mk 27-07-2023

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [41]:
```
a=pd.read_csv(r"C:\Users\user\Downloads\13_placement.csv")
a
```

Out[41]:

	cgpa	placement_exam_marks	placed
0	7.19	26.0	1
1	7.46	38.0	1
2	7.54	40.0	1
3	6.42	8.0	1
4	7.23	17.0	0
...	...	...	...
995	8.87	44.0	1
996	9.12	65.0	1
997	4.89	34.0	0
998	8.62	46.0	1
999	4.90	10.0	1

1000 rows × 3 columns

In [42]:
```
a.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 3 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   cgpa                 1000 non-null   float64
 1   placement_exam_marks  1000 non-null   float64
 2   placed               1000 non-null   int64  
dtypes: float64(2), int64(1)
memory usage: 23.6 KB
```

In [43]:
```
a.columns
```

Out[43]:
```
Index(['cgpa', 'placement_exam_marks', 'placed'], dtype='object')
```

In [44]:
```
a.head()
```

Out[44]:

	cgpa	placement_exam_marks	placed
0	7.19	26.0	1
1	7.46	38.0	1
2	7.54	40.0	1
3	6.42	8.0	1
4	7.23	17.0	0

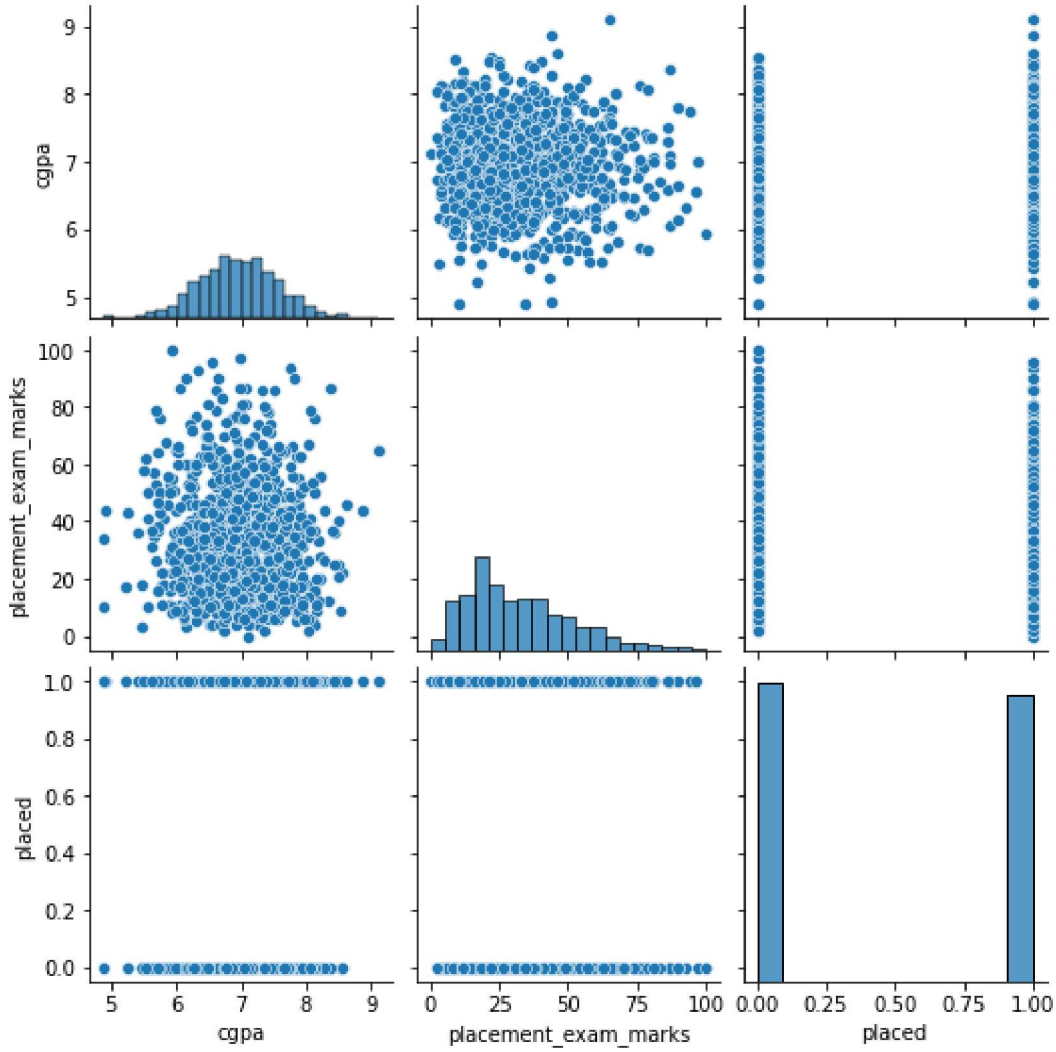
In [45]:
```
a.describe()
```

Out[45]:

	cgpa	placement_exam_marks	placed
count	1000.000000	1000.000000	1000.000000
mean	6.961240	32.225000	0.489000
std	0.615898	19.130822	0.500129
min	4.890000	0.000000	0.000000
25%	6.550000	17.000000	0.000000
50%	6.960000	28.000000	0.000000
75%	7.370000	44.000000	1.000000
max	9.120000	100.000000	1.000000

In [46]:
```
sns.pairplot(a)
```

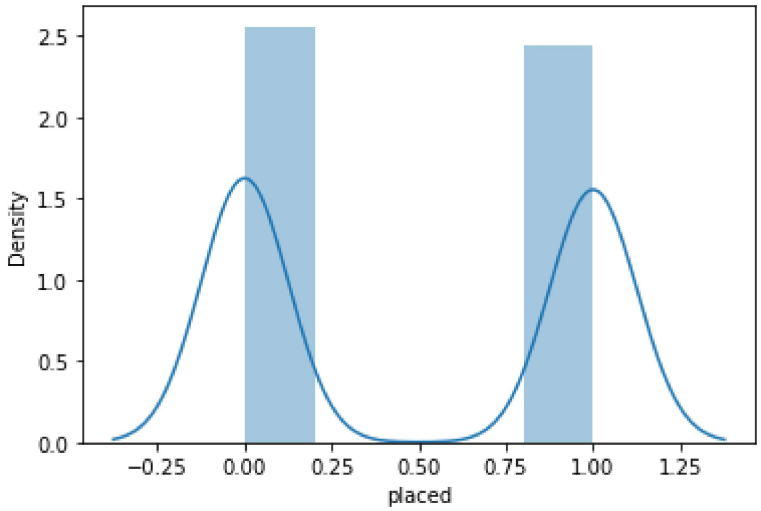
Out[46]:
```
<seaborn.axisgrid.PairGrid at 0x25f067f7070>
```



In [48]:
```
sns.distplot(a[ 'placed'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: Fut
ureWarning: `distplot` is a deprecated function and will be removed in a futu
re version. Please adapt your code to use either `displot` (a figure-level fu
nction with similar flexibility) or `histplot` (an axes-level function for hi
stograms).
  warnings.warn(msg, FutureWarning)
```

Out[48]: <AxesSubplot:xlabel='placed', ylabel='Density'>



In [50]:
```
x1=a[['cgpa', 'placement_exam_marks']]
```

In [29]:
```
sns.heatmap(x1.corr())
```

Out[29]: <AxesSubplot:>

In [52]:
```
x=a[['cgpa', 'placement_exam_marks']]
y=a[ 'placed']
```

In [53]:
```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

In [54]:
```
from sklearn.linear_model import LinearRegression
lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[54]:
```
LinearRegression()
```

In [55]:
```
print(lr.intercept_)
```

```
0.32256820690507815
```

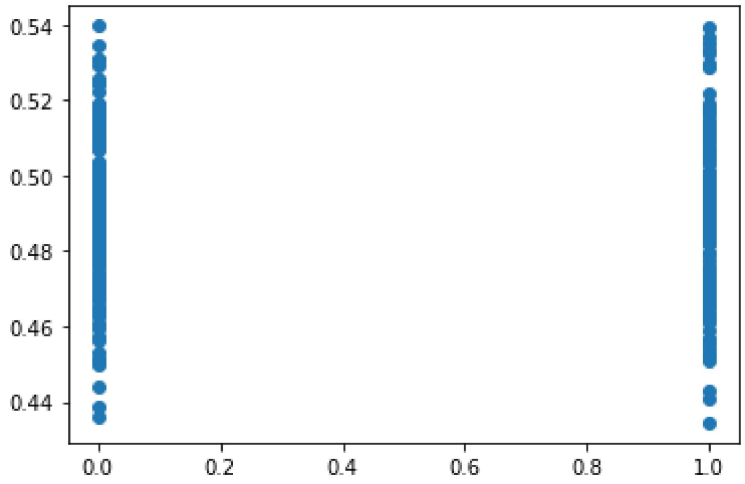
In [56]:
```
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
coeff
```

Out[56]:

	Co-efficient
cgpa	0.026793
placement_exam_marks	-0.000516

In [57]:
```
prediction=lr.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[57]:
```
<matplotlib.collections.PathCollection at 0x25f05ae76d0>
```



In [58]:
```
print(lr.score(x_test,y_test))
```

```
0.0003092921778353741
```