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

import warnings
warnings.filterwarnings('ignore')

%matplotlib inline


df = pd.read_csv('C:/Users/saswa/OneDrive/Desktop/Pinaki_Bank_Marketing/bank-additional/bank-additional/
df.rename(columns={'y':'deposit'}, inplace=True)
df.head()
```

	age	job	marital	education	default	housing	loan	contact	month
0	30	blue-collar	married	basic.9y	no	yes	no	cellular	may
1	39	services	single	high.school	no	no	no	telephone	may
2	25	services	married	high.school	no	yes	no	telephone	jun
3	38	services	married	basic.9y	no	unknown	unknown	telephone	jun
4	47	admin.	married	university.degree	no	yes	no	cellular	nov

5 rows × 21 columns

```
df.head()
```

	age	job	marital	education	default	housing	loan	contact	month
0	30	blue-collar	married	basic.9y	no	yes	no	cellular	may
1	39	services	single	high.school	no	no	no	telephone	may
2	25	services	married	high.school	no	yes	no	telephone	jun
3	38	services	married	basic.9y	no	unknown	unknown	telephone	jun
4	47	admin.	married	university.degree	no	yes	no	cellular	nov

5 rows × 21 columns

```
df.tail()
```

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...
4114	30	admin.	married	basic.6y	no	yes	yes	cellular	jul	thu	...
4115	39	admin.	married	high.school	no	yes	no	telephone	jul	fri	...
4116	27	student	single	high.school	no	no	no	cellular	may	mon	...
4117	58	admin.	married	high.school	no	no	no	cellular	aug	fri	...
4118	34	management	single	high.school	no	yes	no	cellular	nov	wed	...

5 rows × 21 columns

```
df.shape
```

```
(4119, 21)
```

```
df.columns
```

```
Index(['age', 'job', 'marital', 'education', 'default', 'housing', 'loan',
       'contact', 'month', 'day_of_week', 'duration', 'campaign', 'pdays',
       'previous', 'poutcome', 'emp.var.rate', 'cons.price.idx',
       'cons.conf.idx', 'euribor3m', 'nr.employed', 'deposit'],
      dtype='object')
```

```
df.dtypes
```

```
age                int64
job               object
marital           object
education         object
default           object
housing           object
loan              object
contact           object
month             object
day_of_week       object
duration           int64
campaign           int64
pdays              int64
previous           int64
poutcome          object
emp.var.rate     float64
cons.price.idx   float64
cons.conf.idx    float64
euribor3m        float64
nr.employed      float64
deposit           object
dtype: object
```

```
df.dtypes.value_counts()
```

```
object     11
int64       5
float64     5
dtype: int64
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4119 entries, 0 to 4118
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   age             4119 non-null   int64  
 1   job             4119 non-null   object 
 2   marital         4119 non-null   object 
 3   education       4119 non-null   object 
 4   default         4119 non-null   object 
 5   housing         4119 non-null   object 
 6   loan            4119 non-null   object 
 7   contact         4119 non-null   object 
 8   month           4119 non-null   object 
 9   day_of_week     4119 non-null   object 
 10  duration        4119 non-null   int64  
 11  campaign        4119 non-null   int64  
 12  pdays           4119 non-null   int64  
 13  previous        4119 non-null   int64
```

```
 14  poutcome        4119 non-null   object
 15  emp.var.rate    4119 non-null   float64
 16  cons.price.idx  4119 non-null   float64
 17  cons.conf.idx   4119 non-null   float64
 18  euribor3m       4119 non-null   float64
 19  nr.employed     4119 non-null   float64
 20  deposit         4119 non-null   object
dtypes: float64(5), int64(5), object(11)
memory usage: 675.9+ KB
```

```
df.duplicated().sum()
```

```
0
```

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

```
age               0
job               0
marital           0
education         0
default           0
housing           0
loan              0
contact           0
month             0
day_of_week       0
duration          0
campaign          0
pdays             0
previous          0
poutcome          0
emp.var.rate      0
cons.price.idx    0
cons.conf.idx     0
euribor3m         0
nr.employed       0
deposit           0
dtype: int64
```

```
cat_cols = df.select_dtypes(include='object').columns
print(cat_cols)

num_cols = df.select_dtypes(exclude='object').columns
print(num_cols)
```

```
Index(['job', 'marital', 'education', 'default', 'housing', 'loan', 'contact',
       'month', 'day_of_week', 'poutcome', 'deposit'],
      dtype='object')
Index(['age', 'duration', 'campaign', 'pdays', 'previous', 'emp.var.rate',
       'cons.price.idx', 'cons.conf.idx', 'euribor3m', 'nr.employed'],
      dtype='object')
```

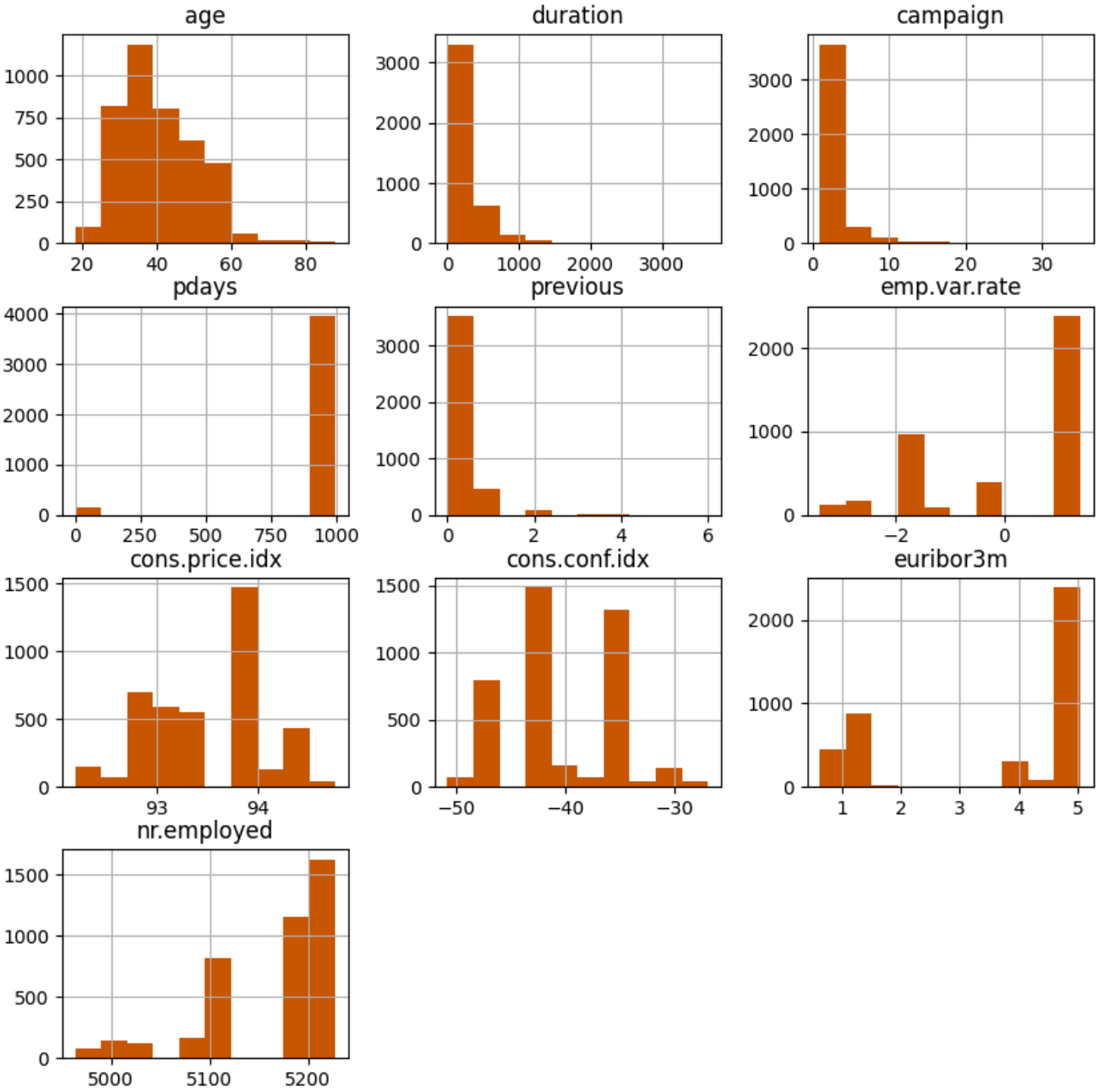
```
df.describe()
```

	age	duration	campaign	pdays	previous	emp.var.rate	cons.price.idx	c
count	4119.000000	4119.000000	4119.000000	4119.000000	4119.000000	4119.000000	4119.000000	
mean	40.113620	256.788055	2.537266	960.422190	0.190337	0.084972	93.579704	
std	10.313362	254.703736	2.568159	191.922786	0.541788	1.563114	0.579349	
min	18.000000	0.000000	1.000000	0.000000	0.000000	-3.400000	92.201000	
25%	32.000000	103.000000	1.000000	999.000000	0.000000	-1.800000	93.075000	
50%	38.000000	181.000000	2.000000	999.000000	0.000000	1.100000	93.749000	
75%	47.000000	317.000000	3.000000	999.000000	0.000000	1.400000	93.994000	
max	88.000000	3643.000000	35.000000	999.000000	6.000000	1.400000	94.767000	

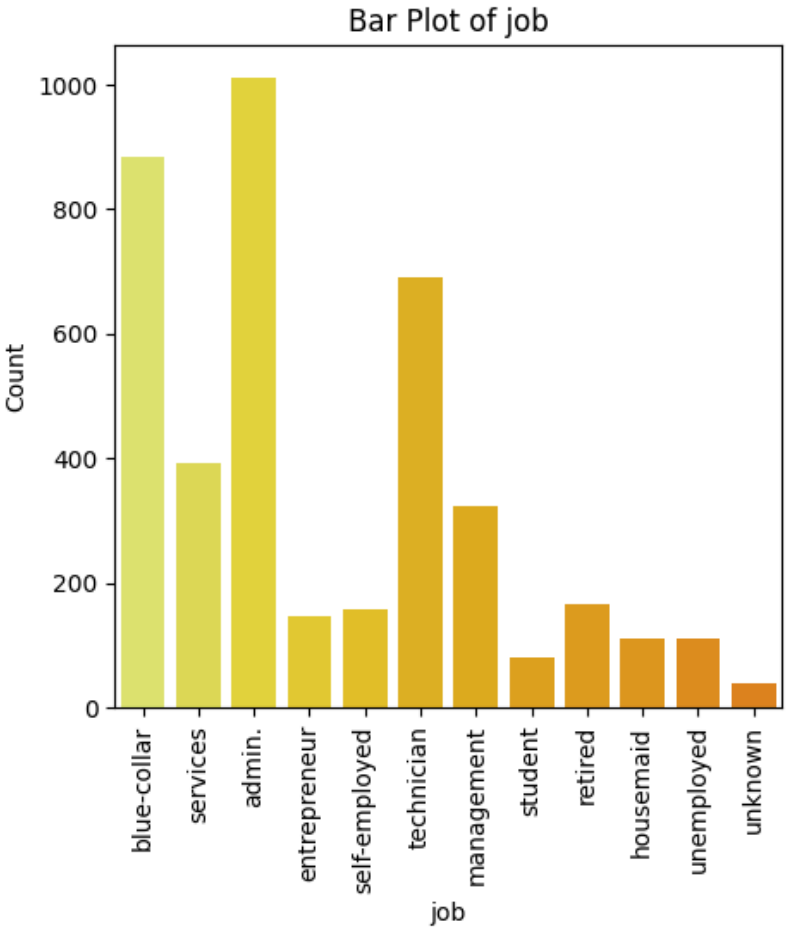
```
df.describe(include='object')
```

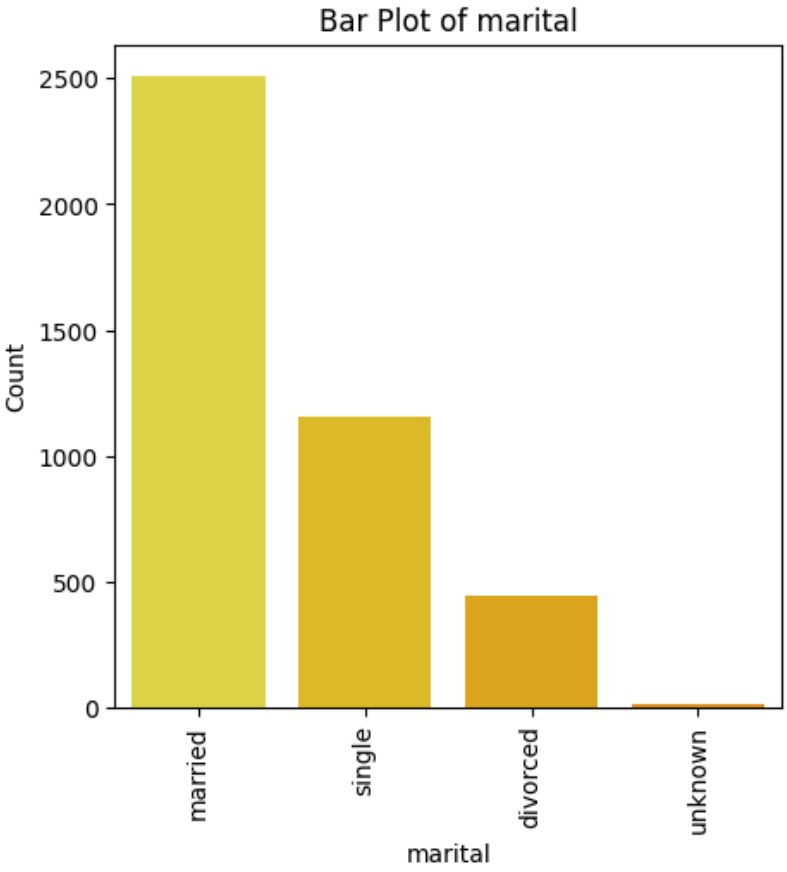
	job	marital	education	default	housing	loan	contact	month	day_of_week	poutco
count	4119	4119	4119	4119	4119	4119	4119	4119	4119	41
unique	12	4	8	3	3	3	2	10	5	
top	admin.	married	university.degree	no	yes	no	cellular	may	thu	nonexiste
freq	1012	2509	1264	3315	2175	3349	2652	1378	860	35

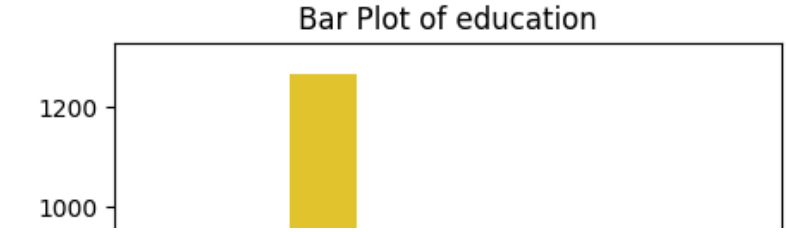
```
df.hist(figsize=(10,10),color='#cc5500')
plt.show()
```

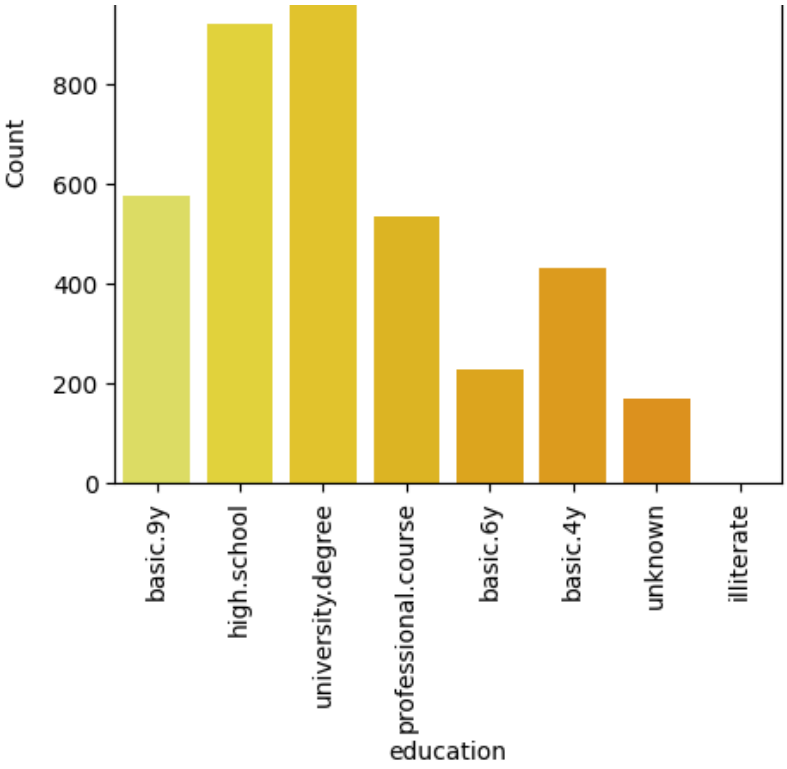


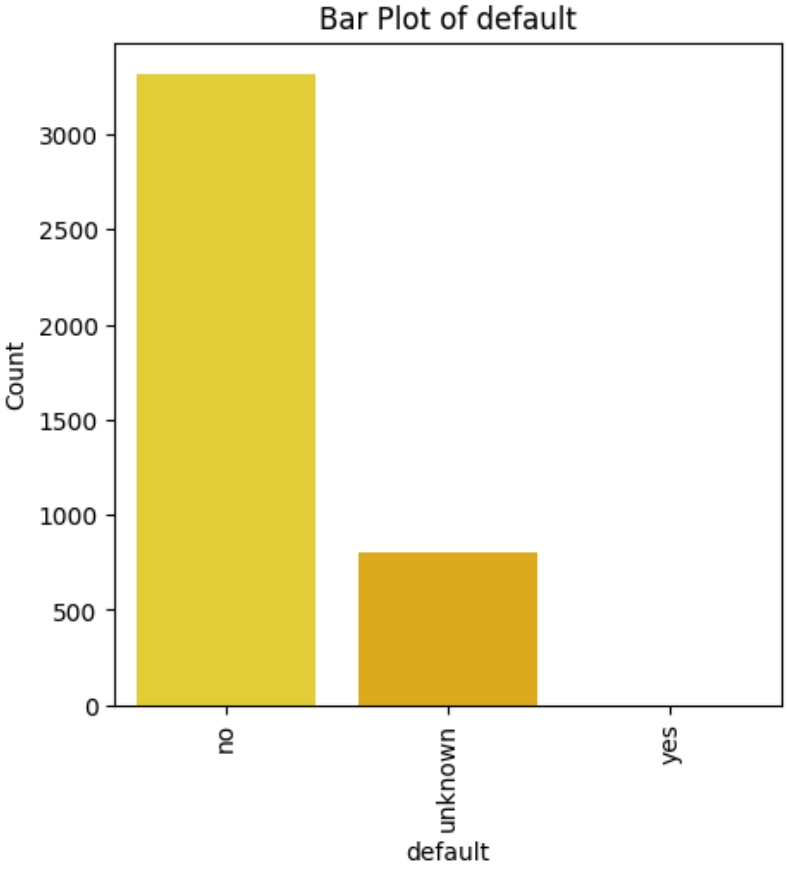

```
for feature in cat_cols:
    plt.figure(figsize=(5,5))  # Adjust the figure size as needed
    sns.countplot(x=feature, data=df, palette='Wistia')
    plt.title(f'Bar Plot of {feature}')
    plt.xlabel(feature)
    plt.ylabel('Count')
    plt.xticks(rotation=90)
    plt.show()
```

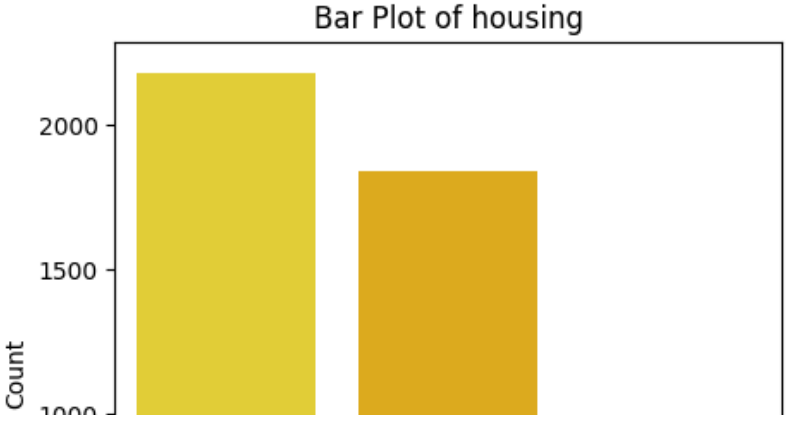

Bar Plot of job
Count
1000
800
600
400
200
0
blue-collar
services
admin.
entrepreneur
self-employed
technician
management
student
retired
housemaid
unemployed
unknown
job

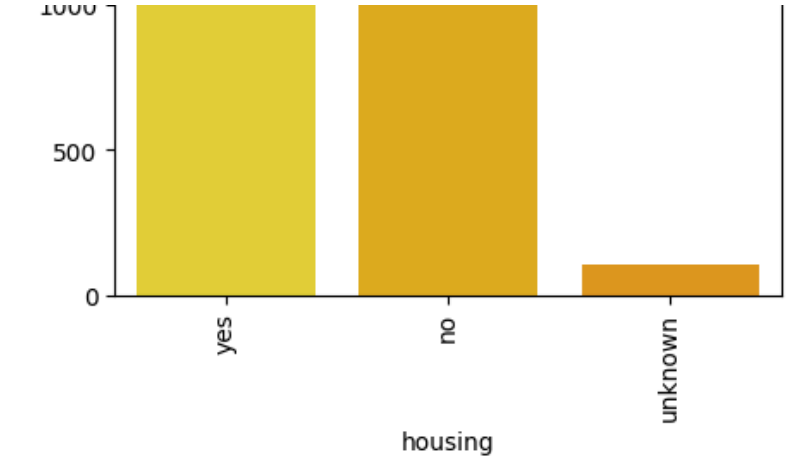

Bar Plot of marital
Count
2500
2000
1500
1000
500
0
married
single
divorced
unknown
marital

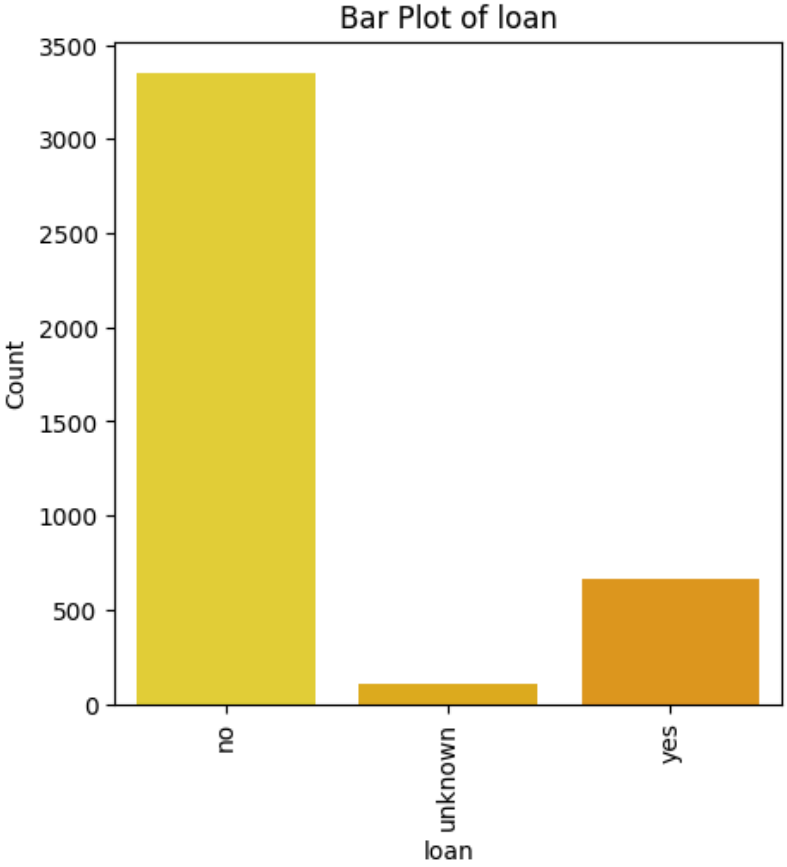

Bar Plot of education
1200
1000

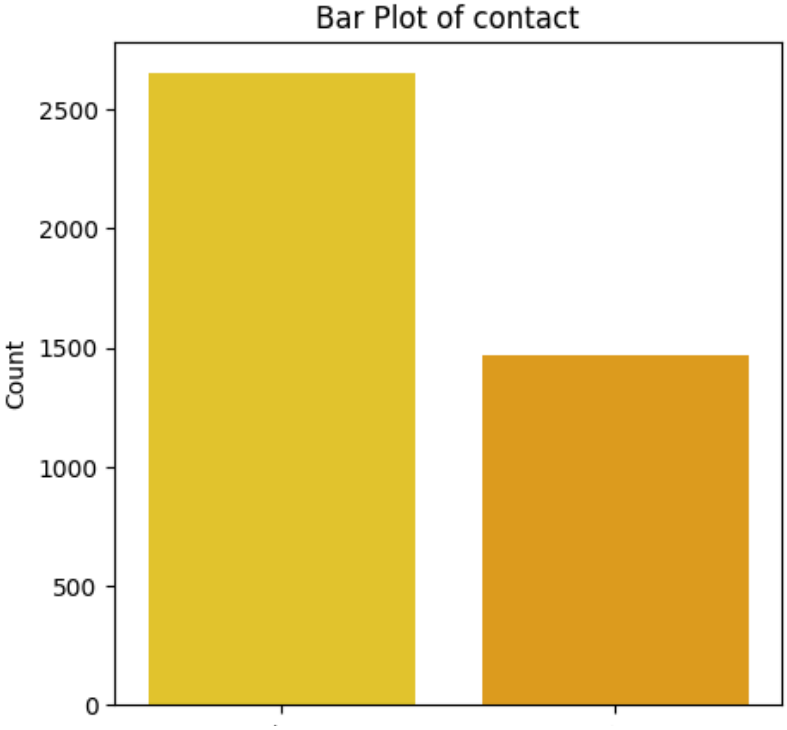

Count
800
600
400
200
0
basic.9y
high.school
university.degree
professional.course
basic.6y
basic.4y
unknown
illiterate
education


Bar Plot of default
Count
3000
2500
2000
1500
1000
500
0
no
unknown
yes
default

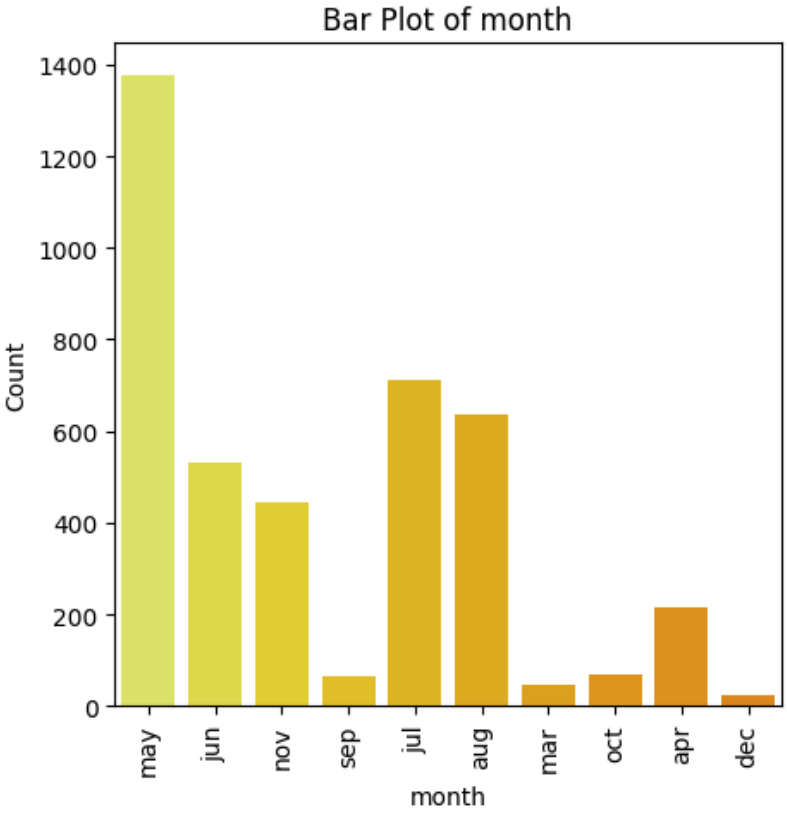

Bar Plot of housing
Count
2000
1500

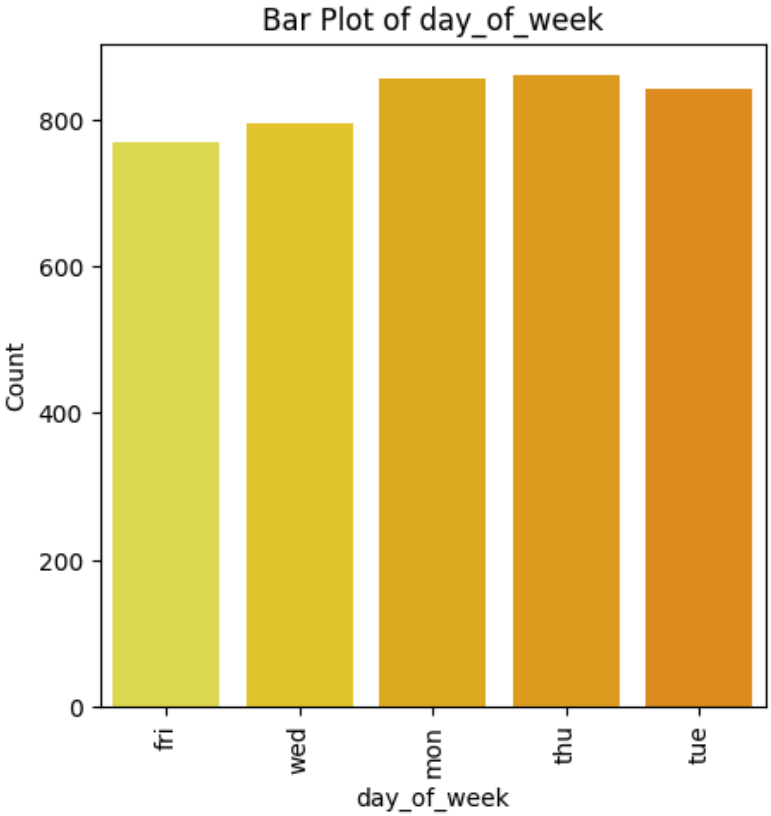

1000
500
0
yes
no
unknown
housing

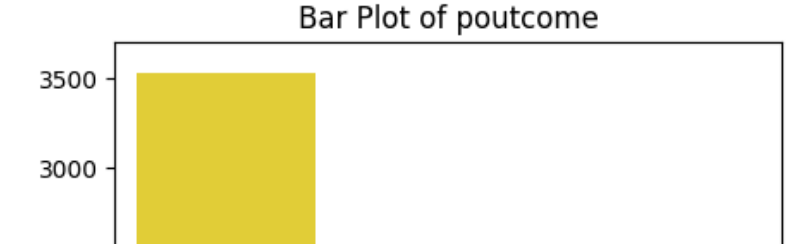

Bar Plot of loan
3500
3000
2500
2000
1500
1000
500
0
Count
no
unknown
yes
loan

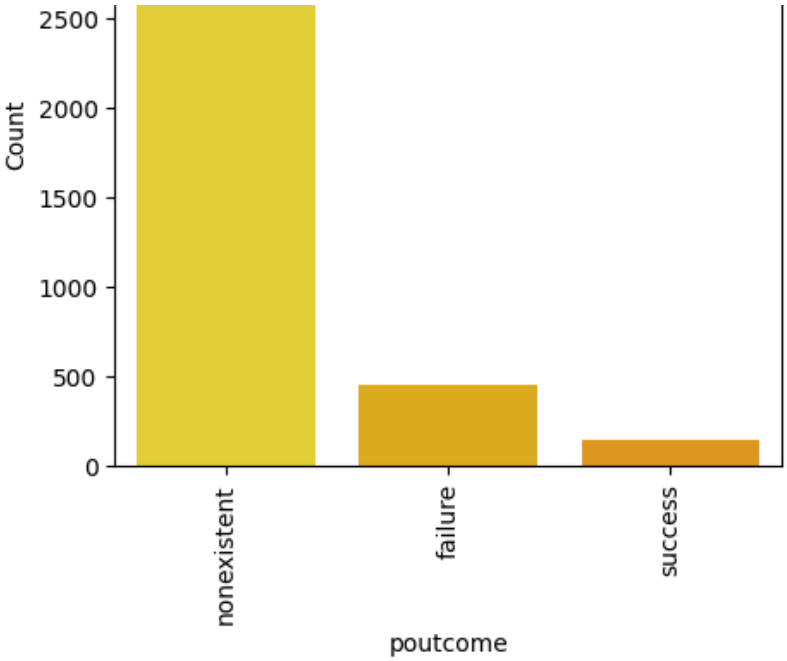

Bar Plot of contact
2500
2000
1500
1000
500
0
Count

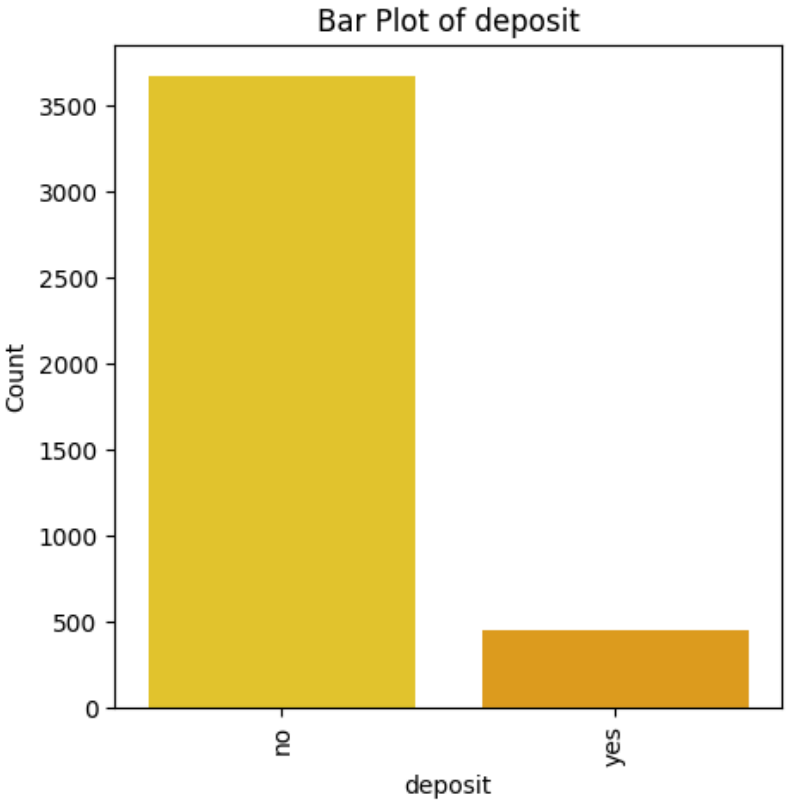

cellular
telephone
contact


Bar Plot of month
Count
month
may
jun
nov
sep
jul
aug
mar
oct
apr
dec

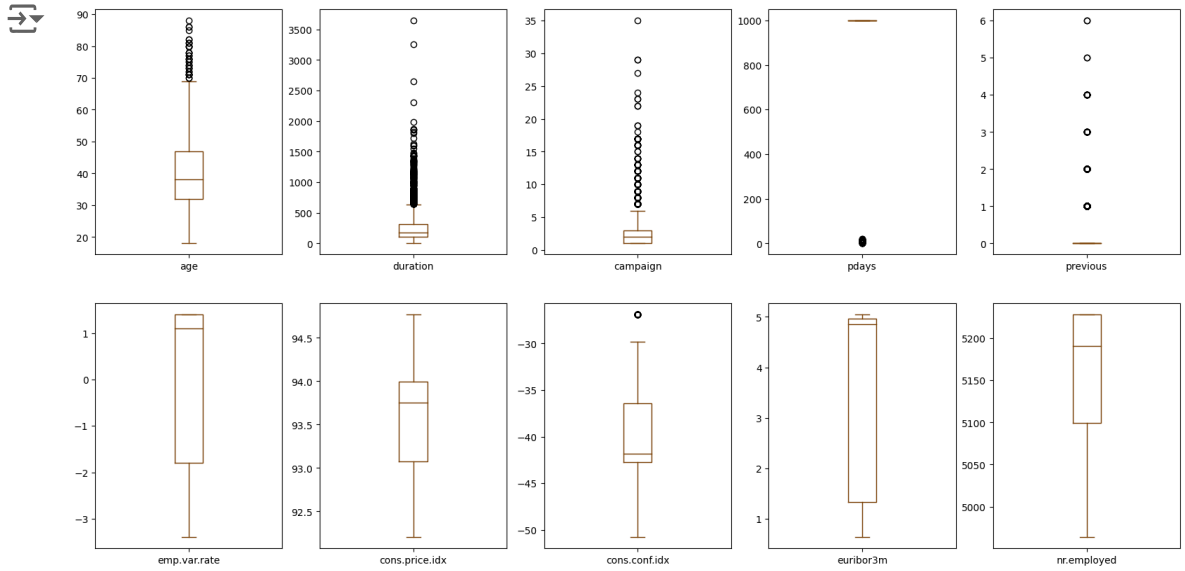

Bar Plot of day_of_week
Count
day_of_week
fri
wed
mon
thu
tue


Bar Plot of poutcome



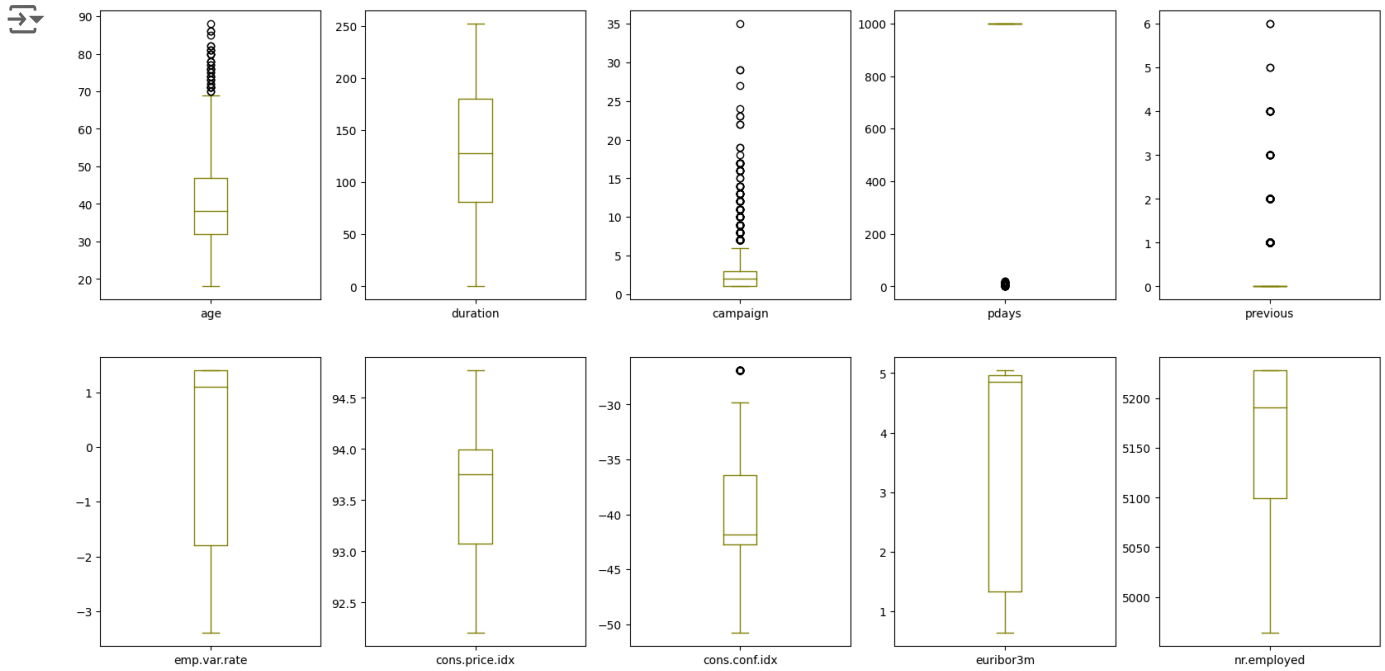



```
df.plot(kind='box', subplots=True, layout=(2,5),figsize=(20,10),color='#7b3f00')
plt.show()
```




```
column = df[['age','campaign','duration']]
q1 = np.percentile(column, 25)
q3 = np.percentile(column, 75)
iqr = q3 - q1
lower_bound = q1 - 1.5 * iqr
upper_bound = q3 + 1.5 * iqr
df[['age','campaign','duration']] = column[(column > lower_bound) & (column < upper_bound)]
```

```
df.plot(kind='box', subplots=True, layout=(2,5),figsize=(20,10),color='#808000')
plt.show()
```




```
corr = df.corr()
print(corr)
corr = corr[abs(corr)>=0.90]
sns.heatmap(corr,annot=True,cmap='Set3',linewidths=0.2)
```

```
plt.show()
```

```
                     age  duration  campaign     pdays  previous  \
age             1.000000  0.014048 -0.014169 -0.043425  0.050931
duration        0.014048  1.000000 -0.218111 -0.093694  0.094206
campaign       -0.014169 -0.218111  1.000000  0.058742 -0.091490
pdays          -0.043425 -0.093694  0.058742  1.000000 -0.587941
previous        0.050931  0.094206 -0.091490 -0.587941  1.000000
emp.var.rate   -0.019192 -0.063870  0.176079  0.270684 -0.415238
cons.price.idx -0.000482 -0.013338  0.145021  0.058472 -0.164922
cons.conf.idx   0.098135  0.045889  0.007882 -0.092090 -0.051420
euribor3m      -0.015033 -0.067815  0.159435  0.301478 -0.458851
nr.employed    -0.041936 -0.097339  0.161037  0.381983 -0.514853

                emp.var.rate  cons.price.idx  cons.conf.idx  euribor3m  \
age                -0.019192       -0.000482       0.098135  -0.015033
duration           -0.063870       -0.013338       0.045889  -0.067815
campaign            0.176079        0.145021       0.007882   0.159435
pdays               0.270684        0.058472      -0.092090   0.301478
previous           -0.415238       -0.164922      -0.051420  -0.458851
emp.var.rate        1.000000        0.755155       0.195022   0.970308
cons.price.idx      0.755155        1.000000       0.045835   0.657159
cons.conf.idx       0.195022        0.045835       1.000000   0.276595
euribor3m           0.970308        0.657159       0.276595   1.000000
nr.employed         0.897173        0.472560       0.107054   0.942589

                nr.employed
age               -0.041936
duration          -0.097339
campaign           0.161037
pdays              0.381983
previous          -0.514853
emp.var.rate       0.897173
cons.price.idx     0.472560
cons.conf.idx      0.107054
euribor3m          0.942589
nr.employed        1.000000
```


```
high_corr_cols = ['emp.var.rate','euribor3m','nr.employed']
```