In [171]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

from xgboost import XGBClassifier

from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import confusion_matrix, accuracy_score, class

import warnings
warnings.filterwarnings("ignore")
```

load the dataset

In [172]:
```
df = pd.read_csv("loan_detection.csv")
df.head()
```

Out[172]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job_blue coll
0	56	1	999	0	1	0	0	
1	57	1	999	0	1	0	0	
2	37	1	999	0	1	0	0	
3	40	1	999	0	1	0	1	
4	56	1	999	0	1	0	0	

5 rows × 60 columns

Basic EDA

In [173]:
```
df.shape
```

Out[173]: (41188, 60)

In [174]:
```
df.describe()
```

Out[174]:

	age	campaign	pdays	previous	no_previous_contact	not_v
count	41188.00000	41188.000000	41188.000000	41188.000000	41188.000000	41188.
mean	40.02406	2.567593	962.475454	0.172963	0.963217	0.
std	10.42125	2.770014	186.910907	0.494901	0.188230	0.
min	17.00000	1.000000	0.000000	0.000000	0.000000	0.
25%	32.00000	1.000000	999.000000	0.000000	1.000000	0.
50%	38.00000	2.000000	999.000000	0.000000	1.000000	0.
75%	47.00000	3.000000	999.000000	0.000000	1.000000	0.
max	98.00000	56.000000	999.000000	7.000000	1.000000	1.

8 rows × 60 columns

In [175]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41188 entries, 0 to 41187
Data columns (total 60 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   age                             41188 non-null  int64
 1   campaign                        41188 non-null  int64
 2   pdays                           41188 non-null  int64
 3   previous                        41188 non-null  int64
 4   no_previous_contact             41188 non-null  int64
 5   not_working                     41188 non-null  int64
 6   job_admin.                      41188 non-null  int64
 7   job_blue-collar                 41188 non-null  int64
 8   job_entrepreneur                41188 non-null  int64
 9   job_housemaid                   41188 non-null  int64
 10  job_management                  41188 non-null  int64
 11  job_retired                     41188 non-null  int64
 12  job_self-employed               41188 non-null  int64
 13  job_services                    41188 non-null  int64
 14  job_student                     41188 non-null  int64
 15  job_technician                  41188 non-null  int64
 16  job_unemployed                  41188 non-null  int64
 17  job_unknown                     41188 non-null  int64
 18  marital_divorced                41188 non-null  int64
 19  marital_married                 41188 non-null  int64
 20  marital_single                  41188 non-null  int64
 21  marital_unknown                 41188 non-null  int64
 22  education_basic.4y              41188 non-null  int64
 23  education_basic.6y              41188 non-null  int64
 24  education_basic.9y              41188 non-null  int64
 25  education_high.school           41188 non-null  int64
```

```
 25  education_high.school         41188 non-null  int64
 26  education_illiterate          41188 non-null  int64
 27  education_professional.course 41188 non-null  int64
 28  education_university.degree   41188 non-null  int64
 29  education_unknown             41188 non-null  int64
 30  default_no                    41188 non-null  int64
 31  default_unknown               41188 non-null  int64
 32  default_yes                   41188 non-null  int64
 33  housing_no                    41188 non-null  int64
 34  housing_unknown               41188 non-null  int64
 35  housing_yes                   41188 non-null  int64
 36  loan_no                       41188 non-null  int64
 37  loan_unknown                  41188 non-null  int64
 38  loan_yes                      41188 non-null  int64
 39  contact_cellular              41188 non-null  int64
 40  contact_telephone             41188 non-null  int64
 41  month_apr                     41188 non-null  int64
 42  month_aug                     41188 non-null  int64
 43  month_dec                     41188 non-null  int64
 44  month_jul                     41188 non-null  int64
 45  month_jun                     41188 non-null  int64
 46  month_mar                     41188 non-null  int64
 47  month_may                     41188 non-null  int64
 48  month_nov                     41188 non-null  int64
 49  month_oct                     41188 non-null  int64
 50  month_sep                     41188 non-null  int64
 51  day_of_week_fri               41188 non-null  int64
 52  day_of_week_mon               41188 non-null  int64
 53  day_of_week_thu               41188 non-null  int64
 54  day_of_week_tue               41188 non-null  int64
 55  day_of_week_wed               41188 non-null  int64
 56  poutcome_failure              41188 non-null  int64
 57  poutcome_nonexistent          41188 non-null  int64
 58  poutcome_success              41188 non-null  int64
 59  Loan_Status_label             41188 non-null  int64
dtypes: int64(60)
memory usage: 18.9 MB
```

In [176]:
```
df.nunique()
```

Out[176]:
```
age                            78
campaign                       42
pdays                          27
previous                        8
no_previous_contact             2
not_working                     2
job_admin.                      2
job_blue-collar                 2
job_entrepreneur                2
job_housemaid                   2
job_management                  2
job_retired                     2
job_self-employed               2
job_services                    2
```

```
job_services                      2
job_student                       2

job_technician                    2
job_unemployed                    2
job_unknown                       2
marital_divorced                  2
marital_married                   2
marital_single                    2
marital_unknown                   2
education_basic.4y                2
education_basic.6y                2
education_basic.9y                2
education_high.school             2
education_illiterate              2
education_professional.course     2
education_university.degree       2
education_unknown                 2
default_no                        2
default_unknown                   2
default_yes                       2
housing_no                        2
housing_unknown                   2
housing_yes                       2
loan_no                           2
loan_unknown                      2
loan_yes                          2
contact_cellular                  2
contact_telephone                 2
month_apr                         2
month_aug                         2
month_dec                         2
month_jul                         2
month_jun                         2
month_mar                         2
month_may                         2
month_nov                         2
month_oct                         2
month_sep                         2
day_of_week_fri                   2
day_of_week_mon                   2
day_of_week_thu                   2
day_of_week_tue                   2
day_of_week_wed                   2
poutcome_failure                  2
poutcome_nonexistent              2
poutcome_success                  2
Loan_Status_label                 2
dtype: int64
```

In [177]:
```
df.columns
```

Out[177]:
```
Index(['age', 'campaign', 'pdays', 'previous', 'no_previous_contac
t',
       'not_working', 'job_admin.', 'job_blue-collar', 'job_entrep
reneur',
       'job_housemaid', 'job_management', 'job_retired', 'job_self
-employed',
       'job_services', 'job_student', 'job_technician', 'job_unemp
loyed',
       'job_unknown', 'marital_divorced', 'marital_married', 'mari
tal_single',
       'marital_unknown', 'education_basic.4y', 'education_basic.6
y',
       'education_basic.9y', 'education_high.school', 'education_i
lliterate',
       'education_professional.course', 'education_university.degr
ee',
       'education_unknown', 'default_no', 'default_unknown', 'defa
ult_yes',
       'housing_no', 'housing_unknown', 'housing_yes', 'loan_no',
       'loan_unknown', 'loan_yes', 'contact_cellular', 'contact_te
lephone',
       'month_apr', 'month_aug', 'month_dec', 'month_jul', 'month_
jun',
       'month_mar', 'month_may', 'month_nov', 'month_oct', 'month_
sep',
       'day_of_week_fri', 'day_of_week_mon', 'day_of_week_thu',
       'day_of_week_tue', 'day_of_week_wed', 'poutcome_failure',
       'poutcome_nonexistent', 'poutcome_success', 'Loan_Status_la
bel'],
      dtype='object')
```

Handling Missing Values

In [178]:
```
df.isnull().sum()
```

Out[178]:
```
age                              0
campaign                         0
pdays                            0
previous                         0
no_previous_contact              0
not_working                      0
job_admin.                       0
job_blue-collar                  0
job_entrepreneur                 0
job_housemaid                    0
job_management                   0
job_retired                      0
job_self-employed                0
job_services                     0
```

```
job_student                      0
job_technician                   0
job_unemployed                   0
job_unknown                      0
marital_divorced                 0
marital_married                  0
marital_single                   0
marital_unknown                  0
education_basic.4y               0
education_basic.6y               0
education_basic.9y               0
education_high.school            0
education_illiterate             0
education_professional.course    0
education_university.degree      0
education_unknown                0
default_no                       0
default_unknown                  0
default_yes                      0
housing_no                       0
housing_unknown                  0
housing_yes                      0
loan_no                          0
loan_unknown                     0
loan_yes                         0
contact_cellular                 0
contact_telephone                0
month_apr                        0
month_aug                        0
month_dec                        0
month_jul                        0
month_jun                        0
month_mar                        0
month_may                        0
month_nov                        0
month_oct                        0
month_sep                        0
day_of_week_fri                  0
day_of_week_mon                  0
day_of_week_thu                  0
day_of_week_tue                  0
day_of_week_wed                  0
poutcome_failure                 0
poutcome_nonexistent             0
poutcome_success                 0
Loan_Status_label                0
dtype: int64
```

In [179]:
```
round(df.isnull().mean() * 100, 2)
```

Out[179]:
```
age                              0.0
campaign                         0.0
pdays                            0.0
previous                         0.0
```

```
previous                         0.0
no_previous_contact              0.0

not_working                      0.0
job_admin.                       0.0
job_blue-collar                  0.0
job_entrepreneur                 0.0
job_housemaid                    0.0
job_management                   0.0
job_retired                      0.0
job_self-employed                0.0
job_services                     0.0
job_student                      0.0
job_technician                   0.0
job_unemployed                   0.0
job_unknown                      0.0
marital_divorced                 0.0
marital_married                  0.0
marital_single                   0.0
marital_unknown                  0.0
education_basic.4y               0.0
education_basic.6y               0.0
education_basic.9y               0.0
education_high.school            0.0
education_illiterate             0.0
education_professional.course    0.0
education_university.degree      0.0
education_unknown                0.0
default_no                       0.0
default_unknown                  0.0
default_yes                      0.0
housing_no                       0.0
housing_unknown                  0.0
housing_yes                      0.0
loan_no                          0.0
loan_unknown                     0.0
loan_yes                         0.0
contact_cellular                 0.0
contact_telephone                0.0
month_apr                        0.0
month_aug                        0.0
month_dec                        0.0
month_jul                        0.0
month_jun                        0.0
month_mar                        0.0
month_may                        0.0
month_nov                        0.0
month_oct                        0.0
month_sep                        0.0
day_of_week_fri                  0.0
day_of_week_mon                  0.0
day_of_week_thu                  0.0
day_of_week_tue                  0.0
day_of_week_wed                  0.0
poutcome_failure                 0.0
```

```
poutcome_nonexistent          0.0

poutcome_success              0.0
Loan_Status_label             0.0
dtype: float64
```

In [180]:
```
df.duplicated().sum()
```

Out[180]: 2417

In [181]:
```
df[df.duplicated()]
```

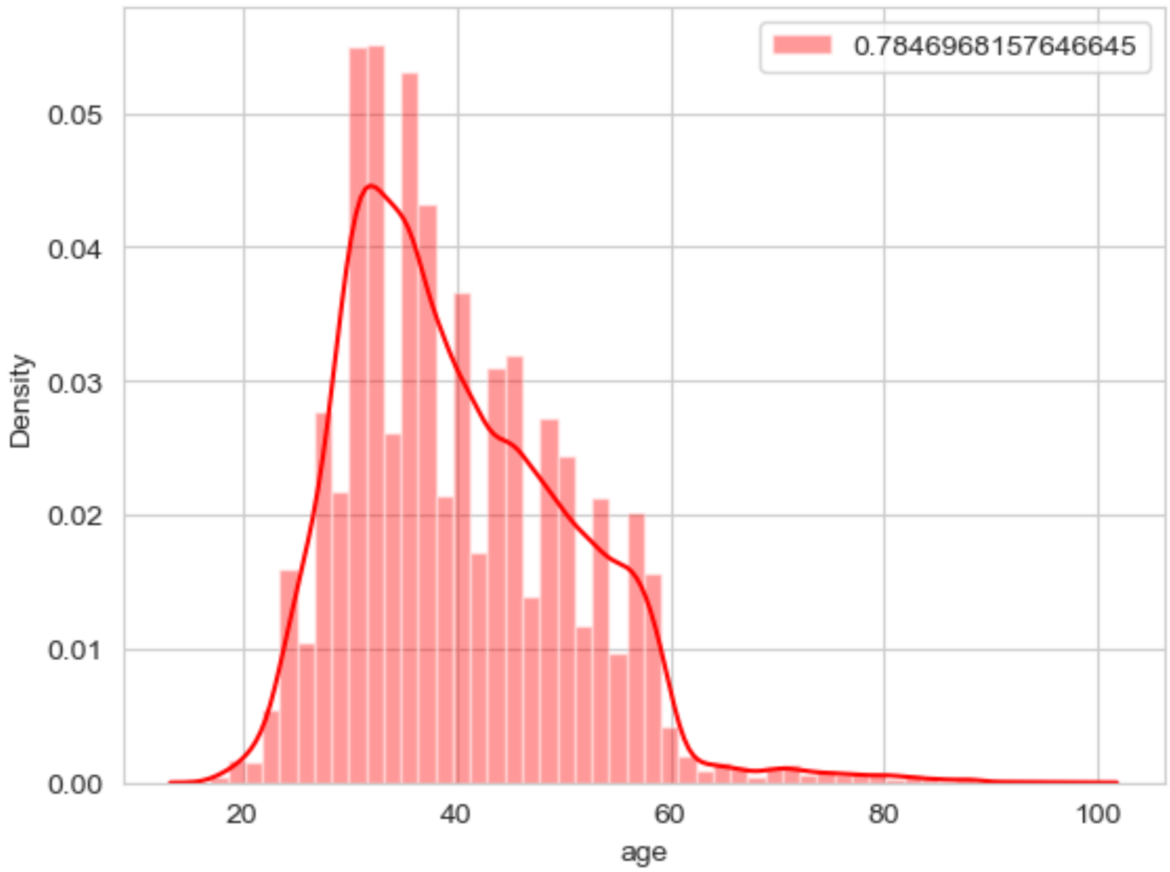
Out[181]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
10	41	1	999	0	1	0	0	
11	25	1	999	0	1	0	0	
16	35	1	999	0	1	0	0	
31	59	1	999	0	1	0	0	
104	52	1	999	0	1	0	1	
...	...	...	...	...	...	...	...	
40928	21	1	999	0	1	1	0	
41131	58	1	999	0	1	0	0	
41167	32	3	999	0	1	0	1	
41172	31	1	999	0	1	0	1	
41181	37	1	999	0	1	0	1	

2417 rows × 60 columns

In [182]:
```
sns.distplot(df['age'], color='r', label=df.age.skew())
plt.legend()
```

Out[182]: <matplotlib.legend.Legend at 0x16e3b6150>



In [183]:
```
df.age.mean()
```

Out[183]: 40.02406040594348

In [184]:
```
df.age.median()
```

Out[184]: 38.0

In [185]:
```
for i in df:
    if df[i].isna().sum() > 0:
        print(f' {i} : {df[i].mean()}')
        df[i].fillna(df[i].mean(), inplace=True)
```

Duplicate Data

In [186]:
```
df.drop_duplicates(keep="first", inplace=True)
```

In [187]:
```
df.duplicated().sum()
```

Out[187]:
```
0
```

In [188]:
```
df[df.duplicated()]
```

Out[188]:

age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job_blue-collar

0 rows × 60 columns

In [189]:
```
df.shape
```

Out[189]:
```
(38771, 60)
```

Outliers or Anomalies

Using IQR

In [190]:
```
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1
IQR
```

Out[190]:
```
age                              15.0
campaign                          2.0
pdays                             0.0
previous                          0.0
no_previous_contact               0.0
not_working                       0.0
job_admin.                        0.0
job_blue-collar                   0.0
job_entrepreneur                  0.0
job_housemaid                     0.0
job_management                    0.0
job_retired                       0.0
job_self-employed                 0.0
job_services                      0.0
job_student                       0.0
job_technician                    0.0
job_unemployed                    0.0
job_unknown                       0.0
marital_divorced                  0.0
marital_married                   1.0
marital_single                    1.0
marital_unknown                   0.0
education_basic.4y                0.0
education_basic.6y                0.0
```

```
education_basic.9y               0.0
education_high.school            0.0
education_illiterate             0.0
education_professional.course    0.0
education_university.degree      1.0
education_unknown                0.0
default_no                       0.0
default_unknown                  0.0
default_yes                      0.0
housing_no                       1.0
housing_unknown                  0.0
housing_yes                      1.0
loan_no                          0.0
loan_unknown                     0.0
loan_yes                         0.0
contact_cellular                 1.0
contact_telephone                1.0
month_apr                        0.0
month_aug                        0.0
month_dec                        0.0
month_jul                        0.0
month_jun                        0.0
month_mar                        0.0
month_may                        1.0
month_nov                        0.0
month_oct                        0.0
month_sep                        0.0
day_of_week_fri                  0.0
day_of_week_mon                  0.0
day_of_week_thu                  0.0
day_of_week_tue                  0.0
day_of_week_wed                  0.0
poutcome_failure                 0.0
poutcome_nonexistent             0.0
poutcome_success                 0.0
Loan_Status_label                0.0
dtype: float64
```

In [191]:
```
print(Q1 - 1.5 * IQR)
print()
print(Q3 + 1.5 * IQR)
```

```
age                           9.5
campaign                     -2.0
pdays                       999.0
previous                      0.0
no_previous_contact           1.0
not_working                   0.0
job_admin.                    0.0
job_blue-collar               0.0
job_entrepreneur              0.0
job_housemaid                 0.0
job_management                0.0
job_retired                   0.0
job_self-employed             0.0
job_services                  0.0
job_student                   0.0
job_technician                0.0
job_unemployed                0.0
job_unknown                   0.0
marital_divorced              0.0
```

In [192]:
```
lower_bound = (Q1 - 1.5 * IQR)
upper_bound = (Q3 + 1.5 * IQR)
```

In [193]:
```
upper_bound
```

Out[193]:
```
age                          69.5
campaign                      6.0
pdays                       999.0
previous                      0.0
no_previous_contact           1.0
not_working                   0.0
job_admin.                    0.0
job_blue-collar               0.0
job_entrepreneur              0.0
job_housemaid                 0.0
job_management                0.0
job_retired                   0.0
job_self-employed             0.0
job_services                  0.0
job_student                   0.0
job_technician                0.0
job_unemployed                0.0
job_unknown                   0.0
marital_divorced              0.0
marital_married               2.5
marital_single                2.5
marital_unknown               0.0
education_basic.4y            0.0
education_basic.6y            0.0
```

```
education_basic.9y                0.0

education_high.school             0.0
education_illiterate              0.0
education_professional.course     0.0
education_university.degree       2.5
education_unknown                 0.0
default_no                        1.0
default_unknown                   0.0
default_yes                       0.0
housing_no                        2.5
housing_unknown                   0.0
housing_yes                       2.5
loan_no                           1.0
loan_unknown                      0.0
loan_yes                          0.0
contact_cellular                  2.5
contact_telephone                 2.5
month_apr                         0.0
month_aug                         0.0
month_dec                         0.0
month_jul                         0.0
month_jun                         0.0
month_mar                         0.0
month_may                         2.5
month_nov                         0.0
month_oct                         0.0
month_sep                         0.0
day_of_week_fri                   0.0
day_of_week_mon                   0.0
day_of_week_thu                   0.0
day_of_week_tue                   0.0
day_of_week_wed                   0.0
poutcome_failure                  0.0
poutcome_nonexistent              1.0
poutcome_success                  0.0
Loan_Status_label                 0.0
dtype: float64
```

In [194]:
```
Q1 = df["age"].quantile(0.25)
Q3 = df["age"].quantile(0.75)
```

In [195]:
```
IQR = Q3 - Q1
IQR
```

Out[195]:
```
15.0
```

In [196]:
```
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
```

In [197]:
```
lower_bound
```

Out[197]: 9.5

In [198]:
```
upper_bound
```

Out[198]: 69.5

In [199]:
```
df_filtered = df[(df['age'] >= lower_bound) & (df['age'] <= upper_b
```

In [200]:
```
print(f"Q1: {Q1}, Q3: {Q3}")
print(f"IQR: {IQR}")
print(f"Lower Bound: {lower_bound}")
print(f"Upper Bound: {upper_bound}")
```

```
Q1: 32.0, Q3: 47.0
IQR: 15.0
Lower Bound: 9.5
Upper Bound: 69.5
```

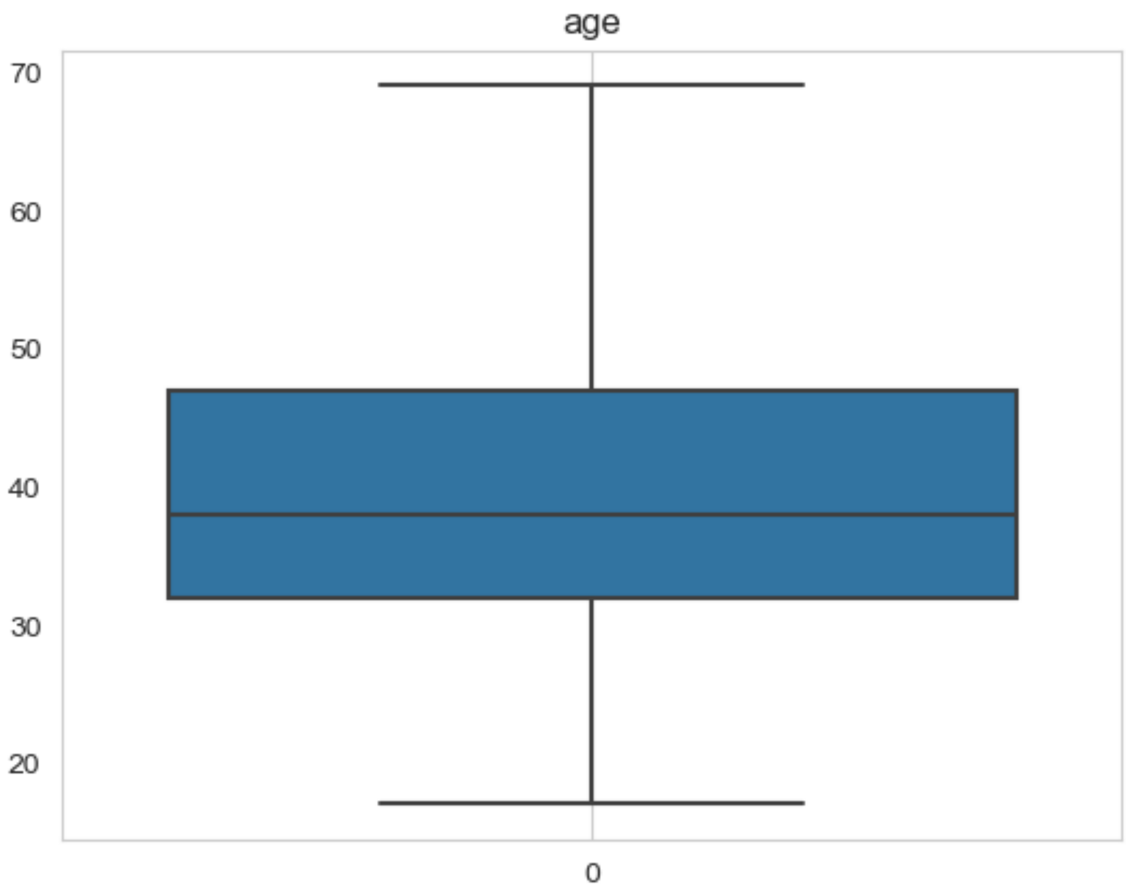
In [201]:
```
df_filtered
```

Out[201]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
0	56	1	999	0	1	0	0	
1	57	1	999	0	1	0	0	
2	37	1	999	0	1	0	0	
3	40	1	999	0	1	0	1	
4	56	1	999	0	1	0	0	
...	...	...	...	...	...	...	...	
41180	36	2	999	0	1	0	1	
41182	29	1	9	1	0	1	0	
41184	46	1	999	0	1	0	0	
41185	56	2	999	0	1	1	0	
41186	44	1	999	0	1	0	0	

38314 rows × 60 columns

In [202]:
```
sns.boxplot(df_filtered['age'])
plt.title("age")
plt.grid()
plt.show()
```



Feature Selection

In [203]:
```
df= df_filtered
df
```

Out[203]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
0	56	1	999	0	1	0	0	
1	57	1	999	0	1	0	0	
2	37	1	999	0	1	0	0	
3	40	1	999	0	1	0	1	
4	56	1	999	0	1	0	0	
...	...	...	...	...	...	...	...	
41180	36	2	999	0	1	0	1	
41182	29	1	9	1	0	1	0	
41184	46	1	999	0	1	0	0	
41185	56	2	999	0	1	1	0	
41186	44	1	999	0	1	0	0	

38314 rows × 60 columns

In [204]:
```
corr_matrix = df.corr()
corr_matrix
```

Out[204]:

	age	campaign	pdays	previous	no_previous_conta
age	1.000000	0.010066	0.007008	-0.025589	0.0070
campaign	0.010066	1.000000	0.056404	-0.086045	0.0563
pdays	0.007008	0.056404	1.000000	-0.582789	0.9999
previous	-0.025589	-0.086045	-0.582789	1.000000	-0.5827
no_previous_contact	0.007062	0.056386	0.999992	-0.582737	1.0000
not_working	0.104183	-0.014212	-0.091269	0.077921	-0.0912
job_admin.	-0.079238	0.014212	-0.034821	0.030180	-0.0348
job_blue-collar	-0.008970	-0.001496	0.063590	-0.052354	0.0635
job_entrepreneur	0.038534	-0.004649	0.019040	-0.013706	0.0190
job_housemaid	0.083795	0.002668	0.002466	-0.015794	0.0024
job_management	0.074676	-0.011820	-0.001543	0.007911	-0.0015
job_retired	0.333065	0.003053	-0.034072	0.019583	-0.0340
job_self-employed	0.004886	0.003518	0.014810	-0.010931	0.0148

job_services	-0.057882	0.001326	0.029145	-0.009117	0.0291
job_student	-0.213602	-0.027017	-0.101166	0.110906	-0.1012
job_technician	-0.050104	0.001932	-0.001163	-0.011389	-0.0011
job_unemployed	0.001529	-0.002850	-0.024804	0.009154	-0.0247
job_unknown	0.049796	0.001078	-0.012548	-0.005160	-0.0125
marital_divorced	0.152305	0.003643	0.019334	-0.010093	0.0193
marital_married	0.287972	0.003598	0.035714	-0.050791	0.0357
marital_single	-0.420439	-0.007394	-0.051779	0.061364	-0.0518
marital_unknown	0.003199	0.009140	-0.006306	0.009484	-0.0063
education_basic.4y	0.205589	0.008350	0.024893	-0.035569	0.0248
education_basic.6y	0.013295	-0.003027	0.023116	-0.020882	0.0231
education_basic.9y	-0.030256	-0.005622	0.035960	-0.026555	0.0359
education_high.school	-0.101134	-0.002061	0.001550	0.018612	0.0015
education_illiterate	0.014968	-0.002155	0.004127	-0.004954	0.0041
education_professional.course	0.009831	0.002703	-0.005233	-0.006435	-0.0052
education_university.degree	-0.056410	0.000140	-0.045412	0.034817	-0.0453
education_unknown	0.058205	0.000890	-0.018807	0.016314	-0.0188
default_no	-0.186083	-0.032442	-0.085162	0.107392	-0.0851
default_unknown	0.186056	0.032535	0.085136	-0.107468	0.0851
default_yes	0.002364	-0.004080	0.001734	0.002827	0.0017
housing_no	0.002841	0.010368	0.011876	-0.024133	0.0118
housing_unknown	-0.001142	-0.003938	0.002510	0.002365	0.0025
housing_yes	-0.002472	-0.009093	-0.012624	0.023305	-0.0125
loan_no	0.012039	0.004704	-0.007371	0.007338	-0.0073
loan_unknown	-0.001142	-0.003938	0.002510	0.002365	0.0025
loan_yes	-0.012275	-0.003295	0.006737	-0.008798	0.0067
contact_cellular	-0.026496	-0.072020	-0.121645	0.221270	-0.1216
contact_telephone	0.026496	0.072020	0.121645	-0.221270	0.1216
month_apr	0.000340	-0.062005	-0.007524	0.080018	-0.0074
month_aug	0.077554	0.024883	-0.007060	-0.046063	-0.0070
month_dec	0.023882	-0.011619	-0.075564	0.055893	-0.0755
month_jul	-0.033330	0.105323	0.049808	-0.118071	0.0497
month_jun	-0.000753	0.066691	0.015832	-0.073067	0.0158
month_mar	-0.025366	-0.017288	-0.082528	0.074531	-0.0826

month_may	-0.053272	-0.037978	0.071321	-0.012326	0.0712
month_nov	0.035375	-0.082584	-0.017164	0.085976	-0.0170
month_oct	0.010380	-0.050288	-0.127852	0.125191	-0.1278
month_sep	0.002033	-0.035082	-0.157506	0.156739	-0.1575
day_of_week_fri	0.007443	0.025347	0.016316	0.002871	0.0163
day_of_week_mon	0.021426	0.009435	0.001188	-0.002120	0.0011
day_of_week_thu	-0.021091	0.008502	-0.009174	0.000868	-0.0091
day_of_week_tue	0.015010	-0.022871	-0.005305	-0.000369	-0.0052
day_of_week_wed	-0.022740	-0.020652	-0.002725	-0.001192	-0.0027
poutcome_failure	-0.026651	-0.075592	0.006908	0.687826	0.0066
poutcome_nonexistent	0.026140	0.095532	0.487108	-0.880192	0.4871
poutcome_success	-0.004742	-0.054634	-0.949831	0.518177	-0.9494
Loan_Status_label	-0.022105	-0.072041	-0.318589	0.221417	-0.3185

60 rows × 60 columns

Model Building

split independent and dependent data

In [205]:
```
X = df.iloc[:, :-1]
X
```

Out[205]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
0	56	1	999	0	1	0	0	
1	57	1	999	0	1	0	0	
2	37	1	999	0	1	0	0	
3	40	1	999	0	1	0	1	
4	56	1	999	0	1	0	0	
...	...	...	...	...	...	...	...	
41180	36	2	999	0	1	0	1	
41182	29	1	9	1	0	1	0	
41184	46	1	999	0	1	0	0	
41185	56	2	999	0	1	1	0	
41186	44	1	999	0	1	0	0	

38314 rows × 59 columns

In [206]:
```
y = df['Loan_Status_label']
y
```

Out[206]:
```
0        0
1        0
2        0
3        0
4        0
        ..
41180    0
41182    0
41184    0
41185    0
41186    1
Name: Loan_Status_label, Length: 38314, dtype: int64
```

In [207]:
```
y.value_counts()
```

Out[207]:
```
Loan_Status_label
0    33939
1     4375
Name: count, dtype: int64
```

In [209]:
```
X_train, X_test, y_train , y_test = train_test_split(X, y , test_si
```

In [210]:
```
X_train
```

Out[210]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
38446	64	1	999	1	1	0	1	
16109	23	1	999	0	1	0	0	
1320	37	2	999	0	1	0	0	
29221	44	1	999	1	1	0	0	
8319	39	12	999	0	1	0	1	
...	...	...	...	...	...	...	...	
6625	35	2	999	0	1	1	0	
11843	31	3	999	0	1	0	1	
41026	65	2	12	1	0	1	0	
885	43	1	999	0	1	0	0	
16784	43	2	999	0	1	0	0	

30651 rows × 59 columns

In [211]:
```
X_test
```

Out[211]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
21256	41	3	999	0	1	0	0	
26340	44	1	999	0	1	1	0	
15626	60	3	999	0	1	1	0	
37031	23	1	999	1	1	1	0	
36150	32	1	999	0	1	0	1	
...	...	...	...	...	...	...	...	
34886	53	1	999	1	1	0	0	
22433	29	1	999	0	1	0	1	
36847	37	1	999	0	1	0	0	
14411	39	2	999	0	1	0	0	
37559	54	3	999	0	1	0	1	

7663 rows × 59 columns

Feature Scaling

In [212]:
```
sc = StandardScaler()
X_train_sc = sc.fit_transform(X_train)
X_test_sc = sc.fit_transform(X_test)
```

In [213]:
```
X_test_sc
```

Out[213]:
```
array([[ 0.13398938,  0.11414988,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ],
       [ 0.44214084, -0.555885  ,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ],
       [ 2.08561527,  0.11414988,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ],
       ...,
       [-0.27687922, -0.555885  ,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ],
       [-0.07144492, -0.22086756,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ],
       [ 1.46931236,  0.11414988,  0.19903267, ..., -0.34189557,
         0.40241877, -0.1896326 ]])
```

In [214]:
```
X_train_sc
```

Out[214]:
```
array([[ 2.49272944, -0.58448757,  0.1951058 , ...,  2.90313209,
        -2.48695613, -0.18506455],
       [-1.72043172, -0.58448757,  0.1951058 , ..., -0.34445556,
         0.40209797, -0.18506455],
       [-0.28179132, -0.22804563,  0.1951058 , ..., -0.34445556,
         0.40209797, -0.18506455],
       ...,
       [ 2.59548947, -0.22804563, -5.09317464, ..., -0.34445556,
        -2.48695613,  5.40352011],
       [ 0.33476884, -0.58448757,  0.1951058 , ..., -0.34445556,
         0.40209797, -0.18506455],
       [ 0.33476884, -0.22804563,  0.1951058 , ..., -0.34445556,
         0.40209797, -0.18506455]])
```

Model Selection

Using Logistic Regression

In [215]:
```
lr = LogisticRegression()
lr.fit(X_train , y_train)
```

Out[215]:
```
▾ LogisticRegression
LogisticRegression()
```

In [216]:
```
print(f'Training Accuracy : {lr.score(X_train, y_train)}')
print(f'Test Accuracy : {lr.score(X_test, y_test)}')
```

```
Training Accuracy : 0.8954357117222929
Test Accuracy : 0.8974292052720867
```

In [217]:
```
# On Scaled Data
lr = LogisticRegression()
lr.fit(X_train_sc, y_train)
```

Out[217]:
```
▾ LogisticRegression
LogisticRegression()
```

In [218]:
```
print(f'Training Accuracy : {lr.score(X_train_sc, y_train)}')
print(f'Test Accuracy : {lr.score(X_test_sc, y_test)}')
```

```
Training Accuracy : 0.8957945907148217
Test Accuracy : 0.8941667754143285
```

Using Decision Tree Classifier

In [219]:
```
dt = DecisionTreeClassifier(max_depth=5)
dt.fit(X_train, y_train)
```

Out[219]:
```
▾ DecisionTreeClassifier
DecisionTreeClassifier(max_depth=5)
```

In [220]:
```
print(f'Training Accuracy : {dt.score(X_train, y_train)}')
print(f'Test Accuracy : {dt.score(X_test, y_test)}')
```

```
Training Accuracy : 0.8984372451143519
Test Accuracy : 0.8948192613858802
```

Using RandomForestClassifier

In [221]:
```
rfc = RandomForestClassifier(n_estimators=100, random_state=42)
rfc.fit(X_train, y_train)
```

Out[221]:
```
RandomForestClassifier
RandomForestClassifier(random_state=42)
```

In [222]:
```
print(f'Training Accuracy : {rfc.score(X_train, y_train)}')
print(f'Test Accuracy : {rfc.score(X_test, y_test)}')
```

```
Training Accuracy : 0.9928876708753385
Test Accuracy : 0.8839879942581235
```

Using XGB Classifier

In [223]:
```
xgb = XGBClassifier(gamma=0.7, reg_alpha=0.5, reg_lambda=0.2)
xgb.fit(X_train ,y_train)
```

Out[223]:
```
XGBClassifier
XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_
rounds=None,
              enable_categorical=False, eval_metric=None, feature
_types=None,
              gamma=0.7, grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=None, m
ax_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
```

In [224]:
```
print(f'Training Accuracy : {xgb.score(X_train, y_train)}')
print(f'Test Accuracy : {xgb.score(X_test, y_test)}')
```

```
Training Accuracy : 0.9065935858536426
Test Accuracy : 0.8957327417460524
```

In [225]:
```
y_pred_xgtr = xgb.predict(X_train)
y_pred_xgts = xgb.predict(X_test)
```

In [226]:
```
X_train[:3]
```

Out[226]:

	age	campaign	pdays	previous	no_previous_contact	not_working	job_admin.	job
38446	64	1	999	1	1	0	1	
16109	23	1	999	0	1	0	0	
1320	37	2	999	0	1	0	0	

3 rows × 59 columns

In [227]:
```
y_train[:3]
```

Out[227]:
```
38446    1
16109    0
1320     0
Name: Loan_Status_label, dtype: int64
```

In [228]:
```
y_pred_xgtr[:3]
```

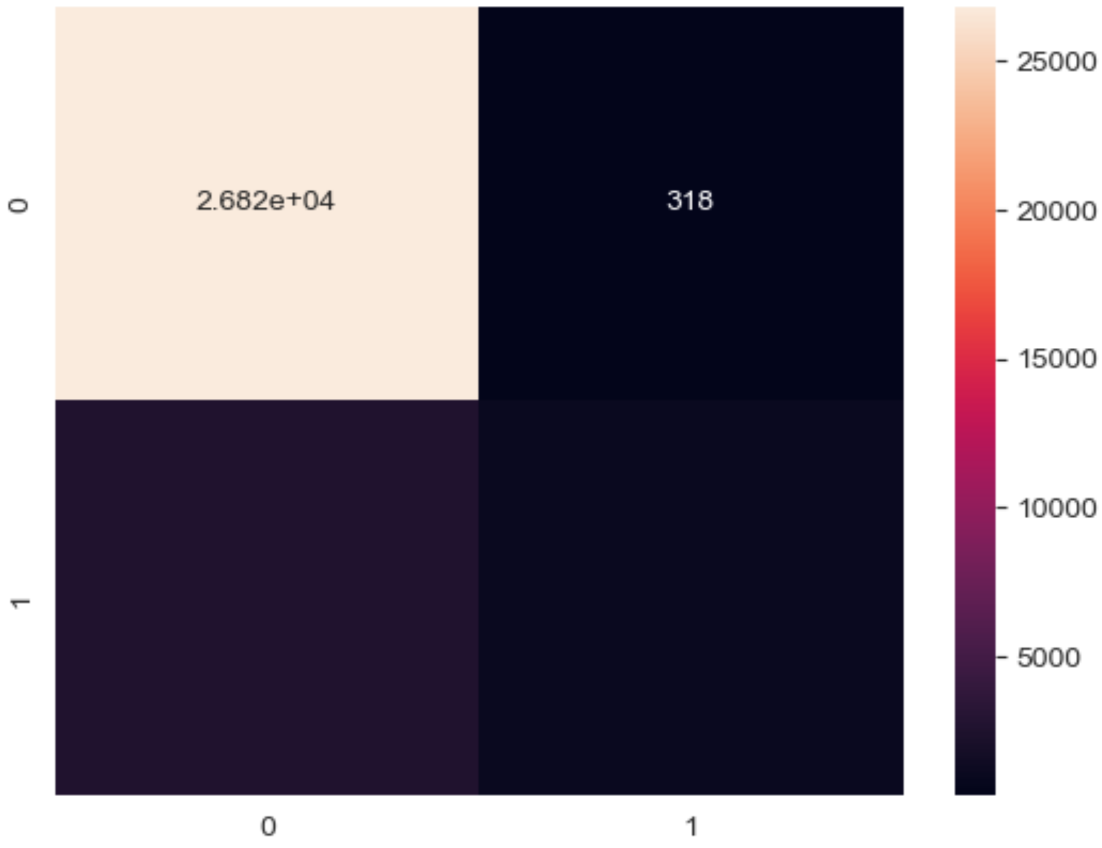
Out[228]:
```
array([1, 0, 0])
```

In [229]:
```
confusion_matrix(y_train, y_pred_xgtr)
```

Out[229]:
```
array([[26822,   318],
       [ 2545,   966]])
```

In [230]:
```
sns.heatmap(confusion_matrix(y_train, y_pred_xgtr), annot=True, fmt
```

Out[230]: <Axes: >



In [231]:
```
accuracy_score(y_train, y_pred_xgtr)
```

Out[231]: 0.9065935858536426

In [232]:
```
print(classification_report(y_train, y_pred_xgtr))
```

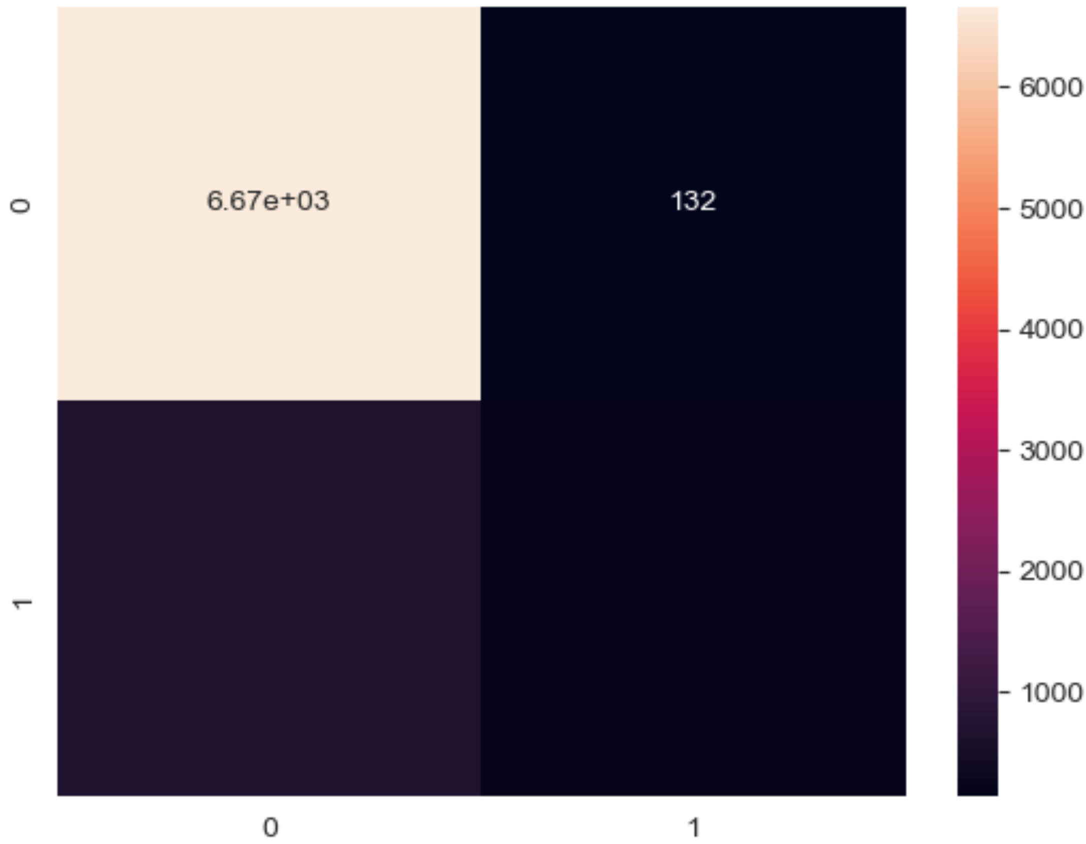
```
              precision    recall  f1-score   support

           0       0.91      0.99      0.95     27140
           1       0.75      0.28      0.40      3511

    accuracy                           0.91     30651
   macro avg       0.83      0.63      0.68     30651
weighted avg       0.89      0.91      0.89     30651
```

In [233]:
```
#TEST
confusion_matrix(y_test, y_pred_xgts)
```

Out[233]:
```
array([[6667,  132],
       [ 667,  197]])
```

In [234]:
```
sns.heatmap(confusion_matrix(y_test, y_pred_xgts), annot=True, fmt=
```

Out[234]: <Axes: >



In [235]:
```
accuracy_score(y_test, y_pred_xgts)
```

Out[235]: 0.8957327417460524

Hyperparameter Tuning

In [236]:
```
parameters = {
    'n_estimators' : [100, 200],
    'learning_rate' : [0.1,0.01,1.0,0.05],
    'max_depth' : [3,4,5],
    'gamma' : [0.2,0.3],
    'reg_alpha' : [0.1,1,0.2],
    'reg_lambda' : [0.1,1]
}
parameters
```

Out[236]:
```
{'n_estimators': [100, 200],
 'learning_rate': [0.1, 0.01, 1.0, 0.05],
 'max_depth': [3, 4, 5],
 'gamma': [0.2, 0.3],
 'reg_alpha': [0.1, 1, 0.2],
 'reg_lambda': [0.1, 1]}
```

In [237]:
```
# perform GridSearchCV
grid_search = GridSearchCV(estimator=xgb, param_grid=parameters, sc
grid_search.fit(X_train, y_train)
```

```
ors=100, reg_alpha=0.2, reg_lambda=0.1;, score=0.890 total time=
0.2s
[CV 3/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=0.1;, score=0.889 total time=
0.2s
[CV 4/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=0.1;, score=0.892 total time=
0.2s
[CV 5/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=0.1;, score=0.890 total time=
0.1s
[CV 1/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=1;, score=0.890 total time=
0.2s
[CV 2/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=1;, score=0.890 total time=
0.1s

[CV 3/5] END gamma=0.3, learning_rate=0.01, max_depth=4, n_estimat
ors=100, reg_alpha=0.2, reg_lambda=1;, score=0.889 total time=
0.2s
```

In [238]:
```
print(f'Best Selected Hyperparamters : \n\n{grid_search.best_params_
print(f'Best Estimators : \n\n{grid_search.best_estimator_}')
```

```
Best Selected Hyperparamters :

{'gamma': 0.2, 'learning_rate': 0.05, 'max_depth': 3, 'n_estimator
s': 200, 'reg_alpha': 1, 'reg_lambda': 0.1}

Best Estimators :

XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_r
ounds=None,
              enable_categorical=False, eval_metric=None, feature_
types=None,
              gamma=0.2, grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=0.05, ma
x_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
              max_delta_step=None, max_depth=3, max_leaves=None,
              min_child_weight=None, missing=nan, monotone_constra
ints=None,
              multi_strategy=None, n_estimators=200, n_jobs=None,
              num_parallel_tree=None, random_state=None, ...)
```

In [239]:
```
print(f'Training Accuracy : {grid_search.score(X_train, y_train)}')
print(f'Test Accuracy : {grid_search.score(X_test, y_test)}')
```

```
Training Accuracy : 0.8989918762846236
Test Accuracy : 0.8972987080777763
```