

Monthly Internship Program for Professionals@ THE CODERS READY

THE CODERS READY

Data Science and Business Analytics

Name:Ashima Sapehia

Trainee Intern @CodersReady

Task 2:

Data of Students based on number of Study hours

Level=Basic

Problem Statement: 1.Predict the percentage of an student based on the no. of study hours. 2.This is a simple linear regression task as it involves just 2 variable. 3.As a data analyst,try to find out the prediction for score. 4.What will be the predicted score if a student studenries for 8.5hrs/day? 5.You can use Python tool to perform this analysis.

In [2]:  
import pandas as pd  
import numpy as np

In [1]:  
import seaborn as sns

In [3]:  
import matplotlib.pyplot as plt  
import seaborn as sn  
from sklearn.linear\_model import LinearRegression  
from sklearn.model\_selection import train\_test\_split  
%matplotlib inline

In [3]:  
data=pd.read\_csv("student\_score.txt",sep='\t')  
data

Out[3]:

	Hours	Scores
0	2.5	21
1	5.1	47
2	3.2	27
3	8.5	75
4	3.5	30
5	1.5	20
6	9.2	88
7	5.5	60
8	8.3	81
9	2.7	25
10	7.7	85
11	5.9	62
12	4.5	41
13	3.3	42
14	1.1	17
15	8.9	95
16	2.5	30
17	1.9	24
18	6.1	67
19	7.4	69
20	2.7	30
21	4.8	54
22	3.8	35
23	6.9	76
24	7.8	86

Visualization with line plot

In [25]:  
plt.style.use('fivethirtyeight')  
data.plot(kind='line')  
plt.title('Hours vs Scores')  
plt.xlabel('Hours')  
plt.ylabel('Scores')  
plt.show()

Data Visualization area plot

In [27]:  
xmin=min(data.Hours)  
xmax=max(data.Hours)  
data.plot(kind='area',alpha=0.8,stacked=True,figsize=(15,10),xlim=(xmin,xmax))  
plt.title('Hours vs Scores',size=14)  
plt.xlabel('Hours',size=15)  
plt.ylabel('Scores',size=15)  
plt.show()

Data Visualizing with scatter plot

In [29]:  
data.plot(kind='scatter',x='Hours',y='Scores',color='b',figsize=(10,8))  
plt.title('Hours vs Percentage')  
plt.xlabel('Hours')  
plt.ylabel('Scores')  
plt.show()

By Visualization we come to know that this problem can be solved by linear regression

Regression Plot

In [37]:  
sns.regplot(x= data['Hours'], y= data['Scores'])  
plt.title('Regression Plot',size=20)  
plt.ylabel('Marks Percentage', size=12)  
plt.xlabel('Hours Studied', size=12)  
plt.show()  
print(data.corr())

	Hours	Scores
Hours	1.000000	0.976191
Scores	0.976191	1.000000

In [5]:  
sns.countplot(data['Hours']);

In [6]:  
sns.countplot(data['Scores']);

In [7]:  
import plotly.express as px  
import plotly.graph\_objects as go

In [9]:  
fig=px.bar(data,x="Hours",y="Scores",title="Details")  
fig.show()

In [11]:  
fig=px.histogram(data,x="Hours")  
fig.show()

In [12]:  
fig=px.histogram(data,x="Scores")  
fig.show()

In [14]:  
fig=px.sunburst(data,path=["Hours","Scores"])  
fig.show()

In [15]:  
fig=px.pie(data,values='Hours',names='Scores')  
fig.show()

Predicting the score with the single input value

In [9]:  
hours=8.5  
predicted\_score=regressor.predict([[hours]])  
print(f'NO. OF HOURS={hours}')  
print(f'predicted Score={predicted\_score[0]}')  
  
NO. OF HOURS=8.5  
predicted Score=[86.0031392]

In [ ]: