

IRIS FLOWER CLASSIFICATION

Installing packages, Loading and inspecting dataset to have a sneak peek

```
In [1]: import numpy as np

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

In [2]: df=pd.read_csv(r"D:\IRIS_FLOWER_CLASSIFICATION.csv")
df
```

0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica
150 rows x 6 columns						
df . shape						

150 rows × 6 columns

```
In [3]: df.shape

Out[3]: (150, 6)

In [4]: df.head(4)

Out[4]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa

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

Out[5]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

```
In [6]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
#   column      Non-Null Count  Dtype
---  ---
0    Id          150 non-null      int64
1    SepalLengthCm  150 non-null      float64
2    SepalWidthCm   150 non-null      float64
3    PetalLengthCm  150 non-null      float64
4    PetalWidthCm   150 non-null      float64
5    Species        150 non-null      object
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB

In [7]: df.isnull().sum()

Id          0
SepalLengthCm  0
SepalWidthCm  0
PetalLengthCm  0
PetalWidthCm  0
Species       0
dtype: int64

In [8]: df.describe
```

<bound method NDFrame.describe of						Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	\
0	1	5.1	3.5	1.4	0.2						
1	2	4.9	3.0	1.4	0.2						
2	3	4.7	3.2	1.3	0.2						
3	4	4.6	3.1	1.5	0.2						
4	5	5.0	3.6	1.4	0.2						
..						
145	146	6.7	3.0	5.2	2.3						
146	147	6.3	2.5	5.0	1.9						
147	148	6.5	3.0	5.2	2.0						
148	149	6.2	3.4	5.4	2.3						
149	150	5.9	3.0	5.1	1.8						
Species											
0	Iris-setosa										
1	Iris-setosa										
2	Iris-setosa										
3	Iris-setosa										
4	Iris-setosa										
..	...										
145	Iris-virginica										
146	Iris-virginica										
147	Iris-virginica										
148	Iris-virginica										
149	Iris-virginica										

[150 rows x 6 columns]>

```
In [9]: df.describe()

Out[9]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

```
In [10]: df.columns

Out[10]: Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
              'Species'],
              dtype='object')

In [11]: list(df)

Out[11]: ['Id',
'SepalLengthCm',
'SepalWidthCm',
'PetalLengthCm',
'PetalWidthCm',
'Species']

In [12]: df.Species.value_counts()

Out[12]: Iris-setosa      50
Iris-versicolor      50
Iris-virginica       50
Name: Species, dtype: int64

In [13]: df.drop('Id',axis=1,inplace=True)

In [14]: df.head(6)

Out[14]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	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
5	5.4	3.9	1.7	0.4	Iris-setosa

```
In [15]: column = df.select_dtypes(include=['number'])

data = column.corr()
data

Out[15]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
SepalLengthCm	1.000000	-0.109369	0.871754	0.817954
SepalWidthCm	-0.109369	1.000000	-0.420516	-0.356544
PetalLengthCm	0.871754	-0.420516	1.000000	0.962757
PetalWidthCm	0.817954	-0.356544	0.962757	1.000000



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

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  ---
0    SepalLengthCm  150 non-null      float64
1    SepalWidthCm   150 non-null      float64
2    PetalLengthCm  150 non-null      float64
3    PetalWidthCm   150 non-null      float64
4    Species        150 non-null      object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB

In [18]: df.shape[1]

Out[18]: 5

In [19]: df.shape[0]

Out[19]: 150

In [20]: df.describe().T

Out[20]:
```

	count	mean	std	min	25%	50%	75%	max
SepalLengthCm	150.0	5.843333	0.828066	4.3	5.1	5.80	6.4	7.9
SepalWidthCm	150.0	3.054000	0.433594	2.0	2.8	3.00	3.3	4.4
PetalLengthCm	150.0	3.758667	1.764420	1.0	1.6	4.35	5.1	6.9
PetalWidthCm	150.0	1.198667	0.763161	0.1	0.3	1.30	1.8	2.5

```
In [21]: df['Species'].value_counts()

Iris-setosa      50
Iris-versicolor      50
Iris-virginica       50
Name: Species, dtype: int64

In [22]: df.nunique()

Out[22]: SepalLengthCm      35
SepalWidthCm      23
PetalLengthCm      43
PetalWidthCm      22
Species           3
dtype: int64
```

Check missing values

```
In [23]: dataset = df.isnull().sum().sort_values(ascending = False)
dataset

Out[23]: SepalLengthCm      0
SepalWidthCm      0
PetalLengthCm      0
PetalWidthCm      0
Species           0
dtype: int64
```

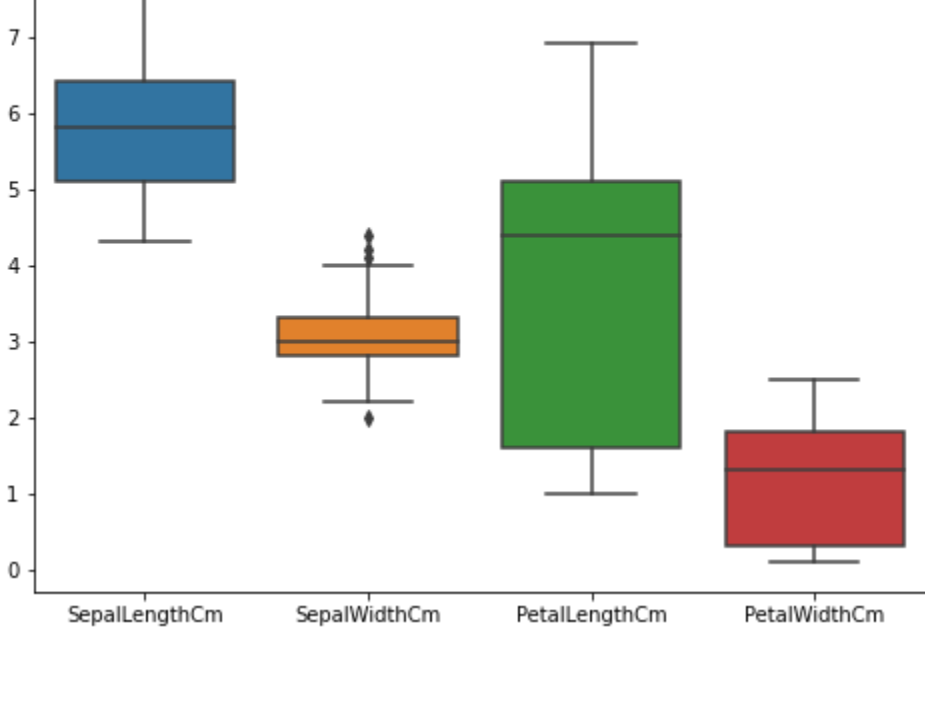
Deal with duplicates values

```
In [24]: df.duplicated().sum()

Out[24]: 3

In [25]: df.drop_duplicates(inplace=True)

In [26]: plt.figure(figsize=(8, 6))
sns.boxplot(data=df)
plt.title("Data with Outliers")
plt.show()
```



Visulization

```
In [27]: sns.pairplot(df,hue = 'Species' , palette = 'Dark2', diag_kind = 'kde')
plt.show()

In [30]: #sns.barplot(df[['SepalWidthCm', 'SepalLengthCm']], fill=True)
#plt.show()

In [32]:

In [ ]:

In [ ]:

In [ ]:
```