Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации
Ордена Трудового Красного Знамени федеральное государственное
бюджетное образовательное учреждение высшего образования
«Московский технический университет связи и информатики»
Кафедра «МКиИТ»

Практическая работа №1

по дисциплине «Data mining»

Москва 2023

Меры центральной тенденции

In [1]:

```
import pandas as pd

bikes = pd.read_pickle('BikesDataVars.pkl')
bikes.head()
```

Out[1]:

	Date	Hour	Temperature	Humidity	Wind speed	Rainfall	Snowfall	Seasons	Holiday	Functioning Day
0	2017-12-01	0	-5.2	37	2.2	0.0	0.0	Winter	0	True
1	2017-12-01	1	-5.5	38	0.8	0.0	0.0	Winter	0	True
2	2017-12-01	2	-6.0	39	1.0	0.0	0.0	Winter	0	True
3	2017-12-01	3	-6.2	40	0.9	0.0	0.0	Winter	0	True
4	2017-12-01	4	-6.0	36	2.3	0.0	0.0	Winter	0	True

In [2]:

```
bikes['Humidity'].value_counts().head(2)
```

Out[2]:

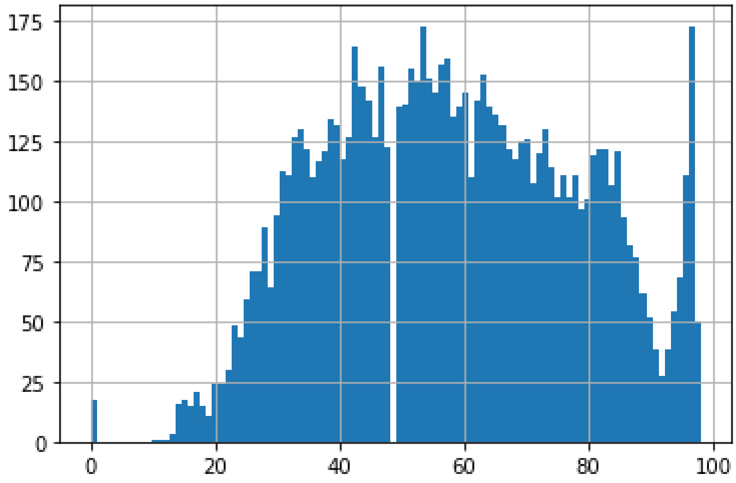
```
53    173
97    173
Name: Humidity, dtype: int64
```

In [3]:

```
bikes['Humidity'].hist(bins=100)
```

Out[3]:

```
<AxesSubplot:>
```



In [4]:

```
bikes.describe()
```

Out[4]:

	Hour	Temperature	Humidity	Wind speed	Rainfall	Snowfall	Holiday
count	8760.000000	8581.000000	8760.000000	8760.000000	8760.000000	8760.000000	8760.000000
mean	11.500000	12.878557	58.226256	1.724909	0.148687	0.075068	0.049315
std	6.922582	11.955551	20.362413	1.036300	1.128193	0.436746	0.216537
min	0.000000	-17.800000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	5.750000	3.400000	42.000000	0.900000	0.000000	0.000000	0.000000
50%	11.500000	13.700000	57.000000	1.500000	0.000000	0.000000	0.000000
75%	17.250000	22.500000	74.000000	2.300000	0.000000	0.000000	0.000000
max	23.000000	39.400000	98.000000	7.400000	35.000000	8.800000	1.000000

In [5]:

```
bikes['Humidity'].mode()
```

Out[5]:

```
0    53
1    97
dtype: int64
```

In [54]:

```
def mode(bikes):

    counts = {}

    for item in (bikes):
        if item in counts:
            counts[item] += 1
        else:
            counts[item] = 1

    return [key for key in counts.keys() if counts[key] == max(counts.values())]
mode((bikes['Humidity']))
```

Out[54]:

```
[53, 97]
```

In [6]:

```
bikes['Humidity'].median()
```

Out[6]:

```
57.0
```

In [43]:

```
index = len(bikes['Humidity']) // 2
if len(bikes['Humidity']) % 2:
    print (sorted(bikes['Humidity'])[index])
else:
    print (sum(sorted(bikes['Humidity'])[index - 1:index + 1]) / 2)
```

```
57.0
```

In [49]:

```
index = len(bikes['Humidity']) // 2
median = sum(sorted(bikes['Humidity'])[index - 1:index + 1]) / 2
if len(bikes['Humidity']) % 2 == 0 : median = sorted(bikes['Humidity'])[index]
print(median)
```

```
57
```

In [7]:

```
bikes['Humidity'].mean()
```

Out[7]:

```
58.226255707762554
```

In [8]:

```
bikes['Humidity'].sum()/len(bikes['Humidity'])
```

Out[8]:

```
58.226255707762554
```