

Pandas 4

Slicing and subsetting with .loc and .iloc

Slicing Rows

- Sort the index before we slice
- `dogs_srt = dogs.set_index(["breed", "color"]).sort_index()`
- We can slide the outer indexes by passing in the indexes in `loc[from: to]`, here to is also included in the slice.

Slicing the outer index level

```
dogs_srt.loc["Chow Chow":"Poodle"]
```

breed	color	name	height_cm	weight_kg
Chow Chow	Brown	Lucy	46	22
Labrador	Black	Max	59	29
	Brown	Bella	56	25
Poodle	Black	Charlie	43	23

The final value **"Poodle"** is included

Full dataset

breed	color	name	height_cm	weight_kg
Chihuahua	Tan	Stella	18	2
Chow Chow	Brown	Lucy	46	22
Labrador	Black	Max	59	29
	Brown	Bella	56	25
Poodle	Black	Charlie	43	23
Schnauzer	Grey	Cooper	49	17
St. Bernard	White	Bernie	77	74

- Here the slicing operation doesn't work with the inner level indexes as we have layered indexes

Slicing the inner index levels badly

```
dogs_srt.loc["Tan":"Grey"]
```

```
Empty DataFrame
Columns: [name, height_cm, weight_kg]
Index: []
```

Full dataset

breed	color	name	height_cm	weight_kg
Chihuahua	Tan	Stella	18	2
Chow Chow	Brown	Lucy	46	22
Labrador	Black	Max	59	29
	Brown	Bella	56	25
Poodle	Black	Charlie	43	23
Schnauzer	Grey	Cooper	49	17
St. Bernard	White	Bernie	77	74

To slice based on the inner level index, we will pass in composite indexes in the slices

```
dogs_srt.loc[("Labrador", "Brown") : ("Schnauzer", "Grey")]
```

Slicing the inner index levels correctly

```
dogs_srt.loc[
    ("Labrador", "Brown") : ("Schnauzer", "Grey")]
```

```
breed      color      name  height_cm  weight_kg
Labrador   Brown    Bella         56         25
Poodle     Black   Charlie         43         23
Schnauzer  Grey     Cooper         49         17
```

Full dataset

breed	color	name	height_cm	weight_kg
Chihuahua	Tan	Stella	18	2
Chow Chow	Brown	Lucy	46	22
Labrador	Black	Max	59	29
	Brown	Bella	56	25
Poodle	Black	Charlie	43	23
Schnauzer	Grey	Cooper	49	17
St. Bernard	White	Bernie	77	74

Slicing Columns

dogs_srt.loc[:, "name": "height_cm"]

- Selecting all rows(indicated by a colon) and only the column height_cm

dogs_srt.loc[("Labrador", "Brown") : ("Schnauzer", "Grey"), "name": "height_cm"]

- Rows from Labrador", "Brown to (including) ("Schnauzer", "Grey") and from this we only select height_cm column.

Slice twice

```
dogs_srt.loc[
    ("Labrador", "Brown"):(("Schnauzer", "Grey"),
    "name":"height_cm"]
```

		name	height_cm
breed	color		
Labrador	Brown	Bella	56
Poodle	Black	Charlie	43
Schnauzer	Grey	Cooper	49

Full dataset

		name	height_cm	weight_kg
breed	color			
Chihuahua	Tan	Stella	18	2
Chow Chow	Brown	Lucy	46	22
Labrador	Black	Max	59	29
	Brown	Bella	56	25
Poodle	Black	Charlie	43	23
Schnauzer	Grey	Cooper	49	17
St. Bernard	White	Bernie	77	74

Slicing using dates:

- We can slice data frames using pandas, here we will set the date as the index of the data frame, then we can pass in the date or the year to get the slice of rows which fall in the condition.

Dog days

```
dogs = dogs.set_index("date_of_birth").sort_index()
print(dogs)
```

	name	breed	color	height_cm	weight_kg
date_of_birth					
2011-12-11	Cooper	Schnauzer	Grey	49	17
2013-07-01	Bella	Labrador	Brown	56	25
2014-08-25	Lucy	Chow Chow	Brown	46	22
2015-04-20	Stella	Chihuahua	Tan	18	2
2016-09-16	Charlie	Poodle	Black	43	23
2017-01-20	Max	Labrador	Black	59	29
2018-02-27	Bernie	St. Bernard	White	77	74

Slicing by dates

```
# Get dogs with date_of_birth between 2014-08-25 and 2016-09-16
dogs.loc["2014-08-25":"2016-09-16"]
```

date_of_birth	name	breed	color	height_cm	weight_kg
2014-08-25	Lucy	Chow Chow	Brown	46	22
2015-04-20	Stella	Chihuahua	Tan	18	2
2016-09-16	Charlie	Poodle	Black	43	23

Slicing by partial dates

```
# Get dogs with date_of_birth between 2014-01-01 and 2016-12-31
dogs.loc["2014":"2016"]
```

date_of_birth	name	breed	color	height_cm	weight_kg
2014-08-25	Lucy	Chow Chow	Brown	46	22
2015-04-20	Stella	Chihuahua	Tan	18	2
2016-09-16	Charlie	Poodle	Black	43	23

Subsetting by row/column number

- We can get a slice of the dataframe using row or column numbers, to do this we will be making use of iloc method, to iloc method we will pass the row and column numbers and this will handle the rest of the slicing
- `iloc[from_row: to_row, from_col:to_col]`
- `dogs.iloc[2:5, 1:4]`
- In iloc the to values are not included in the slice

Subsetting by row/column number

```
print(dogs.iloc[2:5, 1:4])
```

	breed	color	height_cm
2	Chow Chow	Brown	46
3	Schnauzer	Grey	49
4	Labrador	Black	59

Full dataset

	name	breed	color	height_cm	weight_kg
0	Bella	Labrador	Brown	56	25
1	Charlie	Poodle	Black	43	23
2	Lucy	Chow Chow	Brown	46	22
3	Cooper	Schnauzer	Grey	49	17
4	Max	Labrador	Black	59	29
5	Stella	Chihuahua	Tan	18	2
6	Bernie	St. Bernard	White	77	74

Code Snippets

```
# Subset rows from India, Hyderabad to Iraq, Baghdad
print(temperatures_srt.loc[("India", "Hyderabad"):(("Iraq", "Baghdad"))])
```

```
# Subset columns from date to avg_temp_c
print(temperatures_srt.loc[:, "date":"avg_temp_c"])
```

```
# Subset in both directions at once
print(temperatures_srt.loc[("India", "Hyderabad"):(("Iraq", "Baghdad"),
"date":"avg_temp_c")])
```

```
# Use Boolean conditions to subset temperatures for rows in 2010 and 2011
temperatures_bool = temperatures[(temperatures['date'] >= "2010") &
(temperatures['date'] < "2012")]
print(temperatures_bool)
```

```
# Set date as the index and sort the index
temperatures_ind = temperatures.set_index('date').sort_index()
```

```
# Use .loc[] to subset temperatures_ind for rows in 2010 and 2011
print(temperatures_ind.loc["2010" : "2011"])
```

```
# Use .loc[] to subset temperatures_ind for rows from Aug 2010 to Feb 2011
print(temperatures_ind.loc["2010-08-01" : "2011-02-28"])
```

```
# Get 23rd row, 2nd column (index 22, 1)
print(temperatures.iloc[22:23, 1:2])
```

```
# Use slicing to get the first 5 rows  
print(temperatures.iloc[:5])
```

```
# Use slicing to get columns 3 to 4  
print(temperatures.iloc[:, 2:5])
```

```
# Use slicing in both directions at once  
print(temperatures.iloc[:5, 2:4])
```
