In [1]:
```
import pandas as pd
```

In [2]:
```
import numpy as np
```

In [3]:
```
df=pd.read_csv("Iris dataset.csv")
```

In [4]:
```
df
```

Out[4]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

In [5]:
```
df.mean()
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\3698961737.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.mean()
```

Out[5]:
```
sepal_length    5.843333
sepal_width     3.054000
petal_length    3.758667
petal_width     1.198667
dtype: float64
```

In [6]:
```
df.mean(axis=1)[0:3]
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\2281267592.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.mean(axis=1)[0:3]
```

Out[6]:
```
0    2.550
1    2.375
2    2.350
dtype: float64
```

In [7]:
```
df["sepal_length"].mean()
```

Out[7]:
```
5.843333333333335
```

In [8]:
```
df.groupby("species")["sepal_length"].mean()
```

Out[8]:
```
species
Iris-setosa        5.006
Iris-versicolor    5.936
Iris-virginica     6.588
Name: sepal_length, dtype: float64
```

In [9]:
```
df[["sepal_length","sepal_width"]].mean()
```

Out[9]:
```
sepal_length    5.843333
sepal_width     3.054000
dtype: float64
```

In [10]:
```
df.median()
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\530051474.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.median()
```

Out[10]:
```
sepal_length    5.80
sepal_width     3.00
petal_length    4.35
petal_width     1.30
dtype: float64
```

In [11]:
```
df.median(axis=1)[0:2]
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\200861111.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.median(axis=1)[0:2]
```

Out[11]:
```
0    2.45
1    2.20
dtype: float64
```

In [12]:
```
df[["sepal_length","sepal_width"]].median()
```

Out[12]:
```
sepal_length    5.8
sepal_width     3.0
dtype: float64
```

In [13]:
```
df.groupby("species")["sepal_width"].median()
```

Out[13]:
```
species
Iris-setosa        3.4
Iris-versicolor    2.8
Iris-virginica     3.0
Name: sepal_width, dtype: float64
```

In [14]:
```
df.mode()
```

Out[14]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.0	3.0	1.5	0.2	Iris-setosa
1	NaN	NaN	NaN	NaN	Iris-versicolor
2	NaN	NaN	NaN	NaN	Iris-virginica

In [15]:
```
df["petal_length"].mode()
```

Out[15]:
```
0    1.5
Name: petal_length, dtype: float64
```

In [16]:
```
df[["sepal_length","petal_width"]].mode()
```

Out[16]:

	sepal_length	petal_width
0	5.0	0.2

In [17]:
```
df.std()
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\3390915376.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.std()
```

Out[17]:
```
sepal_length    0.828066
sepal_width     0.433594
petal_length    1.764420
petal_width     0.763161
dtype: float64
```

In [18]:
```
df[["petal_length","petal_width"]].std()
```

Out[18]:
```
petal_length    1.764420
petal_width     0.763161
dtype: float64
```

In [19]:
```
df["sepal_width"].std()
```

Out[19]:
```
0.4335943113621737
```

In [20]:
```
df.groupby("species")["petal_length"].std()
```

Out[20]:
```
species
Iris-setosa        0.173511
Iris-versicolor    0.469911
Iris-virginica     0.551895
Name: petal_length, dtype: float64
```

In [21]:
```
df.min()
```

Out[21]:
```
sepal_length            4.3
sepal_width             2.0
petal_length            1.0
petal_width             0.1
species         Iris-setosa
dtype: object
```

In [22]:
```
df["sepal_width"].min()
```

Out[22]:
```
2.0
```

In [23]:
```
df.max()
```

Out[23]:
```
sepal_length              7.9
sepal_width               4.4
petal_length              6.9
petal_width               2.5
species         Iris-virginica
dtype: object
```

In [24]:
```
df["sepal_length"].max()
```

Out[24]:
```
7.9
```

In [25]:
```
df.std(axis=1)[0:2]
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\2883535371.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.std(axis=1)[0:2]
```

Out[25]:
```
0    2.179449
1    2.036950
dtype: float64
```

In [26]:
```
df.max(axis=1)[1:3]
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\2326686098.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.max(axis=1)[1:3]
```

Out[26]:
```
1    4.9
2    4.7
dtype: float64
```

In [27]:
```
df.min(axis=1)[2:4]
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_468\3014756536.py:1: FutureWarnin
g: Dropping of nuisance columns in DataFrame reductions (with 'numeric_only=N
one') is deprecated; in a future version this will raise TypeError.  Select o
nly valid columns before calling the reduction.
  df.min(axis=1)[2:4]
```

Out[27]:
```
2    0.2
3    0.2
dtype: float64
```

In [28]:
```
df_u=df.rename(columns={"petal_length":"petal_len"},inplace=False)
```

In [29]:
```
df_u.groupby("species").petal_len.mean()
```

Out[29]:
```
species
Iris-setosa        1.464
Iris-versicolor    4.260
Iris-virginica     5.552
Name: petal_len, dtype: float64
```

In [30]:
```
df_u.groupby("species")["petal_len"].mean()
```

Out[30]:
```
species
Iris-setosa        1.464
Iris-versicolor    4.260
Iris-virginica     5.552
Name: petal_len, dtype: float64
```

In [31]:
```
df_u
```

Out[31]:

	sepal_length	sepal_width	petal_len	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns