**Assignment -2**

Python Programming

|  |  |
| --- | --- |
| Assignment Date | 19 September 2022 |
| Student Name | KRISHNAPRIYA D |
| Student Roll Number | 2019115048 |
| Maximum Marks | 1. Marks |

## import required libraries

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

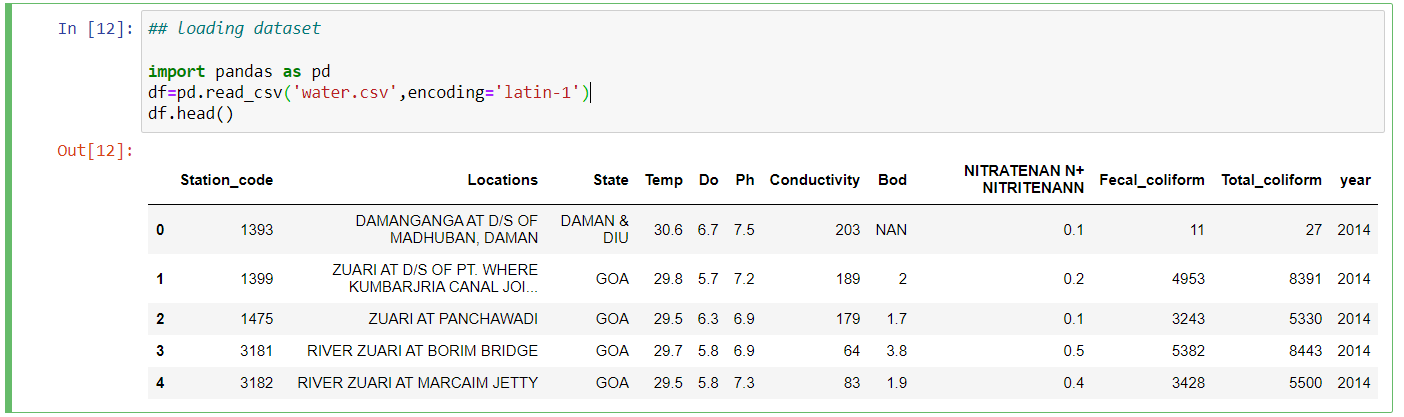
from matplotlib import rcParams

## loading dataset

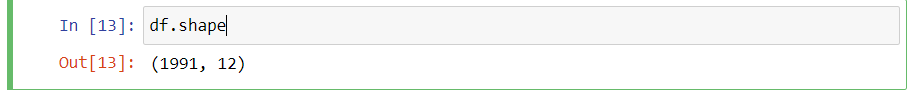
import pandas as pd

df=pd.read\_csv('water.csv',encoding='latin-1')

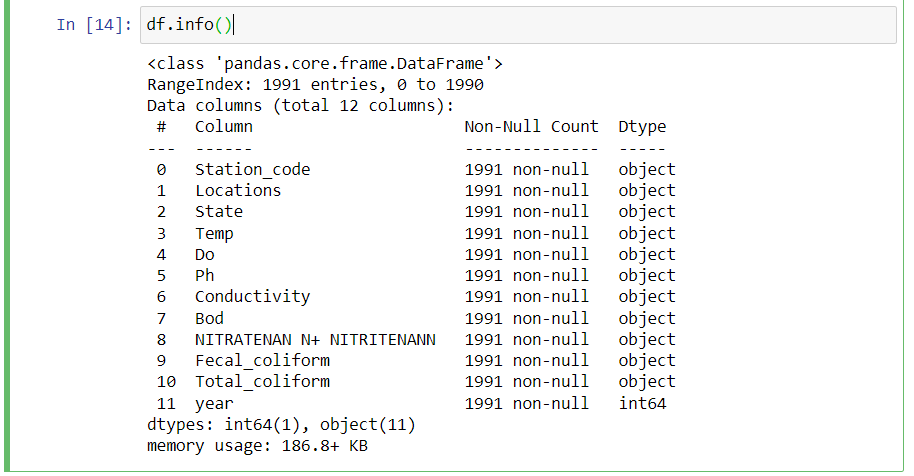
df.head()



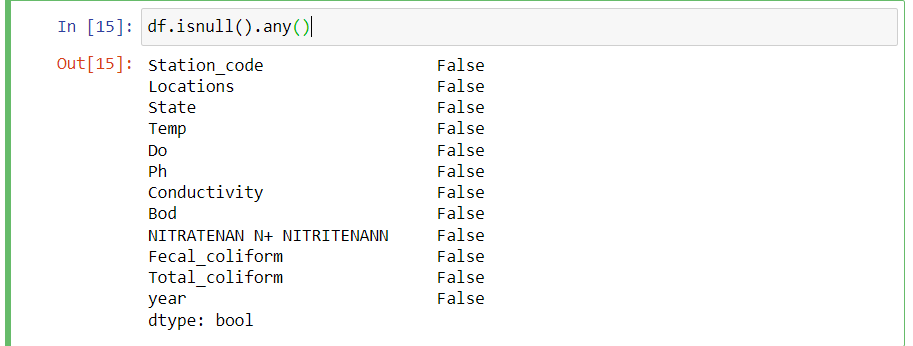
df.shape



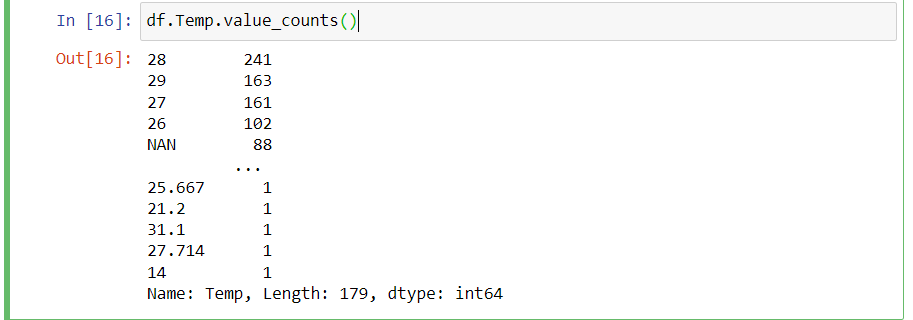
df.info()



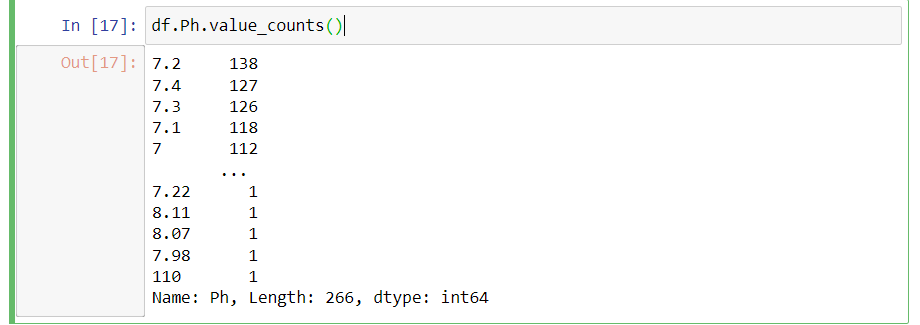
df.isnull().any()



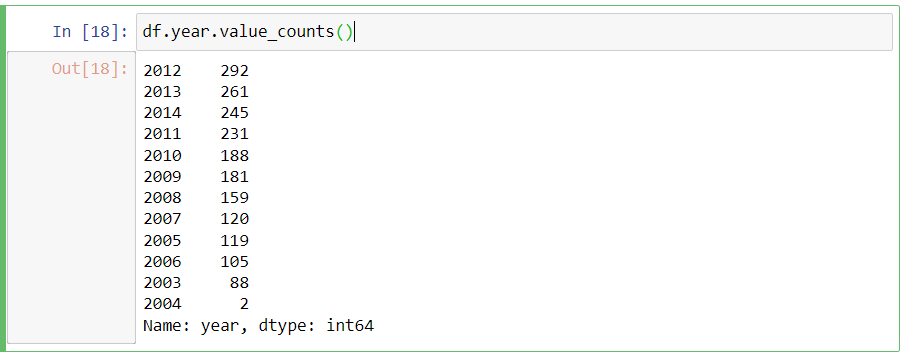
df.Temp.value\_counts()



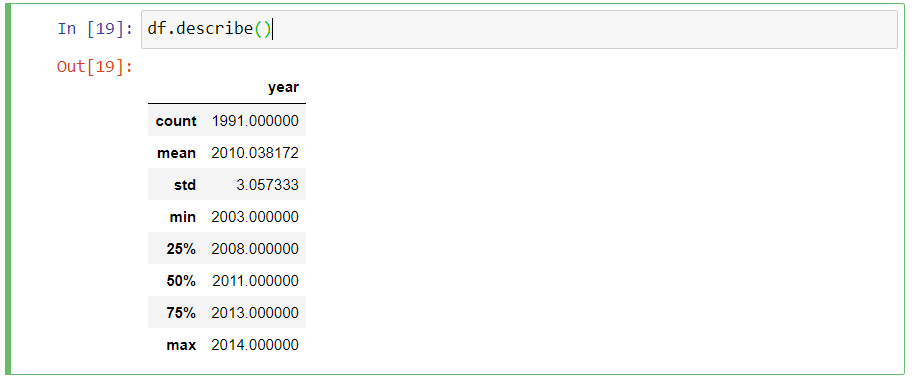
df.Ph.value\_counts()



df.year.value\_counts()

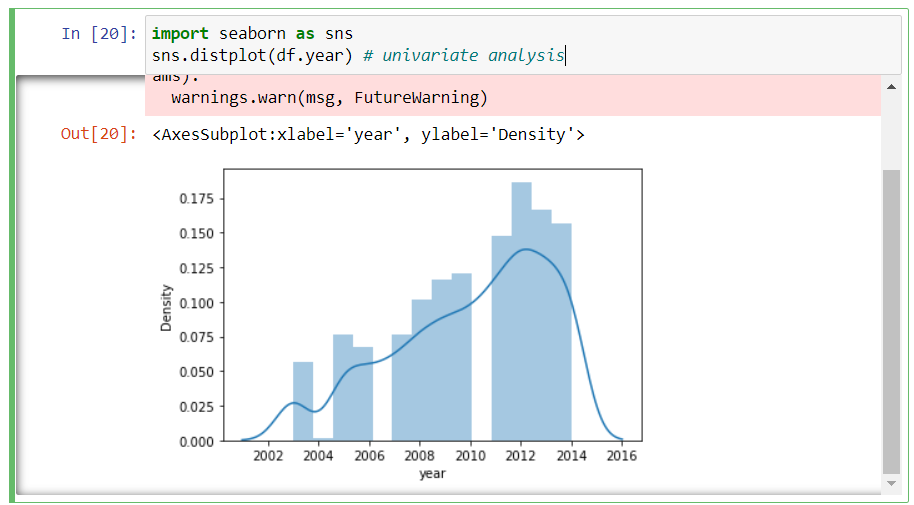


df.describe()

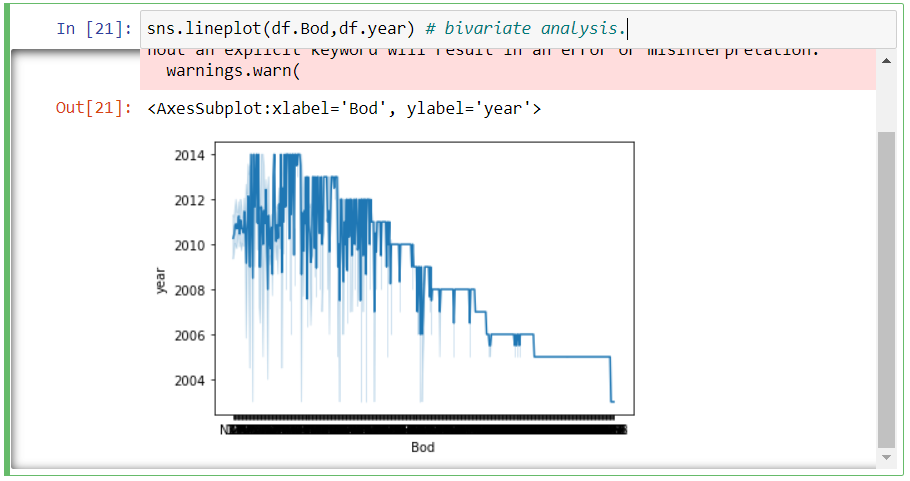


import seaborn as sns

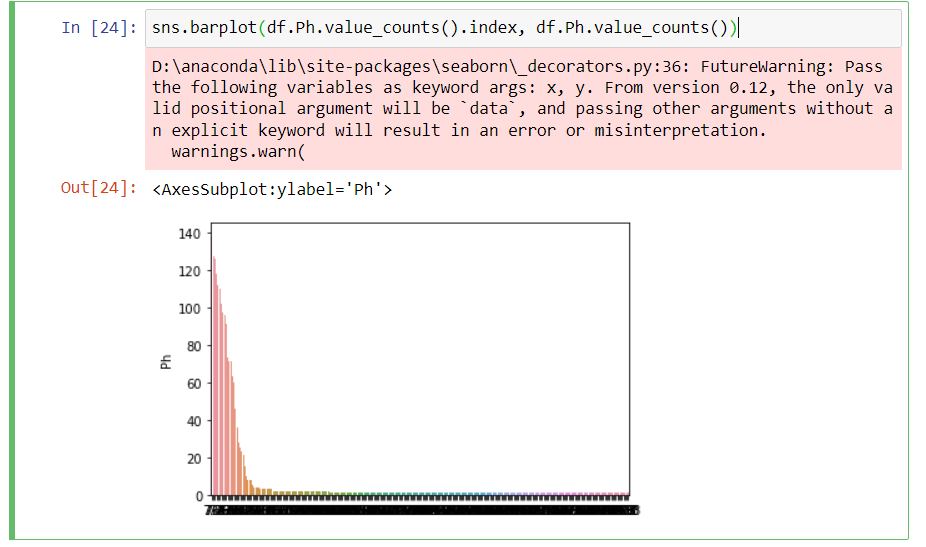
sns.distplot(df.year) # univariate analysis



sns.lineplot(df.Bod,df.year) # bivariate analysis.



sns.barplot(df.Ph.value\_counts().index, df.Ph.value\_counts())



## countplot

import seaborn as sns

import matplotlib.pyplot as plt

def countplot\_of\_2(x,hue,title=None,figsize=(6,5)):

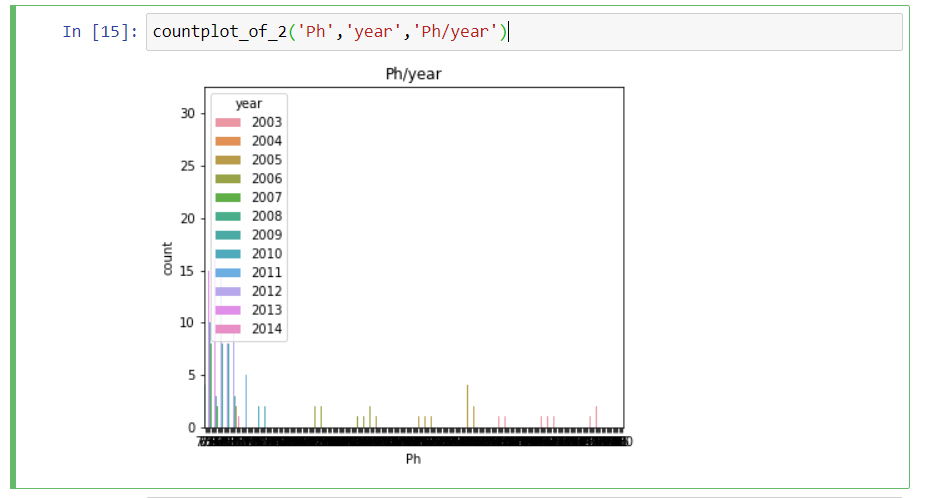
plt.figure(figsize=figsize)

sns.countplot(data=df[[x,hue]],x=x,hue=hue)

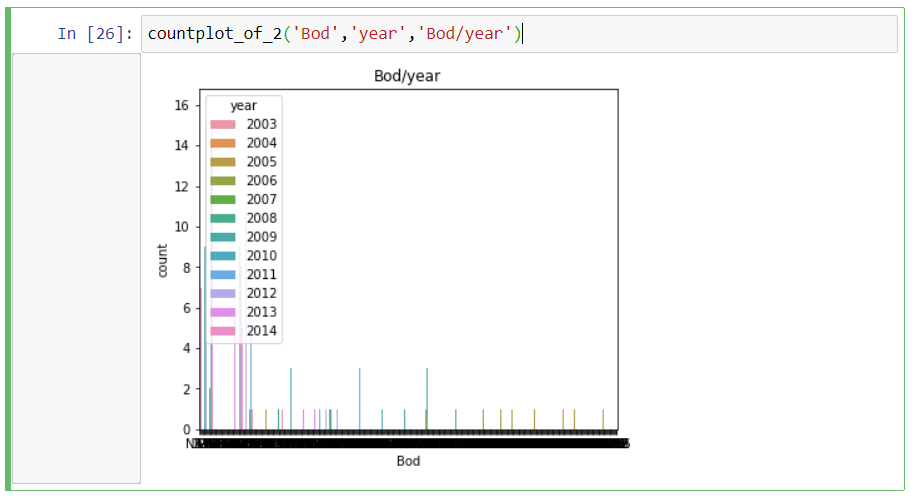
plt.title(title)

plt.show()

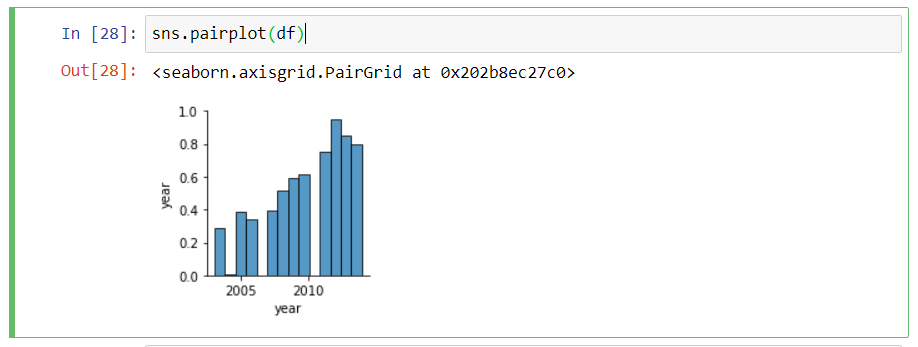
countplot\_of\_2('Ph','year','Ph/year')



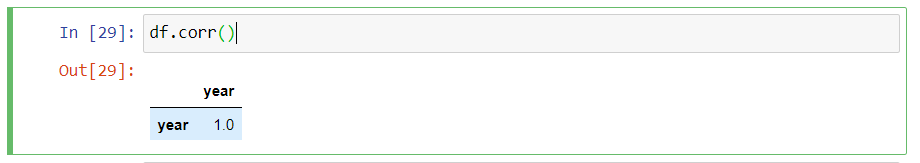
countplot\_of\_2('Bod','year','Bod/year')



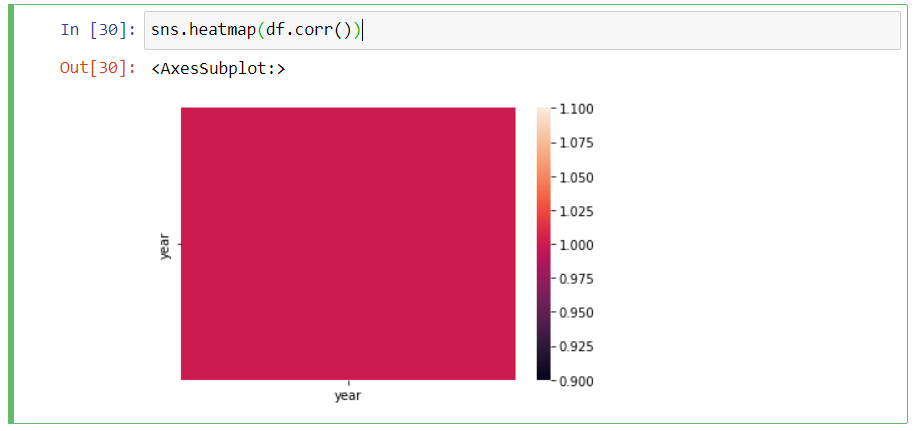
sns.pairplot(df)



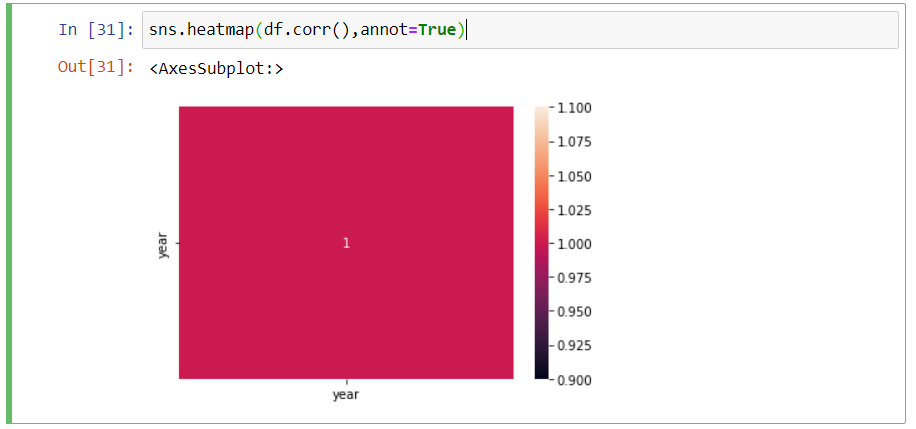
df.corr()



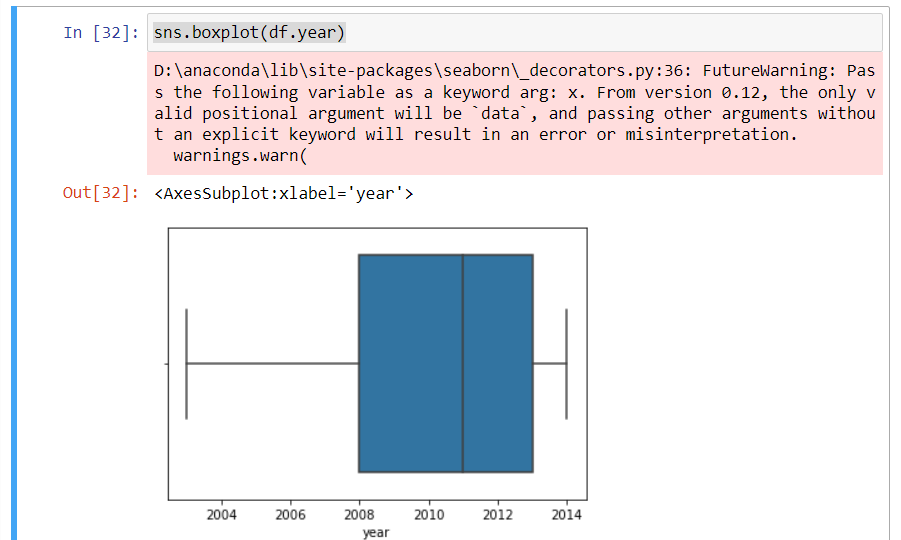
sns.heatmap(df.corr())



sns.heatmap(df.corr(),annot=True)



sns.boxplot(df.year)



**outlier removal- IQR method**

Q1= df.year.quantile(0.25)

Q3=df.year.quantile(0.75)

IQR=Q3-Q1

upper\_limit =Q3 + 1.5\*IQR

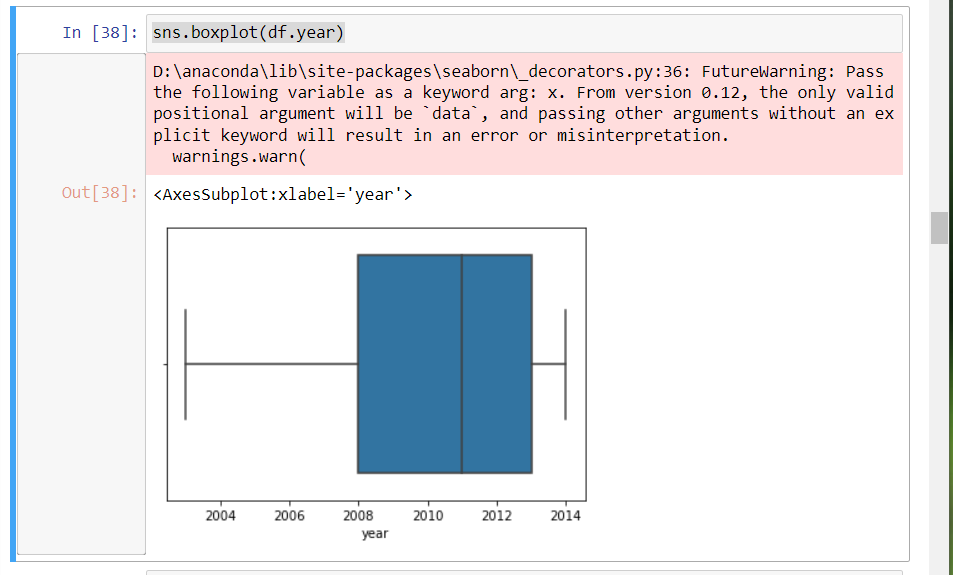
lower\_limit =Q1 - 1.5\*IQR

upper\_limit



df=df[df.year<upper\_limit]

sns.boxplot(df.year)

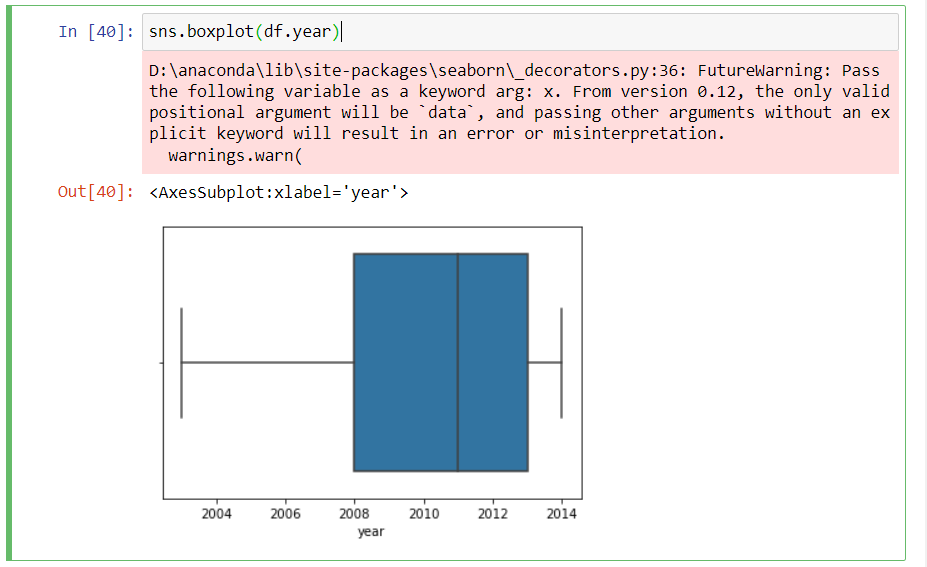


df.shape

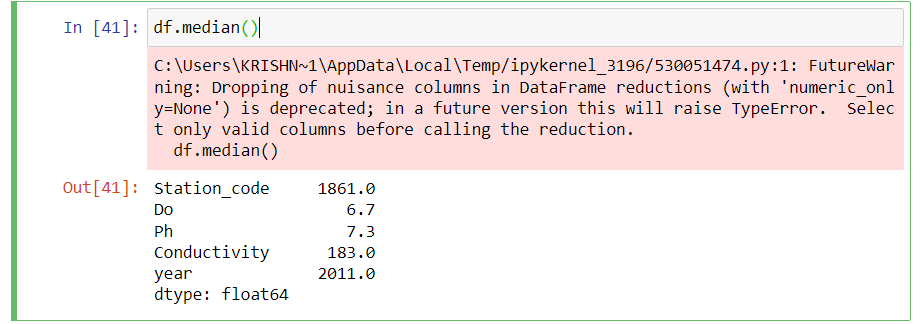


# Outlier replacement using median

sns.boxplot(df.year)



df.median()



Q1= df.year.quantile(0.25)

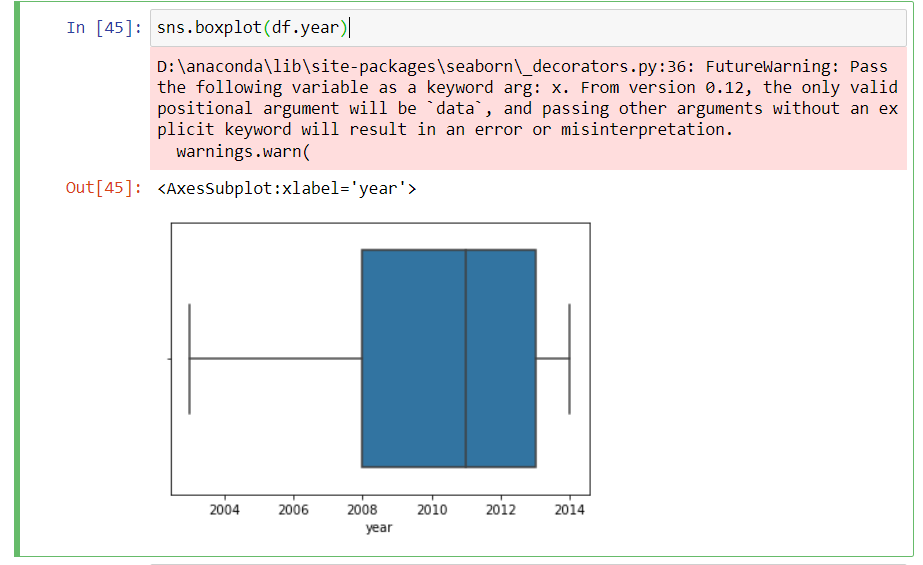
Q3=df.year.quantile(0.75)

IQR=Q3-Q1

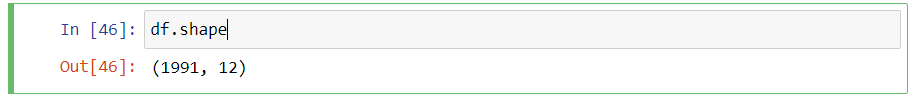
upper\_limit =Q3 + 1.5\*IQR

lower\_limit =Q1 - 1.5\*IQR

sns.boxplot(df.year)

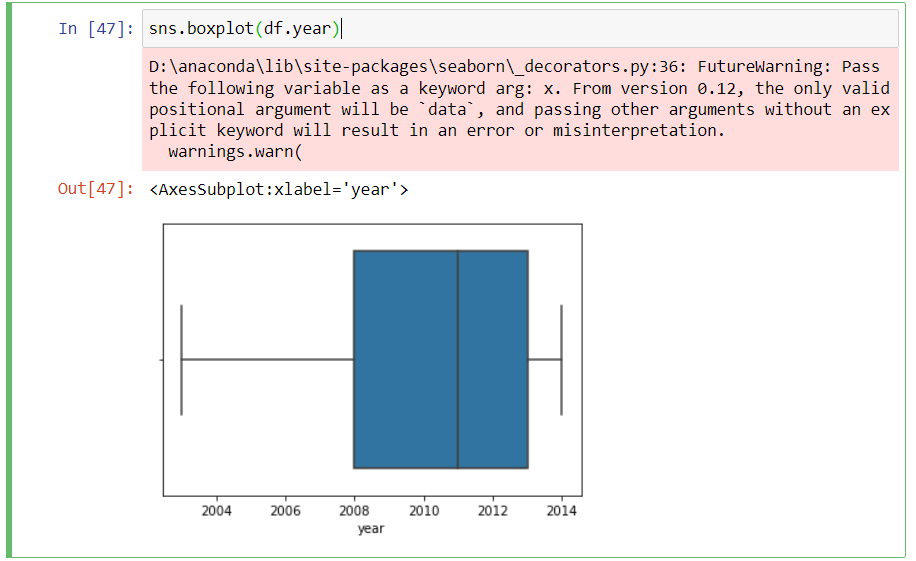


df.shape



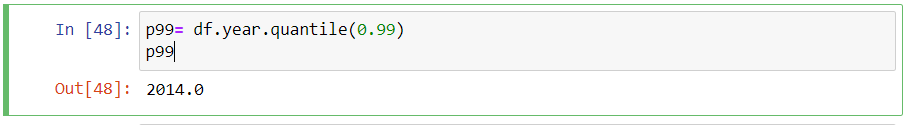
## outlier removal- Percentile method

sns.boxplot(df.year)



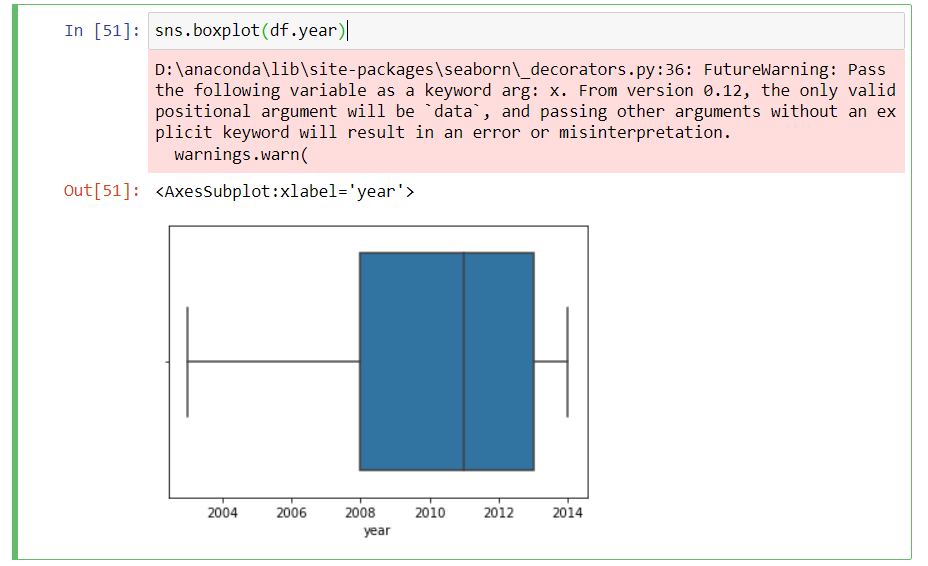
p99= df.year.quantile(0.99)

p99



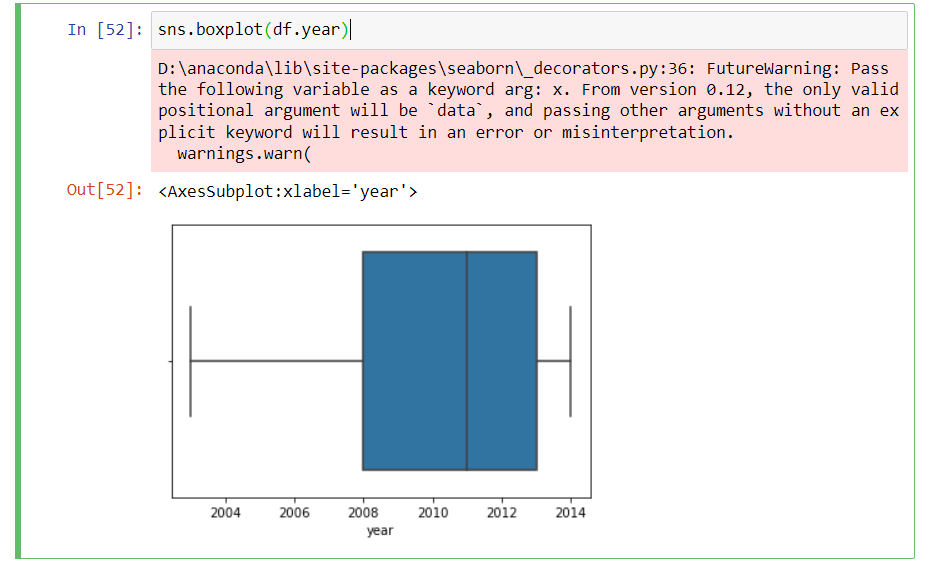
df = df[df.year<=p99]

sns.boxplot(df.year)



## outlier removal- z-score

sns.boxplot(df.year)

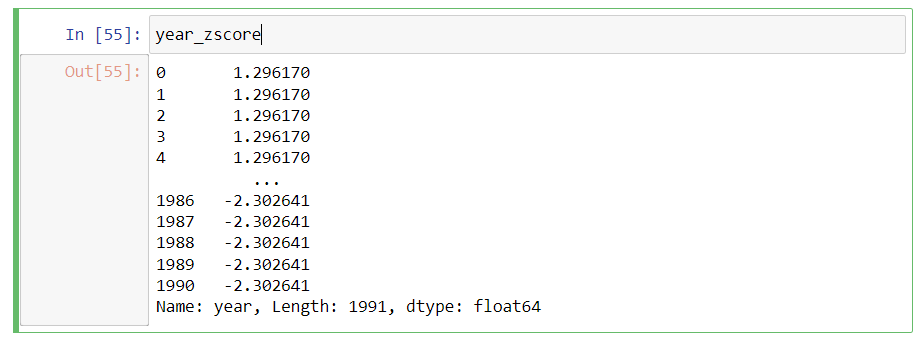


from scipy import stats

from scipy import stats

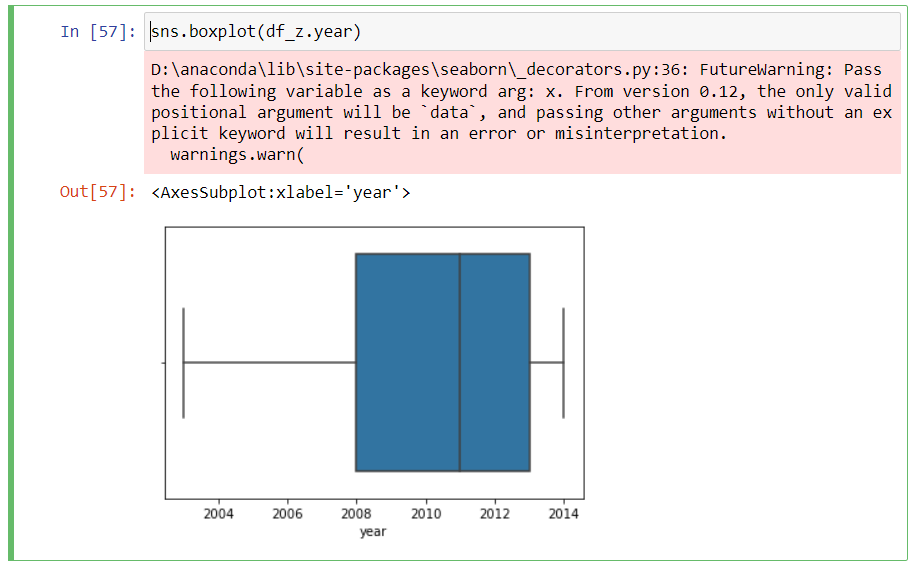
year\_zscore = stats.zscore(df.year)

year\_zscore

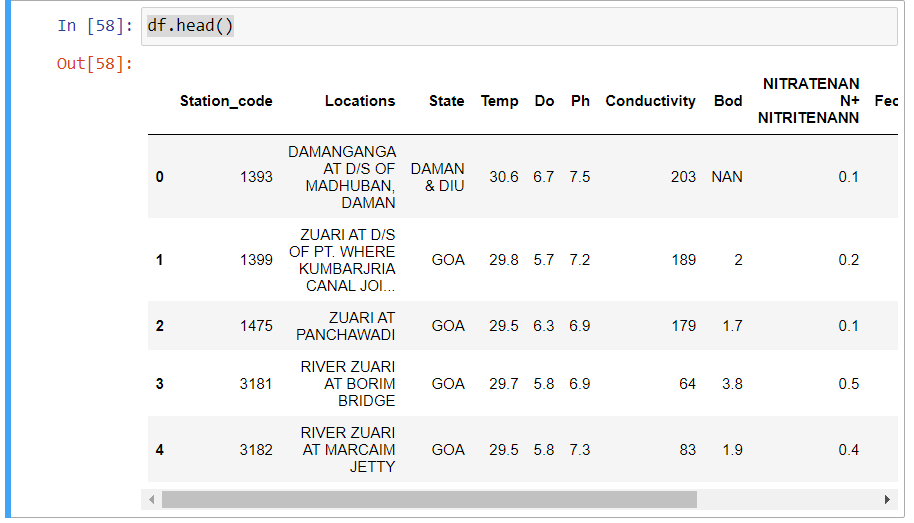


df\_z = df[np.abs(year\_zscore)<=3]

sns.boxplot(df\_z.year)



df.head()



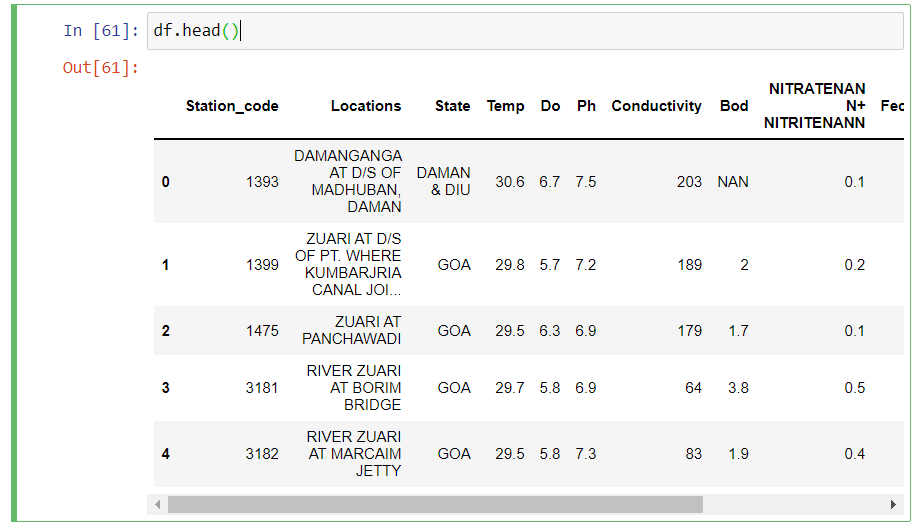
## Encoding Techniques

## 1.Label Encoding[¶](http://localhost:8888/notebooks/Documents/Data%20preprocessing.ipynb#1.Label-Encoding)

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

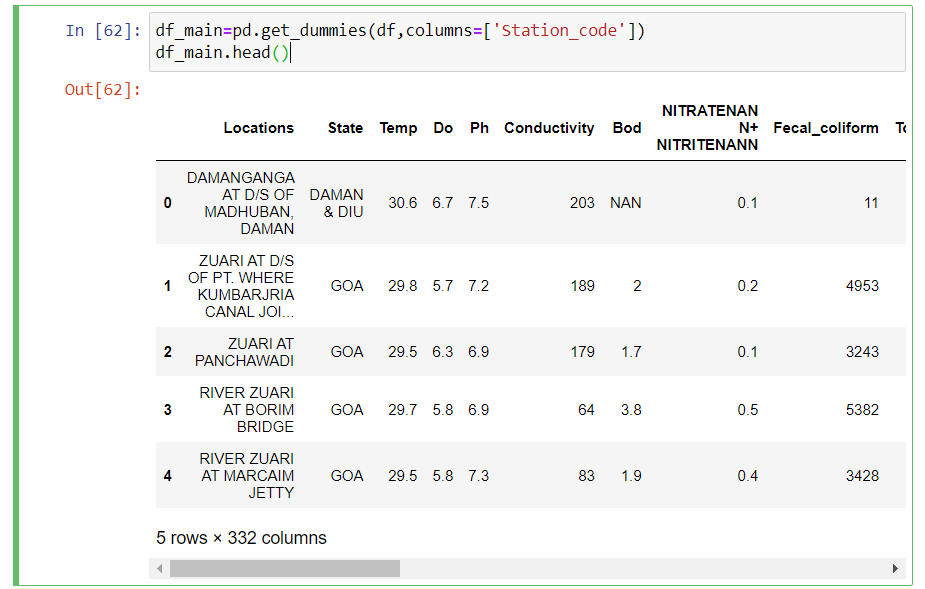
df.head()



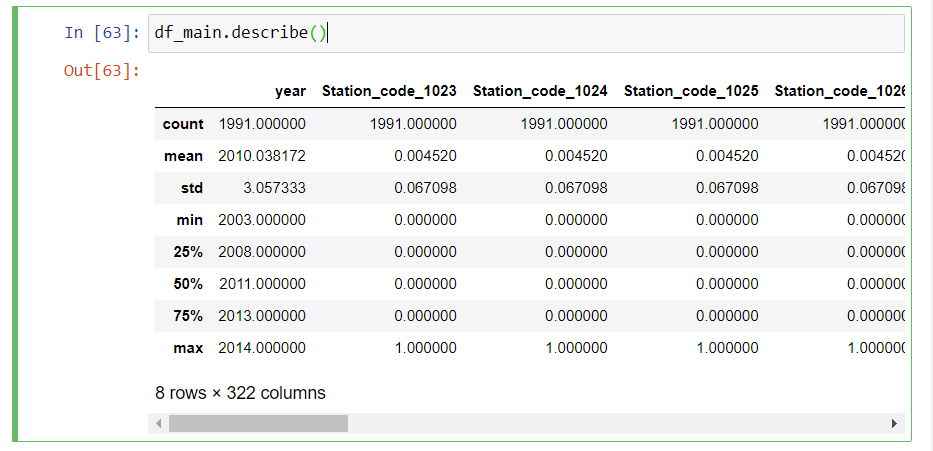
## 2.One hot Encoding[¶](http://localhost:8888/notebooks/Documents/Data%20preprocessing.ipynb#2.One-hot-Encoding)

df\_main=pd.get\_dummies(df,columns=['Station\_code'])

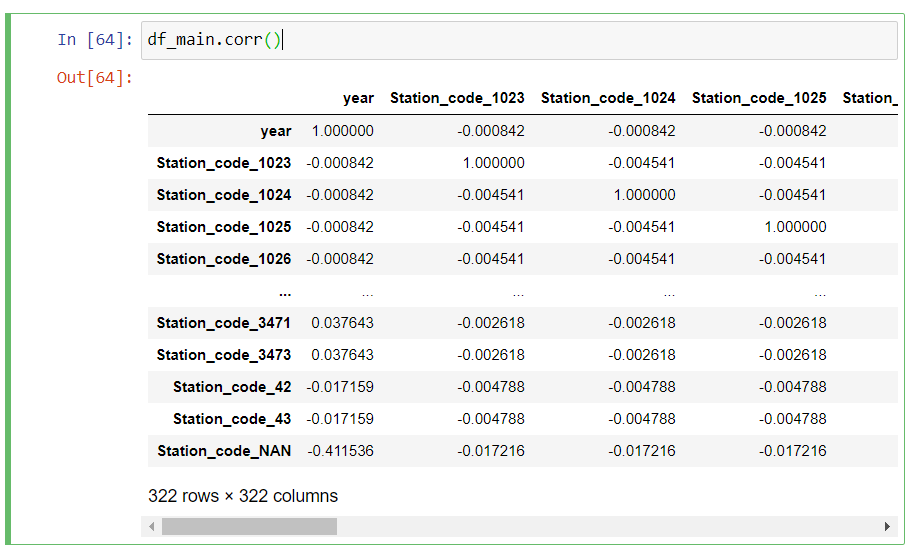
df\_main.head()



df\_main.describe()

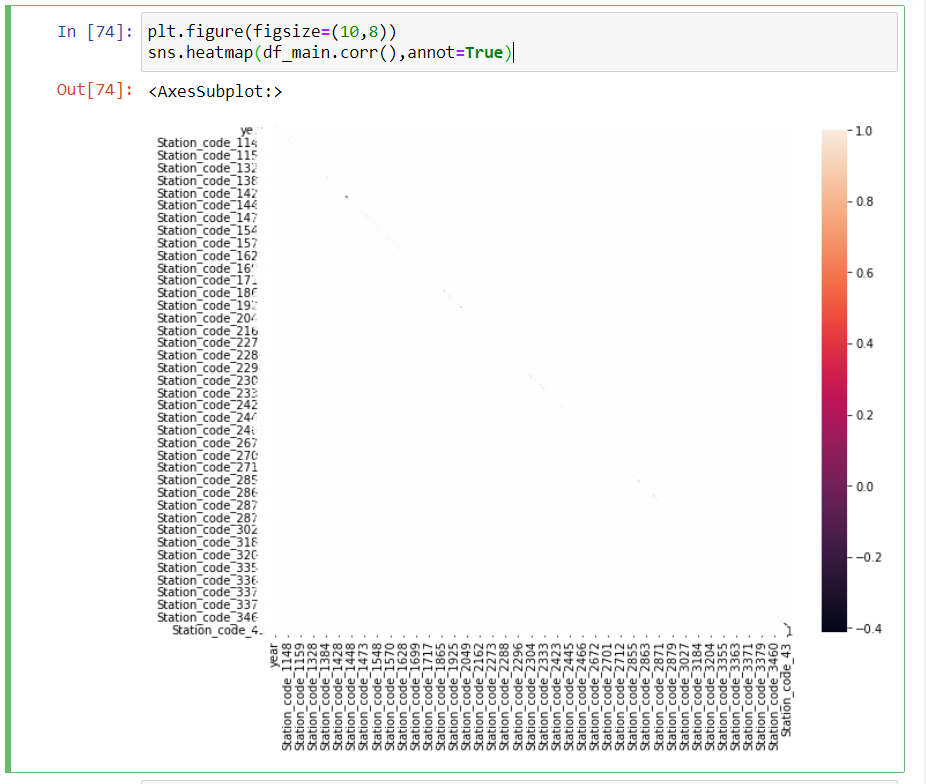


df\_main.corr()

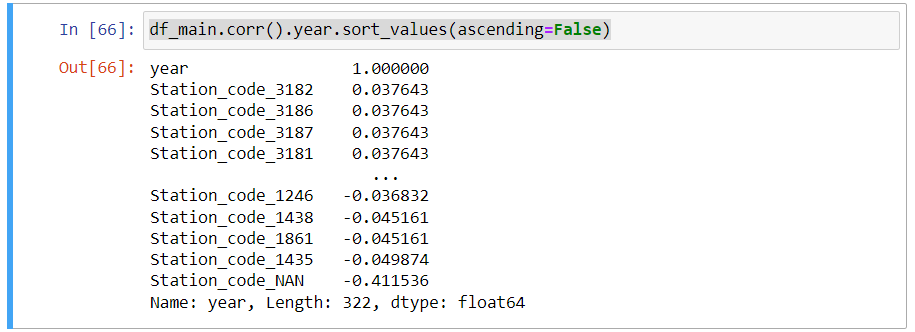


plt.figure(figsize=(10,8))

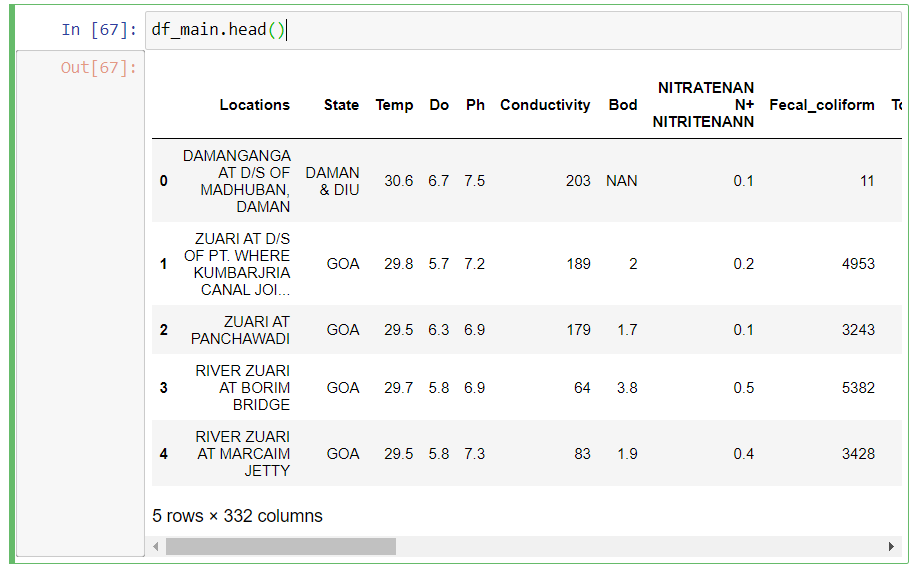
sns.heatmap(df\_main.corr(),annot=True)



df\_main.corr().year.sort\_values(ascending=False)



df\_main.head()

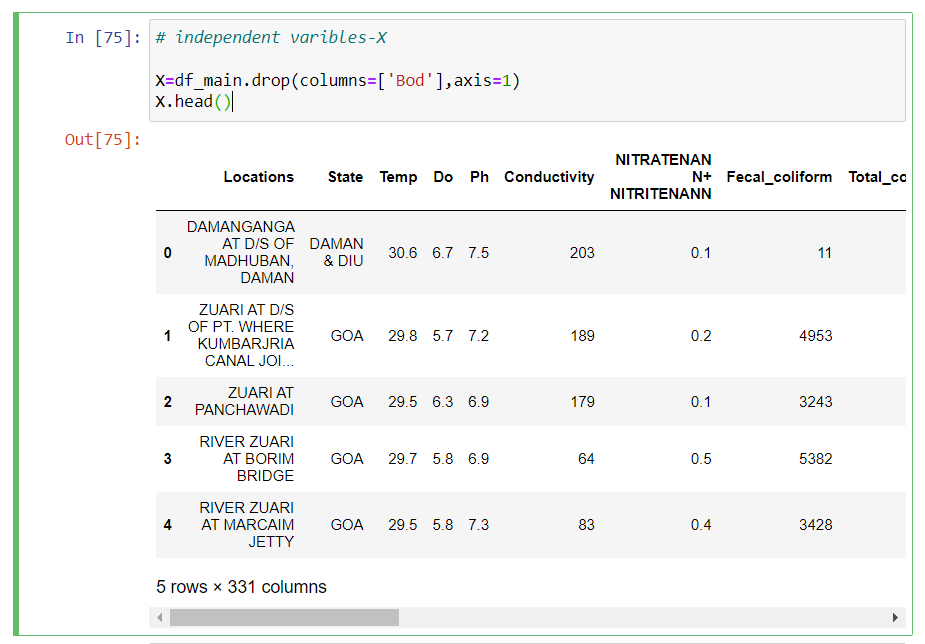


## X and y split

# independent varibles-X

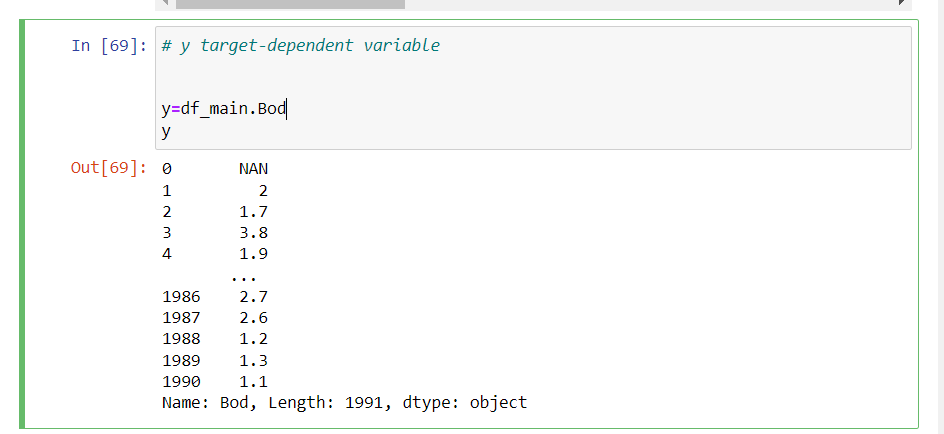
X=df\_main.drop(columns=['Bod'],axis=1)

X.head()

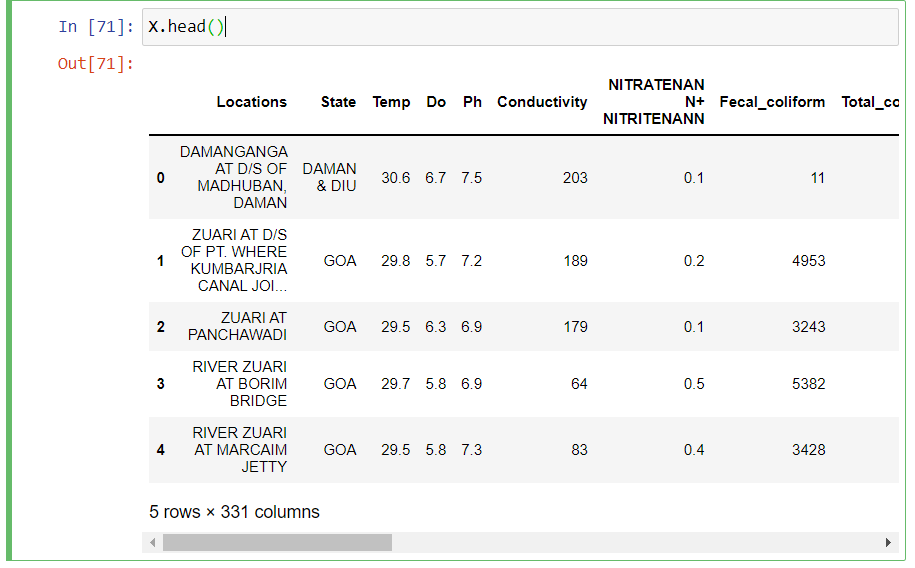


# y target-dependent variable

y=df\_main.Bod



X.head()



## scaling

from sklearn.preprocessing import scale

from sklearn.preprocessing import scale

X=pd.DataFrame(scale(X),columns=X.columns)

X.head()

## Train test split

from sklearn.preprocessing import scale

from sklearn.model\_selection import train\_test\_split

X\_train,X\_test,y\_train,y\_test =train\_test\_split(X\_scaled,y, test\_size=0.3,random\_state=0)

X\_train

X\_train.shape

y\_train.shape

X\_test

X\_test.shape