

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

```
In [2]: df=pd.read_csv("13_placement.csv")
df
```

Out[2]:

	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 [3]: df.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
```

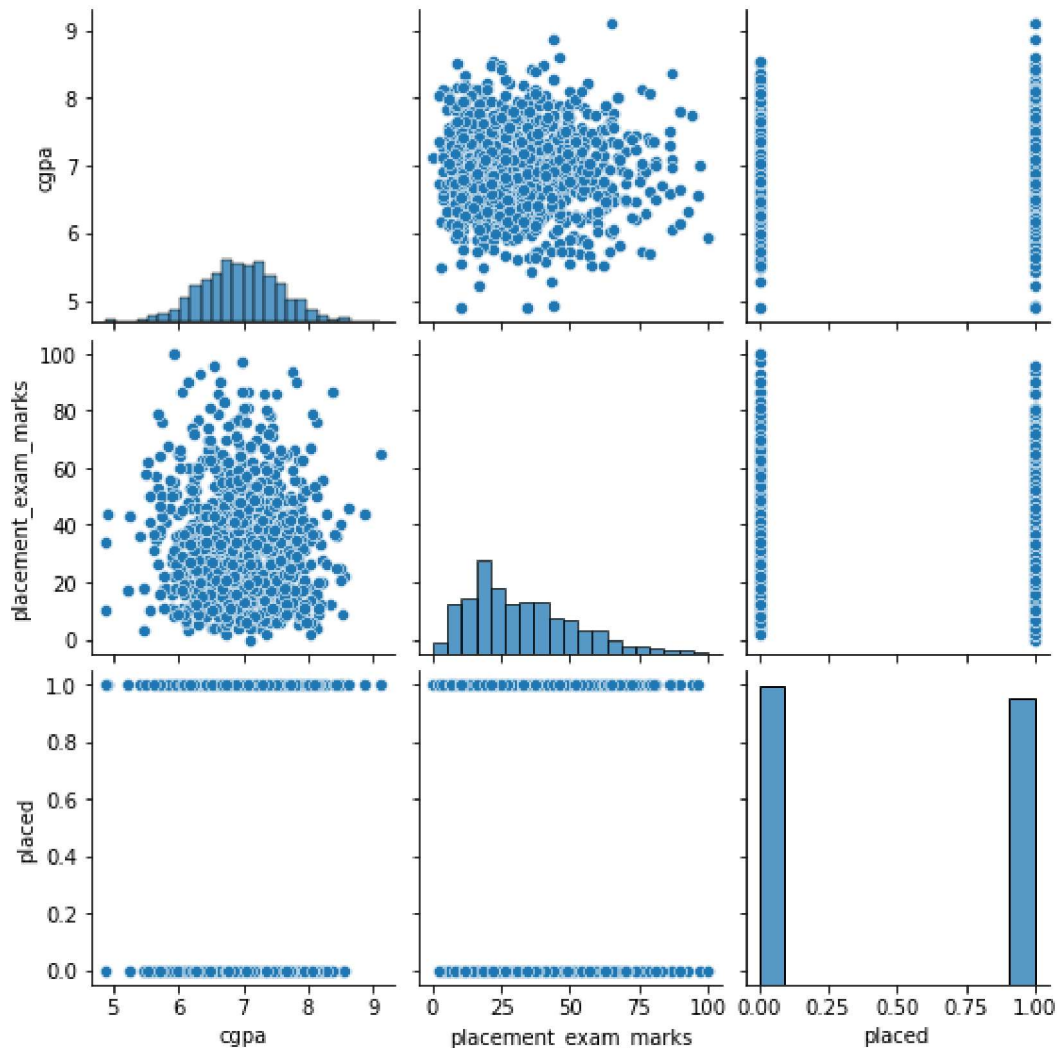
```
In [4]: df.describe()
```

```
Out[4]:
```

	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

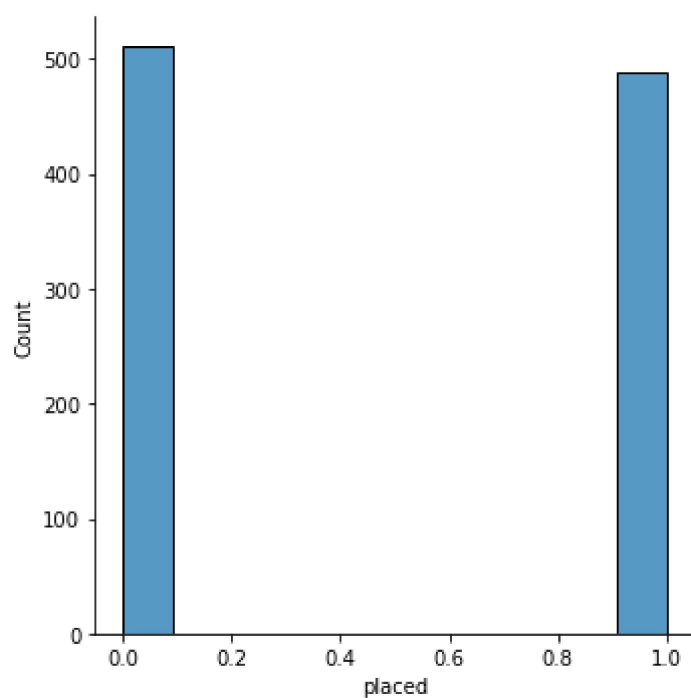
```
In [5]: sns.pairplot(df)
```

```
Out[5]: <seaborn.axisgrid.PairGrid at 0x1440cd48d00>
```



```
In [8]: sns.displot(df['placement_exam_marks'])
```

```
Out[8]: <seaborn.axisgrid.FacetGrid at 0x1440e393fa0>
```



```
In [9]: df1=df.drop(['placed'],axis=1)
df1
```

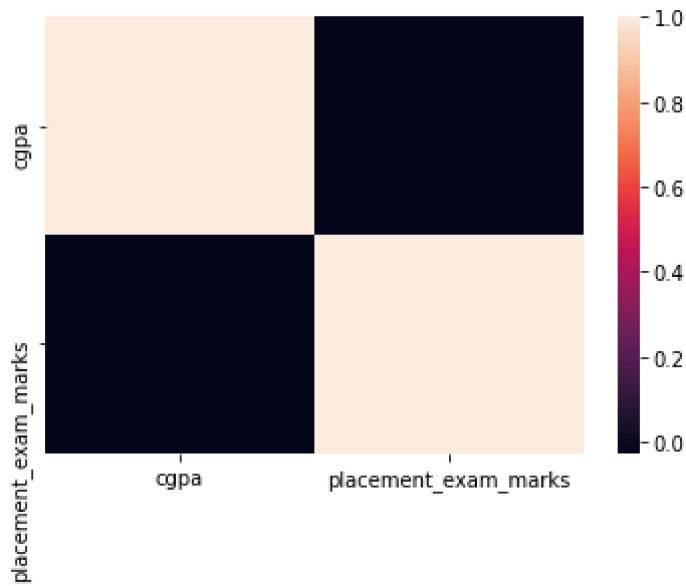
```
Out[9]:
```

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

1000 rows × 2 columns

```
In [10]: sns.heatmap(df1.corr())
```

```
Out[10]: <AxesSubplot:>
```



```
In [11]: from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

```
In [13]: y=df['placement_exam_marks']
x=df1.drop(['placement_exam_marks'],axis=1)
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
print(x_train)
```

```
cgpa
353  6.82
379  7.37
146  6.75
117  7.11
336  7.93
..   ...
144  6.67
644  7.65
236  7.70
519  6.60
682  6.34
```

```
[700 rows x 1 columns]
```

```
In [14]: model=LinearRegression()
model.fit(x_train,y_train)
model.intercept_
```

```
Out[14]: 34.26800206769539
```

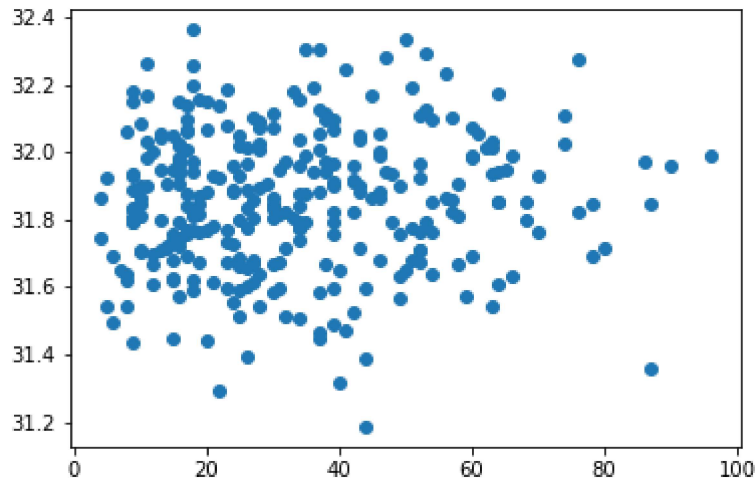
```
In [15]: coeff=pd.DataFrame(model.coef_,x.columns,columns=["Coefficient"])  
coeff
```

Out[15]:

	Coefficient
cgpa	-0.347464

```
In [16]: prediction=model.predict(x_test)  
plt.scatter(y_test,prediction)
```

Out[16]: <matplotlib.collections.PathCollection at 0x1440f1072e0>



```
In [17]: model.score(x_test,y_test)
```

Out[17]: -0.002973135729535503

```
In [18]: from sklearn.linear_model import Ridge,Lasso
```

```
In [19]: rr = Ridge(alpha=10)  
rr.fit(x_train,y_train)
```

Out[19]: Ridge(alpha=10)

```
In [20]: rr.score(x_test,y_test)
```

Out[20]: -0.0030162898246126613

```
In [21]: la = Lasso(alpha=10)  
la.fit(x_train,y_train)
```

Out[21]: Lasso(alpha=10)

```
In [22]: la.score(x_test,y_test)
```

Out[22]: -0.004300952019827831

In []: