QUANTILE IN PYTHON

What is quantile in Python?

quantile() in **Python**. numpy. **quantile**(arr, q, axis = None) : Compute the q^{th} **quantile** of the given data (array elements) along the specified axis. **Quantile** plays a very important role in Statistics when one deals with the Normal Distribution. In the figure given above, Q2 is the median of the normally distributed data.

The word "**quantile**" comes from the word quantity. In simple terms, a **quantile** is where a sample is divided into equal-sized, adjacent, subgroups (that's why it's sometimes called a "fractile"). It can also refer to dividing a probability distribution into areas of equal probability.

DataFrame - quantile() function

The quantile() function is used to get values at the given quantile over requested axis.

Returns: Series or DataFrame
If q is an array, a DataFrame will be returned where the index is q, the columns are the columns of self, and the values are the quantiles.
If q is a float, a Series will be returned where the index is the columns of self and the values are the quantiles.

```
import numpy as np
import pandas as pd
df = pd.DataFrame(np.array([[2, 2], [3, 20], [4, 200], [5, 200]]),
                  columns=['p', 'q'])
df.quantile(.1)
```

OUTPUT:

```
p    2.3
q    7.4
Name: 0.1, dtype: float64
```

```
>>df.quantile([.2, .5])
```

OUTPUT

	p	q
0.2	2.6	12.8
0.5	3.5	110.0

EXAMPLE:

```
import pandas as pd
import numpy as np
df= pd.DataFrame(np.array([[1, 1], [2, 10], [3, 100], [4, 1000]]),columns=['a', 'b'])
print(df)
print(df.quantile(0.5))  #50% quantile or median
```

importing values from an Excel file to create pandas DataFrame

```
import pandas as pd
Cars = pd.read_excel(r'C:\Users\LVISLAPTOP31\Desktop\CARS.xlsx')
df = pd.DataFrame(cars, columns = ['BRAND', 'PRICE'])
max1 = df['Price'].max()
print (df)
```

1. **create pandas DataFrame**
2. **get the maximum price for our Cars**

```
           Brand  Price
0     Honda Civic  22000
1  Toyota Corolla  25000
2      Ford Focus  27000
3         Audi A4  35000

In [5]:
```

SOL:

```
import pandas as pd
cars = {'Brand': ['Honda Civic','Toyota Corolla','Ford Focus','Audi A4'],
        'Price': [22000,25000,27000,35000]
        }
df = pd.DataFrame(cars, columns = ['Brand', 'Price'])
print (df)
max1 = df['Price'].max()
print (max1)
```

3. **ADD YEAR COLUMN IN THE ABOVE DATAFRAME**:

```
            Brand  Price  Year
0     Honda Civic  22000  2015
1  Toyota Corolla  25000  2013
2      Ford Focus  27000  2018
3         Audi A4  35000  2018
```

SOL:

```
import pandas as pd
Cars = {'Brand': ['Honda Civic','Toyota Corolla','Ford Focus','Audi A4'],
        'Price': [22000,25000,27000,35000],
        'Year': [2015,2013,2018,2018]
        }
 df = pd.DataFrame(Cars, columns= ['Brand', 'Price','Year'])
print (df)
```

To sort pandas DataFrame, you may use the **df.sort_values** syntax in Python.

Pandas `sort_values()` function sorts a data frame in Ascending or Descending order of passed Column.

Example 1: Sort Pandas DataFrame in an *ascending* order

Let's say that you want to sort the DataFrame, such that the Brand will be displayed in an ascending order.

Note that unless specified, the values will be sorted in an ascending order *by default*.

df.sort_values(by=['Brand'], inplace=True)

When you run the code, you'll notice that the Brand will indeed get sorted in an ascending order, where Audi A4 would be the first record, while Toyota Corolla would be the last

```
#import pandas as pd

# sort - ascending order

from pandas import DataFrame

 Cars = {'Brand': ['Honda Civic','Toyota Corolla','Ford Focus','Audi A4'],

       'Price': [22000,25000,27000,35000],

       'Year': [2015,2013,2018,2018]

       }

 df = DataFrame(Cars, columns= ['Brand', 'Price','Year'])

# sort Brand - ascending order

df.sort_values(by=['Brand'], inplace=True)

print (df)
```

Example 2: Sort Pandas DataFrame in a *descending* order

Alternatively, you can sort the Brand column in a descending order. To do that, simply add the condition of *ascending=False* in this manner:

df.sort_values(by=['Brand'], inplace=True, ascending=False)

```
# sort - descending order
```

```
from pandas import DataFrame


Cars = {'Brand': ['Honda Civic','Toyota Corolla','Ford Focus','Audi A4'],
        'Price': [22000,25000,27000,35000],
        'Year': [2015,2013,2018,2018]
        }
 df = DataFrame(Cars, columns= ['Brand', 'Price','Year'])
# sort Brand - descending order
df.sort_values(by=['Brand'], inplace=True, ascending=False)
print (df)
```

X----------X----------X