```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv('/content/winequality-red.csv')
df
```

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH
0	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51
1	7.8	0.880	0.00	2.6	0.098	25.0	67.0	0.99680	3.20
2	7.8	0.760	0.04	2.3	0.092	15.0	54.0	0.99700	3.26
3	11.2	0.280	0.56	1.9	0.075	17.0	60.0	0.99800	3.16
4	7.4	0.700	0.00	1.9	0.076	11.0	34.0	0.99780	3.51
...	...	...	...	...	...	...	...	...	...
1594	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45
1595	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52
1596	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42
1597	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57
1598	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39

```
df.head()
```

```
df.tail()
```

```
df.isna().sum()
```

```
fixed acidity           0
volatile acidity        0
citric acid             0
residual sugar          0
chlorides               0
free sulfur dioxide     0
total sulfur dioxide    0
density                 0
pH                      0
sulphates               0
alcohol                 0
quality                 0
dtype: int64
```

```
df.dtypes
```

```
fixed acidity           float64
volatile acidity        float64
citric acid             float64
residual sugar          float64
chlorides               float64
free sulfur dioxide     float64
total sulfur dioxide    float64
density                 float64
pH                      float64
sulphates               float64
alcohol                 float64
quality                   int64
dtype: object
```

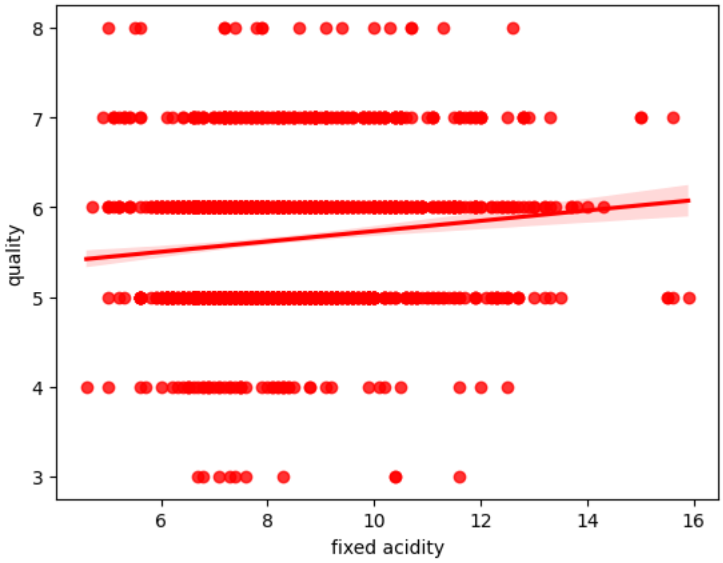
Double-click (or enter) to edit

```
x=df.iloc[:,:-1]
x
```

```
y=df.iloc[:,-1]
y
```

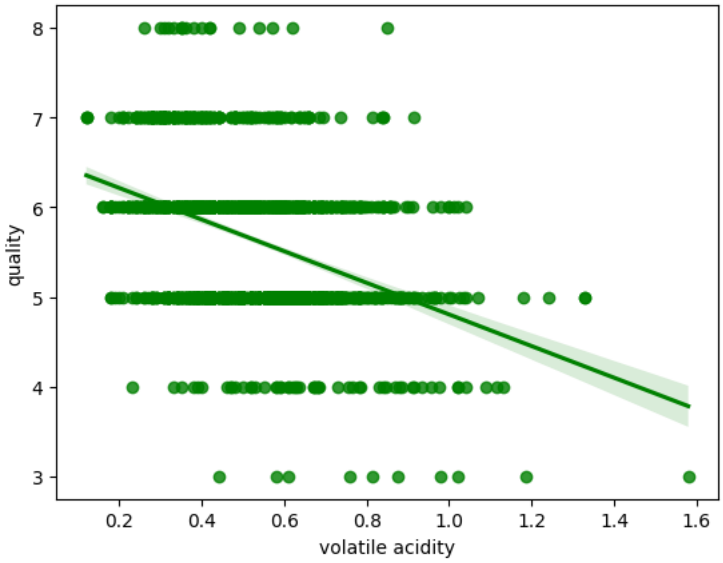
```
sns.regplot(x=df['fixed acidity'],y=y,color='r')
```

```
<Axes: xlabel='fixed acidity', ylabel='quality'>
```



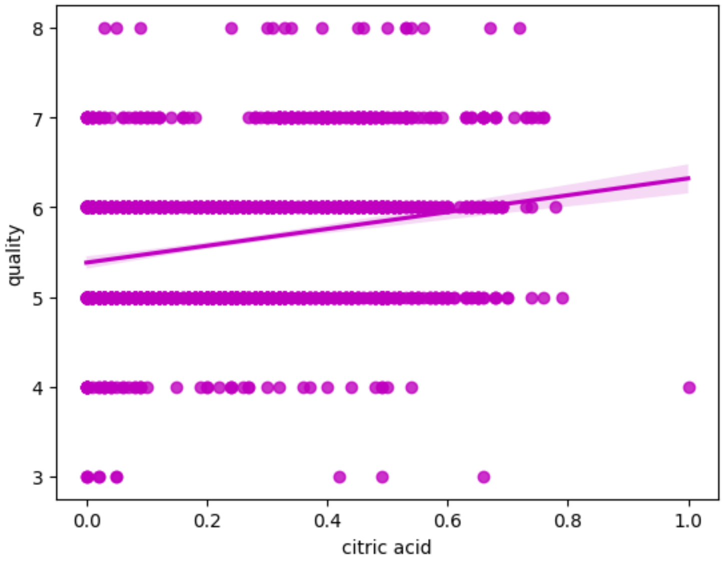
```
sns.regplot(x=df['volatile acidity'],y=y,color='g')
```

```
<Axes: xlabel='volatile acidity', ylabel='quality'>
```



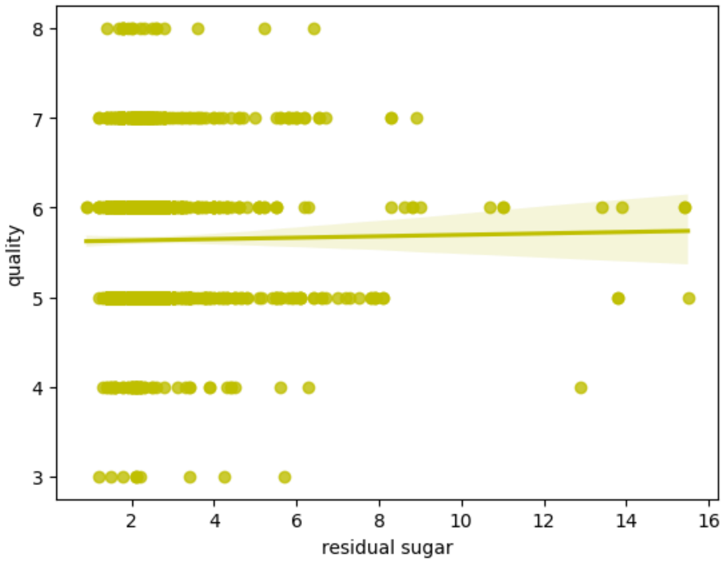
```
sns.regplot(x=df['citric acid'],y=y,color='m')
```

```
<Axes: xlabel='citric acid', ylabel='quality'>
```



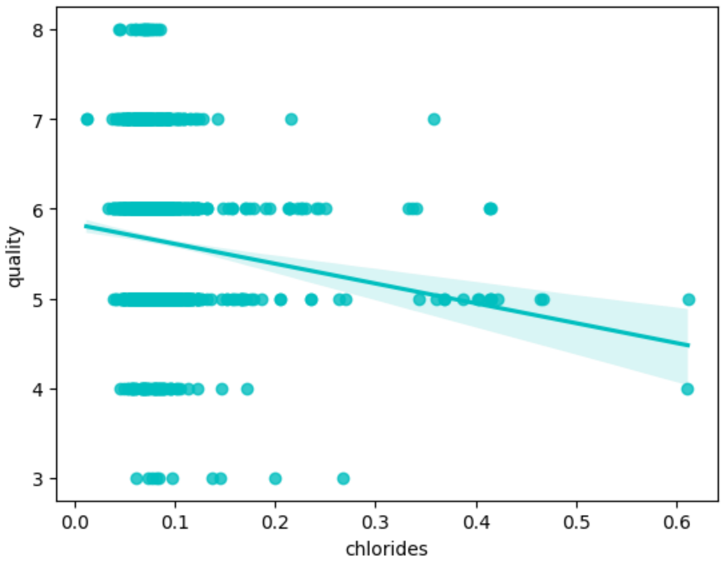
```
sns.regplot(x=df['residual sugar'],y=y,color='y')
```

```
<Axes: xlabel='residual sugar', ylabel='quality'>
```



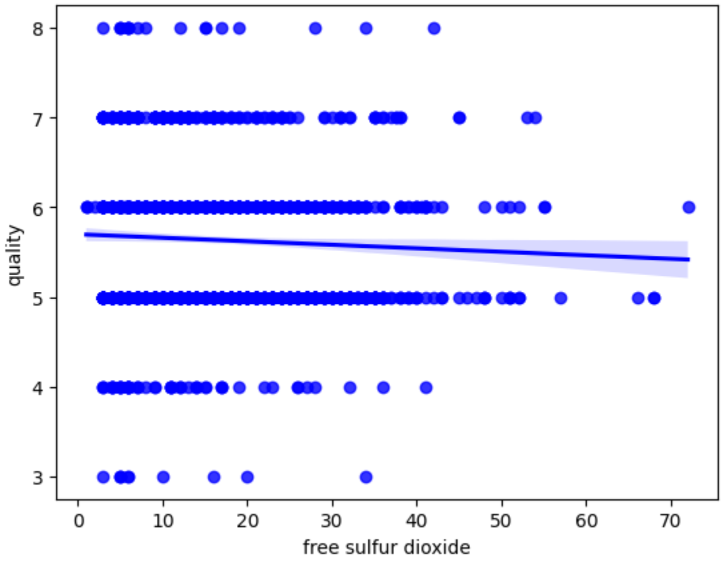
```
sns.regplot(x=df['chlorides'],y=y,color='c')
```

```
<Axes: xlabel='chlorides', ylabel='quality'>
```



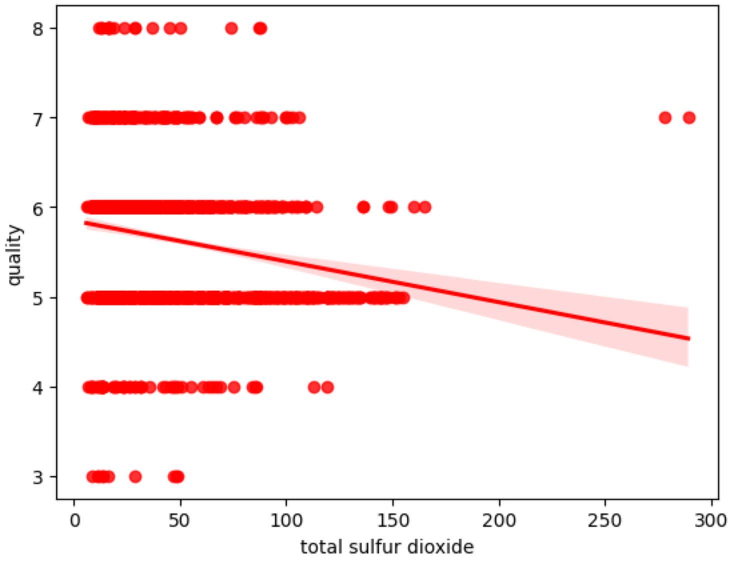
```
sns.regplot(x=df['free sulfur dioxide'],y=y,color='b')
```

```
<Axes: xlabel='free sulfur dioxide', ylabel='quality'>
```



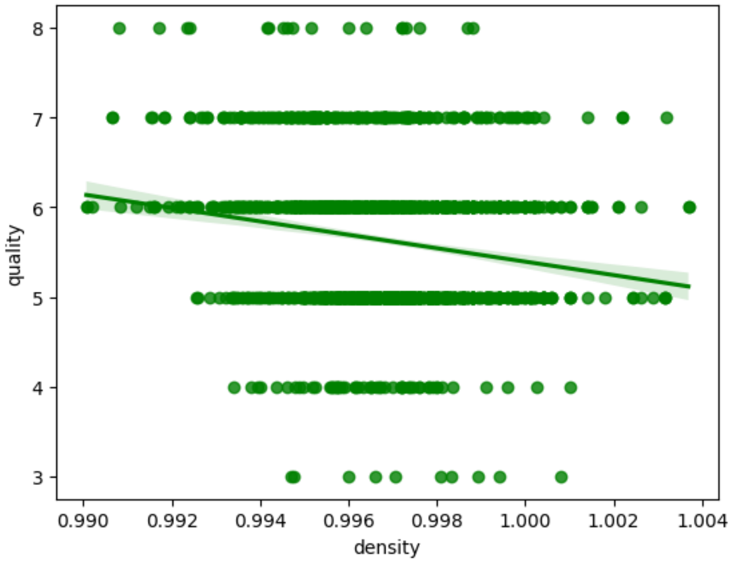
```
sns.regplot(x=df['total sulfur dioxide'],y=y,color='r')
```

```
<Axes: xlabel='total sulfur dioxide', ylabel='quality'>
```



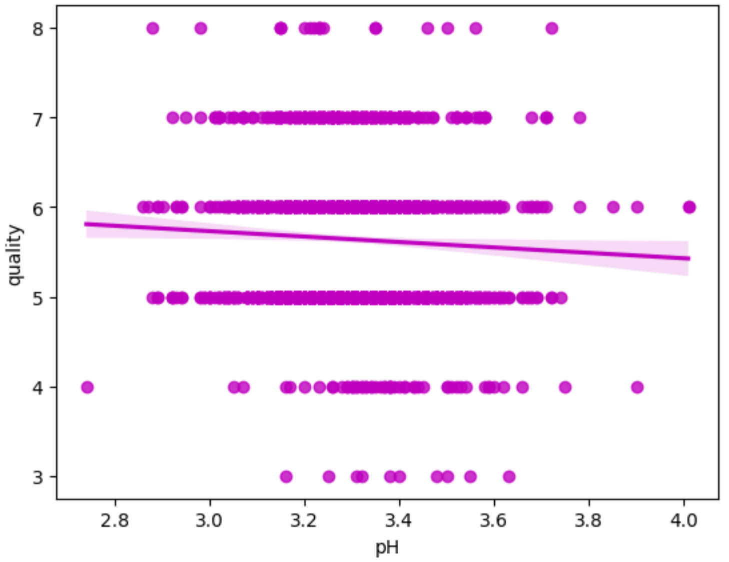
```
sns.regplot(x=df['density'],y=y,color='g')
```

```
<Axes: xlabel='density', ylabel='quality'>
```



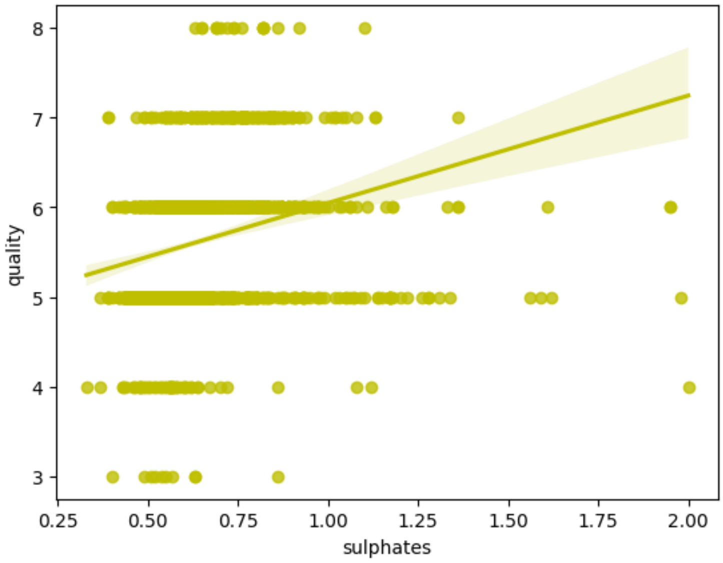
```
sns.regplot(x=df['pH'],y=y,color='m')
```

```
<Axes: xlabel='pH', ylabel='quality'>
```



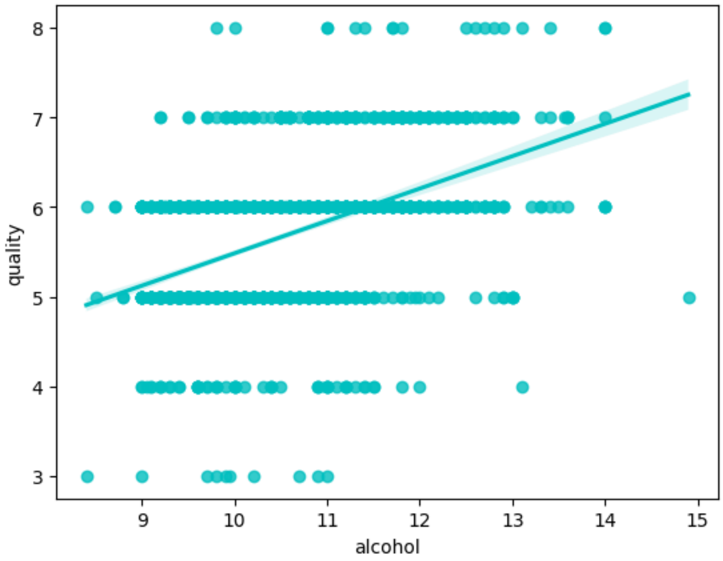
```
sns.regplot(x=df['sulphates'],y=y,color='y')
```

```
<Axes: xlabel='sulphates', ylabel='quality'>
```



```
sns.regplot(x=df['alcohol'],y=y,color='c')
```

```
<Axes: xlabel='alcohol', ylabel='quality'>
```



```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30,random_state=42)
x_train
```

```
x_test
```

```
from sklearn.linear_model import LinearRegression
model=LinearRegression()
model.fit(x_train,y_train)
y_pred=model.predict(x_test)
y_pred
```

```
df1=pd.DataFrame({'Actual_value':y_test,'Predicted_value':y_pred,'Difference':y_test-y_pred})
df1
```

	Actual_value	Predicted_value	Difference
803	6	5.356763	0.643237
124	5	5.090715	-0.090715
350	6	5.625538	0.374462
682	5	5.448861	-0.448861
1326	6	5.744784	0.255216
...	...	...	...
1468	7	5.597986	1.402014
495	8	6.105900	1.894100
1325	6	5.744784	0.255216
514	7	6.101798	0.898202
576	4	5.480730	-1.480730

480 rows × 3 columns

```
print("slope is")
list(zip(x,model.coef_))
```

```
slope is
[('fixed acidity', 0.023470471834254856),
 ('volatile acidity', -1.0996196891580763),
 ('citric acid', -0.24785977563707962),
 ('residual sugar', 0.00773785619809992),
 ('chlorides', -1.6735925141676544),
 ('free sulfur dioxide', 0.004550418153093427),
 ('total sulfur dioxide', -0.0032638916783188643),
 ('density', -14.239556313135678),
 ('pH', -0.31924744442375136),
 ('sulphates', 0.8128247013237453),
 ('alcohol', 0.2919911579408527)]
```

```
print("Contant is",model.intercept_)
```

```
from sklearn.metrics import mean_absolute_error,mean_absolute_percentage_error,mean_squared_error,r2_score
print("MAE is",mean_absolute_error(y_test,y_pred))
print("MAPE is",mean_absolute_percentage_error(y_test,y_pred))
print("MSE is",mean_squared_error(y_test,y_pred))
print("r2_score is", r2_score(y_test,y_pred))
RMSE=np.sqrt(mean_squared_error(y_test,y_pred))
print("RMSE is",RMSE)
```