In [1]:
```
#LOAN DEFAULTER IN BANKING SECTOR AND BASIC PYTHON FRAMEWORK USED
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
pd.options.display.max_columns = None
pd.options.display.max_rows = None
%matplotlib inline
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
apps = pd.read_csv('application_data.csv')
loan_defaulter = pd.read_csv('previous_application.csv')
```

In [3]:
```
apps.head()
```

Out[3]:

| | SK_ID_CURR | TARGET | NAME_CONTRACT_TYPE | CODE_GENDER | FLAG_OWN_CAR | FLAG_OWN_REALTY | CNT |
|---|---|---|---|---|---|---|---|
| **0** | 100002 | 1 | Cash loans | M | N | Y | |
| **1** | 100003 | 0 | Cash loans | F | N | N | |
| **2** | 100004 | 0 | Revolving loans | M | Y | Y | |
| **3** | 100006 | 0 | Cash loans | F | N | Y | |
| **4** | 100007 | 0 | Cash loans | M | N | Y | |

In [4]:
```
# Data pre-procession/Feature selections
apps.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 286.2+ MB
```

In [5]:
```
apps.shape
```

Out[5]:
```
(307511, 122)
```

In [6]:
```
apps.columns
```

Out[6]:
```
Index(['SK_ID_CURR', 'TARGET', 'NAME_CONTRACT_TYPE', 'CODE_GENDER',
       'FLAG_OWN_CAR', 'FLAG_OWN_REALTY', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY',
       ...
       'FLAG_DOCUMENT_18', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'],
      dtype='object', length=122)
```

In [7]:

```
# find out missing values
apps.isnull().sum().sort_values()
```

Out[7]:

```
SK_ID_CURR                         0
HOUR_APPR_PROCESS_START            0
REG_REGION_NOT_WORK_REGION         0
LIVE_REGION_NOT_WORK_REGION        0
REG_CITY_NOT_LIVE_CITY             0
REG_CITY_NOT_WORK_CITY             0
LIVE_CITY_NOT_WORK_CITY            0
ORGANIZATION_TYPE                  0
FLAG_DOCUMENT_21                   0
FLAG_DOCUMENT_20                   0
FLAG_DOCUMENT_19                   0
FLAG_DOCUMENT_18                   0
FLAG_DOCUMENT_17                   0
FLAG_DOCUMENT_16                   0
FLAG_DOCUMENT_15                   0
FLAG_DOCUMENT_14                   0
FLAG_DOCUMENT_13                   0
FLAG_DOCUMENT_12                   0
FLAG_DOCUMENT_11                   0
FLAG_DOCUMENT_10                   0
FLAG_DOCUMENT_9                    0
FLAG_DOCUMENT_8                    0
FLAG_DOCUMENT_7                    0
FLAG_DOCUMENT_6                    0
FLAG_DOCUMENT_5                    0
FLAG_DOCUMENT_4                    0
FLAG_DOCUMENT_3                    0
FLAG_DOCUMENT_2                    0
WEEKDAY_APPR_PROCESS_START         0
REGION_RATING_CLIENT_W_CITY        0
REG_REGION_NOT_LIVE_REGION         0
NAME_HOUSING_TYPE                  0
CNT_CHILDREN                       0
NAME_INCOME_TYPE                   0
NAME_EDUCATION_TYPE                0
NAME_FAMILY_STATUS                 0
REGION_RATING_CLIENT               0
REGION_POPULATION_RELATIVE         0
DAYS_BIRTH                         0
DAYS_EMPLOYED                      0
DAYS_REGISTRATION                  0
DAYS_ID_PUBLISH                    0
AMT_INCOME_TOTAL                   0
FLAG_OWN_REALTY                    0
CODE_GENDER                        0
NAME_CONTRACT_TYPE                 0
FLAG_MOBIL                         0
FLAG_EMP_PHONE                     0
FLAG_WORK_PHONE                    0
FLAG_CONT_MOBILE                   0
FLAG_PHONE                         0
TARGET                             0
FLAG_EMAIL                         0
FLAG_OWN_CAR                       0
AMT_CREDIT                         0
DAYS_LAST_PHONE_CHANGE             1
CNT_FAM_MEMBERS                    2
AMT_ANNUITY                       12
AMT_GOODS_PRICE                  278
EXT_SOURCE_2                     660
DEF_30_CNT_SOCIAL_CIRCLE        1021
DEF_60_CNT_SOCIAL_CIRCLE        1021
OBS_60_CNT_SOCIAL_CIRCLE        1021
OBS_30_CNT_SOCIAL_CIRCLE        1021
                                1292
```

```
AMT_REQ_CREDIT_BUREAU_HOUR       41519
AMT_REQ_CREDIT_BUREAU_DAY        41519
AMT_REQ_CREDIT_BUREAU_MON        41519
AMT_REQ_CREDIT_BUREAU_WEEK       41519
AMT_REQ_CREDIT_BUREAU_YEAR       41519
AMT_REQ_CREDIT_BUREAU_QRT        41519
EXT_SOURCE_3                     60965
OCCUPATION_TYPE                  96391
EMERGENCYSTATE_MODE             145755
TOTALAREA_MODE                  148431
YEARS_BEGINEXPLUATATION_MODE    150007
YEARS_BEGINEXPLUATATION_AVG     150007
YEARS_BEGINEXPLUATATION_MEDI    150007
FLOORSMAX_AVG                   153020
FLOORSMAX_MEDI                  153020
FLOORSMAX_MODE                  153020
HOUSETYPE_MODE                  154297
LIVINGAREA_AVG                  154350
LIVINGAREA_MODE                 154350
LIVINGAREA_MEDI                 154350
ENTRANCES_AVG                   154828
ENTRANCES_MODE                  154828
ENTRANCES_MEDI                  154828
APARTMENTS_MEDI                 156061
APARTMENTS_AVG                  156061
APARTMENTS_MODE                 156061
WALLSMATERIAL_MODE              156341
ELEVATORS_MEDI                  163891
ELEVATORS_AVG                   163891
ELEVATORS_MODE                  163891
NONLIVINGAREA_MODE              169682
NONLIVINGAREA_AVG               169682
NONLIVINGAREA_MEDI              169682
EXT_SOURCE_1                    173378
BASEMENTAREA_MODE               179943
BASEMENTAREA_AVG                179943
BASEMENTAREA_MEDI               179943
LANDAREA_MEDI                   182590
LANDAREA_AVG                    182590
LANDAREA_MODE                   182590
OWN_CAR_AGE                     202929
YEARS_BUILD_MODE                204488
YEARS_BUILD_AVG                 204488
YEARS_BUILD_MEDI                204488
FLOORSMIN_AVG                   208642
FLOORSMIN_MODE                  208642
FLOORSMIN_MEDI                  208642
LIVINGAPARTMENTS_AVG            210199
LIVINGAPARTMENTS_MODE           210199
LIVINGAPARTMENTS_MEDI           210199
FONDKAPREMONT_MODE              210295
NONLIVINGAPARTMENTS_AVG         213514
NONLIVINGAPARTMENTS_MEDI        213514
NONLIVINGAPARTMENTS_MODE        213514
COMMONAREA_MODE                 214865
COMMONAREA_AVG                  214865
COMMONAREA_MEDI                 214865
dtype: int64
```

In [8]:

```
# create new data frame and rename
missing_info =pd.DataFrame(apps.isnull().sum().sort_values()).reset_index()
missing_info.rename(columns={'index':'col_name',0:'null_count'},inplace=True)
missing_info.head()
```

Out[8]:

| | col_name | null_count |
|---|---|---|
| **0** | SK_ID_CURR | 0 |

| | col_name | null_count |
|---|---|---|
| 1 | HOUR_APPR_PROCESS_START | 0 |
| 2 | REG_REGION_NOT_WORK_REGION | 0 |
| 3 | LIVE_REGION_NOT_WORK_REGION | 0 |
| 4 | REG_CITY_NOT_LIVE_CITY | 0 |

In [9]:
```
missing_info['missing_perc'] =missing_info['null_count']/apps.shape[0]*100
missing_info.head()
```

Out[9]:

| | col_name | null_count | missing_perc |
|---|---|---|---|
| 0 | SK_ID_CURR | 0 | 0.0 |
| 1 | HOUR_APPR_PROCESS_START | 0 | 0.0 |
| 2 | REG_REGION_NOT_WORK_REGION | 0 | 0.0 |
| 3 | LIVE_REGION_NOT_WORK_REGION | 0 | 0.0 |
| 4 | REG_CITY_NOT_LIVE_CITY | 0 | 0.0 |

In [10]:
```
# Removed the missing columns
missing_col = missing_info[missing_info['missing_perc']>=40]['col_name'].to_list()
removed_missing = apps.drop(labels=missing_col,axis=1)
removed_missing.shape
```

Out[10]:
```
(307511, 73)
```

In [11]:
```
#list of flag columns
flag_col = []
for col in removed_missing.columns:
    if col.startswith("FLAG_"):
        flag_col.append(col)
len(flag_col)
```

Out[11]:
```
28
```

In [12]:
```
flag_col
```

Out[12]:
```
['FLAG_OWN_CAR',
 'FLAG_OWN_REALTY',
 'FLAG_MOBIL',
 'FLAG_EMP_PHONE',
 'FLAG_WORK_PHONE',
 'FLAG_CONT_MOBILE',
 'FLAG_PHONE',
 'FLAG_EMAIL',
 'FLAG_DOCUMENT_2',
 'FLAG_DOCUMENT_3',
 'FLAG_DOCUMENT_4',
 'FLAG_DOCUMENT_5',
 'FLAG_DOCUMENT_6',
 'FLAG_DOCUMENT_7',
 'FLAG_DOCUMENT_8',
 'FLAG_DOCUMENT_9',
 'FLAG_DOCUMENT_10',
 'FLAG_DOCUMENT_11',
 'FLAG_DOCUMENT_12',
 'T_13',
```

```
'FLAG_DOCUMENT_14',
'FLAG_DOCUMENT_15',
'FLAG_DOCUMENT_16',
'FLAG_DOCUMENT_17',
'FLAG_DOCUMENT_18',
'FLAG_DOCUMENT_19',
'FLAG_DOCUMENT_20',
'FLAG_DOCUMENT_21']
```

In [13]:

```
target_col = removed_missing[flag_col+['TARGET']]
target_col.head()
```

Out[13]:

| | FLAG_OWN_CAR | FLAG_OWN_REALTY | FLAG_MOBIL | FLAG_EMP_PHONE | FLAG_WORK_PHONE | FLAG_CONT_I |
|---|---|---|---|---|---|---|
| **0** | N | Y | 1 | 1 | 0 | |
| **1** | N | N | 1 | 1 | 0 | |
| **2** | Y | Y | 1 | 1 | 1 | |
| **3** | N | Y | 1 | 1 | 0 | |
| **4** | N | Y | 1 | 1 | 0 | |

In [14]:

```
#showing customer 0 is not defaulter and 1 defaulter assumption target values
fig = plt.figure(figsize=(20,25))
for i, col in enumerate(target_col):
    plt.subplot(6,5,i+1)
    sns.countplot(data=target_col,x=col,hue='TARGET')
fig.show()
```


In [15]:
```
#correlation in defaulter
corr_df =['FLAG_OWN_CAR','FLAG_OWN_REALTY','FLAG_MOBIL','FLAG_EMP_PHONE','FLAG_WORK_PHONE'
        'FLAG_PHONE','FLAG_EMAIL','TARGET']
flag_corr_df =removed_missing[corr_df]
```

In [16]:
```
flag_corr_df.groupby(['FLAG_OWN_CAR']).size()
```

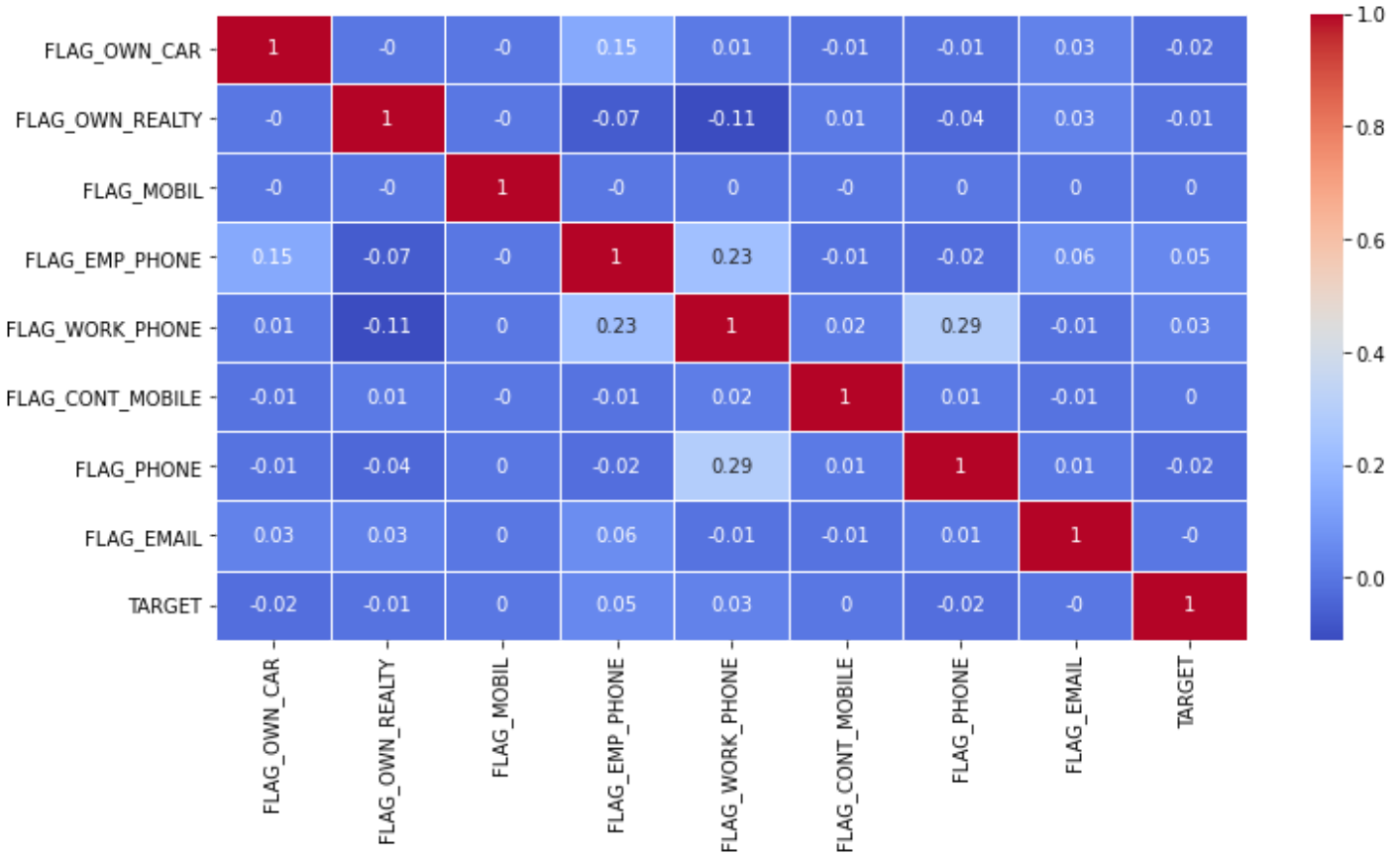
Out[16]:
```
FLAG_OWN_CAR
N    202924
```

```
Y    104587
dtype: int64
```

In [17]:
```
flag_corr_df['FLAG_OWN_CAR']=flag_corr_df['FLAG_OWN_CAR'].replace({'N':0,'Y':1})
flag_corr_df['FLAG_OWN_REALTY']=flag_corr_df['FLAG_OWN_REALTY'].replace({'N':0,'Y':1})
```

In [18]:
```
# check correlaition
corr_df = round(flag_corr_df.corr(),2)
plt.figure(figsize=(12,6))
sns.heatmap(corr_df,cmap='coolwarm',linewidths=.5,annot=True)
```

Out[18]: <AxesSubplot:>

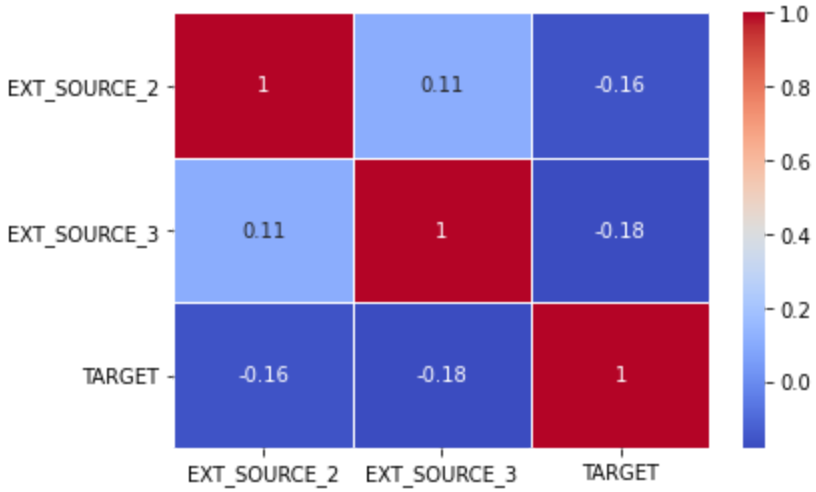



In [19]:
```
# drop flag all columns
data_drop = removed_missing.drop(labels=flag_col,axis=1)
data_drop.shape
```

Out[19]: (307511, 45)

In [20]:
```
sns.heatmap(data=round(data_drop[['EXT_SOURCE_2','EXT_SOURCE_3','TARGET']].corr(),2),cmap=
```

Out[20]: <AxesSubplot:>




In [21]:
```
drop_colums= data_drop.drop(['EXT_SOURCE_2','EXT_SOURCE_3'],axis=1)
drop_colums.shape
```

Out[21]: (307511, 43)

# FEATURE ENGINEERING

In [22]:
```
#check missing values
drop_colums.isnull().sum().sort_values()
```

Out[22]:
```
SK_ID_CURR                        0
ORGANIZATION_TYPE                 0
LIVE_CITY_NOT_WORK_CITY           0
REG_CITY_NOT_WORK_CITY            0
REG_CITY_NOT_LIVE_CITY            0
LIVE_REGION_NOT_WORK_REGION       0
REG_REGION_NOT_WORK_REGION        0
REG_REGION_NOT_LIVE_REGION        0
HOUR_APPR_PROCESS_START           0
WEEKDAY_APPR_PROCESS_START        0
REGION_RATING_CLIENT_W_CITY       0
DAYS_ID_PUBLISH                   0
DAYS_REGISTRATION                 0
DAYS_EMPLOYED                     0
DAYS_BIRTH                        0
REGION_RATING_CLIENT              0
NAME_HOUSING_TYPE                 0
TARGET                            0
NAME_CONTRACT_TYPE                0
REGION_POPULATION_RELATIVE        0
CNT_CHILDREN                      0
AMT_INCOME_TOTAL                  0
AMT_CREDIT                        0
CODE_GENDER                       0
NAME_INCOME_TYPE                  0
NAME_EDUCATION_TYPE               0
NAME_FAMILY_STATUS                0
DAYS_LAST_PHONE_CHANGE            1
CNT_FAM_MEMBERS                   2
AMT_ANNUITY                      12
AMT_GOODS_PRICE                 278
DEF_60_CNT_SOCIAL_CIRCLE       1021
OBS_60_CNT_SOCIAL_CIRCLE       1021
DEF_30_CNT_SOCIAL_CIRCLE       1021
OBS_30_CNT_SOCIAL_CIRCLE       1021
NAME_TYPE_SUITE                1292
_BUREAU_QRT                   41519
```

```
AMT_REQ_CREDIT_BUREAU_HOUR     41519
AMT_REQ_CREDIT_BUREAU_DAY      41519
AMT_REQ_CREDIT_BUREAU_WEEK     41519
AMT_REQ_CREDIT_BUREAU_MON      41519
AMT_REQ_CREDIT_BUREAU_YEAR     41519
OCCUPATION_TYPE                96391
dtype: int64
```

In [23]:
```
# fill missing values ((drop_colums['CNT_FAM_MEMBERS'].mode()[0]),inplace=True)
drop_colums['CNT_FAM_MEMBERS']=drop_colums['CNT_FAM_MEMBERS'].fillna(drop_colums['CNT_FAM_
```

In [24]:
```
drop_colums['CNT_FAM_MEMBERS'].isnull().sum()
```

Out[24]:
```
0
```

In [25]:
```
# convert string to integer values
drop_colums['OCCUPATION_TYPE'] = pd.to_numeric(drop_colums['OCCUPATION_TYPE'], errors='coe
```

In [26]:
```
drop_colums['OCCUPATION_TYPE']=drop_colums['OCCUPATION_TYPE'].fillna(drop_colums['OCCUPATI
```

In [27]:
```
drop_colums['OCCUPATION_TYPE'].isnull().sum()
```

Out[27]:
```
307511
```

In [28]:
```
drop_colums.groupby(['OCCUPATION_TYPE']).size().sort_values()
```

Out[28]:
```
Series([], dtype: int64)
```

In [29]:
```
drop_colums['NAME_TYPE_SUITE'] = pd.to_numeric(drop_colums['NAME_TYPE_SUITE'], errors='coe
```

In [30]:
```
drop_colums['NAME_TYPE_SUITE']=drop_colums['NAME_TYPE_SUITE'].fillna(drop_colums['NAME_TYP
```

In [31]:
```
drop_colums['NAME_TYPE_SUITE'].isnull().sum()
```

Out[31]:
```
307511
```

In [32]:
```
drop_colums.groupby(['NAME_TYPE_SUITE']).size().sort_values()
```

Out[32]:
```
Series([], dtype: int64)
```

In [33]:
```
drop_colums['DAYS_LAST_PHONE_CHANGE']=drop_colums['DAYS_LAST_PHONE_CHANGE'].fillna(drop_co
```

In [34]:
```
drop_colums['DAYS_LAST_PHONE_CHANGE'].isnull().sum()
```

Out[34]:
```
0
```

In [35]:
```
drop_colums['AMT_ANNUITY']=drop_colums['AMT_ANNUITY'].fillna(drop_colums['AMT_ANNUITY'].me
```

In [36]:
```
drop_colums['AMT_ANNUITY'].isnull().sum()
```

Out[36]:
```
0
```

In [37]:
```
amount_credit = []
for col in drop_colums.columns:
    if col.startswith('AMT_REQ_CREDIT_BUREAU_HOUR'):
        amount_credit.append(col)
amount_credit
```

Out[37]:
```
['AMT_REQ_CREDIT_BUREAU_HOUR']
```

In [38]:
```
for col in amount_credit:
    drop_colums[col]=drop_colums[col].fillna((drop_colums[col].median()))
```

In [39]:
```
drop_colums['AMT_REQ_CREDIT_BUREAU_HOUR'].isnull().sum()
```

Out[39]:
```
0
```

In [40]:
```
# DATA ANALYSIS
```

In [41]:
```
drop_colums.dtypes.value_counts()
```

Out[41]:
```
float64    18
int64      15
object      8
Int64       2
dtype: int64
```

In [42]:
```
drop_colums.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Data columns (total 43 columns):
 #   Column                       Non-Null Count   Dtype
---  ------                       --------------   -----
 0   SK_ID_CURR                   307511 non-null  int64
 1   TARGET                       307511 non-null  int64
 2   NAME_CONTRACT_TYPE           307511 non-null  object
 3   CODE_GENDER                  307511 non-null  object
 4   CNT_CHILDREN                 307511 non-null  int64
 5   AMT_INCOME_TOTAL             307511 non-null  float64
 6   AMT_CREDIT                   307511 non-null  float64
 7   AMT_ANNUITY                  307511 non-null  float64
 8   AMT_GOODS_PRICE              307233 non-null  float64
 9   NAME_TYPE_SUITE              0 non-null       Int64
 10  NAME_INCOME_TYPE             307511 non-null  object
 11  NAME_EDUCATION_TYPE          307511 non-null  object
 12  NAME_FAMILY_STATUS           307511 non-null  object
 13  NAME_HOUSING_TYPE            307511 non-null  object
 14  REGION_POPULATION_RELATIVE   307511 non-null  float64
 15  DAYS_BIRTH                   307511 non-null  int64
 16  DAYS_EMPLOYED                307511 non-null  int64
 17  DAYS_REGISTRATION            307511 non-null  float64
 18  DAYS_ID_PUBLISH              307511 non-null  int64
 19  OCCUPATION_TYPE              0 non-null       Int64
 20  CNT_FAM_MEMBERS              307511 non-null  float64
     TING_CLIENT                  307511 non-null  int64
```

```
 22  REGION_RATING_CLIENT_W_CITY  307511 non-null  int64
 23  WEEKDAY_APPR_PROCESS_START   307511 non-null  object
 24  HOUR_APPR_PROCESS_START      307511 non-null  int64
 25  REG_REGION_NOT_LIVE_REGION   307511 non-null  int64
 26  REG_REGION_NOT_WORK_REGION   307511 non-null  int64
 27  LIVE_REGION_NOT_WORK_REGION  307511 non-null  int64
 28  REG_CITY_NOT_LIVE_CITY       307511 non-null  int64
 29  REG_CITY_NOT_WORK_CITY       307511 non-null  int64
 30  LIVE_CITY_NOT_WORK_CITY      307511 non-null  int64
 31  ORGANIZATION_TYPE            307511 non-null  object
 32  OBS_30_CNT_SOCIAL_CIRCLE     306490 non-null  float64
 33  DEF_30_CNT_SOCIAL_CIRCLE     306490 non-null  float64
 34  OBS_60_CNT_SOCIAL_CIRCLE     306490 non-null  float64
 35  DEF_60_CNT_SOCIAL_CIRCLE     306490 non-null  float64
 36  DAYS_LAST_PHONE_CHANGE       307511 non-null  float64
 37  AMT_REQ_CREDIT_BUREAU_HOUR   307511 non-null  float64
 38  AMT_REQ_CREDIT_BUREAU_DAY    265992 non-null  float64
 39  AMT_REQ_CREDIT_BUREAU_WEEK   265992 non-null  float64
 40  AMT_REQ_CREDIT_BUREAU_MON    265992 non-null  float64
 41  AMT_REQ_CREDIT_BUREAU_QRT    265992 non-null  float64
 42  AMT_REQ_CREDIT_BUREAU_YEAR   265992 non-null  float64
dtypes: Int64(2), float64(18), int64(15), object(8)
memory usage: 101.5+ MB
```

In [43]:
```
#categorical data
drop_colums.select_dtypes(include=['object']).columns
```

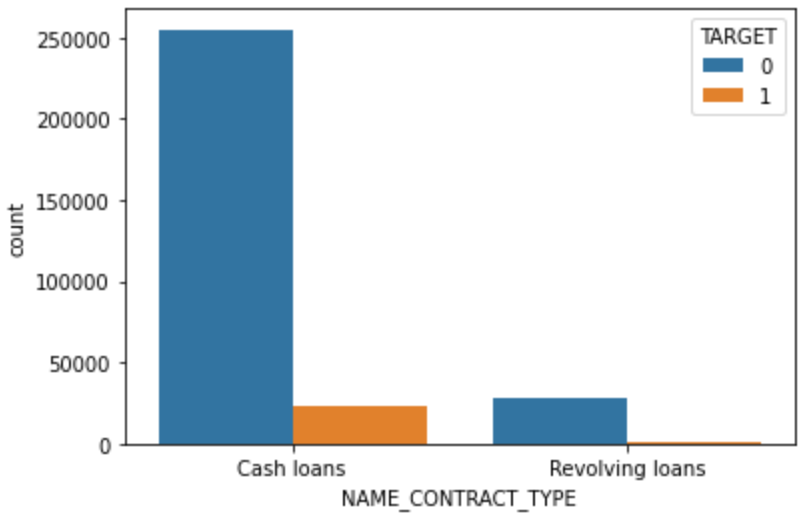
Out[43]:
```
Index(['NAME_CONTRACT_TYPE', 'CODE_GENDER', 'NAME_INCOME_TYPE',
       'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE',
       'WEEKDAY_APPR_PROCESS_START', 'ORGANIZATION_TYPE'],
      dtype='object')
```

In [44]:
```
drop_colums.groupby(['NAME_CONTRACT_TYPE']).size()
```

Out[44]:
```
NAME_CONTRACT_TYPE
Cash loans         278232
Revolving loans     29279
dtype: int64
```

In [45]:
```
sns.countplot(data=drop_colums,x='NAME_CONTRACT_TYPE', hue='TARGET')
```

Out[45]:
```
<AxesSubplot:xlabel='NAME_CONTRACT_TYPE', ylabel='count'>
```




In [46]:
```
data_type = drop_colums[['NAME_CONTRACT_TYPE','TARGET']].groupby(['NAME_CONTRACT_TYPE'],as
data_type
```

Out[46]:

| | NAME_CONTRACT_TYPE | TARGET |
|---|---|---|
| 0 | Cash loans | 0.083459 |
| 1 | Revolving loans | 0.054783 |

In [47]:

```
data_type['PCT'] = data_type['TARGET']*100
data_type
```

Out[47]:

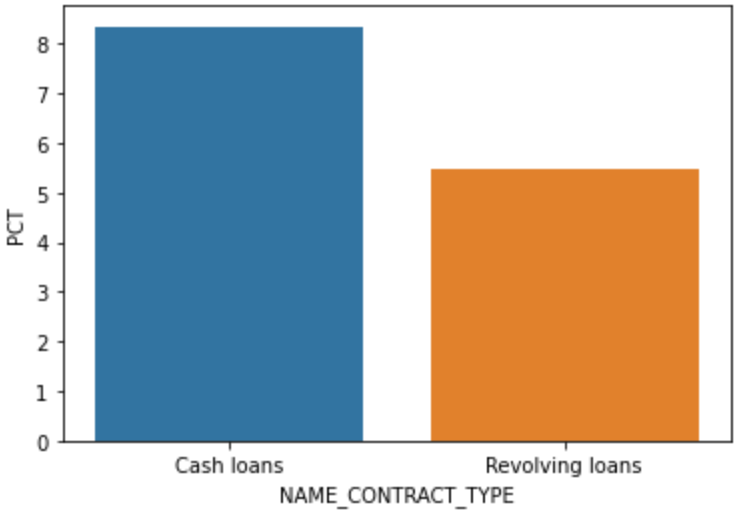
| | NAME_CONTRACT_TYPE | TARGET | PCT |
|---|---|---|---|
| 0 | Cash loans | 0.083459 | 8.345913 |
| 1 | Revolving loans | 0.054783 | 5.478329 |

In [48]:

```
sns.barplot(data=data_type, x = 'NAME_CONTRACT_TYPE', y = 'PCT')
```

Out[48]:

```
<AxesSubplot:xlabel='NAME_CONTRACT_TYPE', ylabel='PCT'>
```

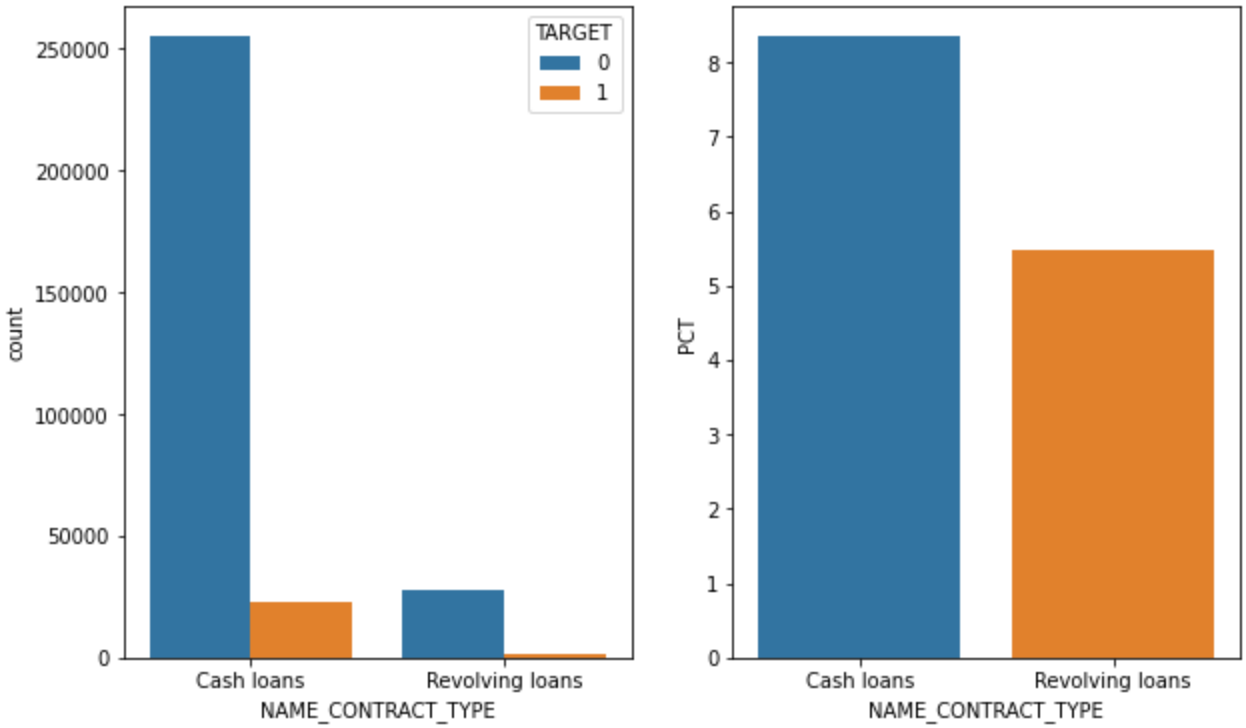



In [49]:

```
plt.figure(figsize=(10,6))
plt.subplot(1,2,1)
sns.countplot(data=drop_colums,x='NAME_CONTRACT_TYPE', hue='TARGET')
plt.subplot(1,2,2)
sns.barplot(data=data_type, x = 'NAME_CONTRACT_TYPE', y = 'PCT')
```

Out[49]:

```
<AxesSubplot:xlabel='NAME_CONTRACT_TYPE', ylabel='PCT'>
```

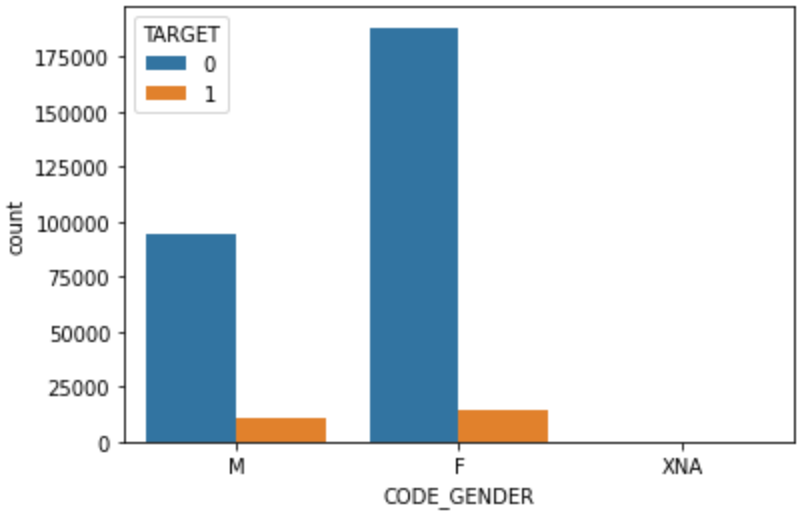



In [50]:
```
drop_colums.groupby(['CODE_GENDER']).size()
```

Out[50]:
```
CODE_GENDER
F      202448
M      105059
XNA         4
dtype: int64
```

In [51]:
```
sns.countplot(data=drop_colums,x='CODE_GENDER', hue='TARGET')
```

Out[51]:
```
<AxesSubplot:xlabel='CODE_GENDER', ylabel='count'>
```




In [52]:
```
data_tool = drop_colums[['CODE_GENDER','TARGET']].groupby(['CODE_GENDER'],as_index=False).
data_tool
```

Out[52]:

| | CODE_GENDER | TARGET |
|---|---|---|
| **0** | F | 0.069993 |
| **1** | M | 0.101419 |
| | A | 0.000000 |

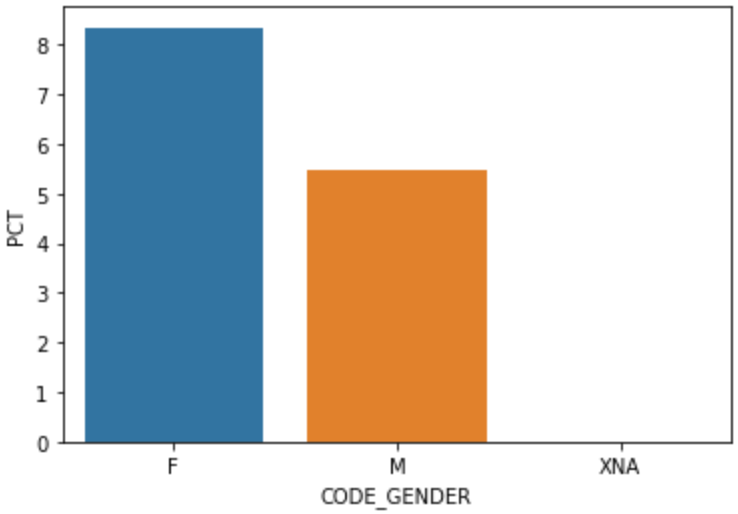
In [53]:
```
data_tool['PCT'] = data_type['TARGET']*100
data_tool
```

Out[53]:

| | CODE_GENDER | TARGET | PCT |
|---|---|---|---|
| **0** | F | 0.069993 | 8.345913 |
| **1** | M | 0.101419 | 5.478329 |
| **2** | XNA | 0.000000 | NaN |

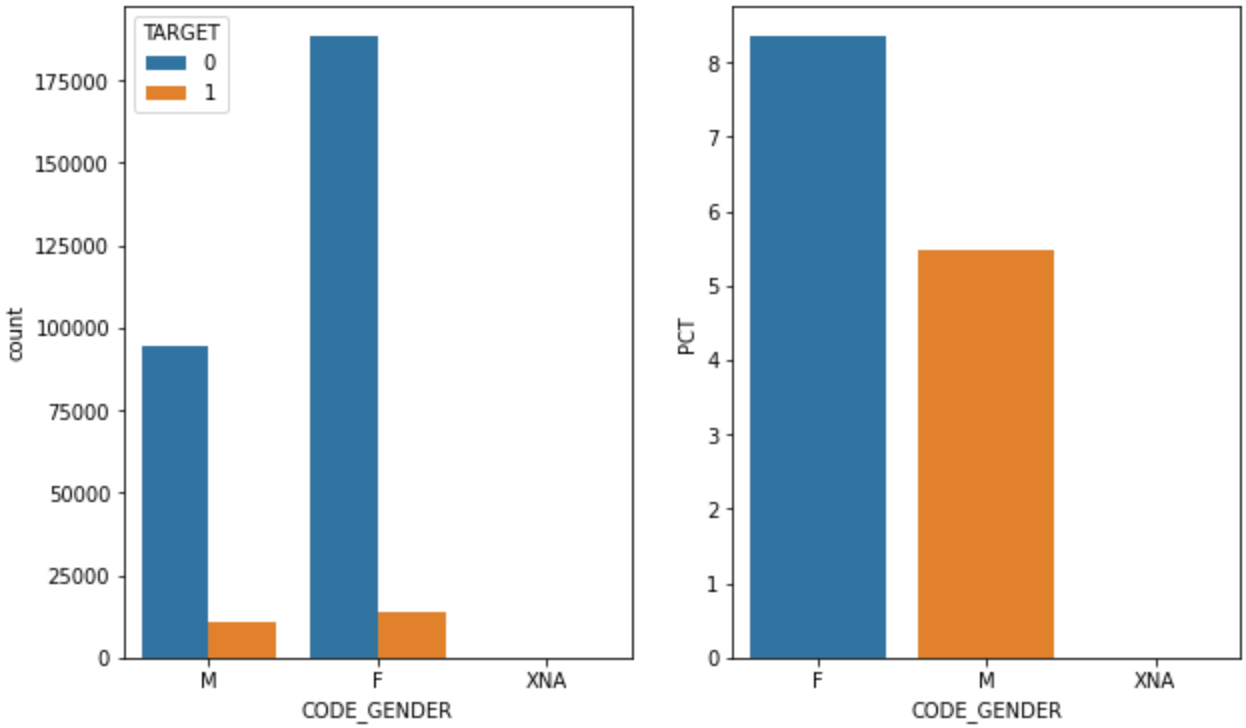
In [54]:
```
sns.barplot(data=data_tool, x = 'CODE_GENDER', y = 'PCT')
```

Out[54]:
```
<AxesSubplot:xlabel='CODE_GENDER', ylabel='PCT'>
```




In [55]:
```
plt.figure(figsize=(10,6))
plt.subplot(1,2,1)
sns.countplot(data=drop_colums,x='CODE_GENDER', hue='TARGET')
plt.subplot(1,2,2)
sns.barplot(data=data_tool, x = 'CODE_GENDER', y = 'PCT')
```

Out[55]:
```
<AxesSubplot:xlabel='CODE_GENDER', ylabel='PCT'>
```




In [56]:
```
drop_colums['NAME_EDUCATION_TYPE'].unique()
```

Out[56]:
```
array(['Secondary / secondary special', 'Higher education',
       'Incomplete higher', 'Lower secondary', 'Academic degree'],
      dtype=object)
```

In [57]:
```
drop_colums.dtypes.value_counts()
```

Out[57]:
```
float64    18
int64      15
object      8
Int64       2
dtype: int64
```

In [58]:
```
#NUMIRICAL VARIABLE DATA TYPES
```

In [59]:
```
num_var = drop_colums.select_dtypes(include=['float64','int64']).columns
num_var
```

Out[59]:
```
Index(['SK_ID_CURR', 'TARGET', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY', 'AMT_GOODS_PRICE', 'NAME_TYPE_SUITE',
       'REGION_POPULATION_RELATIVE', 'DAYS_BIRTH', 'DAYS_EMPLOYED',
       'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH', 'OCCUPATION_TYPE',
       'CNT_FAM_MEMBERS', 'REGION_RATING_CLIENT',
       'REGION_RATING_CLIENT_W_CITY', 'HOUR_APPR_PROCESS_START',
       'REG_REGION_NOT_LIVE_REGION', 'REG_REGION_NOT_WORK_REGION',
       'LIVE_REGION_NOT_WORK_REGION', 'REG_CITY_NOT_LIVE_CITY',
       'REG_CITY_NOT_WORK_CITY', 'LIVE_CITY_NOT_WORK_CITY',
       'OBS_30_CNT_SOCIAL_CIRCLE', 'DEF_30_CNT_SOCIAL_CIRCLE',
       'OBS_60_CNT_SOCIAL_CIRCLE', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'DAYS_LAST_PHONE_CHANGE', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'],
      dtype='object')
```

In [60]:
```
len(num_var)
```

Out[60]: 35

In [61]:
```
# find out percentage of defaulter an dnon defaulter
num_data = drop_colums[num_var]
num_data.groupby(['TARGET']).size()/num_data.shape[0]*100
```

Out[61]:
```
TARGET
0    91.927118
1     8.072882
dtype: float64
```

In [62]:
```
num_data = drop_colums[num_var]
defaulter = num_data[num_data['TARGET']==1].drop(['TARGET'],axis=1)
non_defaulter = num_data[num_data['TARGET']==0].drop(['TARGET'],axis=1)
non_defaulter.head()
```

Out[62]:

| | SK_ID_CURR | CNT_CHILDREN | AMT_INCOME_TOTAL | AMT_CREDIT | AMT_ANNUITY | AMT_GOODS_PRICE | NAME |
|---|---|---|---|---|---|---|---|
| **1** | 100003 | 0 | 270000.0 | 1293502.5 | 35698.5 | 1129500.0 | |
| **2** | 100004 | 0 | 67500.0 | 135000.0 | 6750.0 | 135000.0 | |
| **3** | 100006 | 0 | 135000.0 | 312682.5 | 29686.5 | 297000.0 | |
| **4** | 100007 | 0 | 121500.0 | 513000.0 | 21865.5 | 513000.0 | |
| **5** | 100008 | 0 | 99000.0 | 490495.5 | 27517.5 | 454500.0 | |

In [63]:
```
defaulter.head()
```

Out[63]:

| | SK_ID_CURR | CNT_CHILDREN | AMT_INCOME_TOTAL | AMT_CREDIT | AMT_ANNUITY | AMT_GOODS_PRICE | NAM |
|---|---|---|---|---|---|---|---|
| **0** | 100002 | 0 | 202500.0 | 406597.5 | 24700.5 | 351000.0 | |
| **26** | 100031 | 0 | 112500.0 | 979992.0 | 27076.5 | 702000.0 | |
| **40** | 100047 | 0 | 202500.0 | 1193580.0 | 35028.0 | 855000.0 | |
| **42** | 100049 | 0 | 135000.0 | 288873.0 | 16258.5 | 238500.0 | |
| **81** | 100096 | 0 | 81000.0 | 252000.0 | 14593.5 | 252000.0 | |

In [64]:
```
# corelation of each other data set
```

In [65]:
```
defaulter[['SK_ID_CURR','CNT_CHILDREN','AMT_INCOME_TOTAL']].corr()
```

Out[65]:

| | SK_ID_CURR | CNT_CHILDREN | AMT_INCOME_TOTAL |
|---|---|---|---|
| **SK_ID_CURR** | 1.000000 | -0.005144 | -0.010165 |
| **CNT_CHILDREN** | -0.005144 | 1.000000 | 0.004796 |
| **AMT_INCOME_TOTAL** | -0.010165 | 0.004796 | 1.000000 |

In [66]:
```
defaulters_corr = defaulter.corr()
defaulter_corr_untack = defaulters_corr.where(np.triu(np.ones(defaulters_corr.shape),k=1)
defaulter_corr_untack['corr'] = abs(defaulter_corr_untack['corr'])
defaulter_corr_untack.dropna(subset=['corr']).sort_values(by=['corr'], ascending = False)
```

Out[66]:

| | var_1 | var_2 | corr |
|---|---|---|---|
| **873** | OBS_60_CNT_SOCIAL_CIRCLE | OBS_30_CNT_SOCIAL_CIRCLE | 0.998269 |
| **173** | AMT_GOODS_PRICE | AMT_CREDIT | 0.983103 |
| **524** | REGION_RATING_CLIENT_W_CITY | REGION_RATING_CLIENT | 0.956637 |
| **443** | CNT_FAM_MEMBERS | CNT_CHILDREN | 0.885484 |
| **908** | DEF_60_CNT_SOCIAL_CIRCLE | DEF_30_CNT_SOCIAL_CIRCLE | 0.868994 |
| **664** | LIVE_REGION_NOT_WORK_REGION | REG_REGION_NOT_WORK_REGION | 0.847885 |
| **769** | LIVE_CITY_NOT_WORK_CITY | REG_CITY_NOT_WORK_CITY | 0.778540 |
| **174** | AMT_GOODS_PRICE | AMT_ANNUITY | 0.752699 |
| **139** | AMT_ANNUITY | AMT_CREDIT | 0.752195 |
| **314** | DAYS_EMPLOYED | DAYS_BIRTH | 0.575097 |

In [67]:

```
non_defaulters_corr = non_defaulter.corr()
non_defaulter_corr_untack = non_defaulters_corr.where(np.triu(np.ones(non_defaulters_corr
non_defaulter_corr_untack['corr'] = abs(non_defaulter_corr_untack['corr'])
non_defaulter_corr_untack.dropna(subset=['corr']).sort_values(by=['corr'], ascending = Fal
```

Out[67]:

| | var_1 | var_2 | corr |
|---|---|---|---|
| **873** | OBS_60_CNT_SOCIAL_CIRCLE | OBS_30_CNT_SOCIAL_CIRCLE | 0.998508 |
| **173** | AMT_GOODS_PRICE | AMT_CREDIT | 0.987250 |
| **524** | REGION_RATING_CLIENT_W_CITY | REGION_RATING_CLIENT | 0.950149 |
| **443** | CNT_FAM_MEMBERS | CNT_CHILDREN | 0.878570 |
| **664** | LIVE_REGION_NOT_WORK_REGION | REG_REGION_NOT_WORK_REGION | 0.861861 |
| **908** | DEF_60_CNT_SOCIAL_CIRCLE | DEF_30_CNT_SOCIAL_CIRCLE | 0.859332 |
| **769** | LIVE_CITY_NOT_WORK_CITY | REG_CITY_NOT_WORK_CITY | 0.830381 |
| **174** | AMT_GOODS_PRICE | AMT_ANNUITY | 0.776674 |
| **139** | AMT_ANNUITY | AMT_CREDIT | 0.771297 |
| **314** | DAYS_EMPLOYED | DAYS_BIRTH | 0.618048 |

In [68]:

```
num_data.head()
```

Out[68]:

| | SK_ID_CURR | TARGET | CNT_CHILDREN | AMT_INCOME_TOTAL | AMT_CREDIT | AMT_ANNUITY | AMT_GOODS_PRI |
|---|---|---|---|---|---|---|---|
| **0** | 100002 | 1 | 0 | 202500.0 | 406597.5 | 24700.5 | 351000 |
| **1** | 100003 | 0 | 0 | 270000.0 | 1293502.5 | 35698.5 | 1129500 |
| **2** | 100004 | 0 | 0 | 67500.0 | 135000.0 | 6750.0 | 135000 |
| **3** | 100006 | 0 | 0 | 135000.0 | 312682.5 | 29686.5 | 297000 |
| **4** | 100007 | 0 | 0 | 121500.0 | 513000.0 | 21865.5 | 513000 |

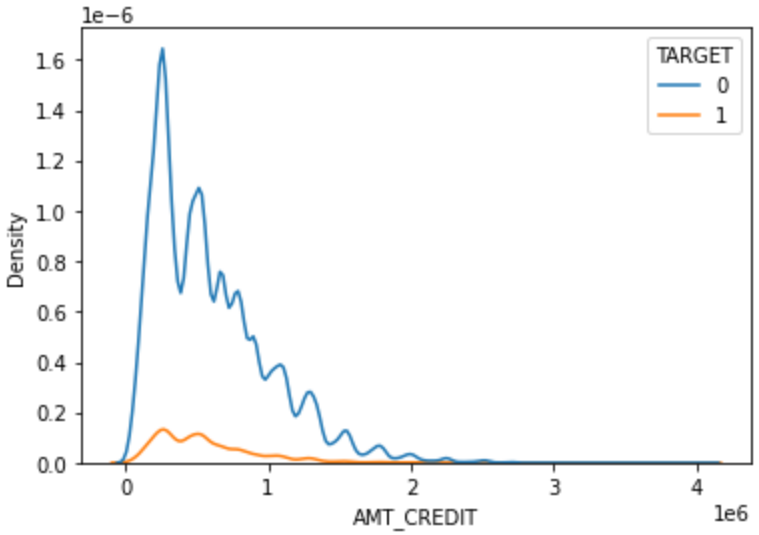
In [69]:

```
amount_var = ['AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY','AMT_GOODS_PRICE']
```

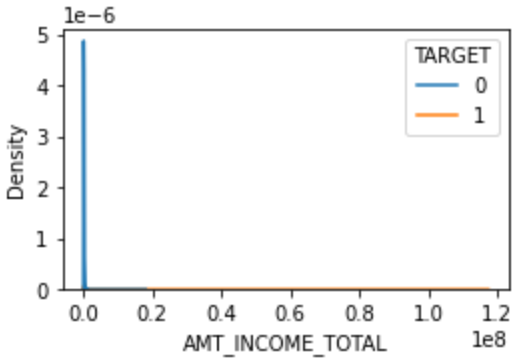
In [70]:

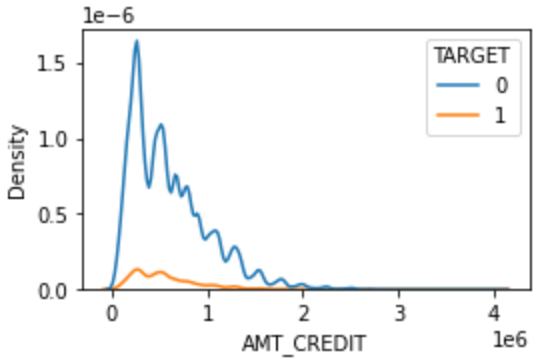
```
sns.kdeplot(data = num_data, x = 'AMT_CREDIT', hue = 'TARGET')
```

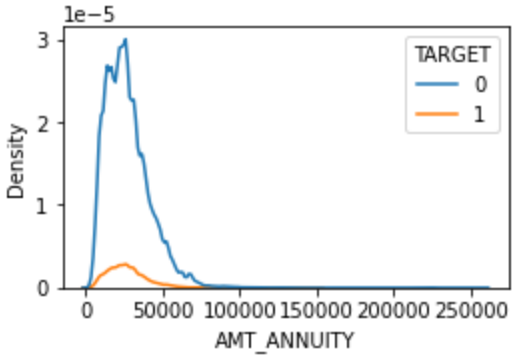
Out[70]: <AxesSubplot:xlabel='AMT_CREDIT', ylabel='Density'>

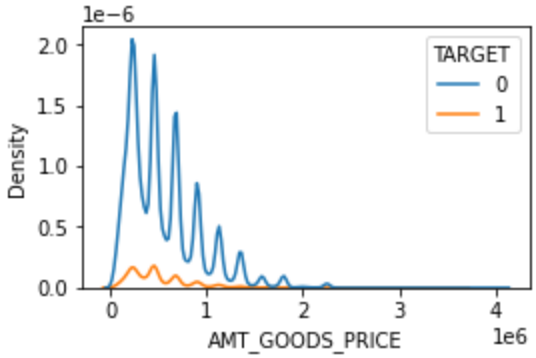



In [71]:
```
# UNIVAVRATE NUMRICAL VARIABLE ANALYSIS
plt.figure(figsize = (10,6))
for i, col in enumerate(amount_var):
    plt.subplot(2,2,i+1)
    sns.kdeplot(data = num_data, x = col, hue = 'TARGET')
    plt.subplots_adjust(wspace = 0.5, hspace = 0.5)
```



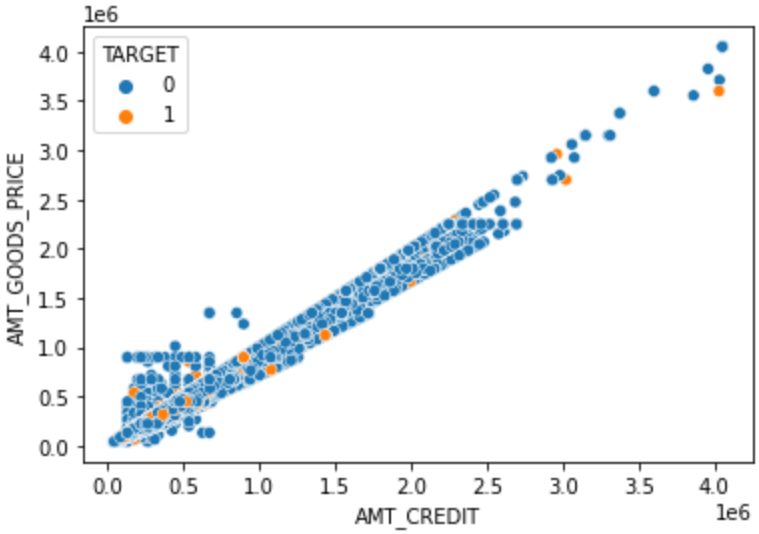



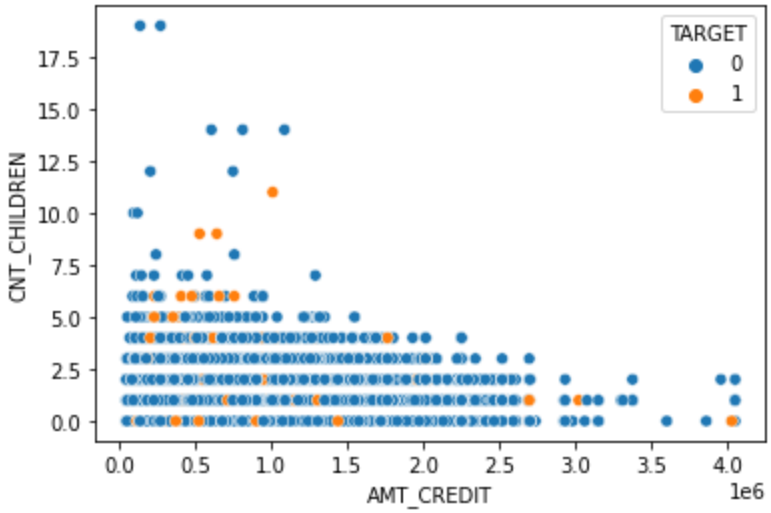

In [72]:
```
#BIVIRAIATE DATA ANALYSIS
sns.scatterplot(data = num_data, x = 'AMT_CREDIT', y = 'AMT_GOODS_PRICE', hue = 'TARGET')
```

Out[72]: <AxesSubplot:xlabel='AMT_CREDIT', ylabel='AMT_GOODS_PRICE'>




In [73]:
```
sns.scatterplot(data = num_data, x = 'AMT_CREDIT', y = 'CNT_CHILDREN', hue = 'TARGET')
```

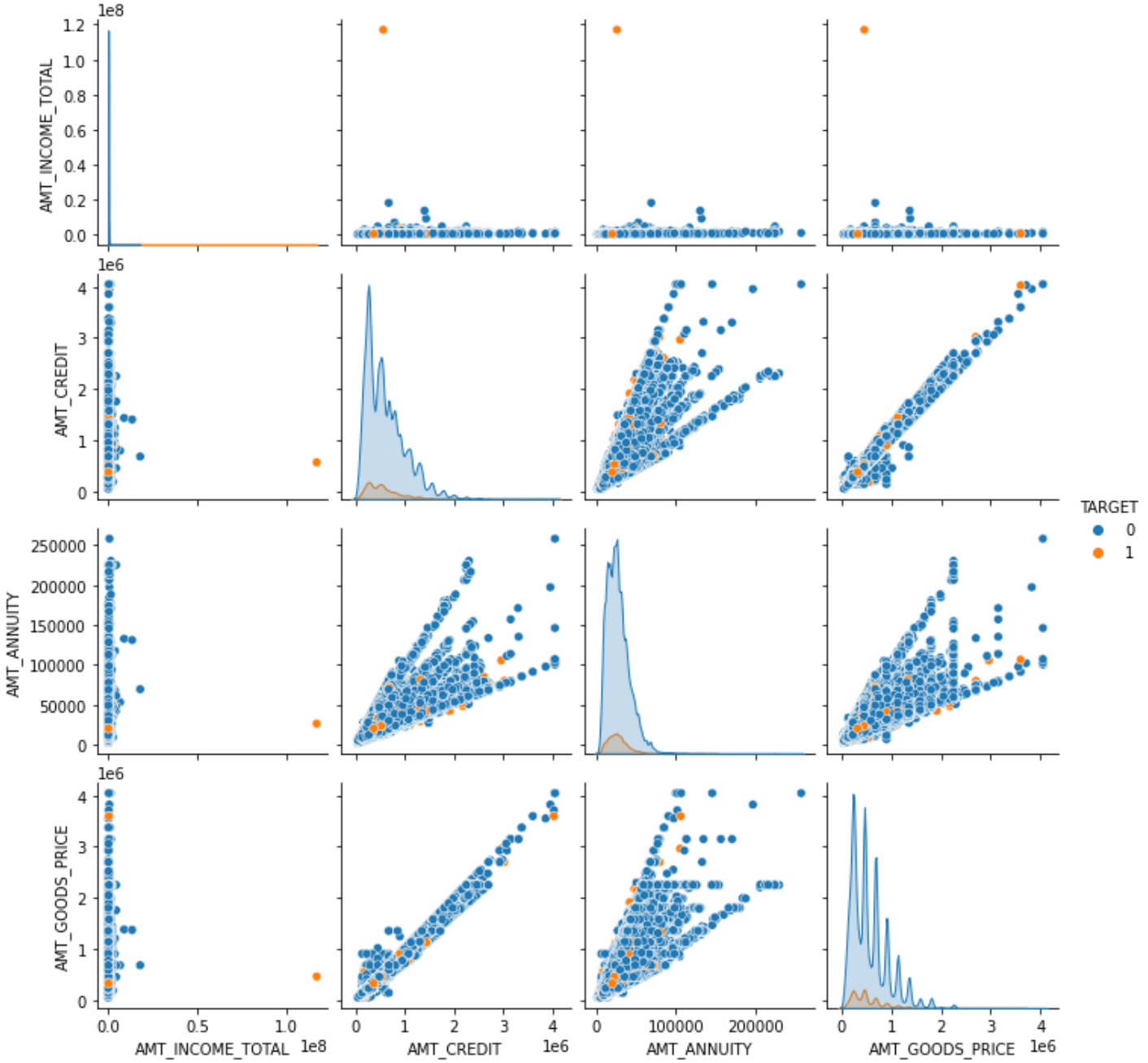
Out[73]:
```
<AxesSubplot:xlabel='AMT_CREDIT', ylabel='CNT_CHILDREN'>
```




In [74]:
```
amount_data = num_data[['AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY','AMT_GOODS_PRICE','T
```

In [75]:
```
sns.pairplot(data = amount_data, hue = 'TARGET')
```

Out[75]:
```
<seaborn.axisgrid.PairGrid at 0x18a6136a820>
```




In [76]:
```
null_value = pd.DataFrame(loan_defaulter.isnull().sum().sort_values(ascending = False)/loa
manage_data = list(null_value[null_value['count_pct']>=50]['var'])
manage_data
```

Out[76]:
```
['RATE_INTEREST_PRIVILEGED',
 'RATE_INTEREST_PRIMARY',
 'AMT_DOWN_PAYMENT',
 'RATE_DOWN_PAYMENT']
```

In [77]:
```
non_data = manage_data+['WEEKDAY_APPR_PROCESS_START','HOUR_APPR_PROCESS_START','FLAG_LAST_
print(non_data)
len(non_data)
```

```
['RATE_INTEREST_PRIVILEGED', 'RATE_INTEREST_PRIMARY', 'AMT_DOWN_PAYMENT', 'RATE_DOWN_PAYME
NT', 'WEEKDAY_APPR_PROCESS_START', 'HOUR_APPR_PROCESS_START', 'FLAG_LAST_APPL_PER_CONTRAC
T', 'NFLAG_LAST_APPL_IN_DAY']
```

Out[77]:
```
8
```

In [78]:
```
len(loan_defaulter.columns)
```

Out[78]: 37

In [79]:
```
pre_app_non = loan_defaulter.drop(labels=non_data,axis = 1)
len(pre_app_non.columns)
```

Out[79]: 29

In [80]:
```
pre_app_non.columns
```

Out[80]:
```
Index(['SK_ID_PREV', 'SK_ID_CURR', 'NAME_CONTRACT_TYPE', 'AMT_ANNUITY',
       'AMT_APPLICATION', 'AMT_CREDIT', 'AMT_GOODS_PRICE',
       'NAME_CASH_LOAN_PURPOSE', 'NAME_CONTRACT_STATUS', 'DAYS_DECISION',
       'NAME_PAYMENT_TYPE', 'CODE_REJECT_REASON', 'NAME_TYPE_SUITE',
       'NAME_CLIENT_TYPE', 'NAME_GOODS_CATEGORY', 'NAME_PORTFOLIO',
       'NAME_PRODUCT_TYPE', 'CHANNEL_TYPE', 'SELLERPLACE_AREA',
       'NAME_SELLER_INDUSTRY', 'CNT_PAYMENT', 'NAME_YIELD_GROUP',
       'PRODUCT_COMBINATION', 'DAYS_FIRST_DRAWING', 'DAYS_FIRST_DUE',
       'DAYS_LAST_DUE_1ST_VERSION', 'DAYS_LAST_DUE', 'DAYS_TERMINATION',
       'NFLAG_INSURED_ON_APPROVAL'],
      dtype='object')
```

In [81]:
```
pre_app_non.isnull().sum().sort_values(ascending=False)/pre_app_non.shape[0]*100
```

Out[81]:
```
NAME_TYPE_SUITE              49.119754
NFLAG_INSURED_ON_APPROVAL    40.298129
DAYS_TERMINATION             40.298129
DAYS_LAST_DUE                40.298129
DAYS_LAST_DUE_1ST_VERSION    40.298129
DAYS_FIRST_DUE               40.298129
DAYS_FIRST_DRAWING           40.298129
AMT_GOODS_PRICE              23.081773
AMT_ANNUITY                  22.286665
CNT_PAYMENT                  22.286366
PRODUCT_COMBINATION           0.020716
AMT_CREDIT                    0.000060
CHANNEL_TYPE                  0.000000
NAME_YIELD_GROUP              0.000000
NAME_SELLER_INDUSTRY          0.000000
SELLERPLACE_AREA              0.000000
SK_ID_PREV                    0.000000
NAME_PRODUCT_TYPE             0.000000
NAME_PORTFOLIO                0.000000
SK_ID_CURR                    0.000000
NAME_CLIENT_TYPE              0.000000
CODE_REJECT_REASON            0.000000
NAME_PAYMENT_TYPE             0.000000
DAYS_DECISION                 0.000000
NAME_CONTRACT_STATUS          0.000000
NAME_CASH_LOAN_PURPOSE        0.000000
AMT_APPLICATION               0.000000
NAME_CONTRACT_TYPE            0.000000
NAME_GOODS_CATEGORY           0.000000
dtype: float64
```

In [82]:
```
pre_app_non['AMT_GOODS_PRICE'].agg(func=['mean','median'])
```

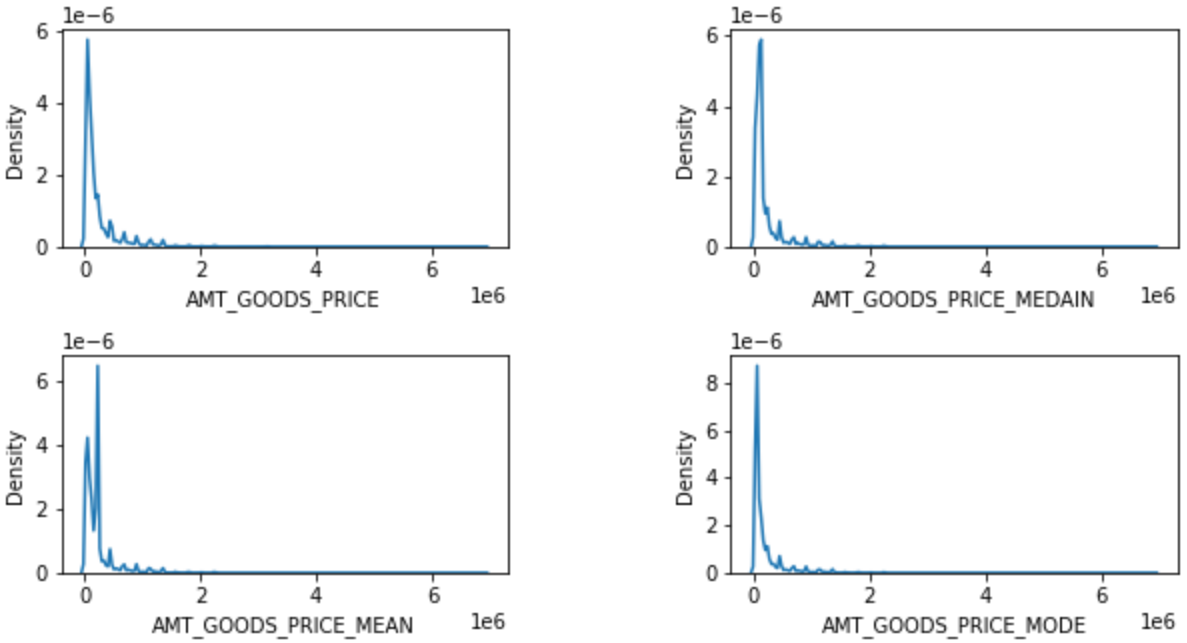
Out[82]:
```
mean      227847.279283
median    112320.000000
Name: AMT_GOODS_PRICE, dtype: float64
```

In [83]:
```
pre_app_non['AMT_GOODS_PRICE_MEDAIN'] = pre_app_non['AMT_GOODS_PRICE'].fillna(pre_app_non[
pre_app_non['AMT_GOODS_PRICE_MEAN'] = pre_app_non['AMT_GOODS_PRICE'].fillna(pre_app_non['A
```

```
pre_app_non['AMT_GOODS_PRICE_MODE'] = pre_app_non['AMT_GOODS_PRICE'].fillna(pre_app_non['A
```

In [84]:
```
all_data = ['AMT_GOODS_PRICE','AMT_GOODS_PRICE_MEDAIN','AMT_GOODS_PRICE_MEAN','AMT_GOODS_P
```

In [85]:
```
plt.figure(figsize=(10,5))
for i, col in enumerate(all_data):
    plt.subplot(2,2,i+1)
    sns.kdeplot(data = pre_app_non,x=col)
    plt.subplots_adjust(wspace=0.5, hspace=0.5)
```




In [86]:
```
pre_app_non['AMT_GOODS_PRICE'] = pre_app_non['AMT_GOODS_PRICE'].fillna(pre_app_non['AMT_GO
```

In [87]:
```
pre_app_non['AMT_GOODS_PRICE'].isnull().sum()
```

Out[87]: 0

In [88]:
```
pre_app_non['AMT_ANNUITY'] = pre_app_non['AMT_ANNUITY'].fillna(pre_app_non['AMT_ANNUITY'],
```

In [89]:
```
pre_app_non['AMT_ANNUITY'].isnull().sum()
```

Out[89]: 0

In [90]:
```
pre_app_non['CNT_PAYMENT'] = pre_app_non['CNT_PAYMENT'].fillna(pre_app_non['CNT_PAYMENT'],
```

In [91]:
```
pre_app_non['CNT_PAYMENT'].isnull().sum()
```

Out[91]: 0

In [92]:
```
pre_app_non['CNT_PAYMENT'].agg(func=['mean','median','max'])
```

Out[92]:
```
mean      15.150574
median    12.000000
```

```
max       84.000000
Name: CNT_PAYMENT, dtype: float64
```

In [93]:
```
pre_app_non['AMT_ANNUITY'].agg(func=['mean','median','max'])
```

Out[93]:
```
mean       14906.506177
median     11250.000000
max       418058.145000
Name: AMT_ANNUITY, dtype: float64
```

In [94]:
```
pre_app_non['PRODUCT_COMBINATION'].head()
```

Out[94]:
```
0    POS mobile with interest
1            Cash X-Sell: low
2           Cash X-Sell: high
3         Cash X-Sell: middle
4           Cash Street: high
Name: PRODUCT_COMBINATION, dtype: object
```

In [95]:
```
pre_app_non['PRODUCT_COMBINATION'] = pre_app_non['PRODUCT_COMBINATION'].fillna(pre_app_nor
```

In [96]:
```
pre_app_non['PRODUCT_COMBINATION'].isnull().sum()
```

Out[96]:
```
0
```

In [97]:
```
pre_app_non['DAYS_FIRST_DRAWING'] = pre_app_non['DAYS_FIRST_DRAWING'].fillna(pre_app_non[
```

In [98]:
```
pre_app_non['DAYS_FIRST_DRAWING'].isnull().sum()
```

Out[98]:
```
0
```

In [99]:
```
pre_app_non['DAYS_FIRST_DUE'] = pre_app_non['DAYS_FIRST_DUE'].fillna(pre_app_non['DAYS_FIF
```

In [100…
```
pre_app_non['DAYS_FIRST_DUE'].isnull().sum()
```

Out[100…
```
0
```

In [101…
```
pre_app_non['DAYS_LAST_DUE'] = pre_app_non['DAYS_LAST_DUE'].fillna(pre_app_non['DAYS_LAST_
```

In [102…
```
pre_app_non['DAYS_LAST_DUE'].isnull().sum()
```

Out[102…
```
0
```

In [103…
```
pre_app_non['NFLAG_INSURED_ON_APPROVAL'] = pre_app_non['NFLAG_INSURED_ON_APPROVAL'].fillna
```

In [104…
```
pre_app_non['NFLAG_INSURED_ON_APPROVAL'].isnull().sum()
```

Out[104…
```
0
```

In [105…
```
pre_app_non['DAYS_TERMINATION'] = pre_app_non['DAYS_TERMINATION'].fillna(pre_app_non['DAYS
```

In [106…
```
pre_app_non['DAYS_TERMINATION'].isnull().sum()
```

Out[106… 0

In [107…
```
pre_app_non['NFLAG_INSURED_ON_APPROVAL'] = pre_app_non['NFLAG_INSURED_ON_APPROVAL'].fillna
```

In [108…
```
pre_app_non['NFLAG_INSURED_ON_APPROVAL'].isnull().sum()
```

Out[108… 0

In [109…
```
pre_app_non['DAYS_LAST_DUE_1ST_VERSION'] = pre_app_non['DAYS_LAST_DUE_1ST_VERSION'].fillna
```

In [110…
```
pre_app_non['DAYS_LAST_DUE_1ST_VERSION'].isnull().sum()
```

Out[110… 0

In [111…
```
pre_app_non['NAME_TYPE_SUITE'] = pre_app_non['NAME_TYPE_SUITE'].fillna(pre_app_non['NAME_T
```

In [112…
```
pre_app_non['NAME_TYPE_SUITE'].isnull().sum()
```

Out[112… 0

In [113…
```
pre_app_non.isnull().sum().sort_values(ascending=False)
```

Out[113…
```
AMT_CREDIT                   1
SK_ID_PREV                   0
CHANNEL_TYPE                 0
AMT_GOODS_PRICE_MEAN         0
AMT_GOODS_PRICE_MEDAIN       0
NFLAG_INSURED_ON_APPROVAL    0
DAYS_TERMINATION             0
DAYS_LAST_DUE                0
DAYS_LAST_DUE_1ST_VERSION    0
DAYS_FIRST_DUE               0
DAYS_FIRST_DRAWING           0
PRODUCT_COMBINATION          0
NAME_YIELD_GROUP             0
CNT_PAYMENT                  0
NAME_SELLER_INDUSTRY         0
SELLERPLACE_AREA             0
NAME_PRODUCT_TYPE            0
SK_ID_CURR                   0
NAME_PORTFOLIO               0
NAME_GOODS_CATEGORY          0
NAME_CLIENT_TYPE             0
NAME_TYPE_SUITE              0
CODE_REJECT_REASON           0
NAME_PAYMENT_TYPE            0
DAYS_DECISION                0
NAME_CONTRACT_STATUS         0
NAME_CASH_LOAN_PURPOSE       0
E                            0
```

```
AMT_APPLICATION              0
AMT_ANNUITY                  0
NAME_CONTRACT_TYPE           0
AMT_GOODS_PRICE_MODE         0
dtype: int64
```

In [114…

```
pre_app_non.isnull().sum().sort_values(ascending=False)/pre_app_non.shape[0]*100
```

Out[114…

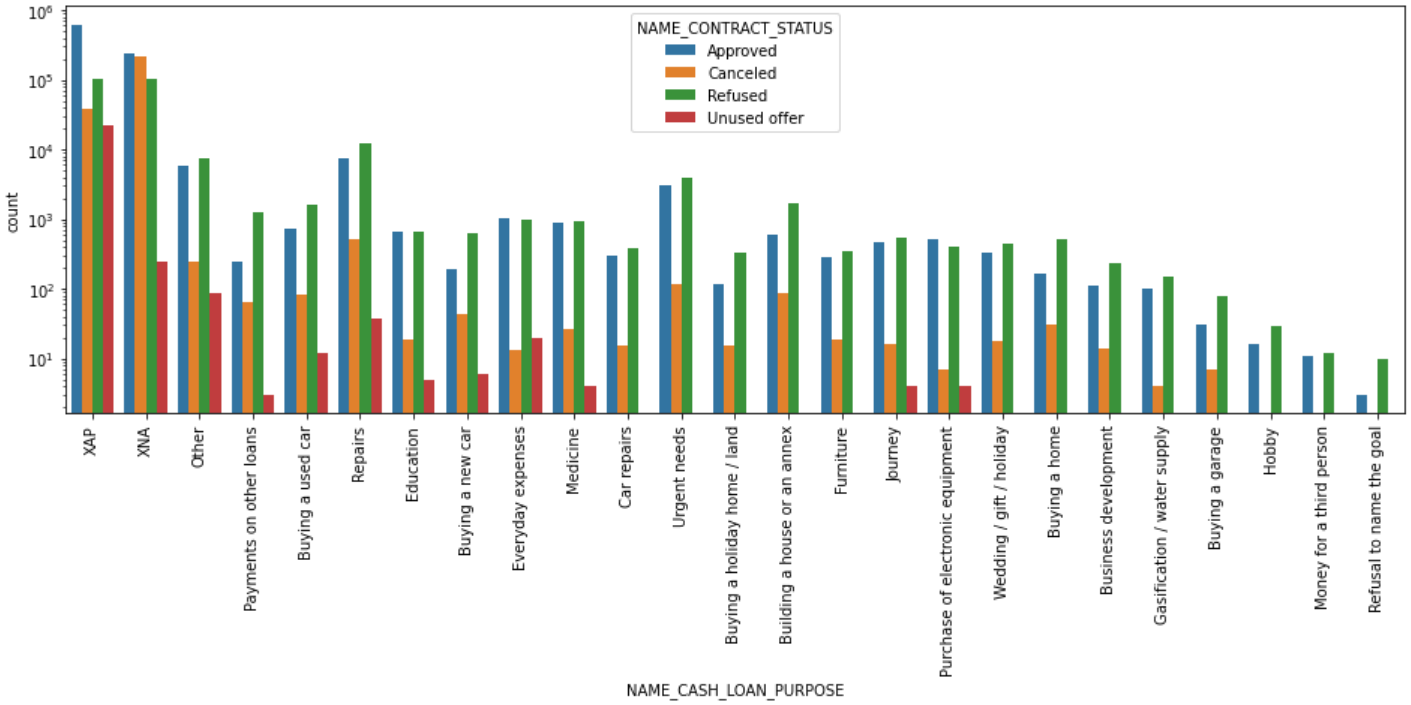
```
AMT_CREDIT                   0.00006
SK_ID_PREV                   0.00000
CHANNEL_TYPE                 0.00000
AMT_GOODS_PRICE_MEAN         0.00000
AMT_GOODS_PRICE_MEDAIN       0.00000
NFLAG_INSURED_ON_APPROVAL    0.00000
DAYS_TERMINATION             0.00000
DAYS_LAST_DUE                0.00000
DAYS_LAST_DUE_1ST_VERSION    0.00000
DAYS_FIRST_DUE               0.00000
DAYS_FIRST_DRAWING           0.00000
PRODUCT_COMBINATION          0.00000
NAME_YIELD_GROUP             0.00000
CNT_PAYMENT                  0.00000
NAME_SELLER_INDUSTRY         0.00000
SELLERPLACE_AREA             0.00000
NAME_PRODUCT_TYPE            0.00000
SK_ID_CURR                   0.00000
NAME_PORTFOLIO               0.00000
NAME_GOODS_CATEGORY          0.00000
NAME_CLIENT_TYPE             0.00000
NAME_TYPE_SUITE              0.00000
CODE_REJECT_REASON           0.00000
NAME_PAYMENT_TYPE            0.00000
DAYS_DECISION                0.00000
NAME_CONTRACT_STATUS         0.00000
NAME_CASH_LOAN_PURPOSE       0.00000
AMT_GOODS_PRICE              0.00000
AMT_APPLICATION              0.00000
AMT_ANNUITY                  0.00000
NAME_CONTRACT_TYPE           0.00000
AMT_GOODS_PRICE_MODE         0.00000
dtype: float64
```

In [ ]:

```
merger_data = pd.merge(apps,loan_defaulter,how='inner',on ='SK_ID_CURR')
merger_data
```

In [121…

```
plt.figure(figsize=(16,5))
sns.countplot(data =merger_data, x = 'NAME_CASH_LOAN_PURPOSE', hue ='NAME_CONTRACT_STATUS'
plt.xticks(rotation=90)
plt.yscale('log')
```

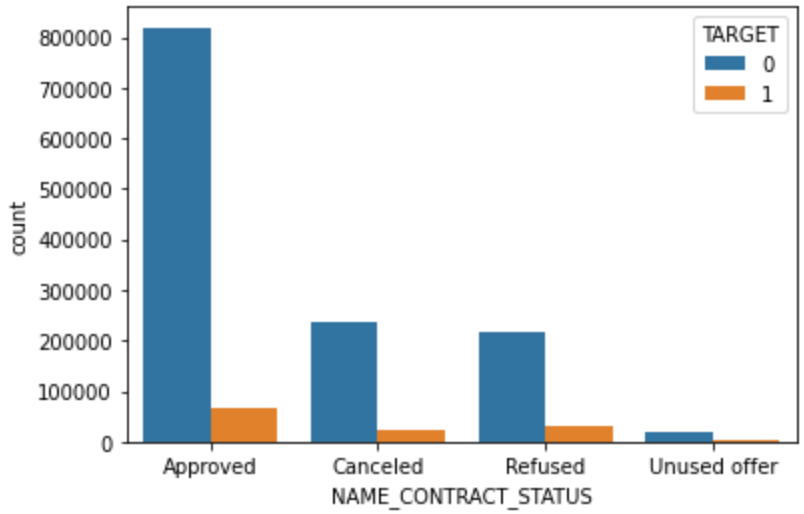



In [123…

```
sns.countplot(data = merger_data, x ='NAME_CONTRACT_STATUS', hue ='TARGET')
```

Out[123…

```
<AxesSubplot:xlabel='NAME_CONTRACT_STATUS', ylabel='count'>
```




In [129…

```
margar_data_combine = merger_data.groupby(['NAME_CONTRACT_STATUS','TARGET']).size().reset_
sum_df = margar_data_combine.groupby(['NAME_CONTRACT_STATUS'])['count'].sum().reset_index(
combine_data = pd.merge(margar_data_combine,sum_df, how ='left',on ='NAME_CONTRACT_STATUS'
combine_data['pct']=round(combine_data['count_x']/combine_data['count_y']*100,2)
combine_data
```

Out[129…

| | NAME_CONTRACT_STATUS | TARGET | count_x | count_y | pct |
|---|---|---|---|---|---|
| **0** | Approved | 0 | 818856 | 886099 | 92.41 |
| **1** | Approved | 1 | 67243 | 886099 | 7.59 |
| **2** | Canceled | 0 | 235641 | 259441 | 90.83 |
| **3** | Canceled | 1 | 23800 | 259441 | 9.17 |
| **4** | Refused | 0 | 215952 | 245390 | 88.00 |
| **5** | Refused | 1 | 29438 | 245390 | 12.00 |
| **6** | Unused offer | 0 | 20892 | 22771 | 91.75 |

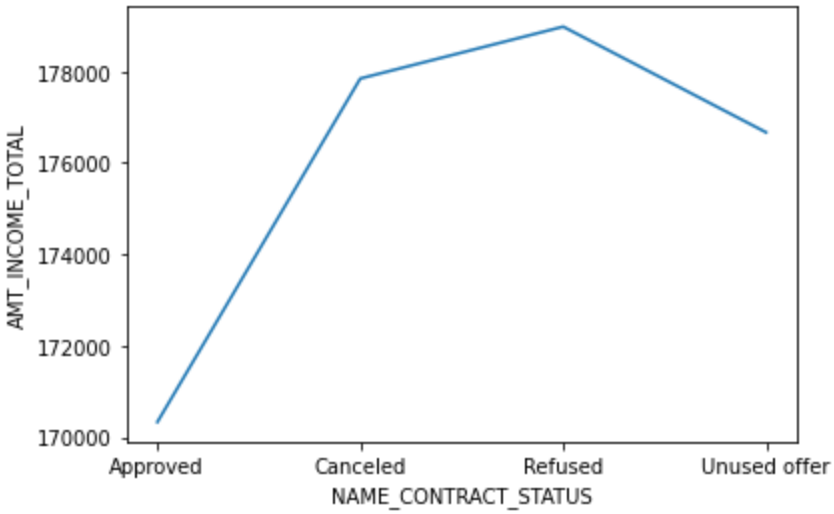
| | NAME_CONTRACT_STATUS | TARGET | count_x | count_y | pct |
|---|---|---|---|---|---|
| **7** | Unused offer | 1 | 1879 | 22771 | 8.25 |

In [130…

```
sns.lineplot(data=merger_data,x='NAME_CONTRACT_STATUS', y='AMT_INCOME_TOTAL',ci=None)
```

Out[130…

```
<AxesSubplot:xlabel='NAME_CONTRACT_STATUS', ylabel='AMT_INCOME_TOTAL'>
```




In [131…

```
loan_defaulter.head()
```

Out[131…

| | SK_ID_PREV | SK_ID_CURR | NAME_CONTRACT_TYPE | AMT_ANNUITY | AMT_APPLICATION | AMT_CREDIT | AMT_D |
|---|---|---|---|---|---|---|---|
| **0** | 2030495 | 271877 | Consumer loans | 1730.430 | 17145.0 | 17145.0 | |
| **1** | 2802425 | 108129 | Cash loans | 25188.615 | 607500.0 | 679671.0 | |
| **2** | 2523466 | 122040 | Cash loans | 15060.735 | 112500.0 | 136444.5 | |
| **3** | 2819243 | 176158 | Cash loans | 47041.335 | 450000.0 | 470790.0 | |
| **4** | 1784265 | 202054 | Cash loans | 31924.395 | 337500.0 | 404055.0 | |

In [ ]: