Engineer new features and select relevant features for model training.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
data = pd.read_csv("/content/heart.csv")
```

```
data.head()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca
0	52	1	0	125	212	0	1	168	0	1.0	2	2
1	53	1	0	140	203	1	0	155	1	3.1	0	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1
4	62	0	0	138	294	1	1	106	0	1.9	1	3

```
data.tail()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope
1020	59	1	1	140	221	0	1	164	1	0.0	2
1021	60	1	0	125	258	0	0	141	1	2.8	1
1022	47	1	0	110	275	0	0	118	1	1.0	1
1023	50	0	0	110	254	0	0	159	0	0.0	2
1024	54	1	0	120	188	0	1	113	0	1.4	1

```
data.columns.values
```

```
array(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg',
       'thalach', 'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'],
      dtype=object)
```

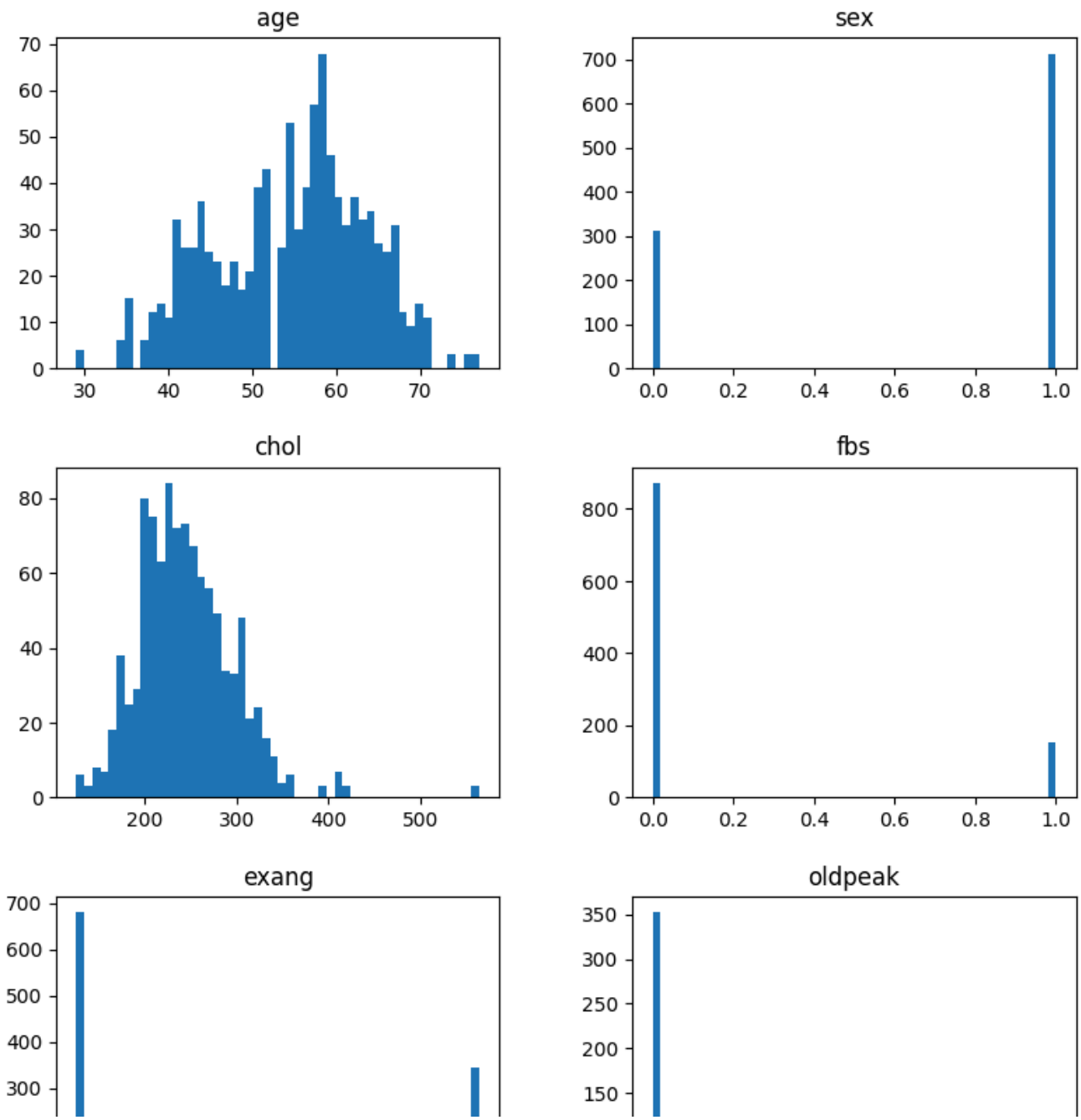
```
data.isna().sum()
```

```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   age       1025 non-null   int64  
 1   sex       1025 non-null   int64  
 2   cp        1025 non-null   int64  
 3   trestbps  1025 non-null   int64  
 4   chol      1025 non-null   int64  
 5   fbs       1025 non-null   int64  
 6   restecg   1025 non-null   int64  
 7   thalach   1025 non-null   int64  
 8   exang     1025 non-null   int64  
 9   oldpeak   1025 non-null   float64
 10  slope     1025 non-null   int64  
 11  ca        1025 non-null   int64  
 12  thal      1025 non-null   int64  
 13  target    1025 non-null   int64  
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
```

```
data.hist(bins = 50, grid = False, figsize=(20,15));
```



```
data.describe()
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	ex
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.000000	1025.000000	1025.000
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.529756	149.114146	0.33
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.527878	23.005724	0.47
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.000000	71.000000	0.000
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.000000	132.000000	0.000
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.000000	152.000000	0.000
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.000000	166.000000	1.000
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.000000	202.000000	1.000

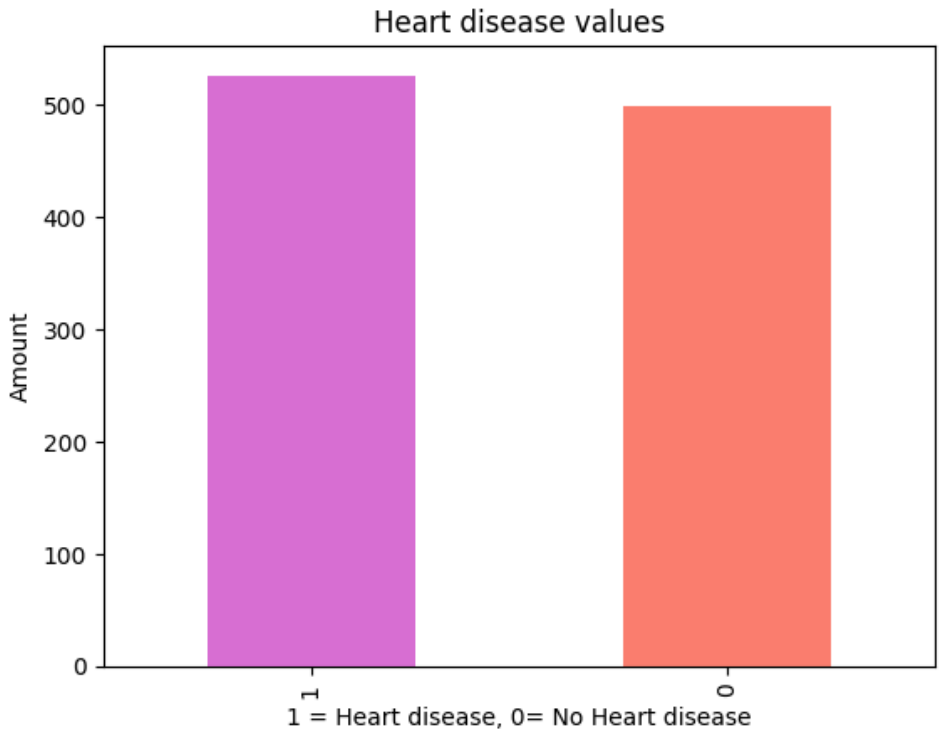
```
questions =["1. How many have heart disease and how many people doesn't have haert disesase? ",
                      "2. People of which sex has most heart disease?",
                      "3. People of which sex has which type of chest pain most?",
                      "4. People with chest pain are most pron to have heart disease?"]
questions
```

```
["1. How many have heart disease and how many people doesn't have haert disesase? ",
 '2. People of which sex has most heart disease?',
 '3. People of which sex has which type of chest pain most?',
 '4. People with chest pain are most pron to have heart disease?']
```

```
# 1. How many have heart disease and how many people doesn't have haert disesase?
data.target.value_counts()
```

```
target
1    526
0    499
Name: count, dtype: int64
```

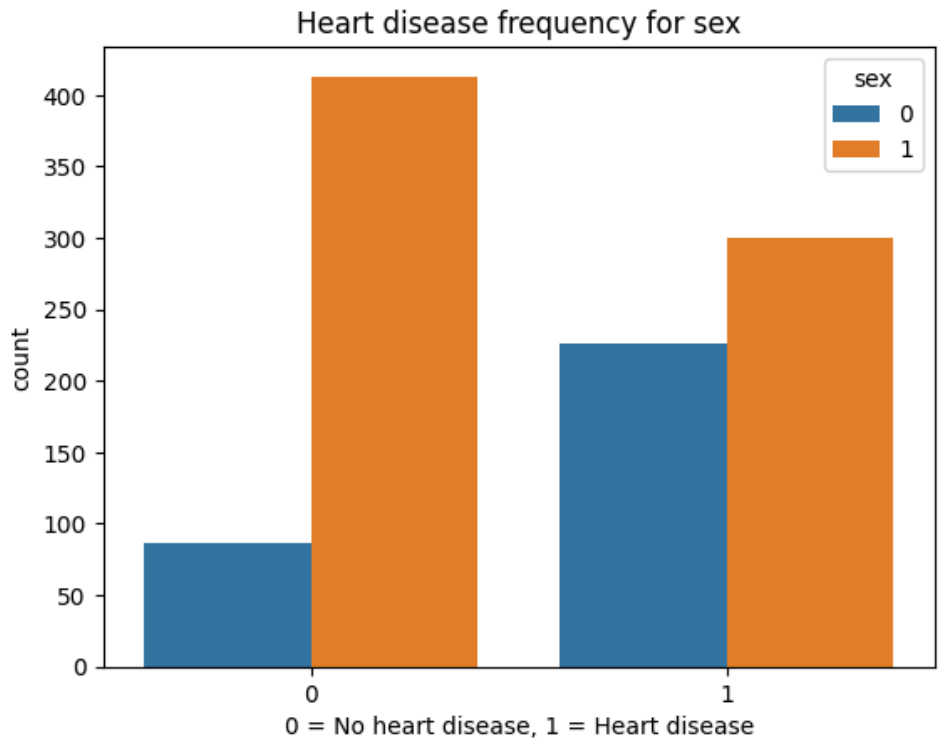
```
data.target.value_counts().plot(kind = "bar", color=["orchid","salmon"])
plt.title("Heart disease values")
plt.xlabel("1 = Heart disease, 0= No Heart disease")
plt.ylabel("Amount");
```




```
#2. People of which sex has most heart disease?
pd.crosstab(data.target,data.sex)
```

target	sex 0	1
0	86	413
1	226	300

```
sns.countplot(x= "target", data=data, hue= "sex")
plt.title("Heart disease frequency for sex")
plt.xlabel("0 = No heart disease, 1 = Heart disease");
```

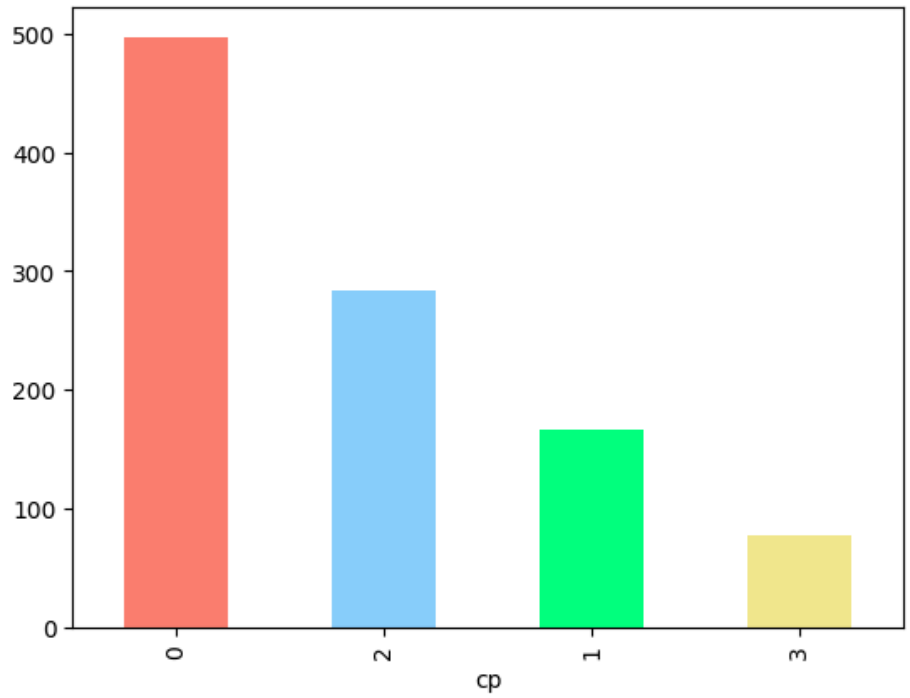


```
#3. People of which sex has which type of chest pain most?
data.cp.value_counts()
```

```
cp
0    497
2    284
1    167
3     77
Name: count, dtype: int64
```

```
data.cp.value_counts().plot(kind = "bar",color = ["salmon", "lightskyblue", "springgreen","khaki"]
```

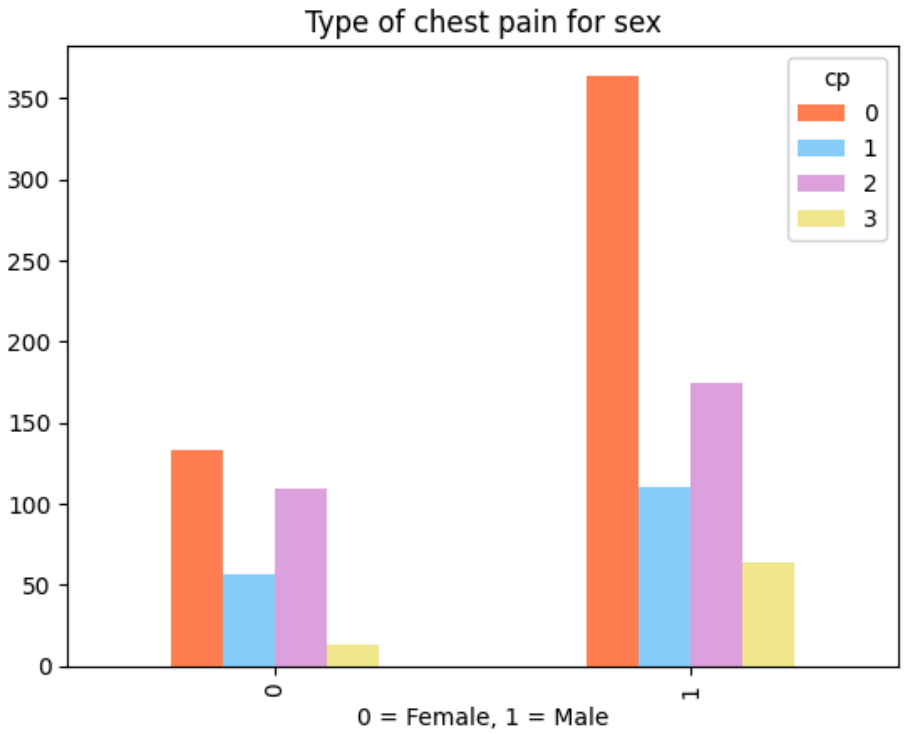
```
<Axes: xlabel='cp'>
```




```
pd.crosstab(data.sex,data.cp)
```

sex \ cp	0	1	2	3
0	133	57	109	13
1	364	110	175	64

```
pd.crosstab(data.sex,data.cp).plot(kind= "bar", color = ["coral","lightskyblue","plum","khaki"])
plt.title("Type of chest pain for sex")
plt.xlabel("0 = Female, 1 = Male");
```

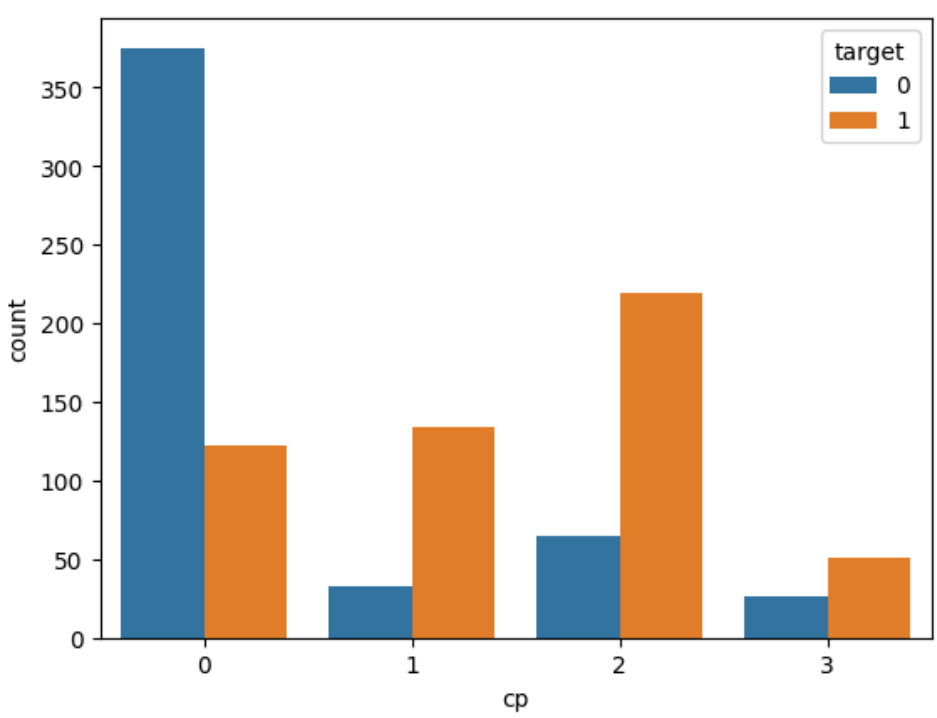


#4. People with chest pain are most pron to have heart disease?

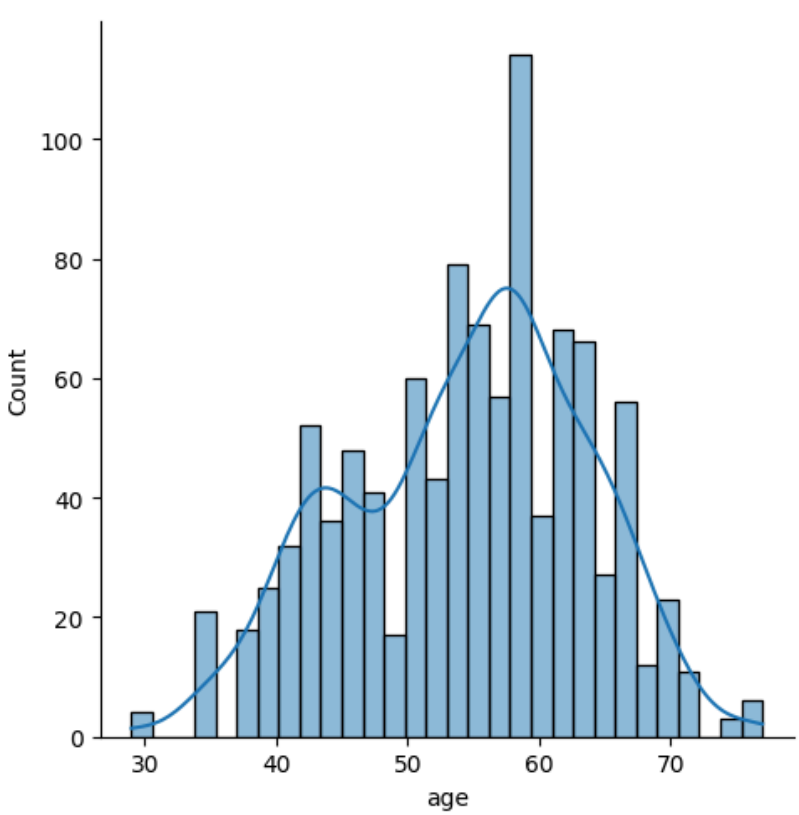
```
pd.crosstab(data.cp,data.target)
```

target	0	1
cp		
0	375	122
1	33	134
2	65	219
3	26	51

```
sns.countplot(x="cp", data = data, hue= "target");
```



```
sns.displot(x="age", data = data, bins = 30, kde= True);
```




```
sns.displot(x="thalach", data = data, bins = 30, kde = True, color = "chocolate");
```