

365 DataScience Dealing with indexing in Python

Step 1 Libraries import

use the well-known convention

```
import pandas as pd
```

Step 2 Create a Series object

```
series_1 = pd.Series([1, 5, 10, 15])
series_1
```

```
0      1
1      5
2     10
3     15
dtype: int64
```

Step 3 Create a dictionary

```
series_1_dict = {'Product A': 22, 'Product B' : 166, 'Product C':156}
series_1_dict
```

```
{'Product A': 22, 'Product B': 166, 'Product C': 156}
```

Step 4 Create a Series using the dictionary

Translate the keys of the dictionary into index values

```
series_2 = pd.Series(series_1_dict)
series_2
```

```
Product A      22
Product B     166
Product C     156
dtype: int64
```

Step 5 Apply the index attribute to verify if the indexing is position-based or label-based

Position-based implicit indexing

```
series_1.index
```

```
RangeIndex(start=0, stop=4, step=1)
```

Label-based explicit indexing

```
series_2.index
```

```
Index(['Product A', 'Product B', 'Product C'], dtype='object')
```

Step 6 Check the content of series_1

```
series_1
```

```
0      1
1      5
2     10
3     15
dtype: int64
```

Step 7 Obtain the value on index position 2

```
series_1[2]
```

```
10
```

Step 8 Check the content of series_2

```
series_2
```

```
Product A      22
Product B     166
Product C     156
dtype: int64
```

Step 9 Obtain the value on index position corresponding to Product B

```
series_2['Product B']
```

```
166
```

Step 10 Obtain the same value using an integer as an index

```
series_2[1]
```

```
166
```

Start your 365 Journey!