print("Draft:", draft)
```

WHAT TO TRACK

Compare shared data with independent numeric storage.

VALIDATE THE RESULT

```
Source: [100 999 300 400]
Draft: [ 50 300]
```

AN INDEX CAN FAIL WHEN A SLICE SUCCEEDS

CODE WALKTHROUGH W04-L03-S07

```
01  import numpy as np
02  values = np.array([10, 20, 30])
03  print("Beyond-end slice:", values[10:20])
04  try:
05      print(values[3])
06  except IndexError as error:
07      print(type(error).__name__)
08      print("Valid positions are 0, 1 and 2")
```

WHAT TO TRACK

Oversized slices can be empty; missing single positions raise.

PREDICT THE RESULT

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

AN INDEX CAN FAIL WHEN A SLICE SUCCEEDS — RESULT

CODE WALKTHROUGH W04-L03-S08

```
01  import numpy as np
02  values = np.array([10, 20, 30])
03  print("Beyond-end slice:", values[10:20])
04  try:
05      print(values[3])
06  except IndexError as error:
07      print(type(error).__name__)
08      print("Valid positions are 0, 1 and 2")
```

WHAT TO TRACK

Oversized slices can be empty; missing single positions raise.

VALIDATE THE RESULT

```
Beyond-end slice: []
IndexError
Valid positions are 0, 1 and
2
```

GUIDED MODIFICATION

LEARNER PRACTICE

Guided modification

Select the first three quantities from [5, 10, 15, 20, 25]. Make a copy, replace its first value with zero and verify the original remains unchanged.

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.

INDEPENDENT APPLICATION

LEARNER PRACTICE

Independent application

From seven daily quantities [4, 6, 8, 10, 12, 14, 16], extract the first five and the final two. Reverse the full series. Print each selection without changing the source.

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

Trace indices from zero and exclude the slice stop. Basic slices share data; a deliberate numeric copy protects independent editing. Check empty results explicitly. Next, we select two-dimensional cells, rows, columns and Boolean masks.

UNDERSTAND APPLY VALIDATE EXPLAIN