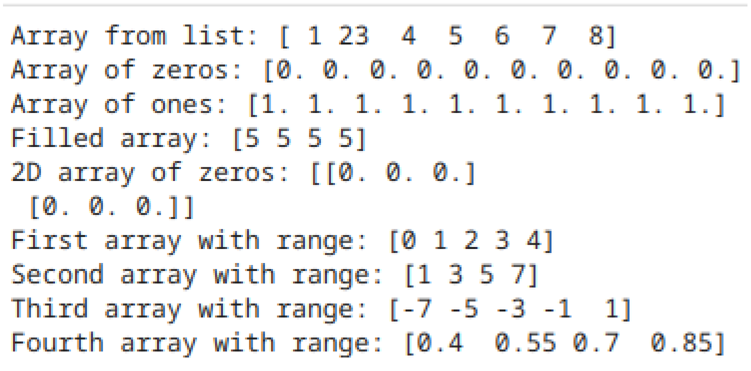
Creation of arrays in numpy

Source code:

```
import numpy as ujwal
def create_arrays():
    # Create a 1D array from a list
    lis = [1, 23, 4, 5, 6, 7, 8]
    ar = ujwal.array(lis)
    print("Array from list:", ar)
    # Create an array with 10 zeros
    arr = ujwal.zeros(10)
    print("Array of zeros:", arr)
    # Create an array with 10 ones
    ones_array = ujwal.ones(10)
    print("Array of ones:", ones_array)
    # Create a filled array of shape (4,) with the value 5
    filled_array = ujwal.full((4), 5)
    print("Filled array:", filled_array)
    # Create a 2D array of zeros with shape (2, 3)
    array_2d = ujwal.zeros((2, 3))
    print("2D array of zeros:", array_2d)
    # Create arrays using arange function
    first = ujwal.arange(0, 5, 1)
    print("First array with range:", first)
    sec = ujwal.arange(1, 8, 2)
    print("Second array with range:", sec)
    thi = ujwal.arange(-7, 2, 2)
    print("Third array with range:", thi)
    fou = ujwal.arange(0.4, 1, 0.15)
    print("Fourth array with range:", fou)
if __name__ == "__main__":
    create_arrays()
```

Output:



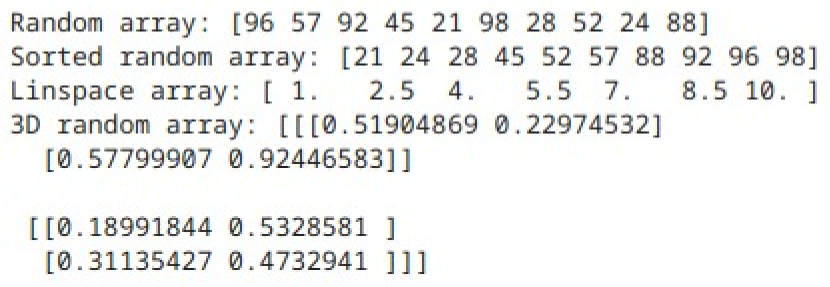
```
Array from list: [ 1 23  4  5  6  7  8]
Array of zeros: [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
Array of ones: [1. 1. 1. 1. 1. 1. 1. 1. 1. 1.]
Filled array: [5 5 5 5]
2D array of zeros: [[0. 0. 0.]
 [0. 0. 0.]]
First array with range: [0 1 2 3 4]
Second array with range: [1 3 5 7]
Third array with range: [-7 -5 -3 -1  1]
Fourth array with range: [0.4  0.55 0.7  0.85]
```

Generating Arrays with random values

Source code:

```
import numpy as ujwal
def random_arrays():
    # Generate a random array of integers between 0 and 100 of size 10
    random_array = ujwal.random.randint(0, 101, size=10)
    print("Random array:", random_array)

    # Sort the random array
    sorted_array = ujwal.sort(random_array)
    print("Sorted random array:", sorted_array)

    # Generate another random array using linspace
    linspace_array = ujwal.linspace(1, 10, 7)
    print("Linspace array:", linspace_array)

    # Generate a 3D random array
    random_3d_array = ujwal.random.rand(2, 2, 2)
    print("3D random array:", random_3d_array)

if __name__ == "__main__":
    random_arrays()
```

Output:



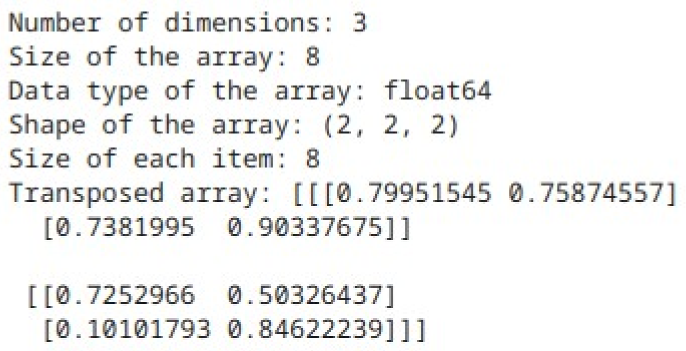
```
Random array: [96 57 92 45 21 98 28 52 24 88]
Sorted random array: [21 24 28 45 52 57 88 92 96 98]
Linspace array: [ 1.   2.5  4.   5.5  7.   8.5 10. ]
3D random array: [[[0.51904869 0.22974532]
  [0.57799907 0.92446583]]

 [[0.18991844 0.5328581 ]
  [0.31135427 0.4732941 ]]]
```

Properties of the array

Source code:

```
import numpy as ujwal
def array_properties():
    # Generate a random 3D array for property checks
    array_example = ujwal.random.rand(2, 2, 2)

    # Print array properties
    print("Number of dimensions:", array_example.ndim)
    print("Size of the array:", array_example.size)
    print("Data type of the array:", array_example.dtype)
    print("Shape of the array:", array_example.shape)
    print("Size of each item:", array_example.itemsize)

    # Transpose the array and print
    transposed_array = array_example.transpose()
    print("Transposed array:", transposed_array)

if __name__ == "__main__":
    array_properties()
```

Output:



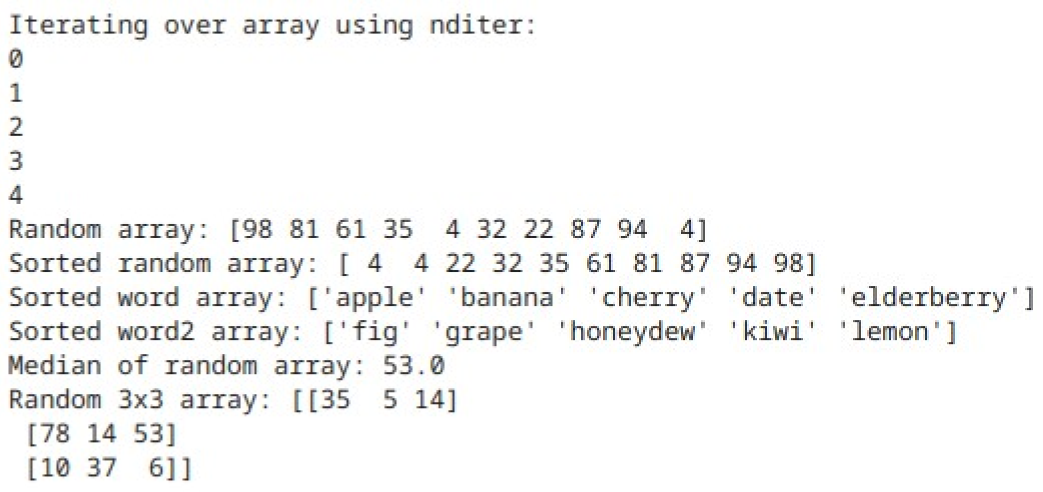
```
Number of dimensions: 3
Size of the array: 8
Data type of the array: float64
Shape of the array: (2, 2, 2)
Size of each item: 8
Transposed array: [[[0.79951545 0.75874557]
  [0.7381995  0.90337675]]

 [[0.7252966  0.50326437]
  [0.10101793 0.84622239]]]
```

Manipulating arrays, indexing and slicing

Source code:

```
import numpy as ujwal
def array_manipulation():
    # Create a 1D array
    first = ujwal.arange(0, 5, 1)
    # Iterate over the array using nditer
    print("Iterating over array using nditer:")
    for x in ujwal.nditer(first):
        print(x)
    # Create a random array and sort it
    random_array = ujwal.random.randint(0, 101, size=10)
    print("Random array:", random_array)
    sorted_array = ujwal.sort(random_array)
    print("Sorted random array:", sorted_array)
    # Original word lists for lexical sorting
    word = ["elderberry", "apple", "banana", "cherry", "date"]
    word2 = ["honeydew", "fig", "grape", "kiwi", "lemon"]
    # Convert lists to NumPy arrays
    word_array = ujwal.array(word)
    word2_array = ujwal.array(word2)
    # Lexically sort the arrays using lexsort
    sorted_indices_word = ujwal.lexsort([word_array])
    sorted_indices_word2 = ujwal.lexsort([word2_array])
    sorted_word = word_array[sorted_indices_word]
    sorted_word2 = word2_array[sorted_indices_word2]
    print("Sorted word array:", sorted_word)
    print("Sorted word2 array:", sorted_word2)
    # Generate a random array for median calculation
    random_for_median = ujwal.random.randint(0, 101, size=10)
    median_value = ujwal.median(random_for_median)
    print("Median of random array:", median_value)
    # Create a random 3x3 array for demonstration
    ranarr = ujwal.random.randint(1, 100, size=(3, 3))
    print("Random 3x3 array:", ranarr)
if __name__ == "__main__":
    array_manipulation()
```

Output:



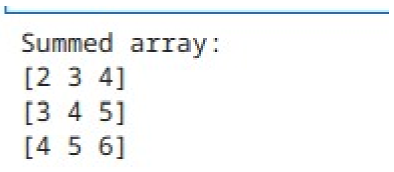
```
Iterating over array using nditer:
0
1
2
3
4
Random array: [98 81 61 35  4 32 22 87 94  4]
Sorted random array: [ 4  4 22 32 35 61 81 87 94 98]
Sorted word array: ['apple' 'banana' 'cherry' 'date' 'elderberry']
Sorted word2 array: ['fig' 'grape' 'honeydew' 'kiwi' 'lemon']
Median of random array: 53.0
Random 3x3 array: [[35  5 14]
 [78 14 53]
 [10 37  6]]
```

Element wise addition in numpy

Source code:

```
import numpy as ujwal

def elementwise_operations():
    # Create two 1D arrays for element-wise addition
    array1 = ujwal.array([1, 2, 3])
    array2 = ujwal.array([[1], [2], [3]])

    # Element-wise addition
    summed = array1 + array2

    # Print the result
    print("Summed array:")
    for x in summed:
        print(x)

if __name__ == "__main__":
    elementwise_operations()
```

Output:



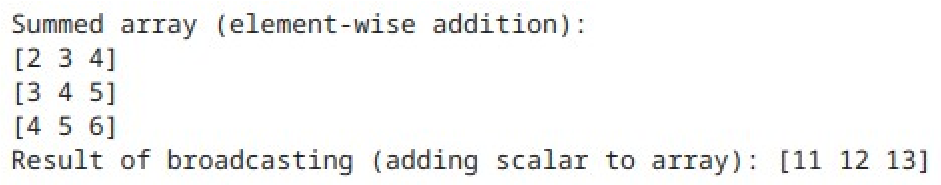
```
Summed array:
[2 3 4]
[3 4 5]
[4 5 6]
```

Array broadcasting

Source code:

```
import numpy as ujwal
def elementwise_operations():
    # Create two 1D arrays for element-wise addition
    array1 = ujwal.array([1, 2, 3])
    array2 = ujwal.array([[1], [2], [3]])  # 2D array with shape (3, 1)

    # Element-wise addition
    summed = array1 + array2

    # Print the result of element-wise addition
    print("Summed array (element-wise addition):")
    for x in summed:
        print(x)

    # Example of array broadcasting
    scalar_value = 10
    broadcasted_result = array1 + scalar_value  # Adding a scalar to the 1D array

    # Print the result of broadcasting
    print("Result of broadcasting (adding scalar to array):", broadcasted_result)
if __name__ == "__main__":
    elementwise_operations()
```

Output:



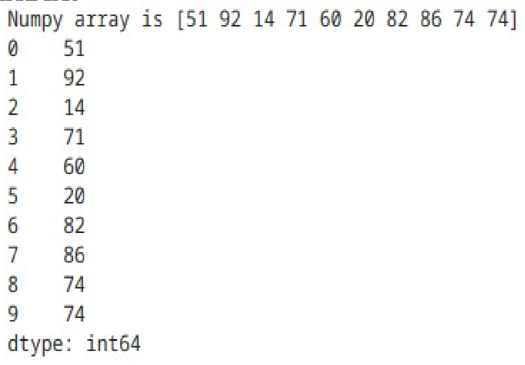
```
Summed array (element-wise addition):
[2 3 4]
[3 4 5]
[4 5 6]
Result of broadcasting (adding scalar to array): [11 12 13]
```

Creating a series in pandas

Source Code:

```
import numpy as np
import pandas as pd
np.random.seed(42)
this=np.random.randint(0, 101, size=10)
print(f"Numpy array is {this}")
that=pd.Series(this)
print(that)
```

Output:



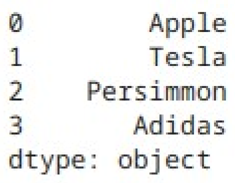
```
Numpy array is [51 92 14 71 60 20 82 86 74 74]
0    51
1    92
2    14
3    71
4    60
5    20
6    82
7    86
8    74
9    74
dtype: int64
```

series using a list

Source Code:

```
import pandas as pd
ls=["Apple","Tesla","Persimmon","Adidas"]
l=pd.Series(ls)
print(l)
```

Output:



```
0        Apple
1        Tesla
2    Persimmon
3       Adidas
dtype: object
```

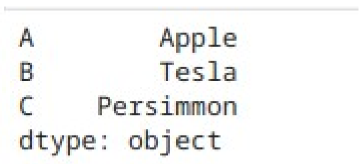
Slicing and Indexing series

Accessing data using built in index

Source Code:

```
import pandas as pd
ls=["Apple","Tesla","Persimmon","Adidas"]
x = pd.Series(ls, index = ["A", "B", "C","D"])
print(x.iloc[0:3])
```

Output:



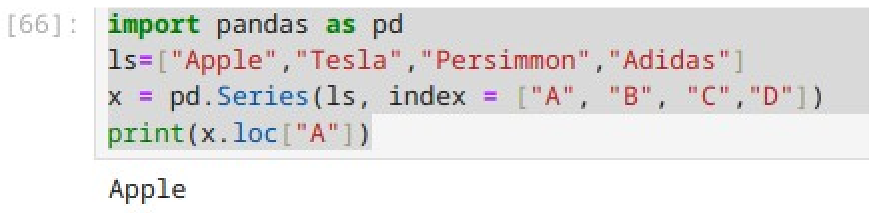
```
A          Apple
B          Tesla
C      Persimmon
dtype: object
```

Accessing data using custom index

Source Code:

```
import pandas as pd
ls=["Apple","Tesla","Persimmon","Adidas"]
x = pd.Series(ls, index = ["A", "B", "C","D"])
print(x.loc["A"])
```

Output:



```
[66]: import pandas as pd
      ls=["Apple","Tesla","Persimmon","Adidas"]
      x = pd.Series(ls, index = ["A", "B", "C","D"])
      print(x.loc["A"])

      Apple
```

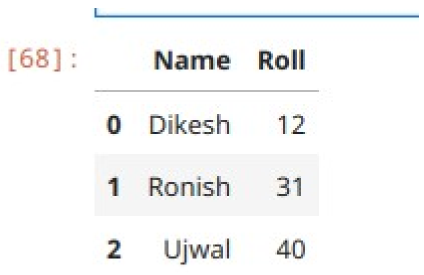
Dataframes

Creating dataframes

Source Code:

```
import pandas as ujwal
dat=[['Dikesh',12],['Ronish',31],['Ujwal',40]]
df=ujwal.DataFrame(dat,columns=['Name','Roll'])
df
```

Output:



[68]:

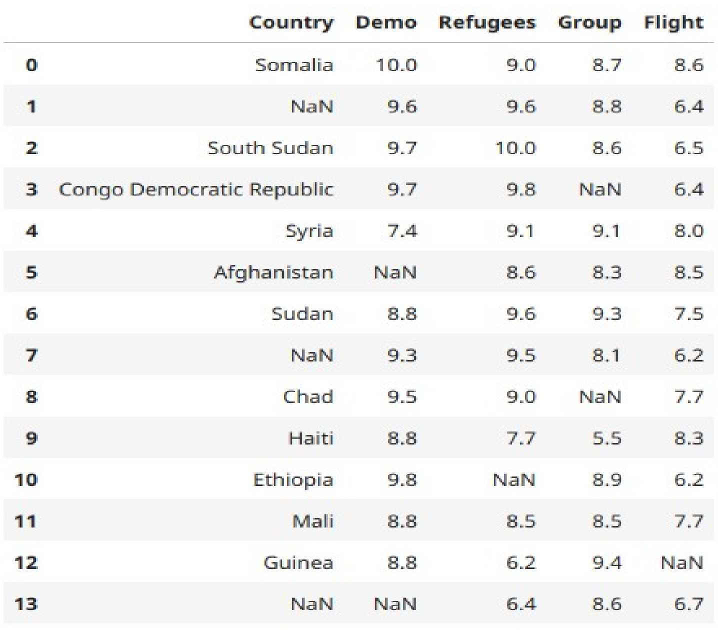
	Name	Roll
0	Dikesh	12
1	Ronish	31
2	Ujwal	40

Creating Dataframes from excel files

Source Code:

```
import pandas as ujwal
fil="/home/nullproj/Documents/BIM-5th-semester/python/practice/pandas/FSI.xlsx"
df=ujwal.read_excel(fil)
df
```

Output:



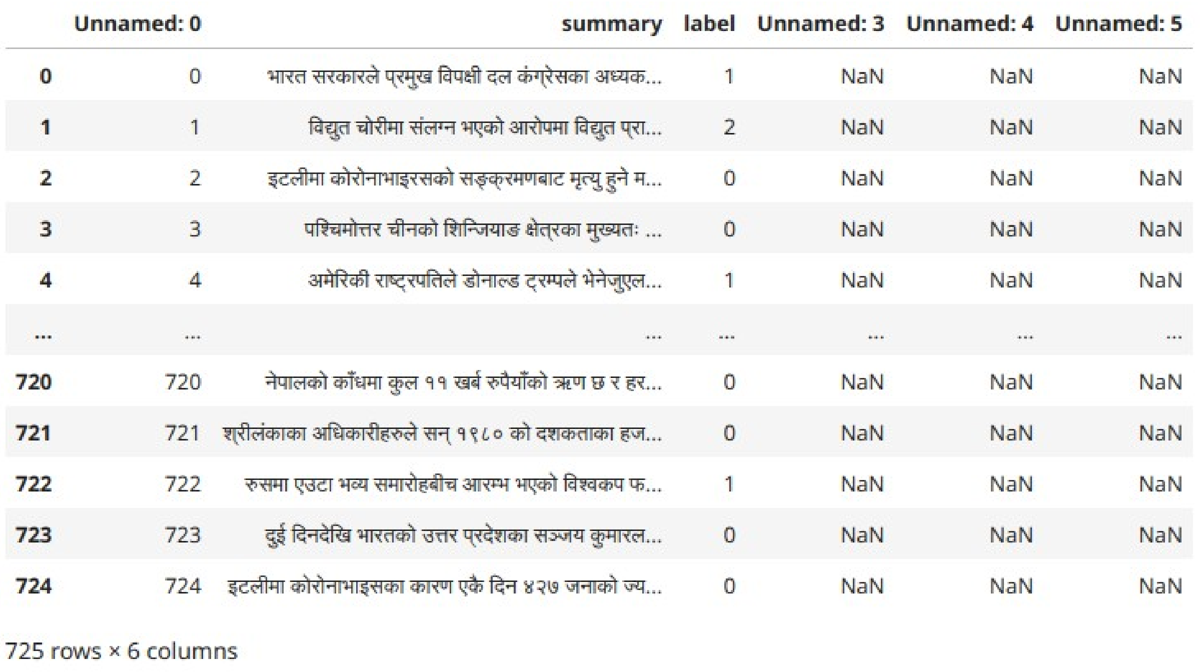
	Country	Demo	Refugees	Group	Flight
0	Somalia	10.0	9.0	8.7	8.6
1	NaN	9.6	9.6	8.8	6.4
2	South Sudan	9.7	10.0	8.6	6.5
3	Congo Democratic Republic	9.7	9.8	NaN	6.4
4	Syria	7.4	9.1	9.1	8.0
5	Afghanistan	NaN	8.6	8.3	8.5
6	Sudan	8.8	9.6	9.3	7.5
7	NaN	9.3	9.5	8.1	6.2
8	Chad	9.5	9.0	NaN	7.7
9	Haiti	8.8	7.7	5.5	8.3
10	Ethiopia	9.8	NaN	8.9	6.2
11	Mali	8.8	8.5	8.5	7.7
12	Guinea	8.8	6.2	9.4	NaN
13	NaN	NaN	6.4	8.6	6.7

Creating Dataframes from CSV file

Source code:

```
import pandas as ujwal
name="/home/nullproj/Documents/BIM-5th-semester/python/practice/pandas/output.csv"
df=ujwal.read_csv(name,on_bad_lines="skip")
df
```

Output:



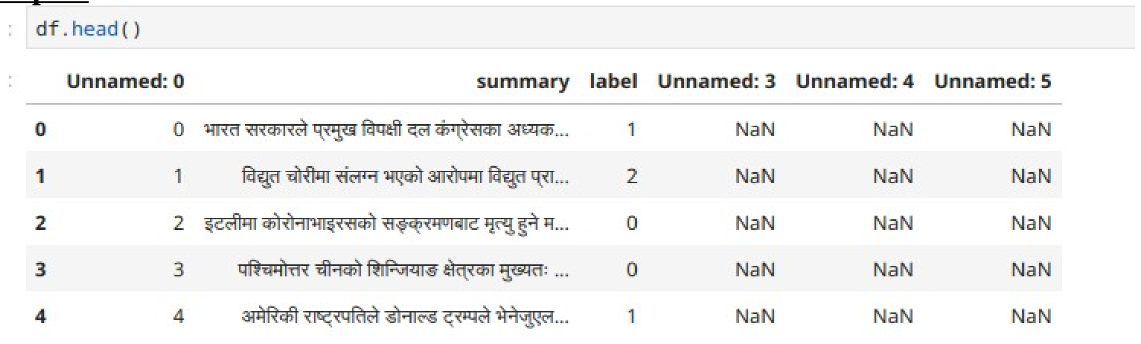
	Unnamed: 0	summary	label	Unnamed: 3	Unnamed: 4	Unnamed: 5
0	0	भारत सरकारले प्रमुख विपक्षी दल कंग्रेसका अध्यक...	1	NaN	NaN	NaN
1	1	विद्युत चोरीमा संलग्न भएको आरोपमा विद्युत प्रा...	2	NaN	NaN	NaN
2	2	इटलीमा कोरोनाभाइरसको सङ्क्रमणबाट मृत्यु हुने म...	0	NaN	NaN	NaN
3	3	पश्चिमोत्तर चीनको शिन्जियाङ क्षेत्रका मुख्यतः ...	0	NaN	NaN	NaN
4	4	अमेरिकी राष्ट्रपतिले डोनाल्ड ट्रम्पले भेनेजुएल...	1	NaN	NaN	NaN
...	...	...	...	...	...	...
720	720	नेपालको काँधमा कुल ११ खर्ब रुपैयाँको ऋण छ र हर...	0	NaN	NaN	NaN
721	721	श्रीलंकाका अधिकारीहरुले सन् १९८० को दशकताका हज...	0	NaN	NaN	NaN
722	722	रुसमा एउटा भव्य समारोहबीच आरम्भ भएको विश्वकप फ...	1	NaN	NaN	NaN
723	723	दुई दिनदेखि भारतको उत्तर प्रदेशका सञ्जय कुमारल...	0	NaN	NaN	NaN
724	724	इटलीमा कोरोनाभाइसका कारण एकै दिन ४२७ जनाको ज्य...	0	NaN	NaN	NaN

725 rows × 6 columns

Dataframe Properties

df.head(): Display top 5 lines of the dataframe

Output:



```
df.head()
```

	Unnamed: 0	summary	label	Unnamed: 3	Unnamed: 4	Unnamed: 5
0	0	भारत सरकारले प्रमुख विपक्षी दल कंग्रेसका अध्यक...	1	NaN	NaN	NaN
1	1	विद्युत चोरीमा संलग्न भएको आरोपमा विद्युत प्रा...	2	NaN	NaN	NaN
2	2	इटलीमा कोरोनाभाइरसको सङ्क्रमणबाट मृत्यु हुने म...	0	NaN	NaN	NaN
3	3	पश्चिमोत्तर चीनको शिन्जियाङ क्षेत्रका मुख्यतः ...	0	NaN	NaN	NaN
4	4	अमेरिकी राष्ट्रपतिले डोनाल्ड ट्रम्पले भेनेजुएल...	1	NaN	NaN	NaN

df.tail(): Display last 5 lines of the dataframe

df.shape(): Display the shape of the dataframe

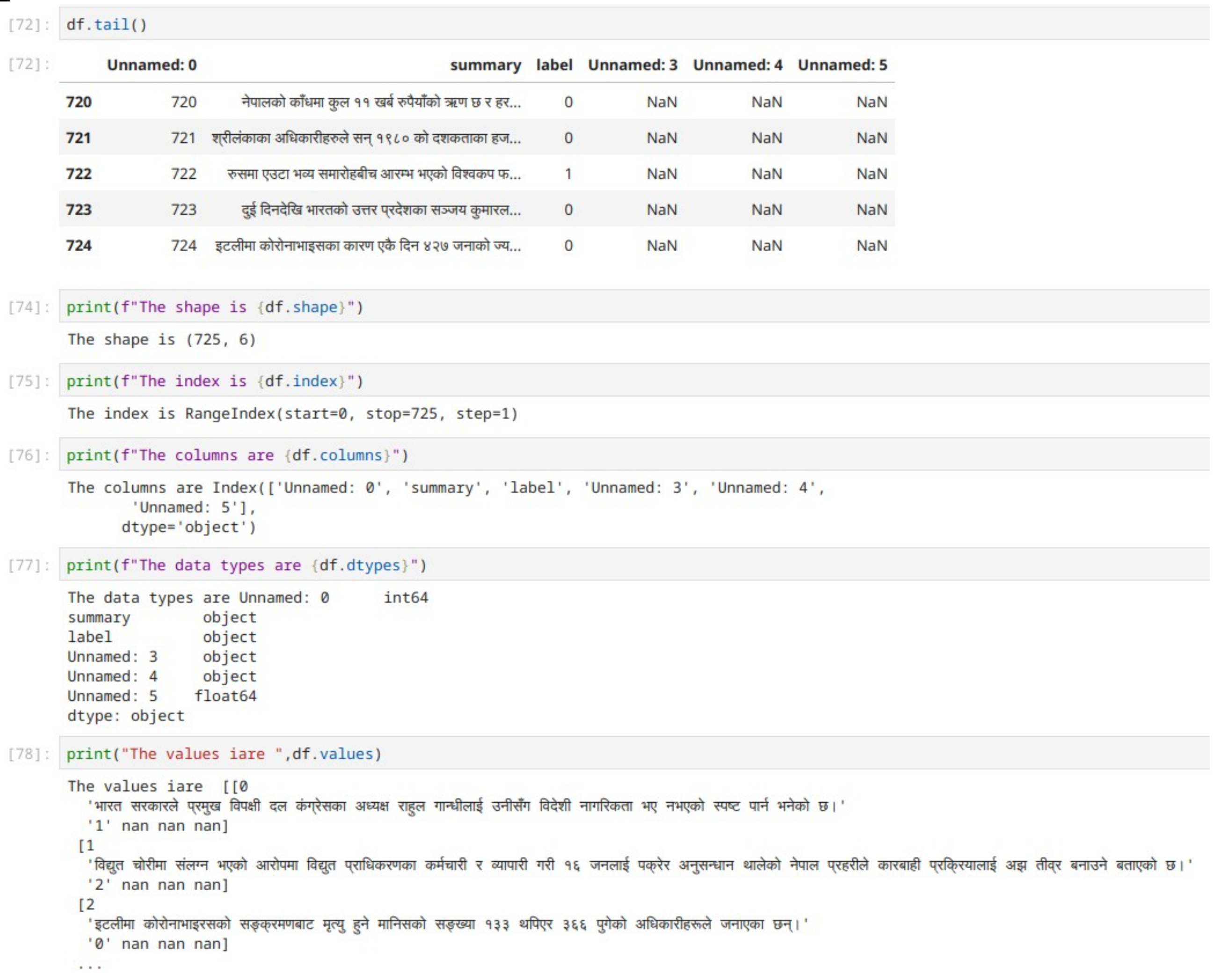
df.index(): Display the index range of the dataframe

df.columns(): Display the columns of the dataframe

df.dtypes(): Display the data types of the dataframe

df.values(): Display the values of the dataframe

Output:



```
[72]: df.tail()
```

[72]:

	Unnamed: 0	summary	label	Unnamed: 3	Unnamed: 4	Unnamed: 5
720	720	नेपालको काँधमा कुल ११ खर्ब रुपैयाँको ऋण छ र हर...	0	NaN	NaN	NaN
721	721	श्रीलंकाका अधिकारीहरुले सन् १९८० को दशकताका हज...	0	NaN	NaN	NaN
722	722	रुसमा एउटा भव्य समारोहबीच आरम्भ भएको विश्वकप फ...	1	NaN	NaN	NaN
723	723	दुई दिनदेखि भारतको उत्तर प्रदेशका सञ्जय कुमारल...	0	NaN	NaN	NaN
724	724	इटलीमा कोरोनाभाइसका कारण एकै दिन ४२७ जनाको ज्य...	0	NaN	NaN	NaN

```
[74]: print(f"The shape is {df.shape}")

      The shape is (725, 6)

[75]: print(f"The index is {df.index}")

      The index is RangeIndex(start=0, stop=725, step=1)

[76]: print(f"The columns are {df.columns}")

      The columns are Index(['Unnamed: 0', 'summary', 'label', 'Unnamed: 3', 'Unnamed: 4',
             'Unnamed: 5'],
            dtype='object')

[77]: print(f"The data types are {df.dtypes}")

      The data types are Unnamed: 0      int64
      summary         object
      label           object
      Unnamed: 3      object
      Unnamed: 4      object
      Unnamed: 5     float64
      dtype: object

[78]: print("The values iare ",df.values)

      The values iare  [[0
        'भारत सरकारले प्रमुख विपक्षी दल कंग्रेसका अध्यक्ष राहुल गान्धीलाई उनीसँग विदेशी नागरिकता भए नभएको स्पष्ट पार्न भनेको छ।'
        '1' nan nan nan]
       [1
        'विद्युत चोरीमा संलग्न भएको आरोपमा विद्युत प्राधिकरणका कर्मचारी र व्यापारी गरी १६ जनलाई पक्रेर अनुसन्धान थालेको नेपाल प्रहरीले कारबाही प्रक्रियालाई अझ तीव्र बनाउने बताएको छ।'
        '2' nan nan nan]
       [2
        'इटलीमा कोरोनाभाइरसको सङ्क्रमणबाट मृत्यु हुने मानिसको सङ्ख्या १३३ थपिएर ३६६ पुगेको अधिकारीहरूले जनाएका छन्।'
        '0' nan nan nan]
       ...
```

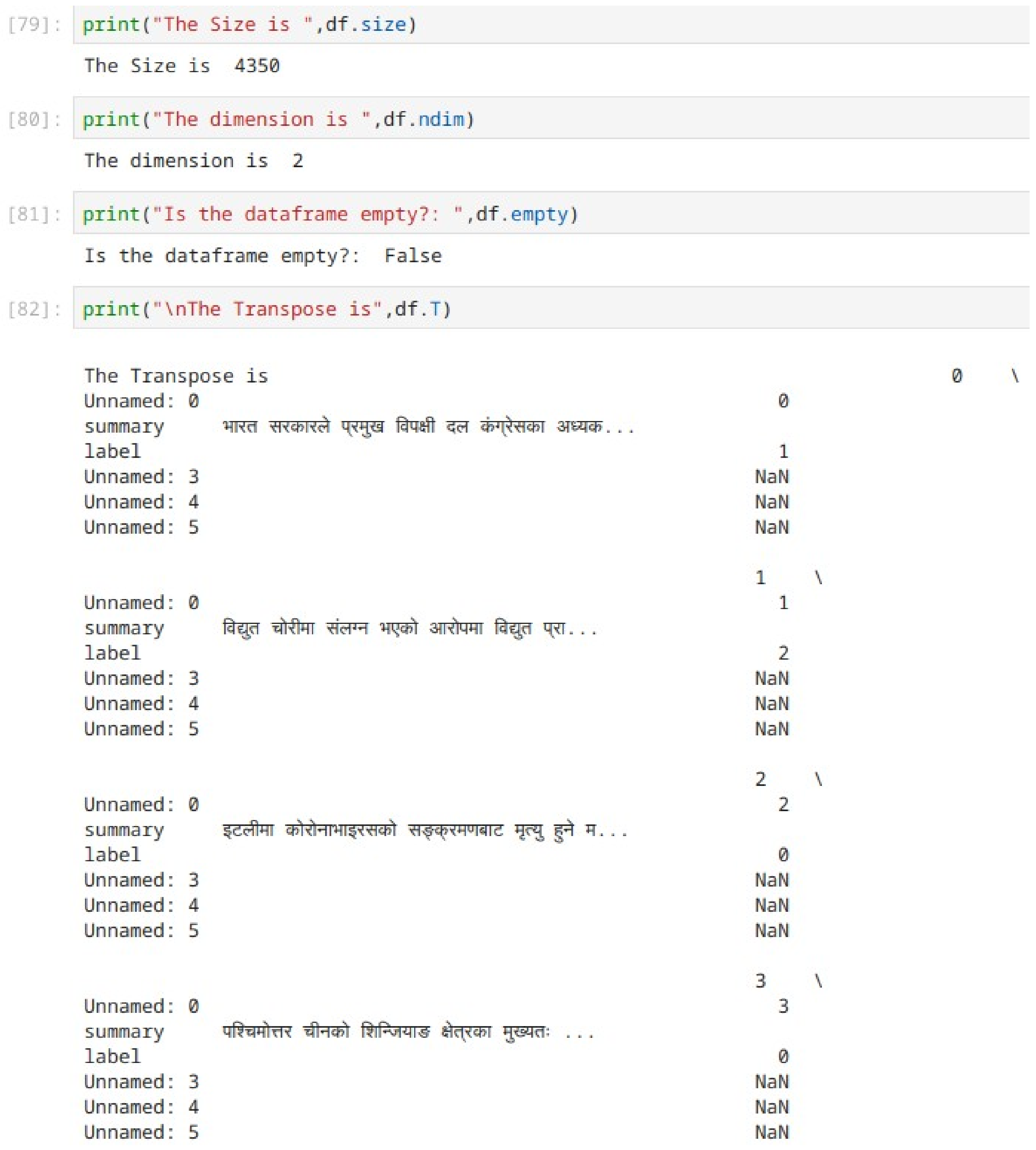
df.size(): Display the size of the dataframe

df.ndim(): Display the dimensions of the dataframe

df.empty(): check if the dataframe is empty

df.T(): Display the Transpose of the dataframe

Output:



```
[79]: print("The Size is ",df.size)

      The Size is  4350

[80]: print("The dimension is ",df.ndim)

      The dimension is  2

[81]: print("Is the dataframe empty?: ",df.empty)

      Is the dataframe empty?:  False

[82]: print("\nThe Transpose is",df.T)

      The Transpose is                                                      0    \
      Unnamed: 0                                                   0
      summary     भारत सरकारले प्रमुख विपक्षी दल कंग्रेसका अध्यक...
      label                                                        1
      Unnamed: 3                                                 NaN
      Unnamed: 4                                                 NaN
      Unnamed: 5                                                 NaN

                                                                 1    \
      Unnamed: 0                                                   1
      summary     विद्युत चोरीमा संलग्न भएको आरोपमा विद्युत प्रा...
      label                                                        2
      Unnamed: 3                                                 NaN
      Unnamed: 4                                                 NaN
      Unnamed: 5                                                 NaN

                                                                 2    \
      Unnamed: 0                                                   2
      summary     इटलीमा कोरोनाभाइरसको सङ्क्रमणबाट मृत्यु हुने म...
      label                                                        0
      Unnamed: 3                                                 NaN
      Unnamed: 4                                                 NaN
      Unnamed: 5                                                 NaN

                                                                 3    \
      Unnamed: 0                                                   3
      summary     पश्चिमोत्तर चीनको शिन्जियाङ क्षेत्रका मुख्यतः  ...
      label                                                        0
      Unnamed: 3                                                 NaN
      Unnamed: 4                                                 NaN
      Unnamed: 5                                                 NaN
```

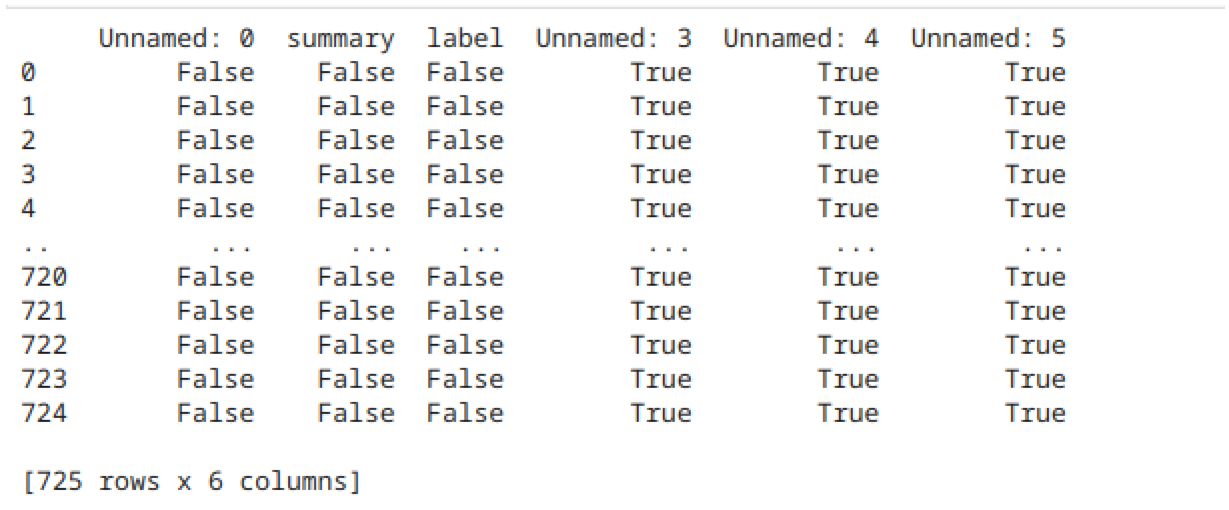
Full result of transpose is not possible to include because it’s too long

Checking Null values in a dataframe

Source Code:

```
import pandas as ujwal

filename="/home/nullproj/Documents/BIM-5th-semester/python/practice/pandas/FSINULL.xlsx"

fd=ujwal.read_excel(filename)

print(df.isnull())
```

Output:



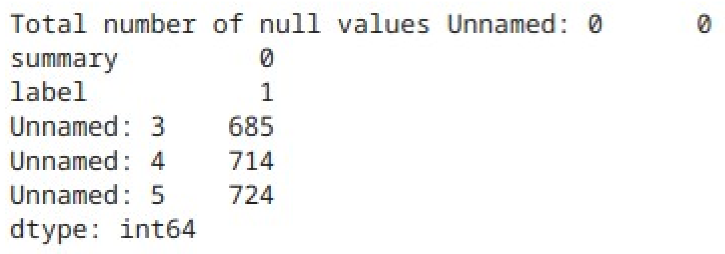
```
     Unnamed: 0  summary  label  Unnamed: 3  Unnamed: 4  Unnamed: 5
0         False    False  False        True        True        True
1         False    False  False        True        True        True
2         False    False  False        True        True        True
3         False    False  False        True        True        True
4         False    False  False        True        True        True
..          ...      ...    ...         ...         ...         ...
720       False    False  False        True        True        True
721       False    False  False        True        True        True
722       False    False  False        True        True        True
723       False    False  False        True        True        True
724       False    False  False        True        True        True

[725 rows x 6 columns]
```

Continuing on the same file

Source code:

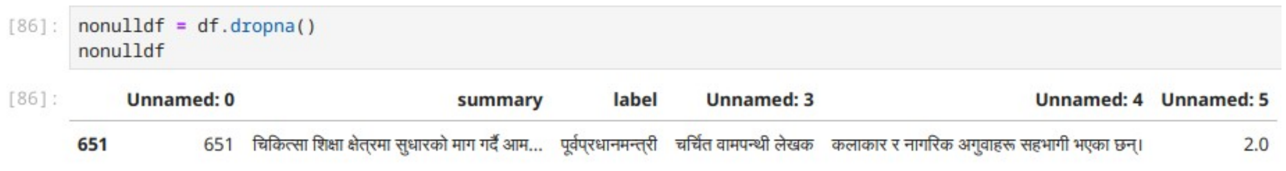
```
#to list the number of null values

totalnull=df.isnull().sum()

print("Total number of null values",totalnull)
```

Output:



```
Total number of null values Unnamed: 0      0
summary         0
label           1
Unnamed: 3    685
Unnamed: 4    714
Unnamed: 5    724
dtype: int64
```

removing null values from a dataframe



```
[86]: nonulldf = df.dropna()
      nonulldf
```

[86]:

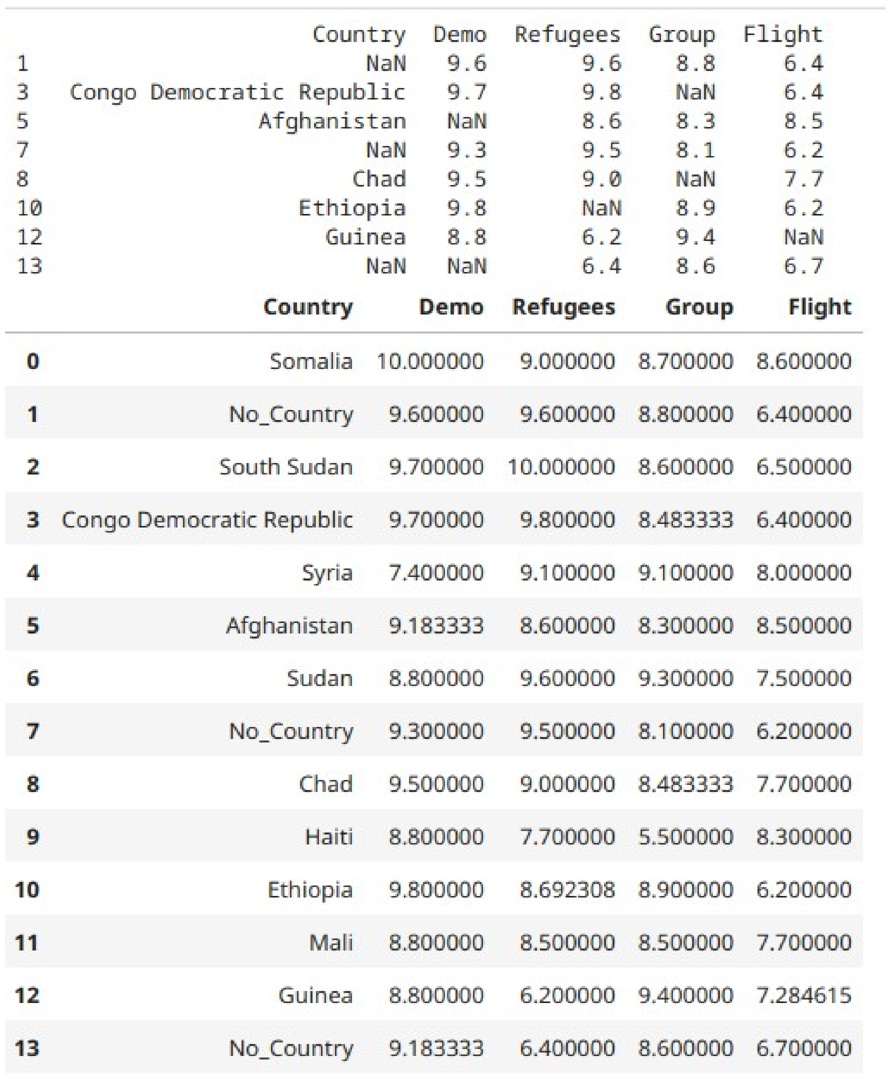
	Unnamed: 0	summary	label	Unnamed: 3	Unnamed: 4	Unnamed: 5
651	651	चिकित्सा शिक्षा क्षेत्रमा सुधारको माग गर्दै आम...	पूर्वप्रधानमन्त्री	चर्चित वामपन्थी लेखक	कलाकार र नागरिक अगुवाहरू सहभागी भएका छन्।	2.0

Filling the null values in the dataframe with mean of each

<u>Source Code:</u>

```
import pandas as pd
filename = "/home/nullproj/Documents/BIM-5th-semester/python/practice/pandas/FSINULL.xlsx"
df = pd.read_excel(filename)
print(df[df.isna().any(axis =1)])
#Filling the missing values with the mean valuesdf
df['Demo'].fillna(df['Demo'].mean(), inplace = True)
df['Refugees'].fillna(df['Refugees'].mean(), inplace = True)
df['Group'].fillna(df['Group'].mean(), inplace = True)
df['Flight'].fillna(df['Flight'].mean(), inplace = True)
df['Country'].fillna('No_Country', inplace = True)
df
```

<u>Output:</u>



```
                      Country  Demo  Refugees  Group  Flight
1                         NaN   9.6       9.6    8.8     6.4
3   Congo Democratic Republic   9.7       9.8    NaN     6.4
5                 Afghanistan   NaN       8.6    8.3     8.5
7                         NaN   9.3       9.5    8.1     6.2
8                        Chad   9.5       9.0    NaN     7.7
10                   Ethiopia   9.8       NaN    8.9     6.2
12                     Guinea   8.8       6.2    9.4     NaN
13                        NaN   NaN       6.4    8.6     6.7
```

	Country	Demo	Refugees	Group	Flight
0	Somalia	10.000000	9.000000	8.700000	8.600000
1	No_Country	9.600000	9.600000	8.800000	6.400000
2	South Sudan	9.700000	10.000000	8.600000	6.500000
3	Congo Democratic Republic	9.700000	9.800000	8.483333	6.400000
4	Syria	7.400000	9.100000	9.100000	8.000000
5	Afghanistan	9.183333	8.600000	8.300000	8.500000
6	Sudan	8.800000	9.600000	9.300000	7.500000
7	No_Country	9.300000	9.500000	8.100000	6.200000
8	Chad	9.500000	9.000000	8.483333	7.700000
9	Haiti	8.800000	7.700000	5.500000	8.300000
10	Ethiopia	9.800000	8.692308	8.900000	6.200000
11	Mali	8.800000	8.500000	8.500000	7.700000
12	Guinea	8.800000	6.200000	9.400000	7.284615
13	No_Country	9.183333	6.400000	8.600000	6.700000

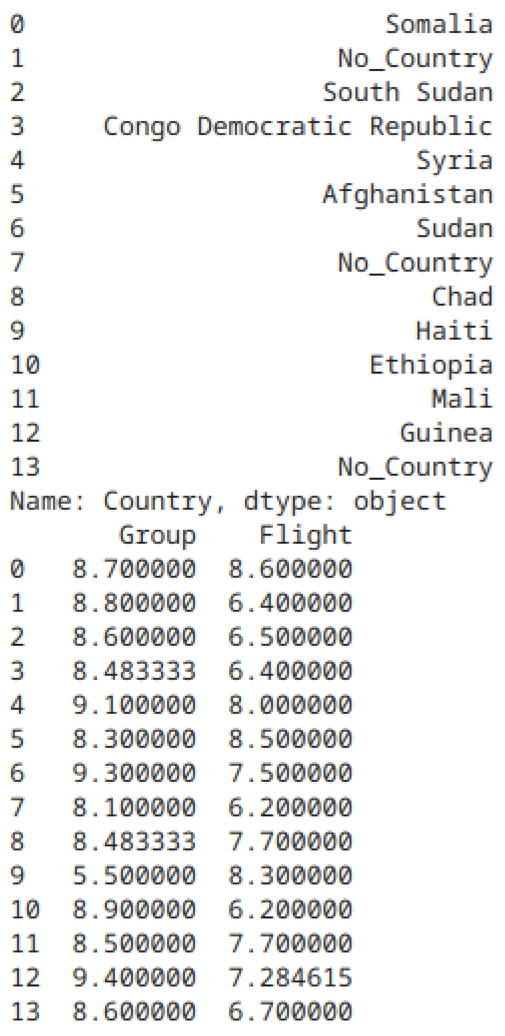
Continuing on the same file

displaying specific column

<u>Source Code:</u>

```
col1 = df['Country']

print(col1)

col2 = df[['Group','Flight']]

print(col2)
```

<u>Output:</u>



```
0                          Somalia
1                       No_Country
2                      South Sudan
3      Congo Democratic Republic
4                            Syria
5                      Afghanistan
6                            Sudan
7                       No_Country
8                             Chad
9                            Haiti
10                        Ethiopia
11                            Mali
12                          Guinea
13                      No_Country
Name: Country, dtype: object
       Group    Flight
0   8.700000  8.600000
1   8.800000  6.400000
2   8.600000  6.500000
3   8.483333  6.400000
4   9.100000  8.000000
5   8.300000  8.500000
6   9.300000  7.500000
7   8.100000  6.200000
8   8.483333  7.700000
9   5.500000  8.300000
10  8.900000  6.200000
11  8.500000  7.700000
12  9.400000  7.284615
13  8.600000  6.700000
```