Assignment 2

Data Wrangling II

Create an “Academic performance” dataset of students and perform the following operations using Python.

1. Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them.
2. Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them.
3. Apply data transformations on at least one of the variables. The purpose of this transformation should be one of the following reasons: to change the scale for better understanding of the variable, to convert a non-linear relation into a linear one, or to decrease the skewness and convert the distribution into a normal distribution.

Importing the required libraries

In [3]:
```
import pandas as pd
import numpy as np
```

In [4]:
```
StudDetails = {'Roll_no':['1','2','3','4','5','6','7','8','9','10'],
               'Name':['Mayur','Mrudul','Pranav','Ketan','Chetan','Sahil','Nik
               'DSBDA':[98,95,97,np.NaN,87,np.NaN,90,8,93,97],
                 'CGPA':[7.03,9.16,np.NaN,8.80,9.09,9.18,9.39,8.90,8.93,9.95],
                  'SGPA':[np.NaN,9.35,np.NaN,np.NaN,np.NaN,np.NaN,9.78,np.NaN,
```

In [5]:
```
df = pd.DataFrame(StudDetails)
```

In [6]:
```
print(df)
```

```
  Roll_no    Name  DSBDA  CGPA  SGPA
0       1   Mayur   98.0  7.03   NaN
1       2  Mrudul   95.0  9.16  9.35
2       3  Pranav   97.0   NaN   NaN
3       4   Ketan    NaN  8.80   NaN
4       5  Chetan   87.0  9.09   NaN
5       6   Sahil    NaN  9.18   NaN
6       7  Nikhil   90.0  9.39  9.78
7       8  Jeevan    8.0  8.90   NaN
8       9    Anuj   93.0  8.93   NaN
9      10   Vicky   97.0  9.95   NaN
```

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10 entries, 0 to 9
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype  
---  ------   --------------  -----  
 0   Roll_no  10 non-null     object 
 1   Name     10 non-null     object 
 2   DSBDA    8 non-null      float64
 3   CGPA     9 non-null      float64
 4   SGPA     2 non-null      float64
dtypes: float64(3), object(2)
memory usage: 528.0+ bytes
```

Inconsistence in datatype

Checking for missing value

In [8]:
```
df.isnull()
```

Out[8]:

	Roll_no	Name	DSBDA	CGPA	SGPA
0	False	False	False	False	True
1	False	False	False	False	False
2	False	False	False	True	True
3	False	False	True	False	True
4	False	False	False	False	True
5	False	False	True	False	True
6	False	False	False	False	False
7	False	False	False	False	True
8	False	False	False	False	True
9	False	False	False	False	True

Yes we have missing values in Coloumn DSBDA CGPA and SGPA

Dealing with Missing Values

In [9]:
```
print("The total missing values in DSBDA column are: ",df['DSBDA'].isnull().sum
```

```
The total missing values in DSBDA column are:  2
```

In [10]:
```
mean_DSBDA = df['DSBDA'].mean()
```

In [11]:
```
print("The mean of DSBDA column is: ",mean_DSBDA)
```

```
The mean of DSBDA column is:  83.125
```

Replace missing values with mean

In [12]:
```
df['DSBDA'].fillna(mean_DSBDA,inplace = True)
```

In [13]:
```
print(df)
```

```
  Roll_no    Name   DSBDA  CGPA  SGPA
0       1   Mayur  98.000  7.03   NaN
1       2  Mrudul  95.000  9.16  9.35
2       3  Pranav  97.000   NaN   NaN
3       4   Ketan  83.125  8.80   NaN
4       5  Chetan  87.000  9.09   NaN
5       6   Sahil  83.125  9.18   NaN
6       7  Nikhil  90.000  9.39  9.78
7       8  Jeevan   8.000  8.90   NaN
8       9    Anuj  93.000  8.93   NaN
9      10   Vicky  97.000  9.95   NaN
```

In [14]:
```
print("The total missing values in CGPA column are: ",df['CGPA'].isnull().sum()
```

```
The total missing values in CGPA column are:  1
```

In [15]:
```
mean_CGPA = df['CGPA'].mean()
```

In [16]:
```
print("The mean of CGPA column is: ",mean_CGPA)
```

```
The mean of CGPA column is:  8.936666666666666
```

In [17]:
```
df['CGPA'].fillna(mean_CGPA,inplace = True)
```

In [18]:
```
df
```

Out[18]:

	Roll_no	Name	DSBDA	CGPA	SGPA
0	1	Mayur	98.000	7.030000	NaN
1	2	Mrudul	95.000	9.160000	9.35
2	3	Pranav	97.000	8.936667	NaN
3	4	Ketan	83.125	8.800000	NaN
4	5	Chetan	87.000	9.090000	NaN
5	6	Sahil	83.125	9.180000	NaN
6	7	Nikhil	90.000	9.390000	9.78
7	8	Jeevan	8.000	8.900000	NaN
8	9	Anuj	93.000	8.930000	NaN
9	10	Vicky	97.000	9.950000	NaN

In [19]:
```
print("The total missing values in SGPA column are: ",df['SGPA'].isnull().sum()
```

```
The total missing values in SGPA column are:  8
```

As SGPA column as more number of missing values we can drop that column

In [25]:
```
df.drop(['SGPA'],axis = 1)
```

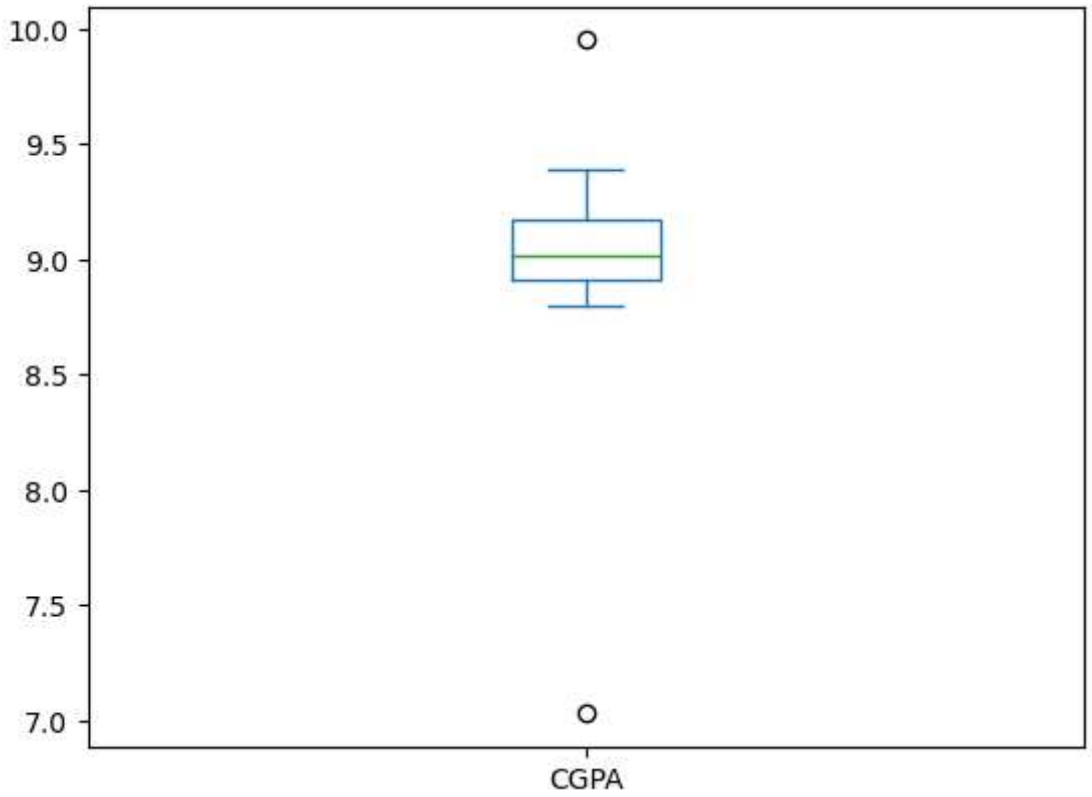
Out[25]:

	Roll_no	Name	DSBDA	CGPA
0	1	Mayur	103.000	7.030000
1	2	Mrudul	100.000	9.160000
2	3	Pranav	102.000	8.936667
3	4	Ketan	88.125	8.800000
4	5	Chetan	92.000	9.090000
5	6	Sahil	88.125	9.180000
6	7	Nikhil	95.000	9.390000
7	8	Jeevan	13.000	8.900000
8	9	Anuj	98.000	8.930000
9	10	Vicky	102.000	9.950000

Finding outliers

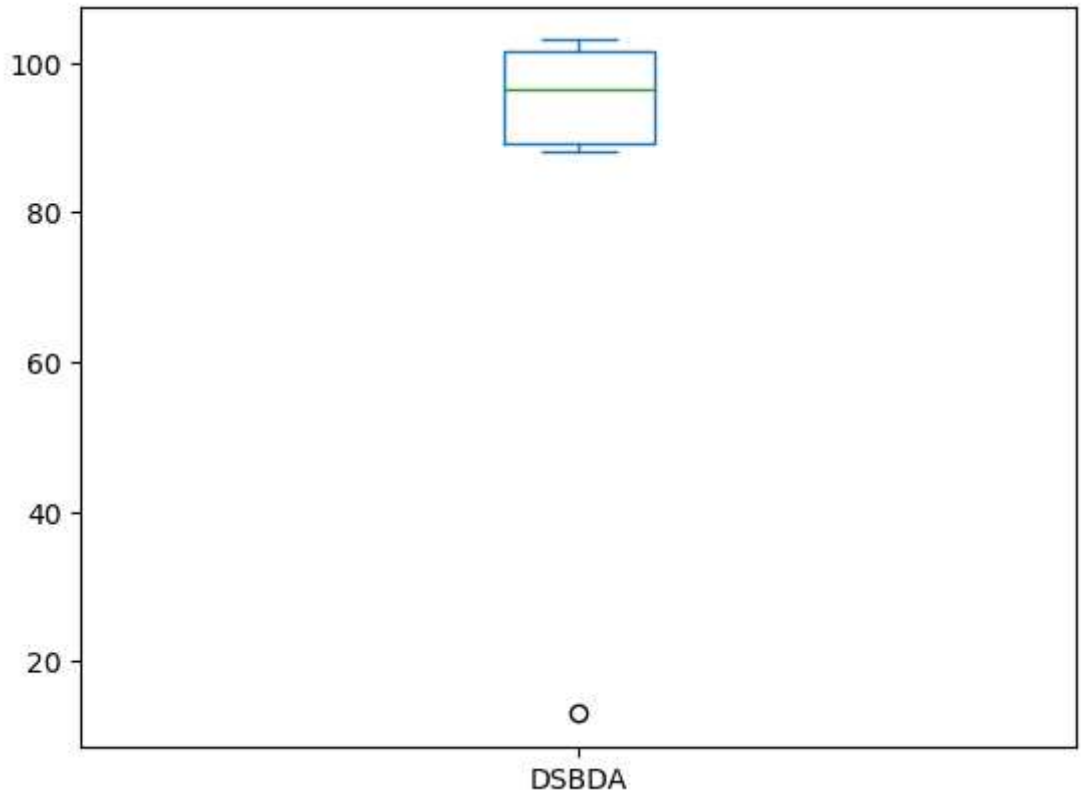
In [26]:
```
df['CGPA'].plot(kind='box')
```

Out[26]: <AxesSubplot:>



In [27]:
```
df['DSBDA'].plot(kind='box')
```

Out[27]: <AxesSubplot:>



In [29]:
```
df['DSBDA']=df['DSBDA'].transform(lambda x:x+5)
df
```

Out[29]:

	Roll_no	Name	DSBDA	CGPA	SGPA
0	1	Mayur	113.000	7.030000	NaN
1	2	Mrudul	110.000	9.160000	9.35
2	3	Pranav	112.000	8.936667	NaN
3	4	Ketan	98.125	8.800000	NaN
4	5	Chetan	102.000	9.090000	NaN
5	6	Sahil	98.125	9.180000	NaN
6	7	Nikhil	105.000	9.390000	9.78
7	8	Jeevan	23.000	8.900000	NaN
8	9	Anuj	108.000	8.930000	NaN
9	10	Vicky	112.000	9.950000	NaN

In [ ]: