Lab 7 exercises

Q1) sort the following key array in **heapsort algorithm** then apply the order to the values array

key = np.array([2, 2, 1, 1, 1])

values = np.array(['2:first', '2:second', '1:first', '1:second','1:third'])

output:

[4 2 3 1 0]

array(['1:third', '1:first', '1:second', '2:second', '2:first'], dtype='<U8')

Q3) create data frame as shown then change any one under 55 to w (warning) and print the enrollment year of those students then print the average for the whole class.

	names	ids	GPAs
0	ahmed	120215568	89
1	mohammed	120206874	77
2	anas	120219874	52
3	foad	120194532	60
4	loay	120212258	58
5	gamal	120214521	54
6	saed	120227452	92

	names	ids	GPAs
0	ahmed	120215568	89
1	mohammed	120206874	77
2	w	w	w
3	foad	120194532	60
4	loay	120212258	58
5	w	w	w
6	saed	120227452	92

Q4) create DataFrame from the following dictionary then add the country column, after that sort the DataFrame by the distance, then create method that get two cities and print the nearest city and how much it is nearer than the other city.

sdata = {'cities': ['Paris', 'Marseille', 'Lyon', 'Toulouse', 'Strasbourg', 'Le Mans'],

'distance': ['1542.1', '8695.5', '2587.5', '1875.7', '5987.0', '11014.9'],}

	cities	distance	country
0	Paris	1542.1	FRANCE
3	Toulouse	1875.7	FRANCE
2	Lyon	2587.5	FRANCE
4	Strasbourg	5987.0	FRANCE
1	Marseille	8695.5	FRANCE
5	Le Mans	11014.9	FRANCE

Q5) Write a program to compare the elements of the two Pandas Series.

Sample Series: [2, 4, 6, 8, 10], [1, 3, 5, 7, 10]

Output:

```
Equals:
0    False
1    False
2    False
3    False
4     True
dtype: bool
Greater than:
0     True
1     True
2     True
3     True
4    False
dtype: bool
Less than:
0    False
1    False
2    False
3    False
4    False
dtype: bool
```

Q6) Write a Pandas program to convert the first column of a DataFrame as a Series, then display most frequent value in a given series and replace everything else as 'Other' in the series.

data = {'col1': [1, 2, 3, 4, 7, 11, 22, 3, 2, 3, 9], 'col2': [4, 5, 6, 9, 5, 0, 4, 6, 7, 8, 0], 'col3': [7, 5, 8, 12, 1, 11, 5 ,8 ,22 , 6, 5]}

Output:

```
Original DataFrame
    col1  col2  col3
0      1     4     7
1      2     5     5
2      3     6     8
3      4     9    12
4      7     5     1
5     11     0    11
6     22     4     5
7      3     6     8
8      2     7    22
9      3     8     6
10     9     0     5
```

```
1st column as a Series:
0      1
1      2
2      3
3      4
4      7
5     11
6     22
7      3
8      2
9      3
10     9
```

```
 Top value Freq:
  3     3
2     2
1     1
4     1
7     1
11    1
22    1
9     1
Name: col1, dtype: int64
```

```
0     Other
1     Other
2         3
3     Other
4     Other
5     Other
6     Other
7         3
8     Other
9         3
10    Other
Name: col1, dtype: object
```