Students performance data analysis and visualization

To understand the influence of various factors like economic, personal and social on the students performance

Inferences would be :

1. How to imporve the students performance in each test ?
2. What are the major factors influencing the test scores ?
3. Effectiveness of test preparation course?

In [24]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

we will set the minimum marks to 40 to pass in a exam

In [25]:
```
passmark = 40
```

In [26]:
```
df = pd.read_csv("../input/StudentsPerformance.csv")
```

We will print top few rows to understand about the various data columns

In [27]:
```
df.head()
```

Out[27]:

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75

Size of data frame

In [28]:
```
print (df.shape)
```

```
(1000, 8)
```

In [29]:
```
df.describe()
```

Out[29]:

	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

Let us check for any missing values

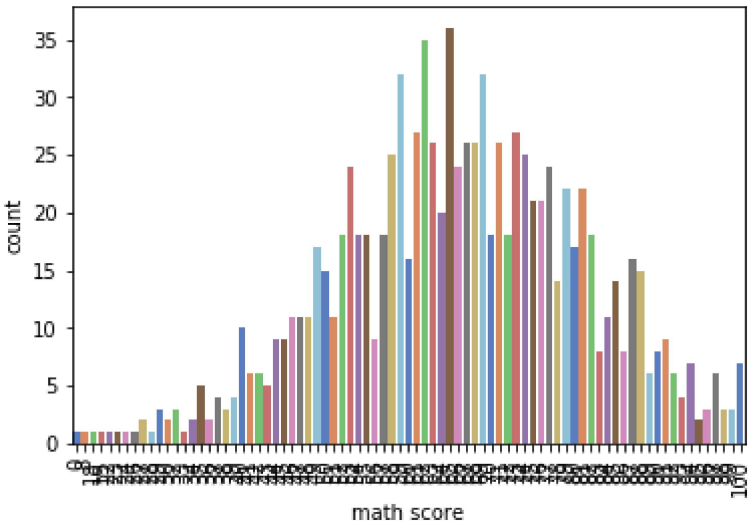
In [30]:
```
df.isnull().sum()
```

Out[30]:
```
gender                       0
race/ethnicity               0
parental level of education  0
lunch                        0
test preparation course      0
math score                   0
reading score                0
writing score                0
dtype: int64
```

As seen above, there are no missing (null) values in this dataframe.

Let us explore the Math Score first

In [31]:
```
p = sns.countplot(x="math score", data = df, palette="muted")
_ = plt.setp(p.get_xticklabels(), rotation=90)
```

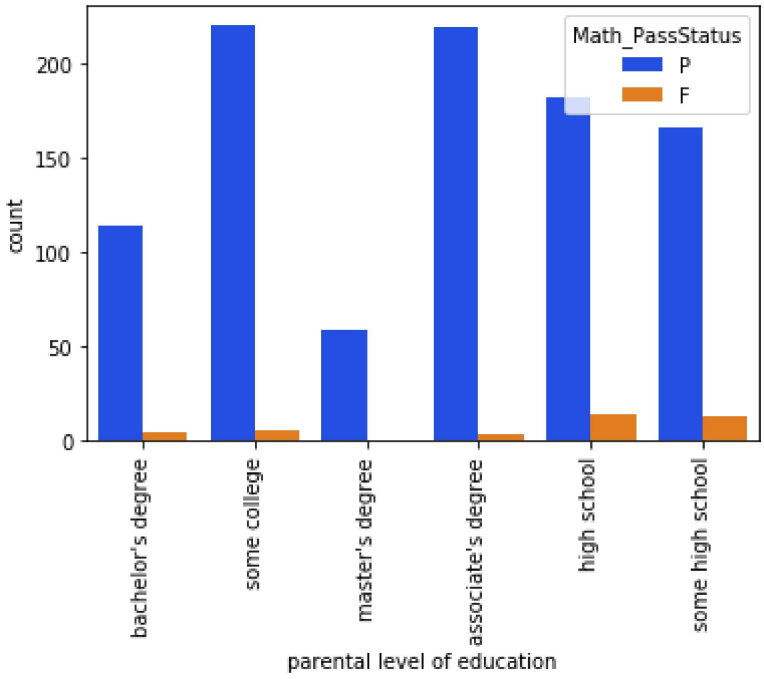


How many students passed in Math exam ?

In [32]:
```
df['Math_PassStatus'] = np.where(df['math score']<passmark, 'F', 'P')
df.Math_PassStatus.value_counts()
```

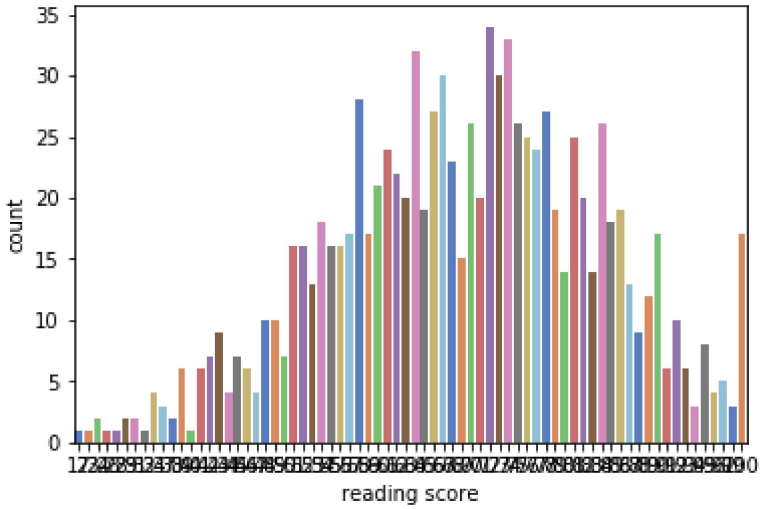
Out[32]:
```
P    960
F     40
Name: Math_PassStatus, dtype: int64
```

In [33]:
```
p = sns.countplot(x='parental level of education', data = df, hue='Math_PassStatus', p
_ = plt.setp(p.get_xticklabels(), rotation=90)
```



Let us explore the Reading score

In [34]:
```
sns.countplot(x="reading score", data = df, palette="muted")
plt.show()
```

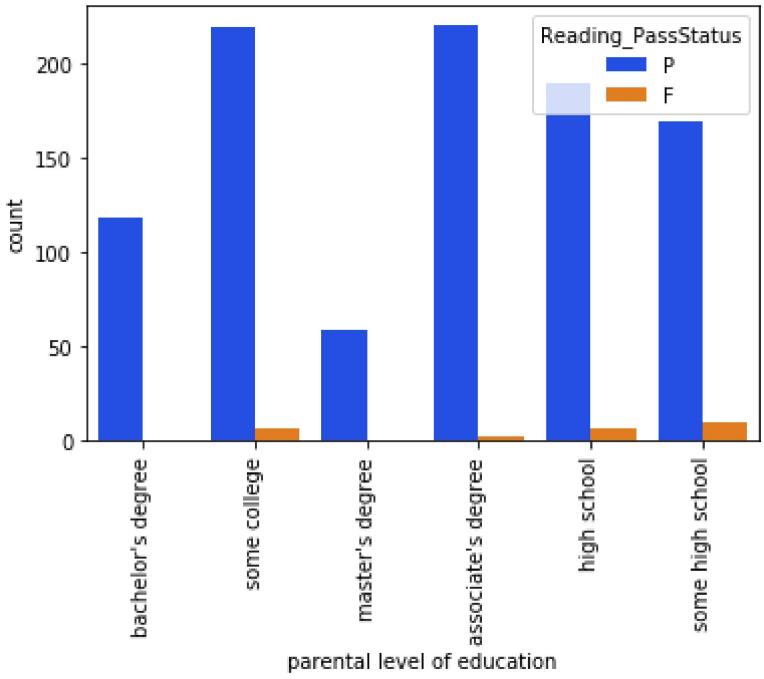


How many students passed in reading ?

In [35]:
```
df['Reading_PassStatus'] = np.where(df['reading score']<passmark, 'F', 'P')
df.Reading_PassStatus.value_counts()
```

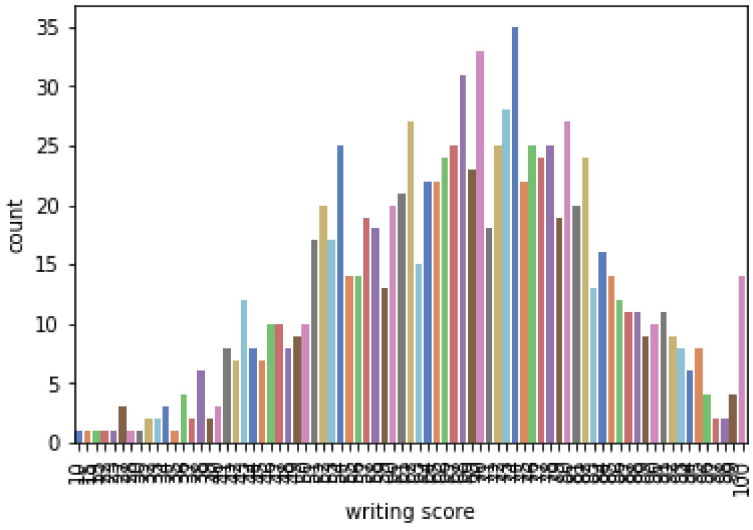
Out[35]:
```
P    974
F     26
Name: Reading_PassStatus, dtype: int64
```

In [36]:
```
p = sns.countplot(x='parental level of education', data = df, hue='Reading_PassStatus'
_ = plt.setp(p.get_xticklabels(), rotation=90)
```



Let us explore writing score

In [37]:
```
p = sns.countplot(x="writing score", data = df, palette="muted")
_ = plt.setp(p.get_xticklabels(), rotation=90)
```

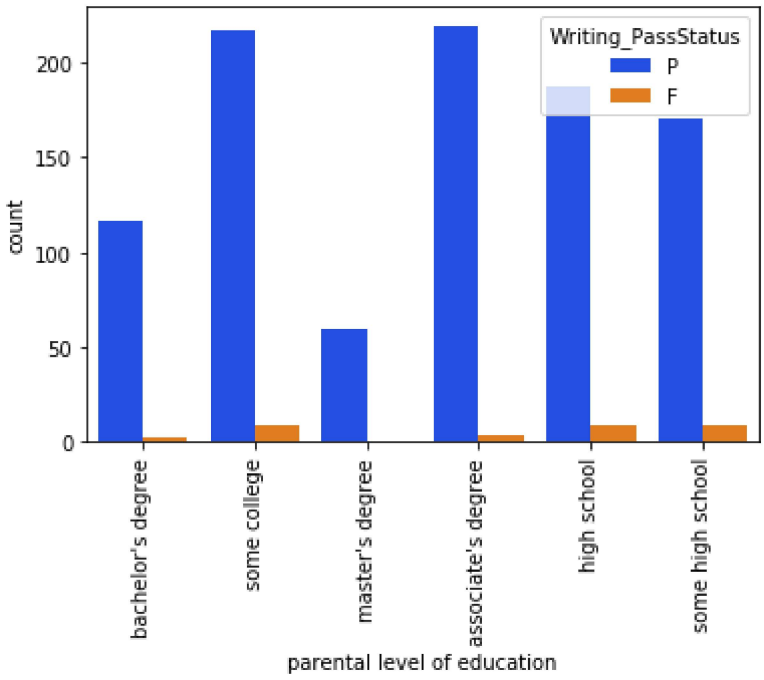


How many students passed writing ?

In [38]:
```
df['Writing_PassStatus'] = np.where(df['writing score']<passmark, 'F', 'P')
df.Writing_PassStatus.value_counts()
```

Out[38]:
```
P    968
F     32
Name: Writing_PassStatus, dtype: int64
```

In [39]:
```
p = sns.countplot(x='parental level of education', data = df, hue='Writing_PassStatus'
_ = plt.setp(p.get_xticklabels(), rotation=90)
```

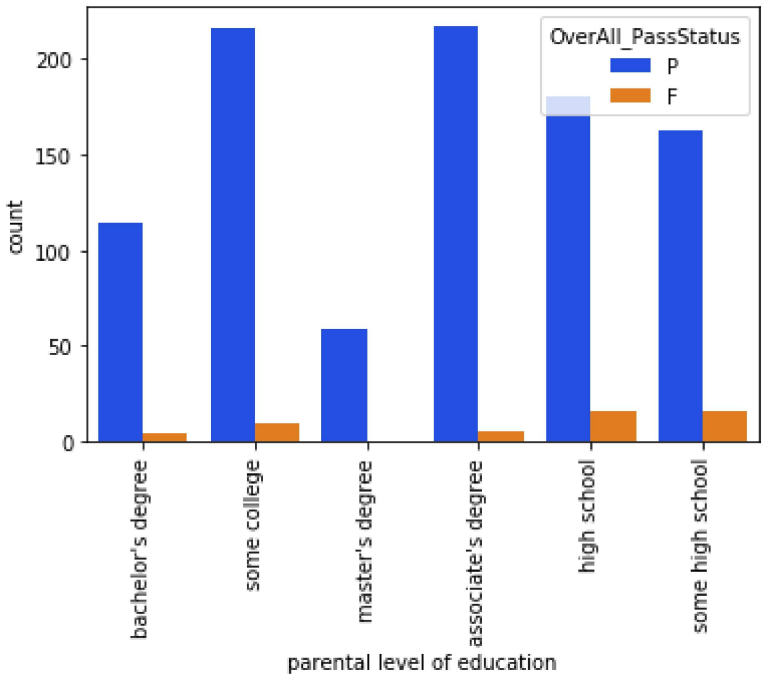


let us check "How many students passed in all the subjects ?"

In [40]:
```
df['OverAll_PassStatus'] = df.apply(lambda x : 'F' if x['Math_PassStatus'] == 'F' or
                                    x['Reading_PassStatus'] == 'F' or x['Writing_PassS

df.OverAll_PassStatus.value_counts()
```

Out[40]:
```
P    949
F     51
Name: OverAll_PassStatus, dtype: int64
```

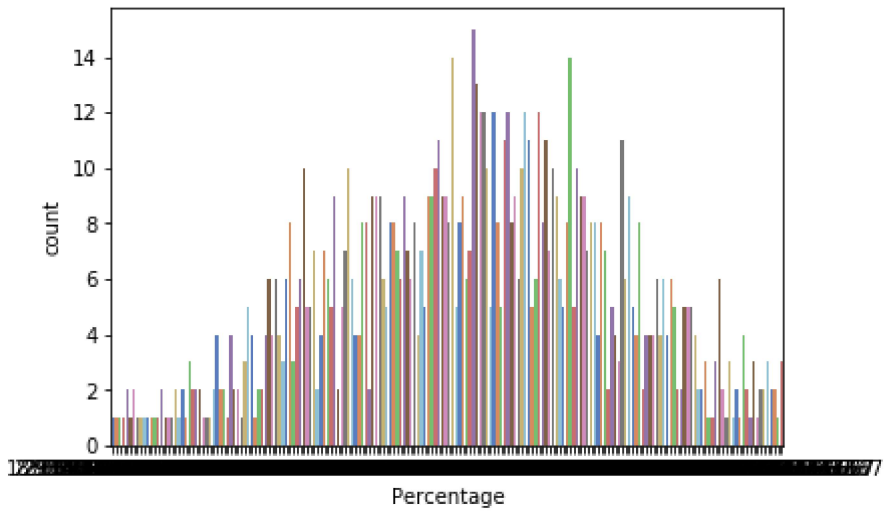
In [41]:
```
p = sns.countplot(x='parental level of education', data = df, hue='OverAll_PassStatus'
_ = plt.setp(p.get_xticklabels(), rotation=90)
```



Find the percentage of marks

In [42]:
```
df['Total_Marks'] = df['math score']+df['reading score']+df['writing score']
df['Percentage'] = df['Total_Marks']/3
```

In [43]:
```
p = sns.countplot(x="Percentage", data = df, palette="muted")
_ = plt.setp(p.get_xticklabels(), rotation=0)
```



Let us assign the grades

Grading

above 80 = A Grade

70 to 80 = B Grade

60 to 70 = C Grade

50 to 60 = D Grade

40 to 50 = E Grade

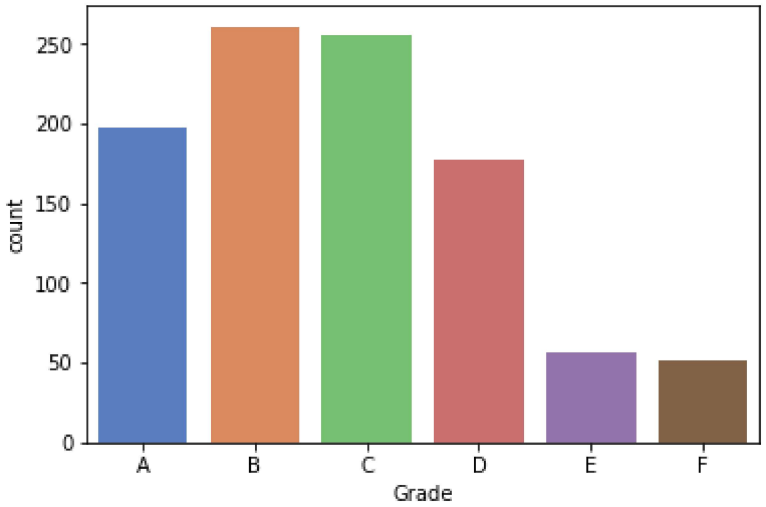
below 40 = F Grade (means Fail)

In [44]:
```
def GetGrade(Percentage, OverAll_PassStatus):
    if ( OverAll_PassStatus == 'F'):
        return 'F'
    if ( Percentage >= 80 ):
        return 'A'
    if ( Percentage >= 70):
        return 'B'
    if ( Percentage >= 60):
        return 'C'
    if ( Percentage >= 50):
        return 'D'
    if ( Percentage >= 40):
        return 'E'
    else:
        return 'F'

df['Grade'] = df.apply(lambda x : GetGrade(x['Percentage'], x['OverAll_PassStatus']),

df.Grade.value_counts()
```

Out[44]:
```
B    261
C    256
A    198
D    178
E     56
F     51
Name: Grade, dtype: int64
```

we will plot the grades obtained in a order

In [45]:
```
sns.countplot(x="Grade", data = df, order=['A','B','C','D','E','F'],  palette="muted")
plt.show()
```



In [46]:
```
p = sns.countplot(x='parental level of education', data = df, hue='Grade', palette='br
_ = plt.setp(p.get_xticklabels(), rotation=90)
```