iteration-of-various-datatypes

May 19, 2024

[1]:
```
#Tuple:
t=(1,2,3)
for i in t:
    print(i)
```

```
1
2
3
```

[12]:
```
#List:
l=[1,2,3]
for i in l:
    print(i)
```

```
1
2
3
```

[2]:
```
#Numpy array:
import numpy as np
a=np.array([[1,2,3,4],[4,5,6,5]])
for i in a:
    print(i)
```

```
[1 2 3 4]
[4 5 6 5]
```

[3]:
```
for i in np.nditer(a):
    print(i)
```

```
1
2
3
4
4
5
6
5
```

[4]:
```
#Dictionary:
l1 = ['Moin', 'Rafi', 'Sid', 'Aneeq']
l2 = [24, 29, 26, 20]
d = dict(zip(l1, l2))
print(d)
```

```
{'Moin': 24, 'Rafi': 29, 'Sid': 26, 'Aneeq': 20}
```

[5]:
```
for i,j in d.items():
    print(i)
    print(j)
```

```
Moin
24
Rafi
29
Sid
26
Aneeq
20
```

[7]:
```
#Pandas Series:
import pandas as pd
series=pd.Series([924, 893, 800, 790],index=['Moin','Irfan','Saif','Zawyan'])
series
for i,j in series.items():
    print(i)
    print(j)
```

```
Moin
924
Irfan
893
Saif
800
Zawyan
790
```

[8]:
```
#Pandas DataFrame:
import numpy as np
import pandas as pd
data=np.array([[1,2,3],[3,2,2],[1,4,5],[5,4,3],[6,7,8],[2,3,4]])
df=pd.DataFrame(data,␣
 ↪columns=['col1','col2','col3'],index=['row1','row2','row3','row4','row5','row6'])
for x in df: #Prints cols.
    print(x)
```

```
col1
```

```
col2
col3
```

[9]:
```
'''To iterate over the columns of the DataFrame:'''
for column_name, series in df.items(): #Same as Dictionary
    print(column_name)
    print(series.values)
```

```
col1
[1 3 1 5 6 2]
col2
[2 2 4 4 7 3]
col3
[3 2 5 3 8 4]
```

[10]:
```
'''iterrows() returns the iterator yielding each index value along with a␣
 ↪series containing the data in each row.'''
for row_index,row in df.iterrows():
    print(row_index,row)
```

```
row1 col1    1
col2    2
col3    3
Name: row1, dtype: int32
row2 col1    3
col2    2
col3    2
Name: row2, dtype: int32
row3 col1    1
col2    4
col3    5
Name: row3, dtype: int32
row4 col1    5
col2    4
col3    3
Name: row4, dtype: int32
row5 col1    6
col2    7
col3    8
Name: row5, dtype: int32
row6 col1    2
col2    3
col3    4
Name: row6, dtype: int32
```

[11]:
```
'''itertuples() method will return an iterator yielding a named tuple for each␣
 ↪row in the DataFrame. The first element of the tuple will be the row's␣
 ↪corresponding index value, while the remaining values are the row values.'''
```

```
for row in df.itertuples():
    print(row)
```

```
Pandas(Index='row1', col1=1, col2=2, col3=3)
Pandas(Index='row2', col1=3, col2=2, col3=2)
Pandas(Index='row3', col1=1, col2=4, col3=5)
Pandas(Index='row4', col1=5, col2=4, col3=3)
Pandas(Index='row5', col1=6, col2=7, col3=8)
Pandas(Index='row6', col1=2, col2=3, col3=4)
```

[ ]: