Name :sandip gadadare. Div:A
Roll no: CO3A09. BATCH:B1

Assignment No : 01

Perform the following operations using Python on any open source dataset (e.g., data.csv)
1. Import all the required Python Libraries.
2. Locate an open source data from the web (e.g., https://www.kaggle.com). Provide a clear description of the data and its source (i.e., URL of the web site).
3. Load the Dataset into pandas dataframe.
4. Data Preprocessing: check for missing values in the data using pandas isnull(), describe() function to get some initial statistics. Provide variable descriptions. Types of variables etc. Check the dimensions of the data frame.
5. Data Formatting and Data Normalization: Summarize the types of variables by checking the data types (i.e., character, numeric, integer, factor, and logical) of the variables in the data set. If variables are not in the correct data type, apply proper type conversions.
6. Turn categorical variables into quantitative variables in Python.

Code :

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

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

In [3]:
```
df=pd.read_csv('iris.csv')
```

In [4]:
```
df.describe()
```
Out[4]:

	150	4	setosa	versicolor	virginica
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333	1.000000
std	0.828066	0.435866	1.765298	0.762238	0.819232
min	4.300000	2.000000	1.000000	0.100000	0.000000
25%	5.100000	2.800000	1.600000	0.300000	0.000000
50%	5.800000	3.000000	4.350000	1.300000	1.000000
75%	6.400000	3.300000	5.100000	1.800000	2.000000
max	7.900000	4.400000	6.900000	2.500000	2.000000

In [5]:
df.isnull()
Out[5]:

	150	4	setosa	versicolor	virginica
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
145	False	False	False	False	False
146	False	False	False	False	False
147	False	False	False	False	False
148	False	False	False	False	False
149	False	False	False	False	False

150 rows × 5 columns

In [6]:
df.notnull()
Out[6]:

	150	4	setosa	versicolor	virginica
0	True	True	True	True	True
1	True	True	True	True	True
2	True	True	True	True	True
3	True	True	True	True	True
4	True	True	True	True	True
...	...	...	...	...	...
145	True	True	True	True	True
146	True	True	True	True	True
147	True	True	True	True	True
148	True	True	True	True	True
149	True	True	True	True	True

150 rows × 5 columns
In [7]:
df.dtypes
Out[7]:
150 float64
4 float64
setosa float64
versicolor float64
virginica int64
dtype: object

In [8]:
```
df.isnull().sum()
```
Out[8]:
```
150          0
4            0
setosa       0
versicolor   0
virginica    0
dtype: int64
```

In [9]:
```
df.size
```
Out[9]:
```
750
```

In [10]:
```
df.ndim
```
Out[10]:
```
2
```

In [11]:
```
df.shape
```
Out[11]:
```
(150, 5)
```

In [12]:
```
df.info
```
Out[12]:
```
<bound method DataFrame.info of      150    4  setosa  versicolor  virginica
0    5.1  3.5     1.4         0.2          0
1    4.9  3.0     1.4         0.2          0
2    4.7  3.2     1.3         0.2          0
3    4.6  3.1     1.5         0.2          0
4    5.0  3.6     1.4         0.2          0
..   ...  ...     ...         ...        ...
145  6.7  3.0     5.2         2.3          2
146  6.3  2.5     5.0         1.9          2
147  6.5  3.0     5.2         2.0          2
148  6.2  3.4     5.4         2.3          2
149  5.9  3.0     5.1         1.8          2

[150 rows x 5 columns]>
```

In [13]:
```
categorical_columns=df.select_dtypes(include=['object','category']).columns
```

In [14]:
```
print(categorical_columns)
```
```
Index([], dtype='object')
```

In [15]:
```
label_encoded_df=df.copy()
```

In [18]:
```
for col in categorical_columns:
label_encoded_df[col]=label_encoded_df[col].astype('category').cat.codes
```

In [19]:
```
print(label_encoded_df.head())
```
```
   150    4  setosa  versicolor  virginica
0  5.1  3.5     1.4         0.2          0
1  4.9  3.0     1.4         0.2          0
2  4.7  3.2     1.3         0.2          0
3  4.6  3.1     1.5         0.2          0
4  5.0  3.6     1.4         0.2          0
```

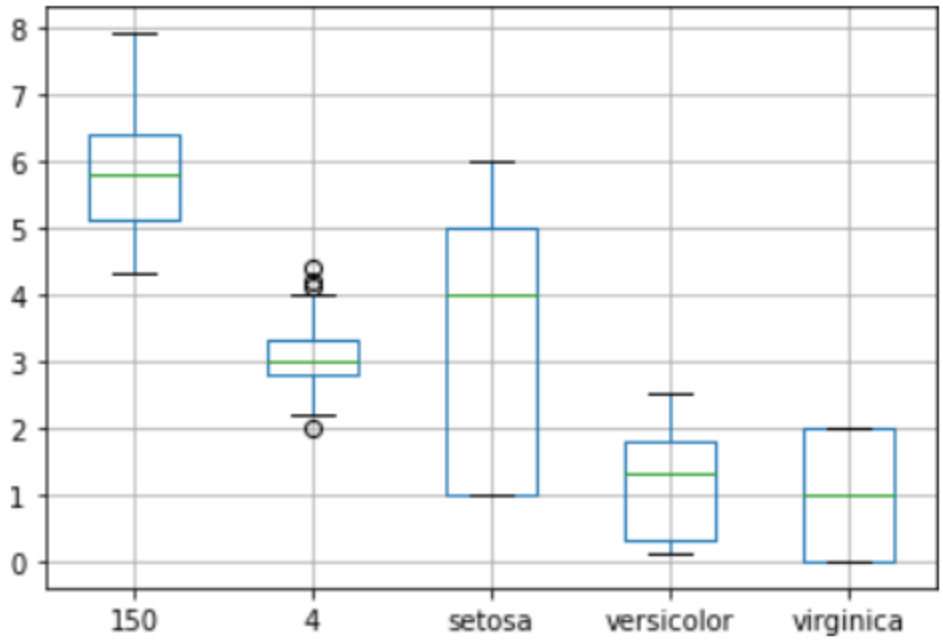
In [21]:
```
df['setosa']=df['setosa'].astype(int)
```

In [22]:
```
df.dtypes
```

Out[22]:
```
150           float64
4             float64
setosa          int32
versicolor    float64
virginica       int64
dtype: object
```

In [23]:
```
df.boxplot()
```
Out[23]:
<AxesSubplot:>

In [ ]:


8
7
6
5
4
3
2
1
0
150
4
setosa
versicolor
virginica