Fizashaikh63 / **FDSL**

Code Issues Pull requests Actions Projects Wiki Security

FDSL / **Assignment2Practice.ipynb**

Fizashaikh63 Add files via upload df70c1e · now

```
446 lines (446 loc) · 9.36 KB
```

In [2]:

```
import pandas as pd
import numpy as np
data=pd.read_csv("student_dataset.csv")
```

In [3]:

```
print(data.head())
```

```
   ID       Name  Age  Marks  Gender
0   1  Student_1   24     94    Male
1   2  Student_2   21     81  Female
2   3  Student_3   22     90    Male
3   4  Student_4   24     56  Female
4   5  Student_5   20     44  Female
```

In [4]:

```
print(data.tail())
```

```
       ID         Name  Age  Marks  Gender
995   996  Student_996   23     82  Female
996   997  Student_997   18     73  Female
997   998  Student_998   24     96  Female
998   999  Student_999   21     65    Male
999  1000 Student_1000   23     80    Male
```

In [5]:

```
print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   ID      1000 non-null   int64 
 1   Name    1000 non-null   object
 2   Age     1000 non-null   int64 
 3   Marks   1000 non-null   int64 
 4   Gender  1000 non-null   object
dtypes: int64(3), object(2)
memory usage: 39.2+ KB
None
```

In [ ]:

```
data['Marks'].mean()
```

Out[ ]:

```
np.float64(70.232)
```

In [7]:

```
data['Marks'].median()
```

Out[7]:

```
np.float64(70.0)
```

In [9]:

```
data['Marks'].max()
```

Out[9]:

```
np.int64(100)
```

In [10]:

```
data['Marks'].min()
```

Out[10]:

```
np.int64(40)
```

In [11]:

```
data['Marks'].sum()
```

Out[11]:

```
np.int64(70232)
```

```
86    26
76    23
79    23
      ..
80    10
66    10
97    10
55     9
87     8
Name: count, Length: 61, dtype: int64
```

In [13]:

```
data.describe()
```

Out[13]:

	ID	Age	Marks
count	1000.000000	1000.0000	1000.000000
mean	500.500000	20.9600	70.232000
std	288.819436	2.0036	17.468638
min	1.000000	18.0000	40.000000
25%	250.750000	19.0000	56.000000
50%	500.500000	21.0000	70.000000
75%	750.250000	23.0000	86.000000
max	1000.000000	24.0000	100.000000

In [14]:

```
data.shape
```

Out[14]:

```
(1000, 5)
```

In [15]:

```
data.dtypes
```

Out[15]:

```
ID         int64
Name      object
Age        int64
Marks      int64
Gender    object
dtype: object
```

In [16]:

```
data.columns
```

Out[16]:

```
Index(['ID', 'Name', 'Age', 'Marks', 'Gender'], dtype='object')
```

```
In [17]: data.index
```

```
Out[17]: RangeIndex(start=0, stop=1000, step=1)
```