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

import warnings
warnings.filterwarnings('ignore')

%matplotlib inline
```

```
from google.colab import files
uploaded=files.upload()
```

Choose Files bank-additional.csv

- **bank-additional.csv**(text/csv) - 583898 bytes, last modified: 5/20/2024 - 100% done

```
Saving bank-additional.csv to bank-additional (1).csv
```

```
df=pd.read_csv('bank-additional.csv',delimiter=';')
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	d
4114	30	admin.	married	basic.6y	no	yes	yes	cellular	jul	
4115	39	admin.	married	high.school	no	yes	no	telephone	jul	
4116	27	student	single	high.school	no	no	no	cellular	may	
4117	58	admin.	married	high.school	no	no	no	cellular	aug	
4118	34	management	single	high.school	no	yes	no	cellular	nov	

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
Name: count, 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')
```

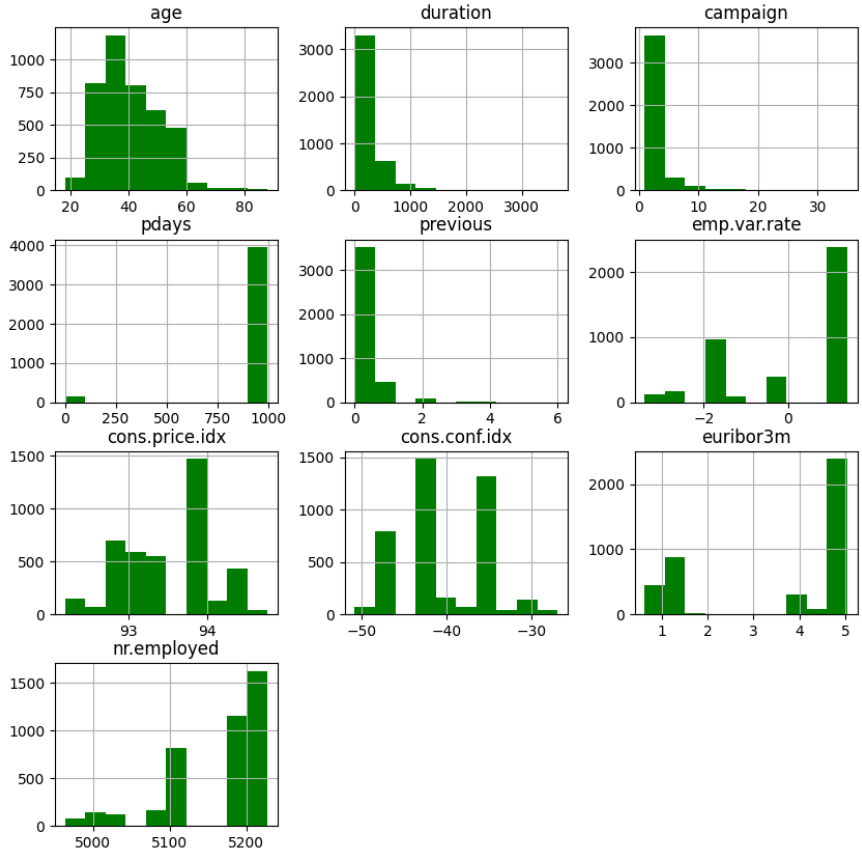
```
df.describe()
```

	age	duration	campaign	pdays	previous	emp.var.rate	co
count	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	
std	10.313362	254.703736	2.568159	191.922786	0.541788	1.563114	
min	18.000000	0.000000	1.000000	0.000000	0.000000	-3.400000	
25%	32.000000	103.000000	1.000000	999.000000	0.000000	-1.800000	
50%	38.000000	181.000000	2.000000	999.000000	0.000000	1.100000	
75%	47.000000	317.000000	3.000000	999.000000	0.000000	1.400000	
max	88.000000	3643.000000	35.000000	999.000000	6.000000	1.400000	

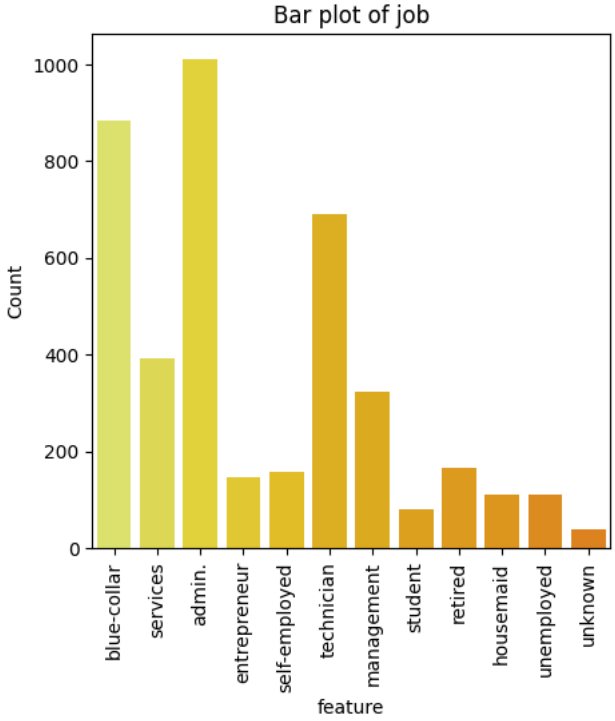
```
df.describe(include='object')
```

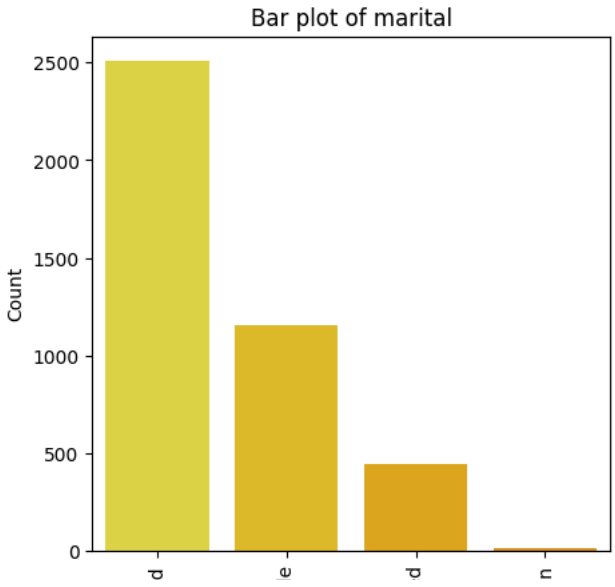
	job	marital	education	default	housing	loan	contact	month	day_o
count	4119	4119	4119	4119	4119	4119	4119	4119	
unique	12	4	8	3	3	3	2	10	
top	admin.	married	university.degree	no	yes	no	cellular	may	
freq	1012	2509	1264	3315	2175	3349	2652	1378	

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

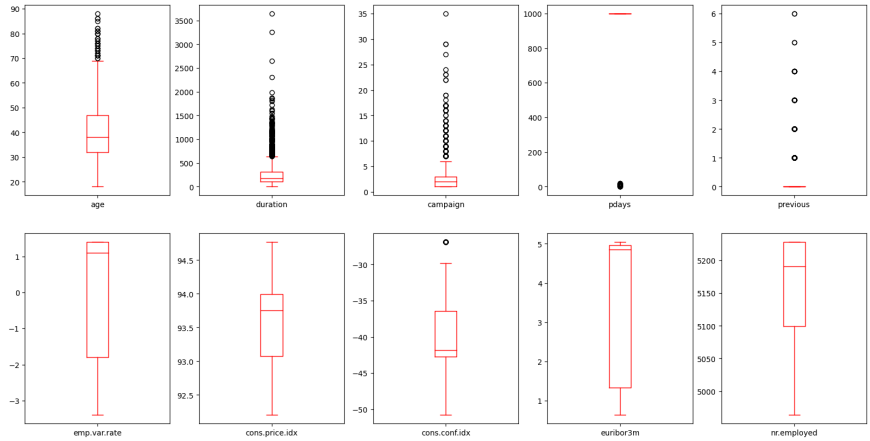



```
for feature in cat_cols:
  plt.figure(figsize=(5,5))
  sns.countplot(x=feature,data=df,palette='Wistia')
  plt.title(f'Bar plot of {feature}')
  plt.xlabel('feature')
  plt.ylabel('Count')
  plt.xticks(rotation=90)
  plt.show()
```


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
feature

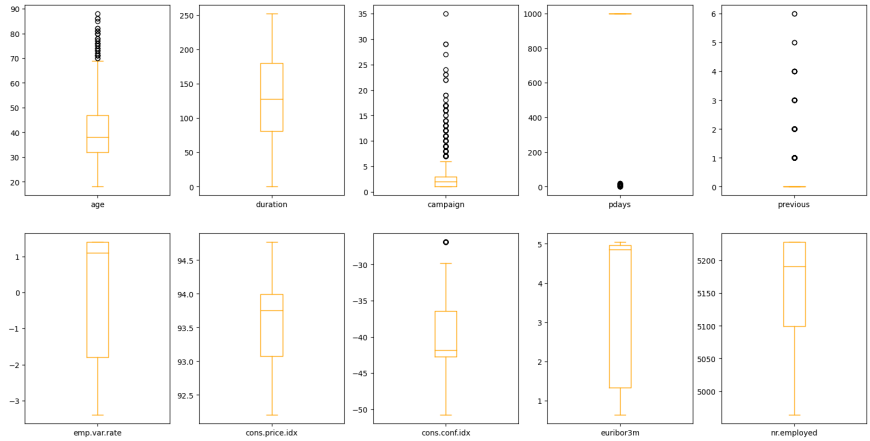

Bar plot of marital
Count
2500
2000
1500
1000
500
0

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




```
string_columns=df.columns[df.dtypes=='object']   #get column names with data type 'object' (string)
#drop columns with data type 'object' (string)
df1=df.drop(columns=string_columns)
print(df1)
```

```
       age  duration  campaign  pdays  previous  emp.var.rate  cons.price.idx  \
0       30       NaN         2    999         0          -1.8          92.893
1       39       NaN         4    999         0           1.1          93.994
2       25     227.0         1    999         0           1.4          94.465
3       38      17.0         3    999         0           1.4          94.465
4       47      58.0         1    999         0          -0.1          93.200
...    ...       ...       ...    ...       ...           ...             ...
4114    30      53.0         1    999         0           1.4          93.918
4115    39     219.0         1    999         0           1.4          93.918
4116    27      64.0         2    999         1          -1.8          92.893
4117    58       NaN         1    999         0           1.4          93.444
4118    34     175.0         1    999         0          -0.1          93.200

      cons.conf.idx  euribor3m  nr.employed
0             -46.2      1.313       5099.1
1             -36.4      4.855       5191.0
2             -41.8      4.962       5228.1
3             -41.8      4.959       5228.1
4             -42.0      4.191       5195.8
...             ...        ...          ...
4114          -42.7      4.958       5228.1
4115          -42.7      4.959       5228.1
4116          -46.2      1.354       5099.1
4117          -36.1      4.966       5228.1
4118          -42.0      4.120       5195.8

[4119 rows x 10 columns]
```

```
corr=df1.corr()
print(corr)
corr=corr[abs(corr)>=0.90]
sns.heatmap(corr,annot=True,cmap='Set3',linewidths=0.2)
plt.show()
```