In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
```

In [2]:
```
sns.set_style("darkgrid")
plt.rcParams["font.size"] = 15
plt.rcParams['figure.figsize'] = (10,7)
plt.rcParams['figure.facecolor'] = '#FFE59A'
```

In [3]:
```
wine_dataset = pd.read_csv("winequality-red.csv")
```

In [4]:
```
wine_dataset.head()
```

Out[4]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sul
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	

In [5]:
```
wine_dataset.shape
```

Out[5]:
```
(1599, 12)
```

In [6]:
```
wine_dataset.head()
```

Out[6]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sul
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	

In [7]:
```
wine_dataset.head()
```

Out[7]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sul
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	

In [8]:
```
wine_dataset.isnull().sum()
```

Out[8]:
```
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
```

In [9]:
```
wine_dataset.describe()
```

Out[9]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free su
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.00
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.87
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.46
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.00
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.00
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.00
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.00
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.00

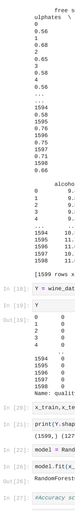
In [10]:
```
plot=plt.figure(figsize=(7,5))
sns.catplot(x = "quality",data = wine_dataset, kind = "count")
```

Out[10]:
```
<seaborn.axisgrid.FacetGrid at 0x22ef6a398e0>
```


In [11]:
```
plot = plt.figure(figsize=(7,5))
sns.barplot(x="quality",y="volatile acidity",data = wine_dataset)
```

Out[11]:
```
<AxesSubplot:xlabel='quality', ylabel='volatile acidity'>
```


In [12]:
```
plot = plt.figure(figsize=(7,5))
sns.barplot(x="quality",y='citric acid', data = wine_dataset)
```

Out[12]:
```
<AxesSubplot:xlabel='quality', ylabel='citric acid'>
```


In [13]:
```
wine_dataset.head()
```

Out[13]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sul
0	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	
1	7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	
2	7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	
3	11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	
4	7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	

In [14]:
```
correlation = wine_dataset.corr()
```

In [15]:
```
plt.figure(figsize=(10,10))
sns.heatmap(correlation,cbar=True,square=True,fmt=".1f",annot=True,annot_kws={'size':8},cmap='Blues')
```

Out[15]:
```
<AxesSubplot:>
```


In [16]:
```
X = wine_dataset.drop('quality',axis=1)
```

In [17]:
```
print(X)
```
```
      fixed acidity  volatile acidity  citric acid  residual sugar  chlo
r  chlorides  \
0               7.4             0.700         0.00             1.
9         0.076
1               7.8             0.880         0.00             2.
6         0.098
2               7.8             0.760         0.04             2.
3         0.092
3              11.2             0.280         0.56             1.
9         0.075
4               7.4             0.700         0.00             1.
9         0.076
...             ...               ...          ...             ...
...         ...
1594            6.2             0.600         0.08             2.
0         0.090
1595            5.9             0.550         0.10             2.
2         0.062
1596            6.3             0.510         0.13             2.
3         0.076
1597            5.9             0.645         0.12             2.
0         0.075
1598            6.0             0.310         0.47             3.
6         0.067

      free sulfur dioxide  total sulfur dioxide  density    pH  s
ulphates  \
0                    11.0                  34.0  0.99780  3.51
0.56
1                    25.0                  67.0  0.99680  3.20
0.68
2                    15.0                  54.0  0.99700  3.26
0.65
3                    17.0                  60.0  0.99800  3.16
0.58
4                    11.0                  34.0  0.99780  3.51
0.56
...                   ...                   ...      ...   ...
...
1594                 32.0                  44.0  0.99490  3.45
0.58
1595                 39.0                  51.0  0.99512  3.52
0.76
1596                 29.0                  40.0  0.99574  3.42
0.75
1597                 32.0                  44.0  0.99547  3.57
0.71
1598                 18.0                  42.0  0.99549  3.39
0.66

      alcohol
0         9.4
1         9.8
2         9.8
3         9.8
4         9.4
...       ...
1594     10.5
1595     11.2
1596     11.0
1597     10.2
1598     11.0

[1599 rows x 11 columns]
```

In [18]:
```
Y = wine_dataset["quality"].apply(lambda y_value: 1 if y_value>=7 else 0)
```

In [19]:
```
Y
```

Out[19]:
```
0       0
1       0
2       0
3       0
4       0
       ..
1594    0
1595    0
1596    0
1597    0
1598    0
Name: quality, Length: 1599, dtype: int64
```

In [20]:
```
x_train,x_test,y_train,y_test = train_test_split(X,Y,test_size=0.2,random_state=3)
```

In [21]:
```
print(Y.shape,y_train.shape,y_test.shape)
```
```
(1599,) (1279,) (320,)
```

In [22]:
```
model = RandomForestClassifier()
```

In [23]:
```
model.fit(x_train,y_train)
```

Out[23]:
```
RandomForestClassifier()
```

In [27]:
```
#Accuracy score
```

In [28]:
```
x_test_prediction = model.predict(x_test)
test_data_accuracy= accuracy_score(x_test_prediction,y_test)
```

In [35]:
```
print("Accuracy: " , test_data_accuracy)
```
```
Accuracy:  0.9375
```

In [37]:
```
input_data = (7.3,0.65,0.0,1.2,0.065,15.0,21.0,0.9946,3.39,0.047,10.0)
```

In [38]:
```
input_data_as_numpy_array =  np.asarray(input_data)
```

In [39]:
```
input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)
```

In [41]:
```
prediction = model.predict(input_data_reshaped)
```
```
C:\Users\kisho\anaconda3\lib\site-packages\sklearn\base.py:450: U
serWarning: X does not have valid feature names, but RandomForest
Classifier was fitted with feature names
  warnings.warn(
```

In [42]:
```
print(prediction)

if (prediction[0]==1):
  print('Good quality wine')

else:
  print('Bad quality wine')
```
```
[1]
Good quality wine
```

In [ ]:
```

```