```
In [75]: import pandas as pd
         import matplotlib.pyplot as plt
```

```
In [82]: pd.read_excel("placement.xlsx");
```

```
In [ ]:
```

```
In [7]: data = pd.read_excel("placement.xlsx");
```

```
In [ ]:
```

```
In [9]: data = pd.read_excel("placement.xlsx",usecols=[0,1])
```

```
In [10]: data
```

Out[10]:

	cgpa	placement_exam_marks
0	7.19	26
1	7.46	38
2	7.54	40
3	6.42	8
4	7.23	17
...	...	...
995	8.87	44
996	9.12	65
997	4.89	34
998	8.62	46
999	4.90	10

1000 rows × 2 columns

```
In [11]: X = data.iloc[:,0:1]
         Y = data.iloc[:,-1]
```

```
In [12]: print(X)
```

Loading [MathJax]/extensions/Safe.js

```
     cgpa
0    7.19
1    7.46
2    7.54
3    6.42
4    7.23
..    ...
995  8.87
996  9.12
997  4.89
998  8.62
999  4.90

[1000 rows x 1 columns]
```

In [13]:
```
print(Y)
```

```
0      26
1      38
2      40
3       8
4      17
       ..
995    44
996    65
997    34
998    46
999    10
Name: placement_exam_marks, Length: 1000, dtype: int64
```

In [19]:
```
from sklearn.model_selection import train_test_split
```

In [20]:
```
x_train,x_test,y_train,y_test = train_test_split(X,Y , test_size=0.2, random
```

In [22]:
```
from sklearn.linear_model import LinearRegression
```

In [30]:
```
lr = LinearRegression()
```

In [31]:
```
lr.fit(x_train,y_train)
```

Out[31]:
```
▼ LinearRegression
LinearRegression()
```

In [35]:
```
lr.predict(x_test.iloc[4].values.reshape(1,1))
```

```
C:\Users\Vinay Partap\AppData\Local\Programs\Python\Python312\Lib\site-packa
ges\sklearn\base.py:465: UserWarning: X does not have valid feature names, b
ut LinearRegression was fitted with feature names
  warnings.warn(
```

Out[35]:
```
array([32.16036528])
```

In [36]:
```
reg = LinearRegression()
```

Loading [MathJax]/extensions/Safe.js

```
In [40]: reg.fit(data[['cgpa']],data.placement_exam_marks)
```

```
Out[40]: ▼ LinearRegression
         LinearRegression()
```

```
In [44]: reg.predict(x_test.iloc[4].values.reshape(1,1))
```

```
C:\Users\Vinay Partap\AppData\Local\Programs\Python\Python312\Lib\site-packa
ges\sklearn\base.py:465: UserWarning: X does not have valid feature names, b
ut LinearRegression was fitted with feature names
  warnings.warn(
```

```
Out[44]: array([32.31957477])
```

```
In [ ]:
```

```
In [51]: data
```

Out[51]:

	cgpa	placement_exam_marks
0	7.19	26
1	7.46	38
2	7.54	40
3	6.42	8
4	7.23	17
...	...	...
995	8.87	44
996	9.12	65
997	4.89	34
998	8.62	46
999	4.90	10

1000 rows × 2 columns

```
In [52]: x_col = data.iloc[:,0].values
```

```
In [53]: y_col = data.iloc[:,1].values
```

```
In [ ]:
```

```
In [ ]:
```

Loading [MathJax]/extensions/Safe.js

In [63]:

```
from numpy import array, zeros,mean
x_mean = mean(x_col)
y_mean= mean(y_col)
```

In [64]:

```
numenator = 0;
denominator = 0;

for i in range(0,len(x_col)):
    numenator = numenator + ((x_col[i]-x_mean)*(y_col[i]-y_mean))
    denominator = denominator + ((x_col[i]-x_mean)**2)
```

In [65]:

```
slope = numenator/denominator
```

In [67]:

```
c = y_mean- slope*x_mean
```

In [69]:

```
print("Equation of Line is : y=",slope,"x +",c)
```

```
Equation of Line is : y= -0.8501867444226203 x + 38.14335397274452
```

In [93]:

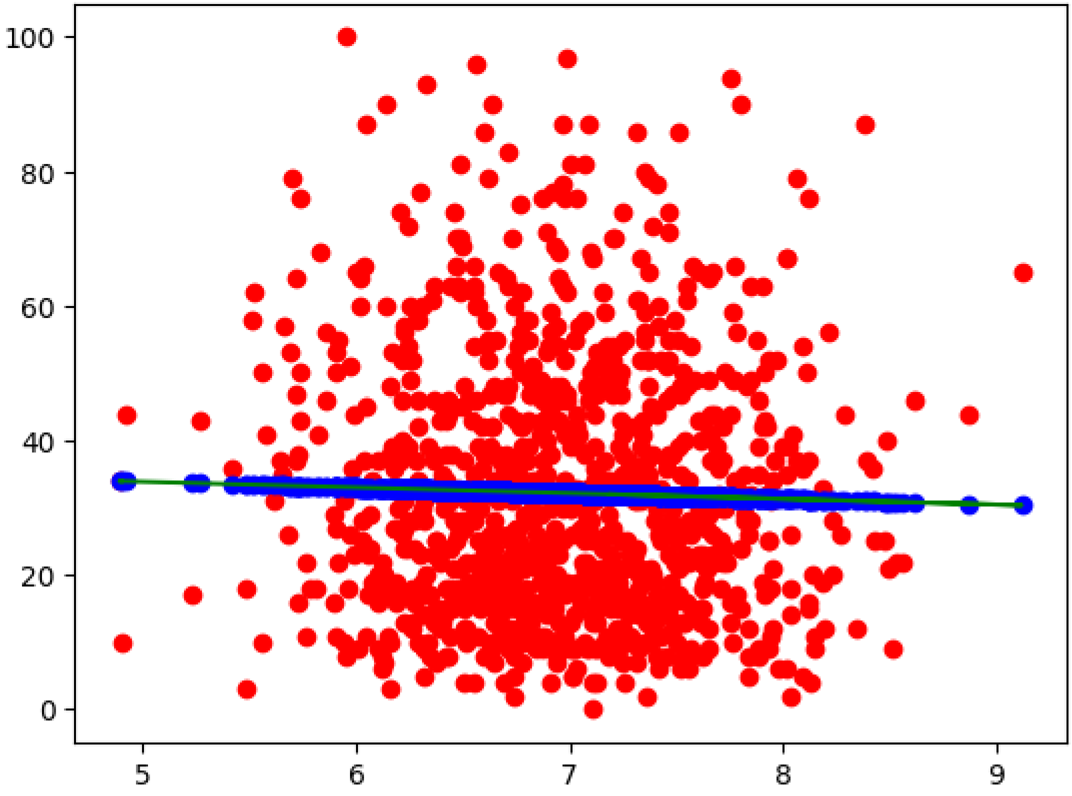
```
import numpy as np
y_predicted = np.array(slope*x_col + c)
```

In [95]:

```
plt.scatter(x_col,y_col,color='red')
plt.scatter(x_col,y_predicted,color='blue')
plt.plot(x_col,y_predicted,color='green')
```

Out[95]:

```
[<matplotlib.lines.Line2D at 0x1337b1704a0>]
```



Loading [MathJax]/extensions/Safe.js

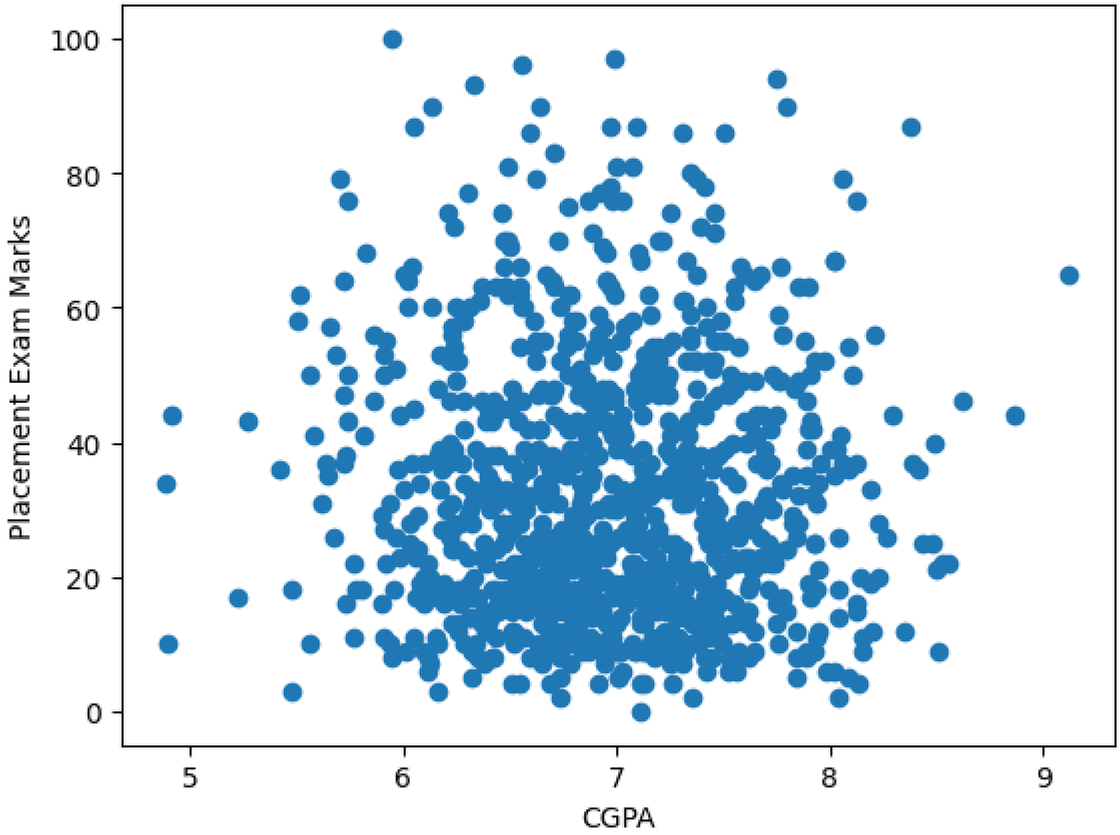
In [85]:

```python
data = pd.read_excel("placement.xlsx", usecols=[0, 1])

# Check the column names to make sure they are correct
print(data.columns)

# Plot the data
plt.scatter(data[['cgpa']], data[['placement_exam_marks']])
plt.xlabel('CGPA')
plt.ylabel('Placement Exam Marks')
plt.show()
```

```
Index(['cgpa', 'placement_exam_marks'], dtype='object')
```



In [ ]:

Loading [MathJax]/extensions/Safe.js