

```
from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

```
df = pd.read_csv('///content/drive/MyDrive/project 2/kidney_disease.csv')
```

```
df.head()
```

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	p
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	...	
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	...	
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	...	
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	...	
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	...	

5 rows × 26 columns

```
df.dtypes
```

	0
id	int64
age	float64
bp	float64
sg	float64
al	float64
su	float64
rbc	object
pc	object
pcc	object
ba	object
bgr	float64
bu	float64
sc	float64
sod	float64
pot	float64
hemo	float64
pcv	object
wc	object
rc	object
htn	object
dm	object
cad	object
appet	object
pe	object
ane	object
classification	object

dtype: object

df.columns

```
Index(['id', 'age', 'bp', 'sg', 'al', 'su', 'rbc', 'pc', 'pcc', 'ba', 'bgr',
      'bu', 'sc', 'sod', 'pot', 'hemo', 'pcv', 'wc', 'rc', 'htn', 'dm',
      'cad',
      'appet', 'pe', 'ane', 'classification'],
      dtype='object')
```

```
df.isnull().sum()
```

	0
id	0
age	9
bp	12
sg	47
al	46
su	49
rbc	152
pc	65
pcc	4
ba	4
bgr	44
bu	19
sc	17
sod	87
pot	88
hemo	52
pcv	70
wc	105
rc	130
htn	2
dm	2
cad	2
appet	1
pe	1
ane	1
classification	0

dtype: int64

```
missing_percentage = df.isnull().mean() * 100  
threshold = 30
```

```

from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder()
for column in df.select_dtypes(include=['object']).columns:
    df[column] = label_encoder.fit_transform(df[column])
df.head()

```

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm
0	0	48.0	80.0	1.020	1.0	0.0	2	1	0	0	...	32	72	34	1	4
1	1	7.0	50.0	1.020	4.0	0.0	2	1	0	0	...	26	56	49	0	3
2	2	62.0	80.0	1.010	2.0	3.0	1	1	0	0	...	19	70	49	0	4
3	3	48.0	70.0	1.005	4.0	0.0	1	0	1	0	...	20	62	19	1	3
4	4	51.0	80.0	1.010	2.0	0.0	1	1	0	0	...	23	68	27	0	3

5 rows × 26 columns

```

for column in df.select_dtypes(include=['float64']).columns:
    mean_value = df[column].mean()
    df[column] = df[column].fillna(mean_value)

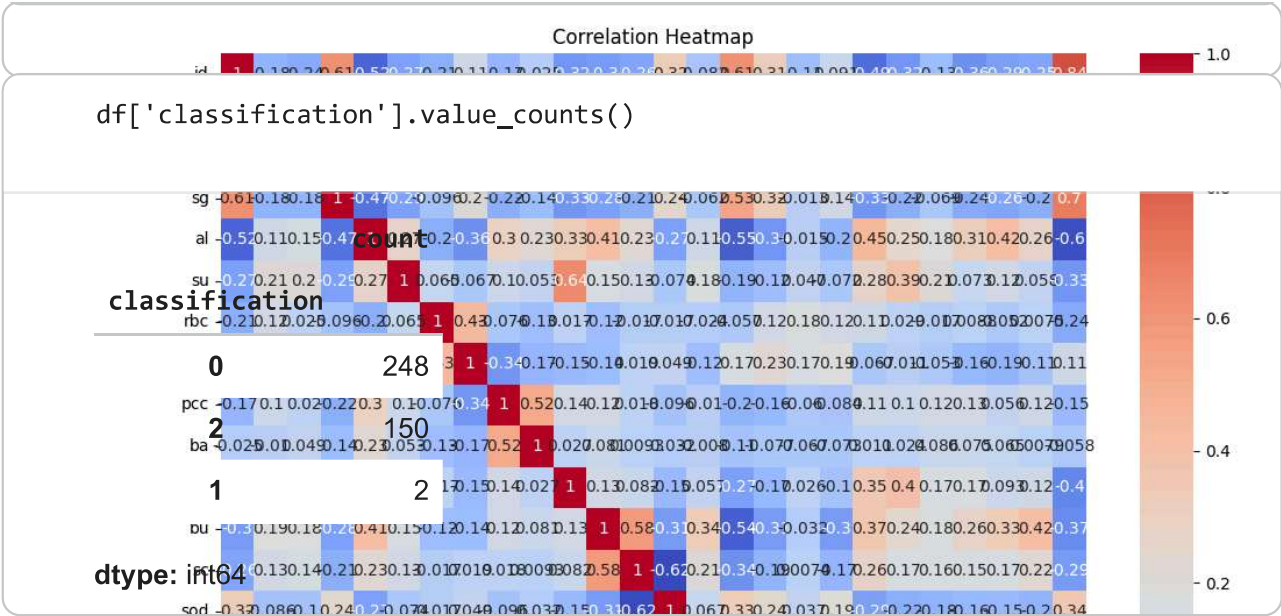
```

```
df.isnull().sum()
```

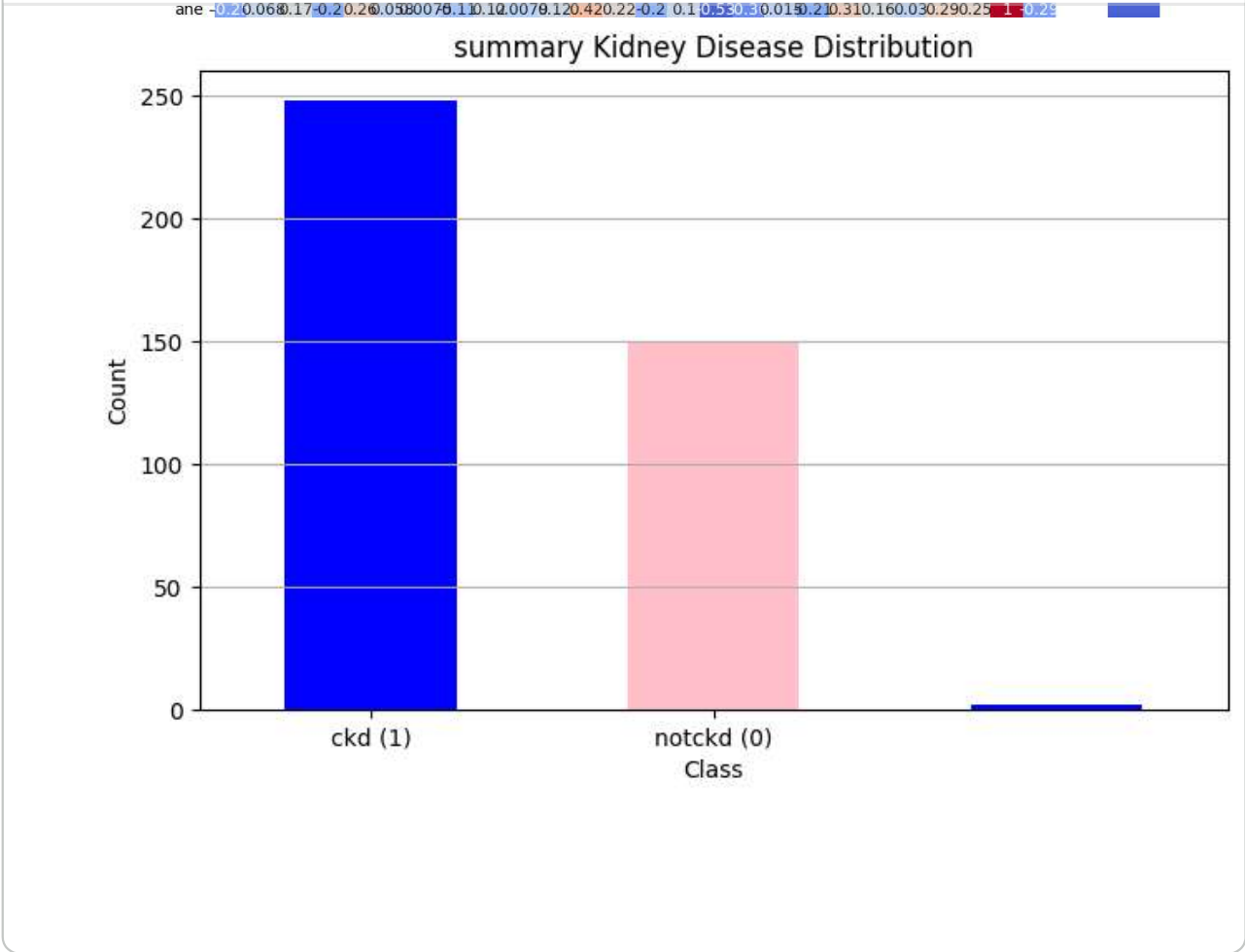
	0
id	0
age	0
bp	0
sg	0
al	0
su	0
rbc	0
pc	0
pcc	0
ba	0
bgr	0
bu	0
sc	0
sod	0
pot	0
hemo	0
pcv	0
wc	0
rc	0
htn	0
dm	0
cad	0
appet	0
pe	0
ane	0
classification	0

dtype: int64

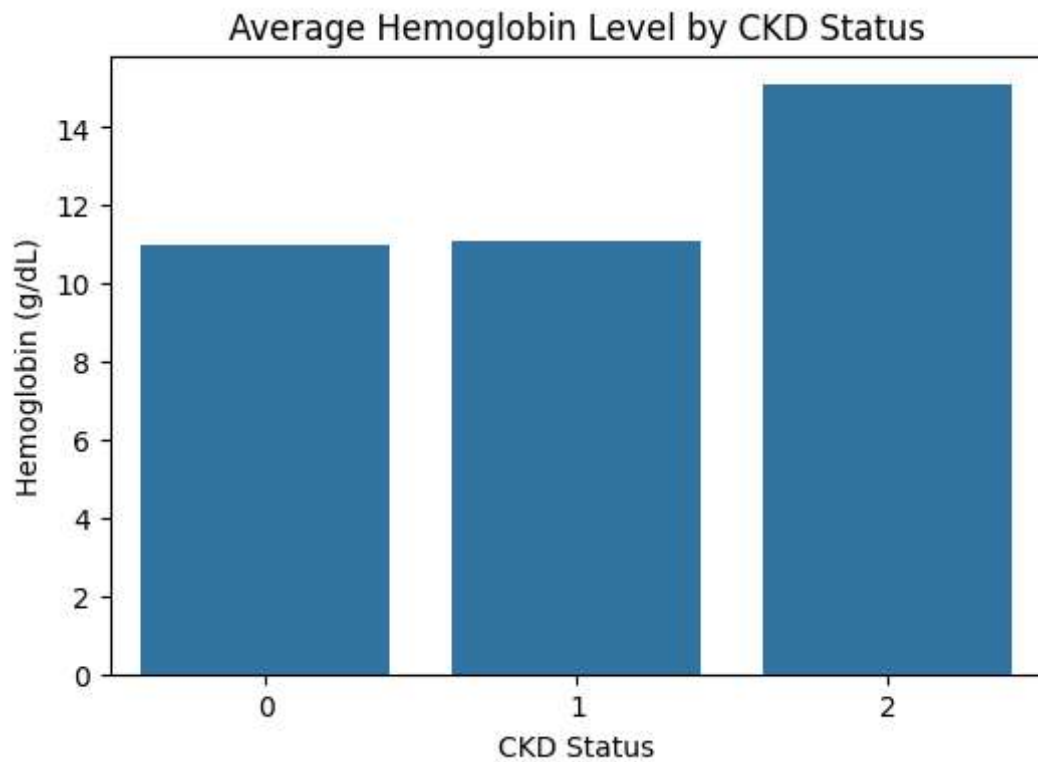
```
plt.figure(figsize=(12, 12))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

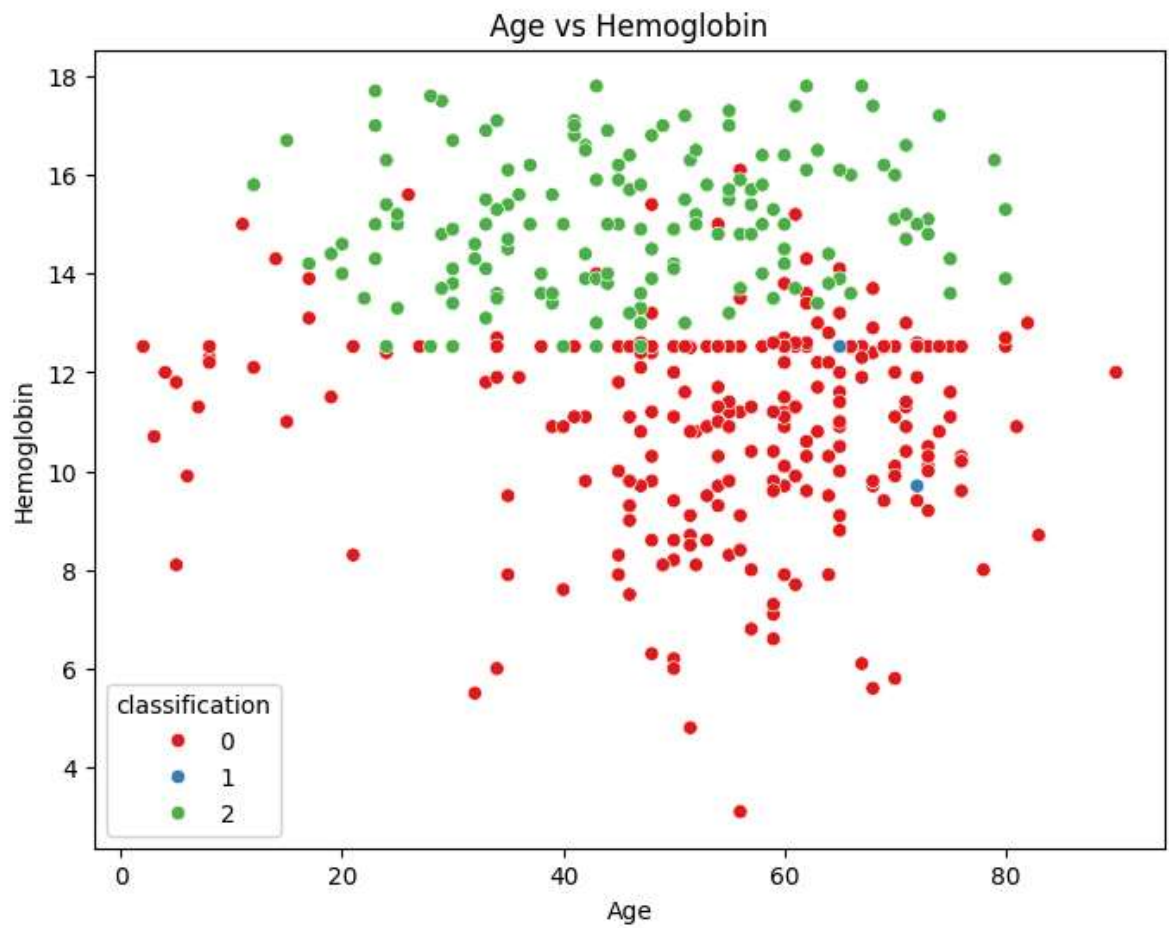
```
plt.figure(figsize=(8, 5))
df['classification'].value_counts().plot(kind='bar', color=['blue',
'pink'])
plt.xticks([0, 1], ['ckd (1)', 'notckd (0)'], rotation=0)
plt.title('summary Kidney Disease Distribution')
plt.xlabel('Class')
plt.ylabel('Count')
plt.grid(axis='y')
plt.show()
```



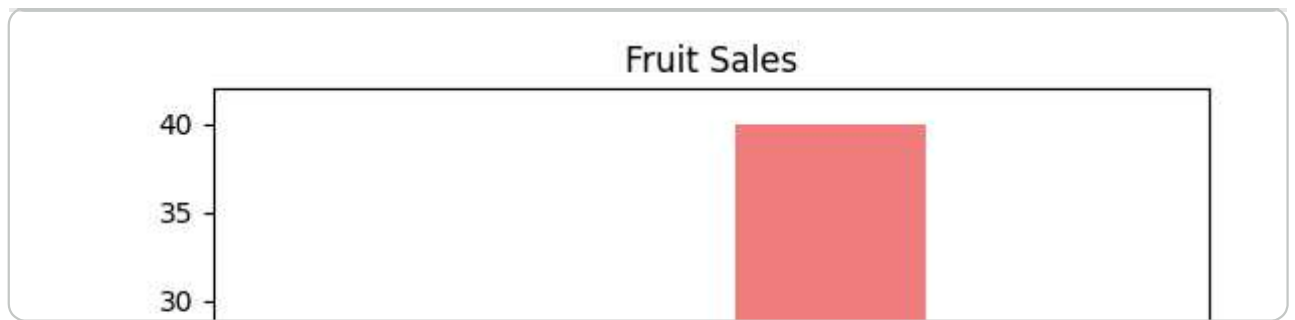
```
import seaborn as sns
plt.figure(figsize=(6,4))
sns.barplot(x='classification', y='hemo', data=df, estimator='mean',
ci=None)
plt.title("Average Hemoglobin Level by CKD Status")
plt.xlabel("CKD Status")
plt.ylabel("Hemoglobin (g/dL)")
plt.show()
```



```
plt.figure(figsize=(8, 6))
sns.scatterplot(x='age', y='hemo', data=df, hue='classification',
palette='Set1')
plt.title('Age vs Hemoglobin')
plt.xlabel('Age')
plt.ylabel('Hemoglobin')
plt.show()
```



```
fruits = ['apple', 'Bananas', 'Cherries', 'Dates']  
sales = [25, 18, 40, 20]  
plt.bar(fruits, sales, color='lightcoral')  
plt.title('Fruit Sales')  
plt.xlabel('Fruit Type')  
plt.ylabel('Quantity Sold')  
plt.show()
```



```
sales_B = [20, 25, 30]
x = np.arange(len(years))
width = 0.35
plt.bar(x - width/2, sales_A, width, label='Product A',
color='orange')
plt.bar(x + width/2, sales_B, width, label='Product B', color='green')
plt.title('Yearly Sales Comparison')
plt.xlabel('Year')
plt.ylabel('Sales')
plt.xticks(x, years)
plt.legend()
plt.show()
```

