

NAME: Yash Arvind Sarode

ROLL NO. : 756

PRN: 202201040115

Batch: G3

Practical No 5: (Graded Assignment) Select any one real-life dataset. Perform data analysis. Identify 10 grains for a given dataset. Develop an interactive dashboard using the matplotlib/Seaborn library. (Use any 10 different graphs with proper titles, legends, axis names, etc. to map identified grains)

CODE:

```
import matplotlib.pyplot as plt
import pandas as pd import numpy
as np
df=pd.read_csv('testmarks.csv')
Roll_no=np.array(df['RollNo'])
EDS=np.array(df['EDS'])
SON=np.array(df['SON'])
DT=np.array(df['DT'])
ET=np.array(df['ET'])
Subject=['EDS','DT','ET','SON']
plt.rcParams['figure.figsize']=[12, 12]
plt.subplots_adjust(left=0.1,
bottom=0.1,right=1,
top=0.9,wspace=0.4,hspace=0.4)

plt.subplot(2,5,1)
plt.plot(Roll_no,EDS,color='green')
plt.xlabel('Roll_no')
plt.ylabel('EDS')
plt.subplot(2,5,2)
```

```
plt.plot(Roll_no, SON, color='hotpink')
plt.xlabel('Roll_no') plt.ylabel('SON')
plt.subplot(2,5,3)
plt.plot(Roll_no, DT, color='cyan')
plt.xlabel('Roll_no')
plt.ylabel('DT')
plt.subplot(2,5,4)
plt.plot(Roll_no, ET, color='magenta')
plt.xlabel('Roll_no')
plt.ylabel('ET')
max=[] max.append(df['EDS'].max())
max.append(df['DT'].max())
max.append(df['ET'].max())
max.append(df['SON'].max())
plt.subplot(2,5,5)
plt.ylabel('max')
plt.xlabel('Subject')
plt.plot(Subject, max, color='black')
min=[]
min.append(df['EDS'].min())
min.append(df['DT'].min())
min.append(df['ET'].min())
min.append(df['SON'].min())
plt.subplot(2,5,6)
plt.ylabel('min')
plt.xlabel('Subject')
plt.plot(Subject, min, color='red')
avg=[]
avg.append(df['EDS'].mean())
avg.append(df['DT'].mean())
avg.append(df['ET'].mean())
avg.append(df['SON'].mean())
plt.subplot(2,5,7)
plt.xlabel('Subject') plt.ylabel('avg')
plt.plot(Subject, avg, color='green')
median=[]
```

```

median.append(df['EDS'].median())
median.append(df['DT'].median())
median.append(df['ET'].median())
median.append(df['SON'].median())
plt.subplot(2,5,8) plt.xlabel('Subject')
plt.ylabel('median')
plt.plot(Subject,median)
    std=[]
std.append(df['EDS'].std())
std.append(df['DT'].std())
std.append(df['ET'].std())
std.append(df['SON'].std())
plt.subplot(2,5,9)
plt.xlabel('Subject')
plt.ylabel('std')
plt.plot(Subject,std)
    mode=[]
mode.append(df['EDS'].mode())
mode.append(df['DT'].mode())
mode.append(df['ET'].mode())
mode.append(df['SON'].mode())
plt.subplot(2,5,10)
plt.xlabel('Subject')
plt.ylabel('mode')
plt.plot(Subject,mode)
    var=[]
var.append(df['EDS'].var())
var.append(df['DT'].var())
var.append(df['ET'].var())
var.append(df['SON'].var())
plt.xlabel('Subject')
plt.ylabel('var')
plt.plot(Subject,var,color='y')
plt.show()

```

OUTPUT:

