Pandas set_index

WHAT YOU'LL LEARN

- How to use the `set_index()` method

- How to "set" an index for a Pandas DataFrame

A quick introduction to Pandas set_index

ALL PANDAS DATAFRAMES HAVE AN INDEX

You can think of an index as being similar to a column ...

(But it's not technically a column, and will not appear when you print a list of columns)

index	employee_id	first_name
0	101	John
1	102	Paul
2	103	George
3	104	Ringo
4	105	Stewart

BY DEFAULT, A DATAFRAME INDEX IS A NUMERIC RANGE OF VALUES STARTING AT 0

If you don't define the index in a different way, all DataFrames have this `RangeIndex` by default
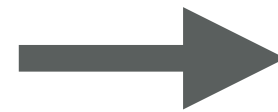
index	employee_id	first_name
0	101	John
1	102	Paul
2	103	George
3	104	Ringo
4	105	Stewart

BUT, WE CAN DEFINE A NEW INDEX BY USING THE SET_INDEX METHOD

`emp_fst_nm.set_index('employee_id')`

emp_fst_nm

index	employee_id	first_name
0	101	John
1	102	Paul
2	103	George
3	104	Ringo
4	105	Stewart

Out:

employee_id	first_name
101	John
102	Paul
103	George
104	Ringo
105	Stewart

Pandas set_index syntax

Syntax: PANDAS SET_INDEX

The name of the DataFrame you want to operate on

The column or list of columns you want to use as the index

```
myDataFrame.set_index(key)
```

The name of the method

Parameters of set_index

THE PARAMETERS OF PANDAS SET_INDEX

Parameter	What it does	Format	Default	Required?
key	Specify the column or columns you want to use as the index	Single variable name or a Python list of variables		Yes
drop=	Specify whether or not the set_index method will "drop" the column that you set as the index	True or False	True	No
inplace=	Specify whether you want to modify your DataFrame "in place"	True or False	False	No

THE OUTPUT OF PANDAS SET_INDEX

- By default, `set_index()` will produce a new DataFrame with the new index
  - by default, the `inplace` parameter is set to `inplace = False`

- If you set `inplace = True,` `set_index()` will directly modify the original DataFrame

Recap

RECAP OF WHAT WE LEARNED

- You can change the index of a DataFrame with `set_index()`

- By default, `set_index()` will create a new DataFrame
  - If you set `inplace = True`, the method will directly modify the original data

- **Next Steps**: Watch the code walkthrough video for step-by-step examples of `set_index()`