FOUNDATION LESSON

WEEK 4 · LESSON 7

Create labelled Series and DataFrames

Series labels align values; a DataFrame organises labelled columns

THE IDEA BEHIND THE OPERATION

MENTAL MODEL

The idea behind the operation

Series labels align values; a DataFrame organises labelled columns

A LABELLED QUANTITY SERIES

CODE WALKTHROUGH W04-L07-S03

```
01  import pandas as pd
02  units = pd.Series([5, 8], index=["A", "B"], name="units")
03  print(units.to_string())
04  print("A:", units.loc["A"])
05  print("Labels:", units.index.tolist())
```

WHAT TO TRACK

Product codes A and B label their quantities.

PREDICT THE RESULT

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

A LABELLED QUANTITY SERIES — RESULT

CODE WALKTHROUGH W04-L07-S04

```
01  import pandas as pd
02  units = pd.Series([5, 8], index=["A", "B"], name="units")
03  print(units.to_string())
04  print("A:", units.loc["A"])
05  print("Labels:", units.index.tolist())
```

WHAT TO TRACK

Product codes A and B label their quantities.

VALIDATE THE RESULT

```
A    5
B    8
A: 5
Labels: ['A', 'B']
```

CREATE A MIXED-COLUMN TABLE

CODE WALKTHROUGH W04-L07-S05

```
01  import pandas as pd
02  df = pd.DataFrame({
03      "product": ["A", "B"],
04      "units": [5, 8],
05      "price": [10.0, 4.5]
06  })
07  print(df.to_string(index=False))
08  print("Shape:", df.shape)
09  print(df.dtypes.to_string())
```

WHAT TO TRACK

Text identifiers, whole quantities and decimal prices.

PREDICT THE RESULT

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

CREATE A MIXED-COLUMN TABLE — RESULT

CODE WALKTHROUGH W04-L07-S06

```
01  import pandas as pd
02  df = pd.DataFrame({
03      "product": ["A", "B"],
04      "units": [5, 8],
05      "price": [10.0, 4.5]
06  })
07  print(df.to_string(index=False))
08  print("Shape:", df.shape)
09  print(df.dtypes.to_string())
```

WHAT TO TRACK

Text identifiers, whole quantities and decimal prices.

VALIDATE THE RESULT

```
product  units  price
      A      5   10.0
      B      8    4.5
Shape: (2, 3)
product     object
units        int64
price      float64
```

ARITHMETIC MATCHES LABELS

CODE WALKTHROUGH W04-L07-S07

```
01  import pandas as pd
02  a = pd.Series([100, 200], index=["A", "B"])
03  b = pd.Series([10, 20], index=["B", "C"])
04  result = a + b
05  print(result.to_string())
```

WHAT TO TRACK

Only label B is present in both inputs.

PREDICT THE RESULT

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

ARITHMETIC MATCHES LABELS — RESULT

CODE WALKTHROUGH W04-L07-S08

```
01  import pandas as pd
02  a = pd.Series([100, 200], index=["A", "B"])
03  b = pd.Series([10, 20], index=["B", "C"])
04  result = a + b
05  print(result.to_string())
```

WHAT TO TRACK

Only label B is present in both inputs.

VALIDATE THE RESULT

```
A      NaN
B    210.0
C      NaN
```

GUIDED MODIFICATION

LEARNER PRACTICE

Guided modification

Create a Series for product codes P and Q with quantities three and seven. Print it, select Q by label and list its index.

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

Add Series [4, 6] labelled [A, B] and Series [10, 20] labelled [B, A]. Predict the aligned result before running. Explain why positional addition would answer a different question.

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

Use Series for labelled values and DataFrames for labelled columns. Arithmetic aligns labels, and missing partners matter. Next, we load a supplied CSV and inspect its structure before trusting it.

UNDERSTAND APPLY VALIDATE EXPLAIN