Data import and basic Exploration

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


pd.options.display.max_columns=None
pd.options.display.max_rows=None
```

In [2]:
```
app = pd.read_csv(r"E:\Data Science Projects\Project Files\Loan Defaulter  Segmentation\archi
prev_app = pd.read_csv(r"E:\Data Science Projects\Project Files\Loan Defaulter  Segmentation\
```

In [3]:
```
app.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]:
```
app.columns
```

Out[4]:
```
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 [5]:
```
app.shape
```

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

In [6]:
```
app.isnull().sum()
```

Out[6]:
```
SK_ID_CURR                      0
TARGET                          0
NAME_CONTRACT_TYPE              0
CODE_GENDER                     0
FLAG_OWN_CAR                    0
FLAG_OWN_REALTY                 0
CNT_CHILDREN                    0
AMT_INCOME_TOTAL                0
AMT_CREDIT                      0
AMT_ANNUITY                    12
AMT_GOODS_PRICE               278
NAME_TYPE_SUITE              1292
NAME_INCOME_TYPE                0
NAME_EDUCATION_TYPE             0
NAME_FAMILY_STATUS              0
NAME_HOUSING_TYPE               0
REGION_POPULATION_RELATIVE      0
DAYS_BIRTH                      0
DAYS_EMPLOYED                   0
```

In [7]:
```
app.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
```

In [8]:
```
msng_info = pd.DataFrame(app.isnull().sum().sort_values()).reset_index()
msng_info
```

Out[8]:

	index	0
0	SK_ID_CURR	0
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
5	REG_CITY_NOT_WORK_CITY	0
6	LIVE_CITY_NOT_WORK_CITY	0
7	ORGANIZATION_TYPE	0
8	FLAG_DOCUMENT_21	0
9	FLAG_DOCUMENT_20	0
10	FLAG_DOCUMENT_19	0

In [9]:
```
msng_info.rename(columns={'index':'col_name',0:'null_count'},inplace=True)
msng_info
```

Out[9]:

	col_name	null_count
0	SK_ID_CURR	0
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
5	REG_CITY_NOT_WORK_CITY	0
6	LIVE_CITY_NOT_WORK_CITY	0
7	ORGANIZATION_TYPE	0
8	FLAG_DOCUMENT_21	0
9	FLAG_DOCUMENT_20	0
10	FLAG_DOCUMENT_19	0

In [10]:
```
msng_info.head()
```

Out[10]:

	col_name	null_count
0	SK_ID_CURR	0
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 [11]:
```
msng_info['msng_pct']=msng_info['null_count']/app.shape[0]*100
msng_info
```

Out[11]:

	col_name	null_count	msng_pct
0	SK_ID_CURR	0	0.000000
1	HOUR_APPR_PROCESS_START	0	0.000000
2	REG_REGION_NOT_WORK_REGION	0	0.000000
3	LIVE_REGION_NOT_WORK_REGION	0	0.000000
4	REG_CITY_NOT_LIVE_CITY	0	0.000000
5	REG_CITY_NOT_WORK_CITY	0	0.000000
6	LIVE_CITY_NOT_WORK_CITY	0	0.000000
7	ORGANIZATION_TYPE	0	0.000000
8	FLAG_DOCUMENT_21	0	0.000000
9	FLAG_DOCUMENT_20	0	0.000000
10	FLAG_DOCUMENT_19	0	0.000000

In [12]:
```
msng_info.to_excel(r"E:\Data Science Projects\Project Files\Loan Defaulter  Segmentation\arch
msng_info.head()
```

Out[12]:

	col_name	null_count	msng_pct
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 [13]:
```
msng_col=msng_info[msng_info['msng_pct']>=40]['col_name'].to_list()
msng_col
```

Out[13]:
```
['EMERGENCYSTATE_MODE',
 'TOTALAREA_MODE',
 'YEARS_BEGINEXPLUATATION_MODE',
 'YEARS_BEGINEXPLUATATION_AVG',
 'YEARS_BEGINEXPLUATATION_MEDI',
 'FLOORSMAX_AVG',
 'FLOORSMAX_MEDI',
 'FLOORSMAX_MODE',
 'HOUSETYPE_MODE',
 'LIVINGAREA_AVG',
 'LIVINGAREA_MODE',
 'LIVINGAREA_MEDI',
 'ENTRANCES_AVG',
 'ENTRANCES_MODE',
 'ENTRANCES_MEDI',
 'APARTMENTS_MEDI',
 'APARTMENTS_AVG',
 'APARTMENTS_MODE',
 'WALLSMATERIAL_MODE',
 'ELEVATORS_MEDI',
 'ELEVATORS_AVG',
 'ELEVATORS_MODE',
 'NONLIVINGAREA_MODE',
 'NONLIVINGAREA_AVG',
 'NONLIVINGAREA_MEDI',
 'EXT_SOURCE_1',
 'BASEMENTAREA_MODE',
 'BASEMENTAREA_AVG',
 'BASEMENTAREA_MEDI',
 'LANDAREA_MEDI',
 'LANDAREA_AVG',
 'LANDAREA_MODE',
 'OWN_CAR_AGE',
 'YEARS_BUILD_MODE',
 'YEARS_BUILD_AVG',
 'YEARS_BUILD_MEDI',
 'FLOORSMIN_AVG',
 'FLOORSMIN_MODE',
 'FLOORSMIN_MEDI',
 'LIVINGAPARTMENTS_AVG',
 'LIVINGAPARTMENTS_MODE',
 'LIVINGAPARTMENTS_MEDI',
 'FONDKAPREMONT_MODE',
 'NONLIVINGAPARTMENTS_AVG',
 'NONLIVINGAPARTMENTS_MEDI',
 'NONLIVINGAPARTMENTS_MODE',
 'COMMONAREA_MODE',
 'COMMONAREA_AVG',
 'COMMONAREA_MEDI']
```

In [14]:
```
len(msng_col)
```

Out[14]:
```
49
```

In [15]:
```
app_msng_rmvd=app.drop(labels=msng_col,axis=1)
app_msng_rmvd.shape
```

Out[15]:
```
(307511, 73)
```

In [16]:
```
app_msng_rmvd.head()
```

Out[16]:

	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 [17]:
```
flag_col=[]

for col in app_msng_rmvd.columns:
    if col.startswith("FLAG_"):
        flag_col.append(col)

flag_col
```

Out[17]:
```
['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',
 'FLAG_DOCUMENT_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 [18]:
```
len(flag_col)
```

Out[18]: 28

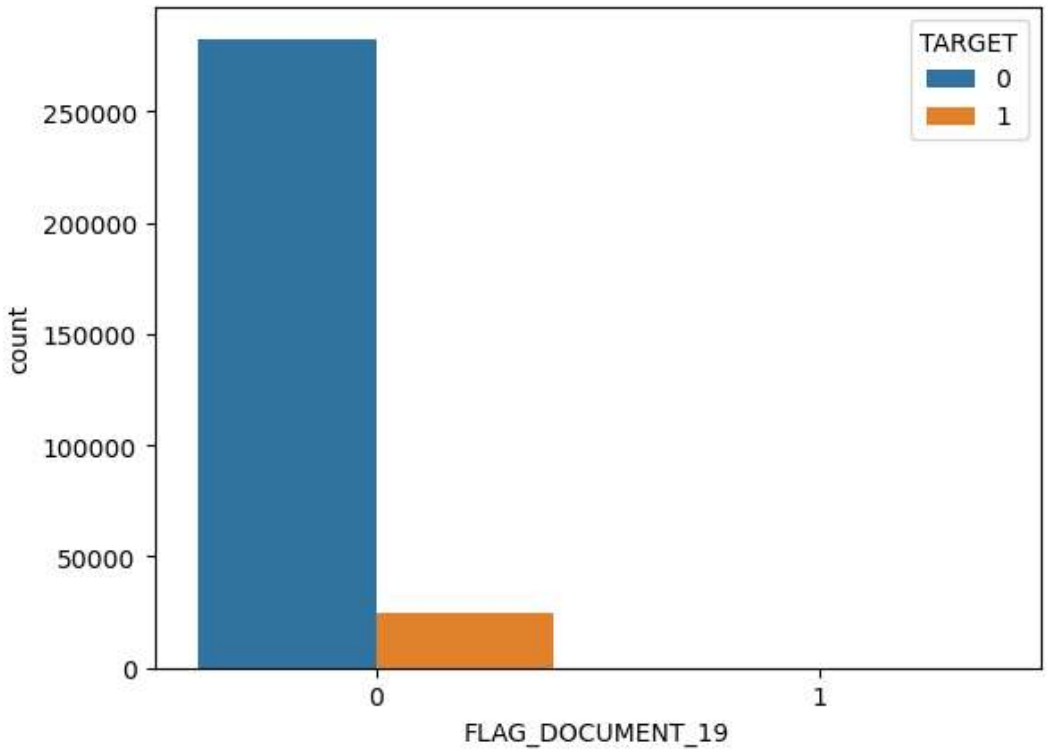
In [19]:
```
flag_tgt_col=app_msng_rmvd[flag_col+['TARGET']]
flag_tgt_col.head()
```

Out[19]:

	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 [20]:
```
sns.countplot(x='FLAG_DOCUMENT_19', data=flag_tgt_col,hue='TARGET') # target 0 means non defa
```

Out[20]:
```
<Axes: xlabel='FLAG_DOCUMENT_19', ylabel='count'>
```



In [21]:
```
plt.figure(figsize=(25,25))

for i, col in enumerate(flag_col):#enumerate print index number with column name
    plt.subplot(7,4,i+1)
    sns.countplot(x=col,hue='TARGET',data=flag_tgt_col)
```

In [22]:
```
flag_col
```

Out[22]:
```
['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',
 'FLAG_DOCUMENT_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 [23]:
```
flag_corr=['FLAG_OWN_CAR','FLAG_OWN_REALTY','FLAG_MOBIL','FLAG_EMP_PHONE','FLAG_WORK_PHONE',
           'FLAG_EMAIL','TARGET']
flag_corr
```

Out[23]:
```
['FLAG_OWN_CAR',
 'FLAG_OWN_REALTY',
 'FLAG_MOBIL',
 'FLAG_EMP_PHONE',
 'FLAG_WORK_PHONE',
 'FLAG_CONT_MOBILE',
 'FLAG_PHONE',
 'FLAG_EMAIL',
 'TARGET']
```

In [24]:
```
flag_corr_df = app_msng_rmvd[flag_corr]
flag_corr_df.head()
```

Out[24]:

	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 [25]:
```
flag_corr_df.groupby(['FLAG_OWN_CAR']).size()
```

Out[25]:
```
FLAG_OWN_CAR
N    202924
Y    104587
dtype: int64
```

In [26]:
```
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})

flag_corr_df.groupby(['FLAG_OWN_CAR']).size()
```

```
C:\Users\mshiv\AppData\Local\Temp\ipykernel_9548\165340972.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  flag_corr_df['FLAG_OWN_CAR']=flag_corr_df['FLAG_OWN_CAR'].replace({'N':0,'Y':1})
C:\Users\mshiv\AppData\Local\Temp\ipykernel_9548\165340972.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)
  flag_corr_df['FLAG_OWN_REALTY']=flag_corr_df['FLAG_OWN_REALTY'].replace({'N':0,'Y':1})
```

Out[26]:
```
FLAG_OWN_CAR
0    202924
1    104587
dtype: int64
```

In [27]:
```
flag_corr_df.corr()
```

Out[27]:

	FLAG_OWN_CAR	FLAG_OWN_REALTY	FLAG_MOBIL	FLAG_EMP_PHONE	FLAG_WORK_PH
FLAG_OWN_CAR	1.000000	-0.002817	-0.002512	0.154659	0.01
FLAG_OWN_REALTY	-0.002817	1.000000	-0.001198	-0.070021	-0.11
FLAG_MOBIL	-0.002512	-0.001198	1.000000	-0.000845	0.00
FLAG_EMP_PHONE	0.154659	-0.070021	-0.000845	1.000000	0.23
FLAG_WORK_PHONE	0.011471	-0.114043	0.000900	0.233801	1.00
FLAG_CONT_MOBILE	-0.006644	0.008526	-0.000078	-0.012819	0.02
FLAG_PHONE	-0.007588	-0.041507	0.001128	-0.016131	0.29
FLAG_EMAIL	0.032105	0.029247	0.000442	0.062542	-0.01
TARGET	-0.021851	-0.006148	0.000534	0.045982	0.02

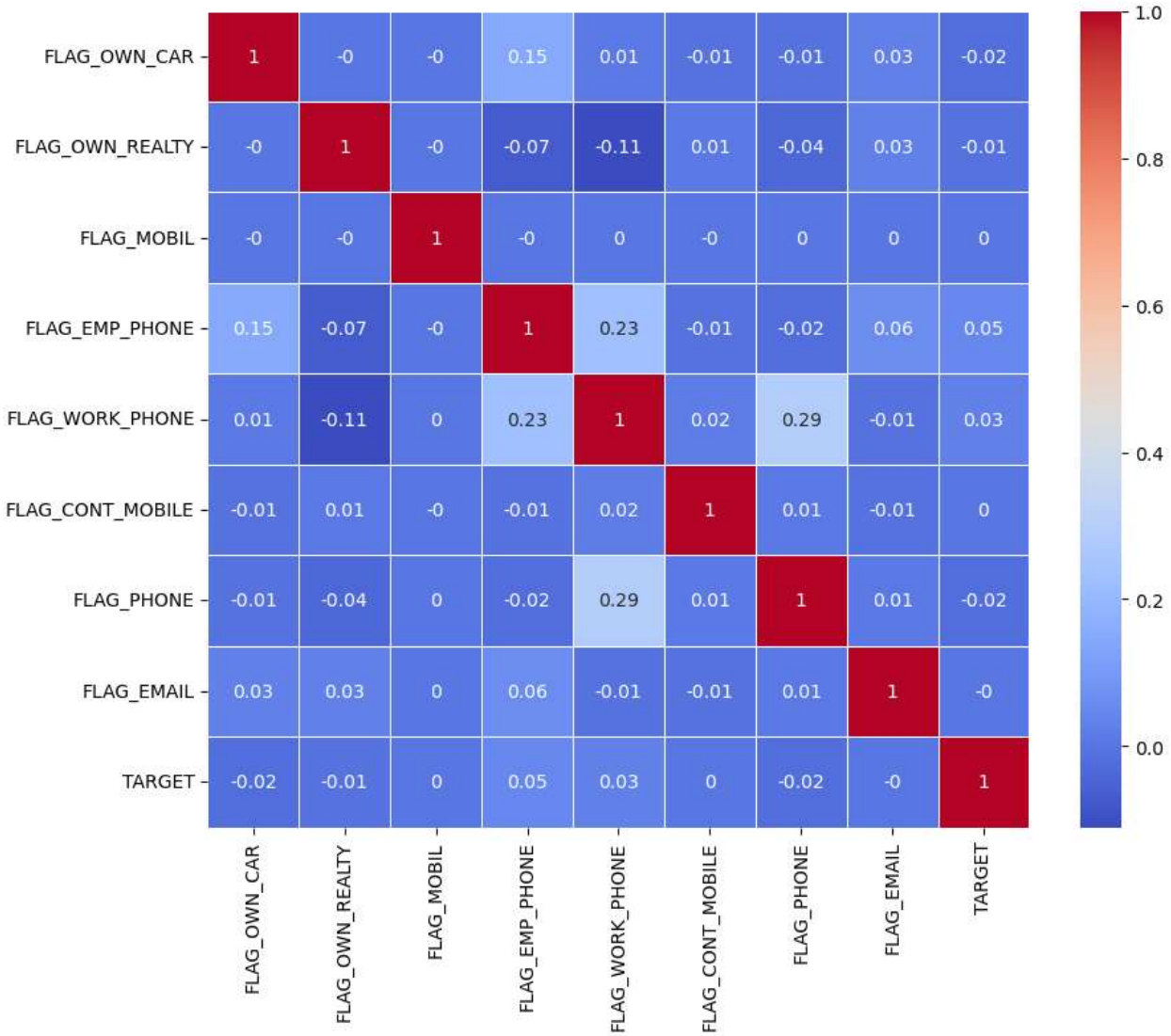
In [28]:
```
corr_df=round(flag_corr_df.corr(),2)
corr_df
```

Out[28]:

	FLAG_OWN_CAR	FLAG_OWN_REALTY	FLAG_MOBIL	FLAG_EMP_PHONE	FLAG_WORK_PH
FLAG_OWN_CAR	1.00	-0.00	-0.0	0.15	
FLAG_OWN_REALTY	-0.00	1.00	-0.0	-0.07	
FLAG_MOBIL	-0.00	-0.00	1.0	-0.00	
FLAG_EMP_PHONE	0.15	-0.07	-0.0	1.00	
FLAG_WORK_PHONE	0.01	-0.11	0.0	0.23	
FLAG_CONT_MOBILE	-0.01	0.01	-0.0	-0.01	
FLAG_PHONE	-0.01	-0.04	0.0	-0.02	
FLAG_EMAIL	0.03	0.03	0.0	0.06	
TARGET	-0.02	-0.01	0.0	0.05	

In [29]:
```
plt.figure(figsize=(10,8))
sns.heatmap(corr_df, linewidths=0.5,annot=True,cmap='coolwarm')
```

Out[29]: <Axes: >



In [30]:
```
app_flag_rmvd = app_msng_rmvd.drop(labels=flag_col,axis=1)
app_flag_rmvd.shape
```

Out[30]: (307511, 45)

In [31]:
```
app_flag_rmvd.head()
```

Out[31]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_
0	100002	1	Cash loans	M	0	202500.0	
1	100003	0	Cash loans	F	0	270000.0	1
2	100004	0	Revolving loans	M	0	67500.0	
3	100006	0	Cash loans	F	0	135000.0	
4	100007	0	Cash loans	M	0	121500.0	

In [32]:
```
app_flag_rmvd[['EXT_SOURCE_2','EXT_SOURCE_3','TARGET']].corr()
```

Out[32]:

	EXT_SOURCE_2	EXT_SOURCE_3	TARGET
EXT_SOURCE_2	1.000000	0.109167	-0.160472
EXT_SOURCE_3	0.109167	1.000000	-0.178919
TARGET	-0.160472	-0.178919	1.000000

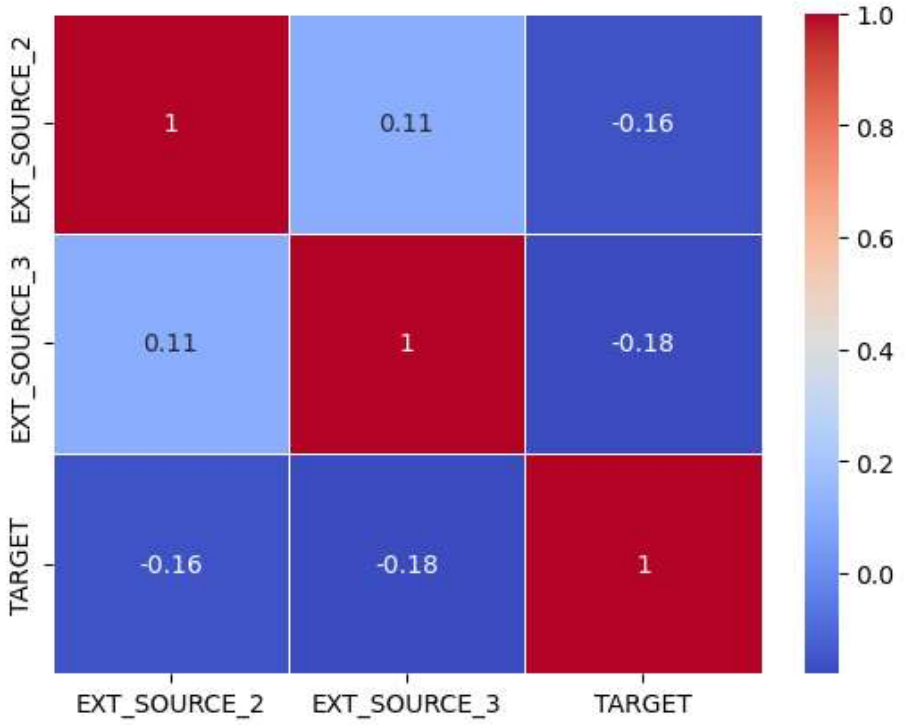
In [33]:
```
round(app_flag_rmvd[['EXT_SOURCE_2','EXT_SOURCE_3','TARGET']].corr(),2)
```

Out[33]:

	EXT_SOURCE_2	EXT_SOURCE_3	TARGET
EXT_SOURCE_2	1.00	0.11	-0.16
EXT_SOURCE_3	0.11	1.00	-0.18
TARGET	-0.16	-0.18	1.00

In [34]:
```
sns.heatmap(round(app_flag_rmvd[['EXT_SOURCE_2','EXT_SOURCE_3','TARGET']].corr(),2),cmap='co
```

Out[34]: <Axes: >



In [35]:
```
app_score_col_rmvd = app_flag_rmvd.drop(['EXT_SOURCE_2','EXT_SOURCE_3'] , axis=1)
```

In [36]:
```
app_score_col_rmvd.shape
```

Out[36]: (307511, 43)

Feature Engineering

In [37]:
```
app_score_col_rmvd.isnull().sum().sort_values()
```

Out[37]:
```
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
AMT_REQ_CREDIT_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 [38]:
```
(app_score_col_rmvd.isnull().sum().sort_values()/app_score_col_rmvd.shape[0])*100 # percenta
```

Out[38]:
```
SK_ID_CURR                    0.000000
ORGANIZATION_TYPE             0.000000
LIVE_CITY_NOT_WORK_CITY       0.000000
REG_CITY_NOT_WORK_CITY        0.000000
REG_CITY_NOT_LIVE_CITY        0.000000
LIVE_REGION_NOT_WORK_REGION   0.000000
REG_REGION_NOT_WORK_REGION    0.000000
REG_REGION_NOT_LIVE_REGION    0.000000
HOUR_APPR_PROCESS_START       0.000000
WEEKDAY_APPR_PROCESS_START    0.000000
REGION_RATING_CLIENT_W_CITY   0.000000
DAYS_ID_PUBLISH               0.000000
DAYS_REGISTRATION             0.000000
DAYS_EMPLOYED                 0.000000
DAYS_BIRTH                    0.000000
REGION_RATING_CLIENT          0.000000
NAME_HOUSING_TYPE             0.000000
TARGET                        0.000000
NAME_CONTRACT_TYPE            0.000000
REGION_POPULATION_RELATIVE    0.000000
CNT_CHILDREN                  0.000000
AMT_INCOME_TOTAL              0.000000
AMT_CREDIT                    0.000000
CODE_GENDER                   0.000000
NAME_INCOME_TYPE              0.000000
NAME_EDUCATION_TYPE           0.000000
NAME_FAMILY_STATUS            0.000000
DAYS_LAST_PHONE_CHANGE        0.000325
CNT_FAM_MEMBERS               0.000650
AMT_ANNUITY                   0.003902
AMT_GOODS_PRICE               0.090403
DEF_60_CNT_SOCIAL_CIRCLE      0.332021
OBS_60_CNT_SOCIAL_CIRCLE      0.332021
DEF_30_CNT_SOCIAL_CIRCLE      0.332021
OBS_30_CNT_SOCIAL_CIRCLE      0.332021
NAME_TYPE_SUITE               0.420148
AMT_REQ_CREDIT_BUREAU_QRT    13.501631
AMT_REQ_CREDIT_BUREAU_HOUR   13.501631
AMT_REQ_CREDIT_BUREAU_DAY    13.501631
AMT_REQ_CREDIT_BUREAU_WEEK   13.501631
AMT_REQ_CREDIT_BUREAU_MON    13.501631
AMT_REQ_CREDIT_BUREAU_YEAR   13.501631
OCCUPATION_TYPE              31.345545
dtype: float64
```

Missing Imputation

In [39]:
```
app_score_col_rmvd['CNT_FAM_MEMBERS'] = app_score_col_rmvd['CNT_FAM_MEMBERS'].fillna(app_sco
```

In [40]:
```
app_score_col_rmvd['CNT_FAM_MEMBERS'].isnull().sum()
```

Out[40]:
```
0
```

In [41]:
```
app_score_col_rmvd.groupby(['OCCUPATION_TYPE']).size().sort_values()
```

Out[41]:
```
OCCUPATION_TYPE
IT staff                  526
HR staff                  563
Realty agents             751
Secretaries              1305
Waiters/barmen staff     1348
Low-skill Laborers       2093
Private service staff    2652
Cleaning staff           4653
Cooking staff            5946
Security staff           6721
Medicine staff           8537
Accountants              9813
High skill tech staff   11380
Drivers                 18603
Managers                21371
Core staff              27570
Sales staff             32102
Laborers                55186
dtype: int64
```

In [42]:
```
app_score_col_rmvd['OCCUPATION_TYPE'].mode()[0]
```

Out[42]:
```
'Laborers'
```

In [43]:
```
app_score_col_rmvd['OCCUPATION_TYPE'] = app_score_col_rmvd['OCCUPATION_TYPE'].fillna(app_sco
```

In [44]:
```
app_score_col_rmvd['OCCUPATION_TYPE'].isnull().sum()
```

Out[44]:
```
0
```

In [45]:
```
app_score_col_rmvd.groupby(['NAME_TYPE_SUITE']).size().sort_values()
```

Out[45]:
```
NAME_TYPE_SUITE
Group of people       271
Other_A               866
Other_B              1770
Children             3267
Spouse, partner     11370
Family              40149
Unaccompanied      248526
dtype: int64
```

In [46]:
```
app_score_col_rmvd['NAME_TYPE_SUITE'].isnull().sum()
```

Out[46]:
```
1292
```

In [47]:
```
app_score_col_rmvd['NAME_TYPE_SUITE'].mode()[0]
```

Out[47]:
```
'Unaccompanied'
```

In [48]:
```
app_score_col_rmvd['NAME_TYPE_SUITE'] = app_score_col_rmvd['NAME_TYPE_SUITE'].fillna(app_sco
```

In [49]:
```
app_score_col_rmvd['NAME_TYPE_SUITE'].isnull().sum()
```

Out[49]:
```
0
```

In [50]:
```
app_score_col_rmvd['AMT_ANNUITY'].describe()
```

Out[50]:
```
count    307499.000000
mean      27108.573909
std       14493.737315
min        1615.500000
25%       16524.000000
50%       24903.000000
75%       34596.000000
max      258025.500000
Name: AMT_ANNUITY, dtype: float64
```

In [51]:
```
app_score_col_rmvd['AMT_ANNUITY'] = app_score_col_rmvd['AMT_ANNUITY'].fillna(app_score_col_r
```

In [52]:
```
app_score_col_rmvd['AMT_ANNUITY'].isnull().sum()
```

Out[52]:
```
0
```

In [53]:
```
app_score_col_rmvd['AMT_REQ_CREDIT_BUREAU_HOUR'].describe()
```

Out[53]:
```
count    265992.000000
mean          0.006402
std           0.083849
min           0.000000
25%           0.000000
50%           0.000000
75%           0.000000
max           4.000000
Name: AMT_REQ_CREDIT_BUREAU_HOUR, dtype: float64
```

In [54]:
```
app_score_col_rmvd['AMT_REQ_CREDIT_BUREAU_HOUR'].values
```

Out[54]:
```
array([0., 0., 0., ..., 1., 0., 0.])
```

In [55]:
```
amt_req_col=[]

for col in app_score_col_rmvd.columns:
    if col.startswith('AMT_REQ_CREDIT_BUREAU'):
        amt_req_col.append(col)
```

In [56]:
```
amt_req_col
```

Out[56]:
```
['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']
```

In [57]:
```
for col in amt_req_col:
    app_score_col_rmvd[col]=app_score_col_rmvd[col].fillna(app_score_col_rmvd[col].median())
```

In [58]:
```
app_score_col_rmvd.isnull().sum().sort_values()
```

Out[58]:
```
SK_ID_CURR                        0
AMT_REQ_CREDIT_BUREAU_QRT         0
REGION_RATING_CLIENT_W_CITY       0
WEEKDAY_APPR_PROCESS_START        0
HOUR_APPR_PROCESS_START           0
REG_REGION_NOT_LIVE_REGION        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
AMT_REQ_CREDIT_BUREAU_HOUR        0
AMT_REQ_CREDIT_BUREAU_DAY         0
AMT_REQ_CREDIT_BUREAU_WEEK        0
AMT_REQ_CREDIT_BUREAU_MON         0
CNT_FAM_MEMBERS                   0
OCCUPATION_TYPE                   0
REGION_RATING_CLIENT              0
DAYS_REGISTRATION                 0
TARGET                            0
NAME_CONTRACT_TYPE                0
CODE_GENDER                       0
CNT_CHILDREN                      0
AMT_INCOME_TOTAL                  0
DAYS_ID_PUBLISH                   0
AMT_ANNUITY                       0
AMT_CREDIT                        0
NAME_INCOME_TYPE                  0
NAME_EDUCATION_TYPE               0
NAME_FAMILY_STATUS                0
NAME_HOUSING_TYPE                 0
REGION_POPULATION_RELATIVE        0
DAYS_BIRTH                        0
DAYS_EMPLOYED                     0
NAME_TYPE_SUITE                   0
AMT_REQ_CREDIT_BUREAU_YEAR        0
DAYS_LAST_PHONE_CHANGE            1
AMT_GOODS_PRICE                 278
OBS_30_CNT_SOCIAL_CIRCLE       1021
DEF_30_CNT_SOCIAL_CIRCLE       1021
OBS_60_CNT_SOCIAL_CIRCLE       1021
DEF_60_CNT_SOCIAL_CIRCLE       1021
dtype: int64
```

In [59]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].agg(['min','max','median'])
#we go with meian because it will give us a exact value and if we take the mean (bigger than
```

Out[59]:
```
min         40500.0
max       4050000.0
median     450000.0
Name: AMT_GOODS_PRICE, dtype: float64
```

In [60]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].mean()
```

Out[60]:
```
538396.2074288895
```

In [61]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'] = app_score_col_rmvd['AMT_GOODS_PRICE'].fillna(app_sco
```

In [62]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].isnull().sum()
```

Out[62]:
```
0
```

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

Out[63]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_
0	100002	1	Cash loans	M	0	202500.0	
1	100003	0	Cash loans	F	0	270000.0	1
2	100004	0	Revolving loans	M	0	67500.0	
3	100006	0	Cash loans	F	0	135000.0	
4	100007	0	Cash loans	M	0	121500.0	

Value Modification

In [64]:
```
days_col=[]

for col in app_score_col_rmvd.columns:
    if col.startswith('DAYS'):
        days_col.append(col)


days_col
```

Out[64]:
```
['DAYS_BIRTH',
 'DAYS_EMPLOYED',
 'DAYS_REGISTRATION',
 'DAYS_ID_PUBLISH',
 'DAYS_LAST_PHONE_CHANGE']
```

In [65]:
```
for col in days_col:
    app_score_col_rmvd[col] = abs(app_score_col_rmvd[col])
```

In [66]:
```
app_score_col_rmvd.head()
```

Out[66]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_
0	100002	1	Cash loans	M	0	202500.0	
1	100003	0	Cash loans	F	0	270000.0	1
2	100004	0	Revolving loans	M	0	67500.0	
3	100006	0	Cash loans	F	0	135000.0	
4	100007	0	Cash loans	M	0	121500.0	

In [67]:
```
len(app_score_col_rmvd.columns)
```

Out[67]: 43

In [68]:
```
app_score_col_rmvd.nunique().sort_values()
```

Out[68]:
```
LIVE_REGION_NOT_WORK_REGION          2
TARGET                               2
NAME_CONTRACT_TYPE                   2
REG_REGION_NOT_LIVE_REGION           2
REG_CITY_NOT_LIVE_CITY               2
REG_CITY_NOT_WORK_CITY               2
LIVE_CITY_NOT_WORK_CITY              2
REG_REGION_NOT_WORK_REGION           2
REGION_RATING_CLIENT_W_CITY          3
REGION_RATING_CLIENT                 3
CODE_GENDER                          3
NAME_EDUCATION_TYPE                  5
AMT_REQ_CREDIT_BUREAU_HOUR           5
NAME_HOUSING_TYPE                    6
NAME_FAMILY_STATUS                   6
WEEKDAY_APPR_PROCESS_START           7
NAME_TYPE_SUITE                      7
NAME_INCOME_TYPE                     8
AMT_REQ_CREDIT_BUREAU_DAY            9
```

In [69]:
```
app_score_col_rmvd.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              307511 non-null  float64
 9   NAME_TYPE_SUITE              307511 non-null  object 
 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              307511 non-null  object 
 20  CNT_FAM_MEMBERS              307511 non-null  float64
 21  REGION_RATING_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       307510 non-null  float64
 37  AMT_REQ_CREDIT_BUREAU_HOUR   307511 non-null  float64
 38  AMT_REQ_CREDIT_BUREAU_DAY    307511 non-null  float64
 39  AMT_REQ_CREDIT_BUREAU_WEEK   307511 non-null  float64
 40  AMT_REQ_CREDIT_BUREAU_MON    307511 non-null  float64
 41  AMT_REQ_CREDIT_BUREAU_QRT    307511 non-null  float64
 42  AMT_REQ_CREDIT_BUREAU_YEAR   307511 non-null  float64
dtypes: float64(18), int64(15), object(10)
memory usage: 100.9+ MB
```

In [70]:
```
app_score_col_rmvd['SK_ID_CURR'].unique()
```

Out[70]:
```
array([100002, 100003, 100004, ..., 456253, 456254, 456255], dtype=int64)
```

In [71]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].unique()
```

Out[71]:
```
array([ 351000. , 1129500. ,  135000. , ...,  453465. ,  143977.5,
        743863.5])
```

Outlier Detection and Treatment

In [72]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].describe()
```

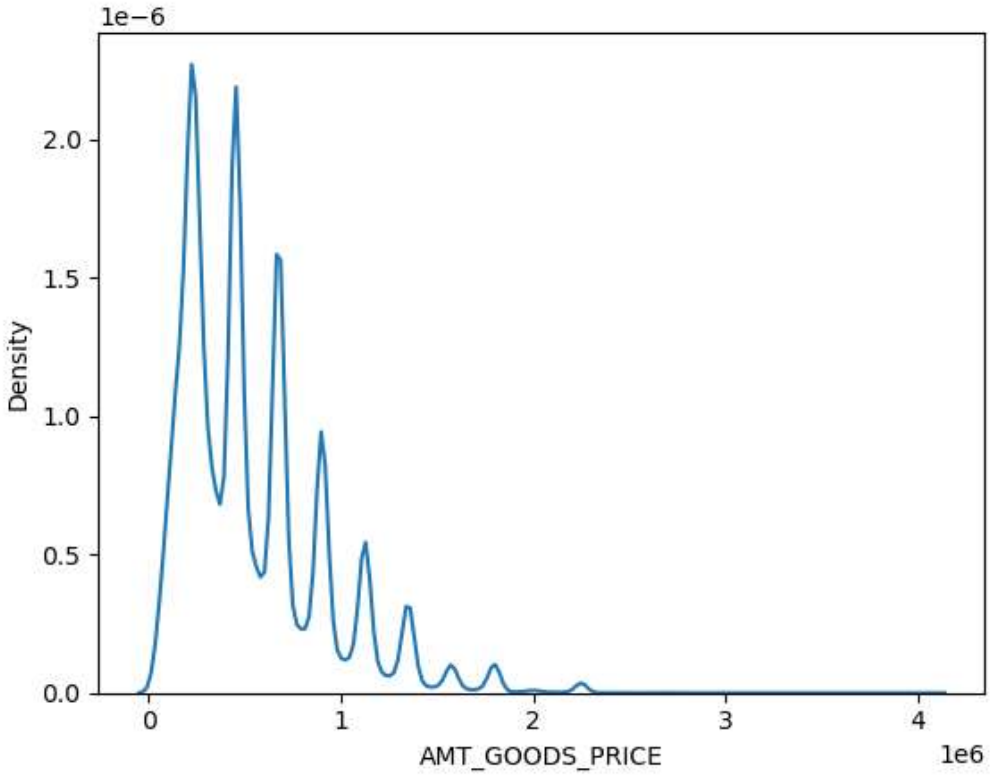
Out[72]:
```
count    3.075110e+05
mean     5.383163e+05
std      3.692890e+05
min      4.050000e+04
25%      2.385000e+05
50%      4.500000e+05
75%      6.795000e+05
max      4.050000e+06
Name: AMT_GOODS_PRICE, dtype: float64
```

In [73]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].agg(['min','max','median'])
```

Out[73]:
```
min         40500.0
max       4050000.0
median     450000.0
Name: AMT_GOODS_PRICE, dtype: float64
```

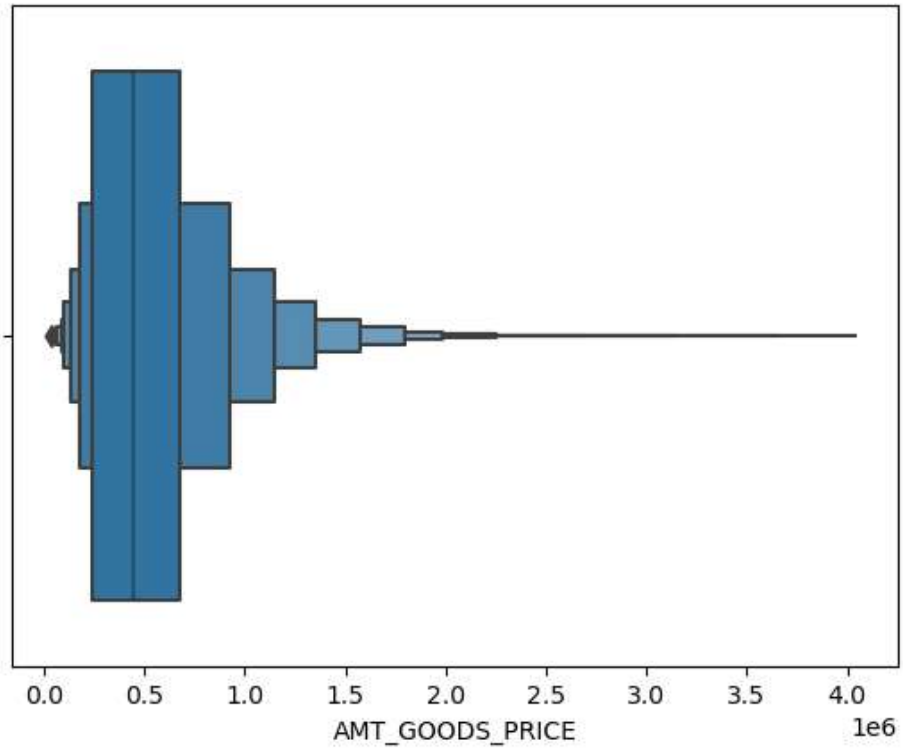
In [74]:
```
sns.kdeplot(data=app_score_col_rmvd, x='AMT_GOODS_PRICE') # Right skewed
```

Out[74]:
```
<Axes: xlabel='AMT_GOODS_PRICE', ylabel='Density'>
```



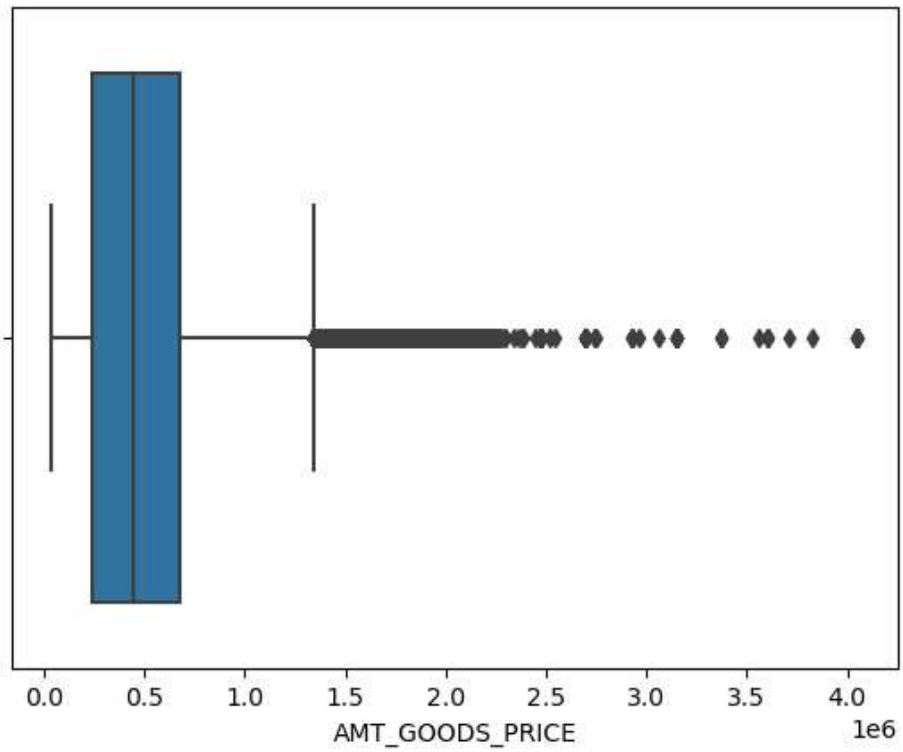
In [75]:
```
sns.boxenplot(data= app_score_col_rmvd, x='AMT_GOODS_PRICE')
```

Out[75]: <Axes: xlabel='AMT_GOODS_PRICE'>



In [76]:
```
sns.boxplot(data=app_score_col_rmvd, x='AMT_GOODS_PRICE')
```

Out[76]: <Axes: xlabel='AMT_GOODS_PRICE'>



In [77]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.
# the 99% of the data is less than the maximumm value
```

Out[77]:
```
0.001      45000.0
0.100     180000.0
0.200     225000.0
0.300     270000.0
0.400     378000.0
0.500     450000.0
0.600     522000.0
0.700     675000.0
0.800     814500.0
0.900    1093500.0
0.990    1800000.0
Name: AMT_GOODS_PRICE, dtype: float64
```

In [78]:
```
app_score_col_rmvd['AMT_GOODS_PRICE'].max()
```

Out[78]:
```
4050000.0
```

In [79]:
```
bins=[0,100000,200000,300000,400000,500000,600000,700000,800000,900000,4050000]
ranges=['0-100k','100k-200k','200k-300k','300k-400k','400k-500k','500k-600k','600k-700k','70

app_score_col_rmvd['AMT_GOODS_PRICE_RANGE']=pd.cut(app_score_col_rmvd['AMT_GOODS_PRICE'],bin
```

In [80]:
```
app_score_col_rmvd.groupby(['AMT_GOODS_PRICE_RANGE']).size()
```

Out[80]:
```
AMT_GOODS_PRICE_RANGE
0-100k         8709
100k-200k     32956
200k-300k     62761
300k-400k     21219
400k-500k     57251
500k-600k     13117
600k-700k     40024
700k-800k      8110
800k-900k     21484
Above 900k    41880
dtype: int64
```

In [81]:
```
app_score_col_rmvd['AMT_GOODS_PRICE_RANGE'].isnull().sum()
```

Out[81]:
```
0
```

In [82]:
```
app_score_col_rmvd['AMT_INCOME_TOTAL'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0
```

Out[82]:
```
0.001     31500.0
0.100     81000.0
0.200     99000.0
0.300    112500.0
0.400    135000.0
0.500    147150.0
0.600    162000.0
0.700    180000.0
0.800    225000.0
0.900    270000.0
0.990    472500.0
Name: AMT_INCOME_TOTAL, dtype: float64
```

In [83]:
```
app_score_col_rmvd['AMT_INCOME_TOTAL'].max()
```

Out[83]:
```
117000000.0
```

In [84]:
```
bins=[0,100000,150000,200000,250000,300000,350000,400000,117000000]
ranges=['0-100k','100k-150k','150k-200k','200k-250k','250k-300k','300k-350k','350k-400k','Ab

app_score_col_rmvd['AMT_INCOME_TOTAL_RANGE']=pd.cut(app_score_col_rmvd['AMT_INCOME_TOTAL'],b
```

In [85]:
```
app_score_col_rmvd.groupby(['AMT_INCOME_TOTAL_RANGE']).size()
```

Out[85]:
```
AMT_INCOME_TOTAL_RANGE
0-100k        63698
100k-150k     91591
150k-200k     64307
200k-250k     48137
250k-300k     17039
300k-350k      8874
350k-400k      5802
Above 400k     8063
dtype: int64
```

In [86]:
```
app_score_col_rmvd['AMT_INCOME_TOTAL_RANGE'].isnull().sum()
```

Out[86]:
```
0
```

In [87]:
```
app_score_col_rmvd['AMT_CREDIT'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.99])
```

Out[87]:
```
0.001      47970.0
0.100     180000.0
0.200     254700.0
0.300     306306.0
0.400     432000.0
0.500     513531.0
0.600     604152.0
0.700     755190.0
0.800     900000.0
0.900    1133748.0
0.990    1854000.0
Name: AMT_CREDIT, dtype: float64
```

In [88]:
```
app_score_col_rmvd['AMT_CREDIT'].max()
```

Out[88]:
```
4050000.0
```

In [89]:
```
bins=[0,200000,400000,600000,800000,900000,1000000,2000000,3000000,4050000]
ranges=['0-200k','200k-400k','400k-600k','600k-800k','800k-900k','900k-1M','1M-2M','2M-3M','/

app_score_col_rmvd['AMT_CREDIT_RANGE']=pd.cut(app_score_col_rmvd['AMT_CREDIT'],bins,labels=r
```

In [90]:
```
app_score_col_rmvd.groupby(['AMT_CREDIT_RANGE']).size()
```

Out[90]:
```
AMT_CREDIT_RANGE
0-200k       36144
200k-400k    81151
400k-600k    66270
600k-800k    43242
800k-900k    21792
900k-1M       8927
1M-2M        47956
2M-3M         1997
Above 3M        32
dtype: int64
```

In [91]:
```
app_score_col_rmvd['AMT_CREDIT_RANGE'].isnull().sum()
```

Out[91]:
```
0
```

In [92]:
```
app_score_col_rmvd['AMT_ANNUITY'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.99])
```

Out[92]:
```
0.001     3933.09
0.100    11074.50
0.200    14701.50
0.300    18189.00
0.400    21870.00
0.500    24903.00
0.600    28062.00
0.700    32004.00
0.800    37516.50
0.900    45954.00
0.990    70006.50
Name: AMT_ANNUITY, dtype: float64
```

In [93]:
```
app_score_col_rmvd['AMT_ANNUITY'].max()
```

Out[93]:
```
258025.5
```

In [94]:
```
bins=[0,25000,50000,100000,150000,200000,258026]
ranges=['0-25k','25k-50k','50k-100k','100k-150k','150k-200k','Above 200k']


app_score_col_rmvd['AMT_ANNUITY_RANGE']=pd.cut(app_score_col_rmvd['AMT_ANNUITY'],bins,labels
```

In [95]:
```
app_score_col_rmvd.groupby(['AMT_ANNUITY_RANGE']).size()
```

Out[95]:
```
AMT_ANNUITY_RANGE
0-25k         154867
25k-50k       131347
50k-100k       20792
100k-150k        437
150k-200k         32
Above 200k        36
dtype: int64
```

In [96]:
```
app_score_col_rmvd['AMT_ANNUITY'].isnull().sum()
```

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

In [97]:
```
app_score_col_rmvd['DAYS_EMPLOYED'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.99
```

Out[97]:
```
0.001        60.0
0.100       392.0
0.200       749.0
0.300      1132.0
0.400      1597.0
0.500      2219.0
0.600      3032.0
0.700      4435.0
0.800      9188.0
0.900    365243.0
0.990    365243.0
Name: DAYS_EMPLOYED, dtype: float64
```

In [98]:
```
app_score_col_rmvd['DAYS_EMPLOYED'].max()
```

Out[98]:
```
365243
```

In [99]:
```
app_score_col_rmvd['DAYS_EMPLOYED'].min()
```

Out[99]:
```
0
```

In [100]:
```
bins=[0,1825,3650,5475,7300,9125,10950,12775,14600,16425,18250,23691,365243]
ranges=['0-5Y','5Y-10Y','10Y-15Y','15Y-20Y','20Y-25Y','25Y-30Y','30Y-35Y','35Y-40Y','40Y-45Y


app_score_col_rmvd['DAYS_EMPLOYED_RANGE']=pd.cut(app_score_col_rmvd['DAYS_EMPLOYED'],bins,lal
```

In [101]:
```
app_score_col_rmvd.groupby(['DAYS_EMPLOYED_RANGE']).size()
```

Out[101]:
```
DAYS_EMPLOYED_RANGE
0-5Y         136309
5Y-10Y        64872
10Y-15Y       27549
15Y-20Y       10849
20Y-25Y        6243
25Y-30Y        3308
30Y-35Y        1939
35Y-40Y         832
40Y-45Y         210
45Y-50Y          24
50Y-65Y           0
Above 65Y     55374
dtype: int64
```

In [102]:
```
app_score_col_rmvd['DAYS_EMPLOYED_RANGE'].isnull().sum()
```

Out[102]:
```
2
```

In [103]:
```
app_score_col_rmvd['DAYS_BIRTH'].quantile([0.001,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,0.99])
```

Out[103]:
```
0.001     7759.0
0.100    10284.0
0.200    11694.0
0.300    13140.0
0.400    14416.0
0.500    15750.0
0.600    17220.0
0.700    18885.0
0.800    20474.0
0.900    22181.0
0.990    24419.0
Name: DAYS_BIRTH, dtype: float64
```

In [104]:
```
app_score_col_rmvd['DAYS_BIRTH'].max()
```

Out[104]:
```
25229
```

In [105]:
```
app_score_col_rmvd['DAYS_BIRTH'].min()
```

Out[105]:
```
7489
```

In [106]:
```
bins=[0,7300,10950,14600,18250,21900,25229]
ranges=['0-20Y','20Y-30Y','30Y-40Y','40Y-50Y','50Y-60Y','Above 60Y']


app_score_col_rmvd['DAYS_BIRTH_RANGE']=pd.cut(app_score_col_rmvd['DAYS_BIRTH'],bins,labels=r
```

In [107]:
```
app_score_col_rmvd.groupby(['DAYS_BIRTH_RANGE']).size()
```

Out[107]:
```
DAYS_BIRTH_RANGE
0-20Y            0
20Y-30Y      45021
30Y-40Y      82308
40Y-50Y      76541
50Y-60Y      68062
Above 60Y    35579
dtype: int64
```

In [108]:
```
app_score_col_rmvd['DAYS_BIRTH_RANGE'].isnull().sum()
```

Out[108]:
```
0
```

Data Analysis

In [109]:
```
app_score_col_rmvd.dtypes.value_counts()
```

Out[109]:
```
float64     18
int64       15
object      10
category     1
category     1
category     1
category     1
category     1
category     1
dtype: int64
```

In [110]:
```
app_score_col_rmvd.select_dtypes(include=object).head()
```

Out[110]:

	NAME_CONTRACT_TYPE	CODE_GENDER	NAME_TYPE_SUITE	NAME_INCOME_TYPE	NAME_EDUCATION_TYPE
0	Cash loans	M	Unaccompanied	Working	Secondary / secondary special
1	Cash loans	F	Family	State servant	Higher education
2	Revolving loans	M	Unaccompanied	Working	Secondary / secondary special
3	Cash loans	F	Unaccompanied	Working	Secondary / secondary special
4	Cash loans	M	Unaccompanied	Working	Secondary / secondary special

In [111]:
```
obj_var=app_score_col_rmvd.select_dtypes(include=object).columns
obj_var
```

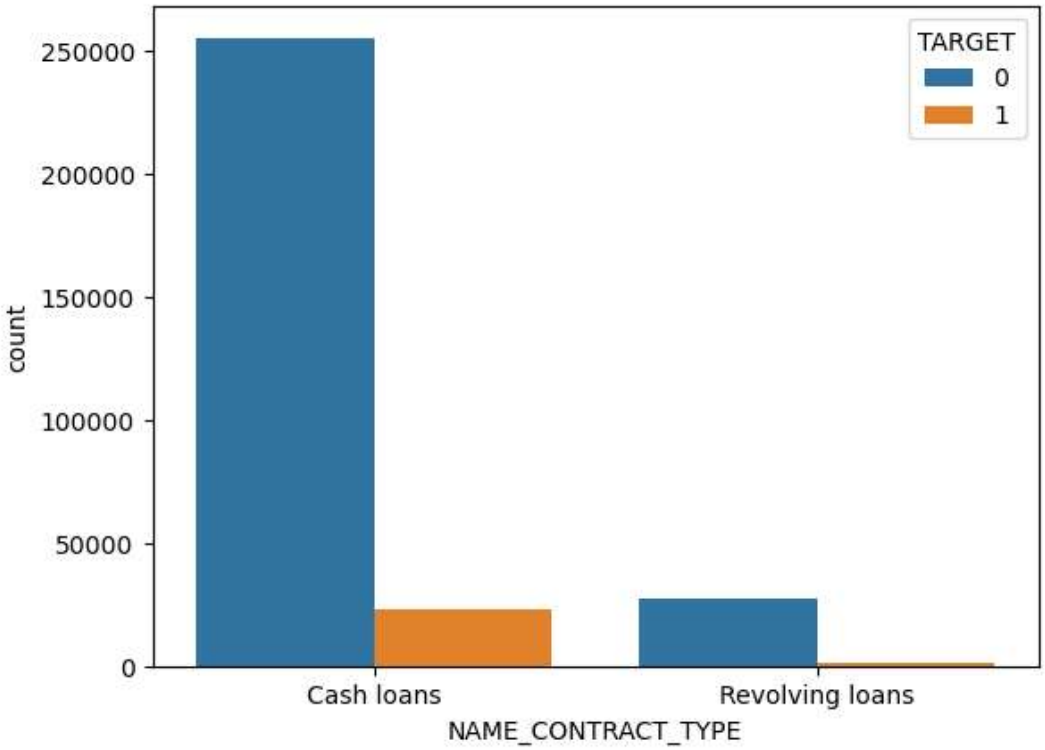
Out[111]:
```
Index(['NAME_CONTRACT_TYPE', 'CODE_GENDER', 'NAME_TYPE_SUITE',
       'NAME_INCOME_TYPE', 'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS',
       'NAME_HOUSING_TYPE', 'OCCUPATION_TYPE', 'WEEKDAY_APPR_PROCESS_START',
       'ORGANIZATION_TYPE'],
      dtype='object')
```

In [112]:
```
app_score_col_rmvd.groupby(['NAME_CONTRACT_TYPE']).size()
```

Out[112]:
```
NAME_CONTRACT_TYPE
Cash loans         278232
Revolving loans     29279
dtype: int64
```

In [113]:
```
#checking proportion of cash loan and revolving loans
sns.countplot(data=app_score_col_rmvd,x='NAME_CONTRACT_TYPE',hue='TARGET') # 0 means no defa
```

Out[113]: <Axes: xlabel='NAME_CONTRACT_TYPE', ylabel='count'>



In [114]:
```
data_pct = app_score_col_rmvd[['NAME_CONTRACT_TYPE','TARGET']].groupby(['NAME_CONTRACT_TYPE'
```

In [115]:
```
data_pct
```

Out[115]:

	NAME_CONTRACT_TYPE	TARGET
0	Cash loans	0.083459
1	Revolving loans	0.054783

In [116]:
```
data_pct['PCT']= data_pct['TARGET']*100
```

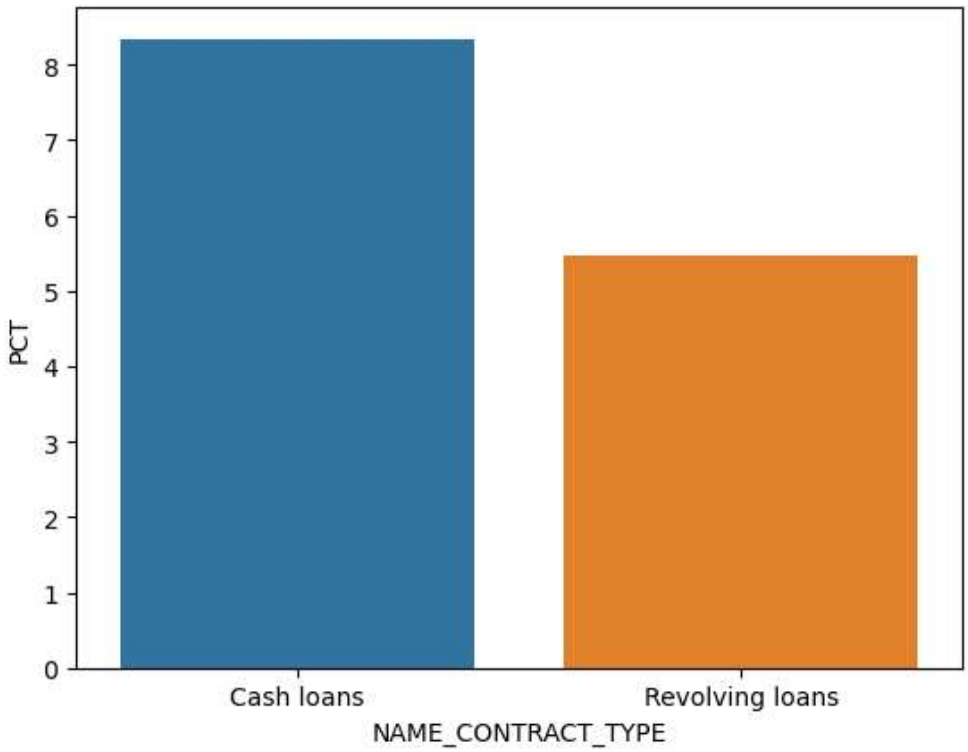
In [117]:
```
data_pct
```

Out[117]:

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

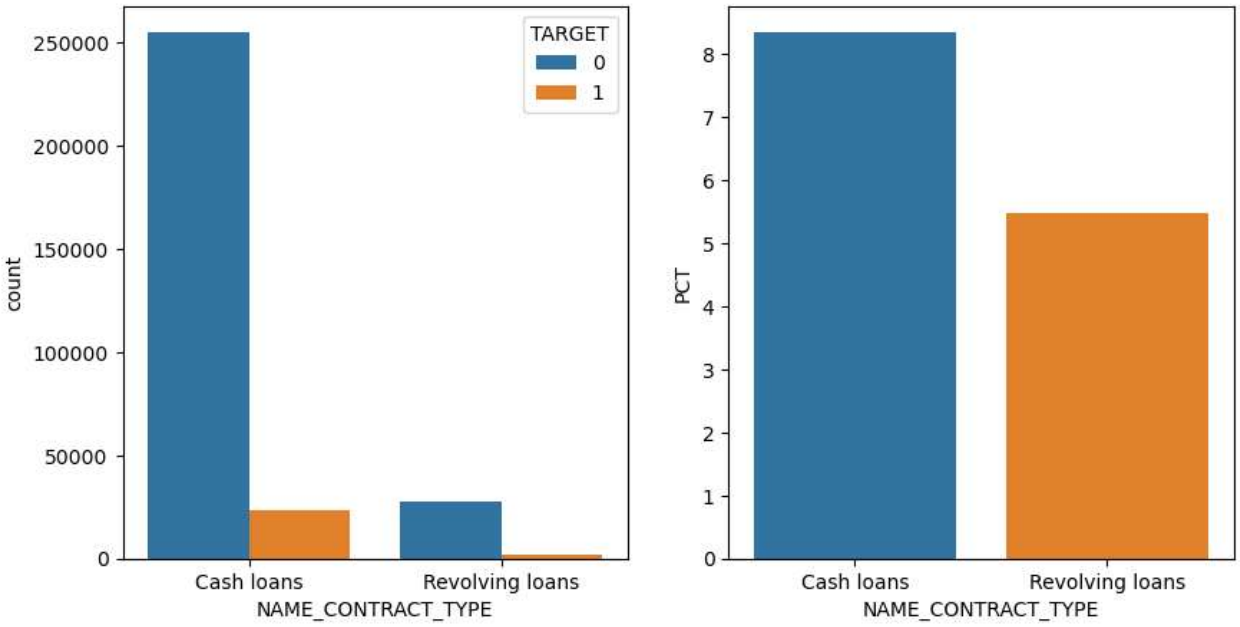
In [118]:
```
sns.barplot(data=data_pct,x='NAME_CONTRACT_TYPE',y='PCT')
```

Out[118]: <Axes: xlabel='NAME_CONTRACT_TYPE', ylabel='PCT'>



In [119]:
```
plt.figure(figsize=(10,5))

plt.subplot(1,2,1)
sns.countplot(data=app_score_col_rmvd,x='NAME_CONTRACT_TYPE',hue='TARGET')


plt.subplot(1,2,2)
sns.barplot(data=data_pct,x='NAME_CONTRACT_TYPE',y='PCT')
```

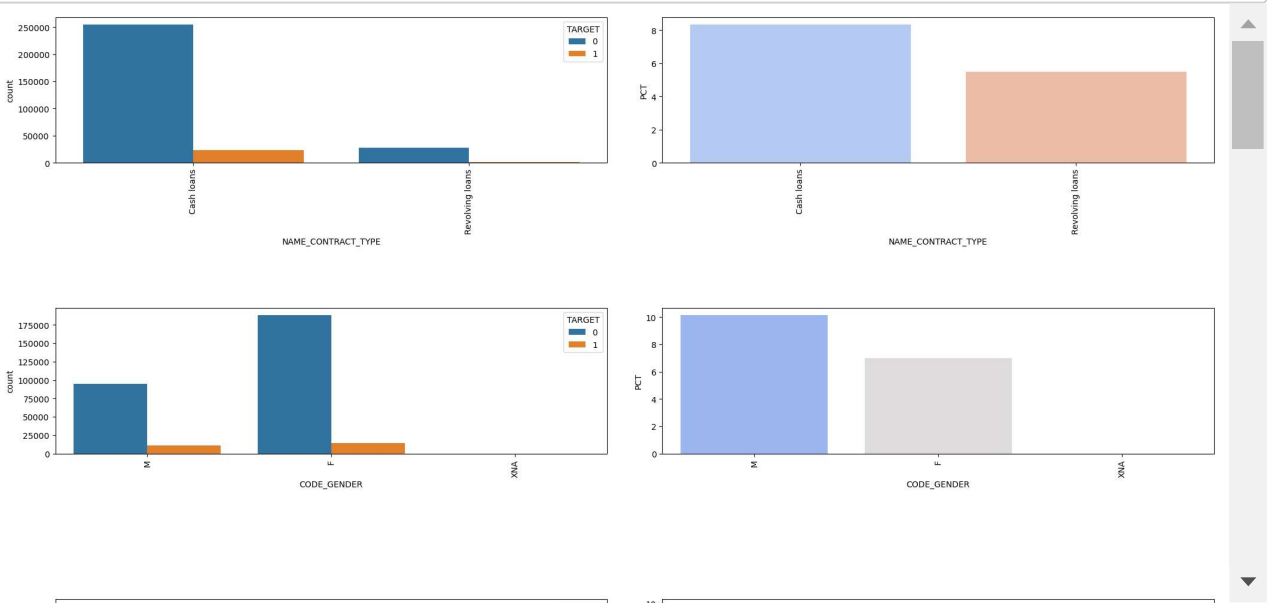
Out[119]: <Axes: xlabel='NAME_CONTRACT_TYPE', ylabel='PCT'>



In [120]:
```
len(obj_var)
```

Out[120]: 10

In [121]:
```
plt.figure(figsize=(25,60))

for i, var in enumerate(obj_var):


    data_pct = app_score_col_rmvd[[var,'TARGET']].groupby([var],as_index=False).mean().sort_
    data_pct['PCT']= data_pct['TARGET']*100


    plt.subplot(10,2,i+i+1)
    plt.subplots_adjust(wspace=0.1,hspace=1)
    sns.countplot(data=app_score_col_rmvd,x=var,hue='TARGET')
    plt.xticks(rotation=90)


    plt.subplot(10,2,i+i+2)
    sns.barplot(data=data_pct,x=var,y='PCT', palette='coolwarm')
    plt.xticks(rotation=90)
```



In [122]:
```
app_score_col_rmvd.dtypes.value_counts()
```

Out[122]:
```
float64     18
int64       15
object      10
category     1
category     1
category     1
category     1
category     1
category     1
dtype: int64
```

In [123]:
```
num_var= app_score_col_rmvd.select_dtypes(include=['float64','int64']).columns
num_cat_var= app_score_col_rmvd.select_dtypes(include=['float64','int64','category']).columns
num_var
```

Out[123]:
```
Index(['SK_ID_CURR', 'TARGET', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY', 'AMT_GOODS_PRICE',
       'REGION_POPULATION_RELATIVE', 'DAYS_BIRTH', 'DAYS_EMPLOYED',
       'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH', '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 [124]:
```
len(num_var)
```

Out[124]:
```
33
```

In [125]:
```
num_data=app_score_col_rmvd[num_var]
num_data.head()
```

Out[125]:

	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 [126]:
```
num_data.groupby(['TARGET']).size()/num_data.shape[0] * 100
```

Out[126]:
```
TARGET
0    91.927118
1     8.072882
dtype: float64
```

In [127]:
```
defaulters= num_data[num_data['TARGET']==1].drop(['TARGET'],axis=1)
defaulters.head()
```

Out[127]:

	SK_ID_CURR	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	REG
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 [128]:
```
repayers= num_data[num_data['TARGET']==0].drop(['TARGET'],axis=1)
repayers.head()
```

Out[128]:

	SK_ID_CURR	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRICE	REGIC
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 [129]:
```
defaulters.corr()
```

Out[129]:

	SK_ID_CURR	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANN
SK_ID_CURR	1.000000	-0.005144	-0.010165	-0.001290	-0.00
CNT_CHILDREN	-0.005144	1.000000	0.004796	-0.001675	0.03
AMT_INCOME_TOTAL	-0.010165	0.004796	1.000000	0.038131	0.04
AMT_CREDIT	-0.001290	-0.001675	0.038131	1.000000	0.75
AMT_ANNUITY	-0.007578	0.031257	0.046421	0.752195	1.00
AMT_GOODS_PRICE	-0.001814	-0.008111	0.037591	0.982783	0.75
REGION_POPULATION_RELATIVE	0.006301	-0.031975	0.009135	0.069161	0.07
DAYS_BIRTH	0.001254	-0.259109	-0.003096	0.135316	0.01
DAYS_EMPLOYED	-0.005161	-0.192864	-0.014977	0.001930	-0.08
DAYS_REGISTRATION	-0.006342	-0.149154	-0.000158	0.025854	-0.03
DAYS_ID_PUBLISH	0.002539	0.032299	0.004215	0.052329	0.01
CNT_FAM_MEMBERS	-0.003816	0.885484	0.006654	0.051224	0.07
REGION_RATING_CLIENT	-0.005936	0.040680	-0.021486	-0.059193	-0.07
REGION_RATING_CLIENT_W_CITY	-0.004135	0.043185	-0.022808	-0.071377	-0.08
HOUR_APPR_PROCESS_START	0.005004	-0.023899	0.013775	0.031782	0.03
REG_REGION_NOT_LIVE_REGION	-0.004249	-0.024322	0.007577	0.019540	0.03
REG_REGION_NOT_WORK_REGION	0.004120	-0.020793	0.014531	0.033260	0.06
LIVE_REGION_NOT_WORK_REGION	0.004303	-0.012073	0.013409	0.033554	0.06
REG_CITY_NOT_LIVE_CITY	0.008328	-0.001174	-0.002223	-0.033034	-0.00
REG_CITY_NOT_WORK_CITY	0.000787	0.046115	-0.003019	-0.037720	0.00
LIVE_CITY_NOT_WORK_CITY	-0.002929	0.053515	-0.001353	-0.016509	0.00
OBS_30_CNT_SOCIAL_CIRCLE	-0.009395	0.025804	-0.004709	0.019098	0.00
DEF_30_CNT_SOCIAL_CIRCLE	-0.005549	0.001448	-0.005186	-0.025979	-0.02
OBS_60_CNT_SOCIAL_CIRCLE	-0.009058	0.025180	-0.004616	0.019487	0.00
DEF_60_CNT_SOCIAL_CIRCLE	-0.009428	-0.005106	-0.004866	-0.030880	-0.02
DAYS_LAST_PHONE_CHANGE	-0.002455	-0.011547	0.002429	0.110851	0.07
AMT_REQ_CREDIT_BUREAU_HOUR	-0.011106	0.000316	0.001079	-0.003771	0.01
AMT_REQ_CREDIT_BUREAU_DAY	-0.007388	-0.011255	0.000135	0.004346	0.00
AMT_REQ_CREDIT_BUREAU_WEEK	-0.003075	-0.009316	0.000941	0.010598	0.02
AMT_REQ_CREDIT_BUREAU_MON	0.005180	-0.008852	0.005718	0.056227	0.04
AMT_REQ_CREDIT_BUREAU_QRT	-0.001614	-0.013029	0.001037	-0.007201	-0.00
AMT_REQ_CREDIT_BUREAU_YEAR	0.006843	-0.027253	0.004516	-0.020698	-0.00

In [130]:
```
defaulters[['SK_ID_CURR','CNT_CHILDREN','AMT_INCOME_TOTAL']].corr()
```

Out[130]:

	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 [131]:
```
defaulters_corr = defaulters.corr()
defaulters_corr.where(np.triu(np.ones(defaulters_corr.shape),k=1).astype(bool))
```

Out[131]:

	SK_ID_CURR	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANN
SK_ID_CURR	NaN	-0.005144	-0.010165	-0.001290	-0.00
CNT_CHILDREN	NaN	NaN	0.004796	-0.001675	0.03
AMT_INCOME_TOTAL	NaN	NaN	NaN	0.038131	0.04
AMT_CREDIT	NaN	NaN	NaN	NaN	0.75
AMT_ANNUITY	NaN	NaN	NaN	NaN	
AMT_GOODS_PRICE	NaN	NaN	NaN	NaN	
REGION_POPULATION_RELATIVE	NaN	NaN	NaN	NaN	
DAYS_BIRTH	NaN	NaN	NaN	NaN	
DAYS_EMPLOYED	NaN	NaN	NaN	NaN	
DAYS_REGISTRATION	NaN	NaN	NaN	NaN	
DAYS_ID_PUBLISH	NaN	NaN	NaN	NaN	
CNT_FAM_MEMBERS	NaN	NaN	NaN	NaN	
REGION_RATING_CLIENT	NaN	NaN	NaN	NaN	
REGION_RATING_CLIENT_W_CITY	NaN	NaN	NaN	NaN	
HOUR_APPR_PROCESS_START	NaN	NaN	NaN	NaN	
REG_REGION_NOT_LIVE_REGION	NaN	NaN	NaN	NaN	
REG_REGION_NOT_WORK_REGION	NaN	NaN	NaN	NaN	
LIVE_REGION_NOT_WORK_REGION	NaN	NaN	NaN	NaN	
REG_CITY_NOT_LIVE_CITY	NaN	NaN	NaN	NaN	
REG_CITY_NOT_WORK_CITY	NaN	NaN	NaN	NaN	
LIVE_CITY_NOT_WORK_CITY	NaN	NaN	NaN	NaN	
OBS_30_CNT_SOCIAL_CIRCLE	NaN	NaN	NaN	NaN	
DEF_30_CNT_SOCIAL_CIRCLE	NaN	NaN	NaN	NaN	
OBS_60_CNT_SOCIAL_CIRCLE	NaN	NaN	NaN	NaN	
DEF_60_CNT_SOCIAL_CIRCLE	NaN	NaN	NaN	NaN	
DAYS_LAST_PHONE_CHANGE	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_HOUR	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_DAY	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_WEEK	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_MON	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_QRT	NaN	NaN	NaN	NaN	
AMT_REQ_CREDIT_BUREAU_YEAR	NaN	NaN	NaN	NaN	

In [132]:
```
defaulters_corr_unstack=defaulters_corr.where(np.triu(np.ones(defaulters_corr.shape),k=1).as

defaulters_corr_unstack.head()
```

Out[132]:

	var1	var2	corr
0	SK_ID_CURR	SK_ID_CURR	NaN
1	SK_ID_CURR	CNT_CHILDREN	NaN
2	SK_ID_CURR	AMT_INCOME_TOTAL	NaN
3	SK_ID_CURR	AMT_CREDIT	NaN
4	SK_ID_CURR	AMT_ANNUITY	NaN

In [133]:
```
defaulters_corr_unstack['corr']=abs(defaulters_corr_unstack['corr'])
defaulters_corr_unstack.sort_values(by=['corr'],ascending=False)
```

Out[133]:

	var1	var2	corr
757	OBS_60_CNT_SOCIAL_CIRCLE	OBS_30_CNT_SOCIAL_CIRCLE	0.998269
163	AMT_GOODS_PRICE	AMT_CREDIT	0.982783
428	REGION_RATING_CLIENT_W_CITY	REGION_RATING_CLIENT	0.956637
353	CNT_FAM_MEMBERS	CNT_CHILDREN	0.885484
790	DEF_60_CNT_SOCIAL_CIRCLE	DEF_30_CNT_SOCIAL_CIRCLE	0.868994
560	LIVE_REGION_NOT_WORK_REGION	REG_REGION_NOT_WORK_REGION	0.847885
659	LIVE_CITY_NOT_WORK_CITY	REG_CITY_NOT_WORK_CITY	0.778540
164	AMT_GOODS_PRICE	AMT_ANNUITY	0.752295
131	AMT_ANNUITY	AMT_CREDIT	0.752195
263	DAYS_EMPLOYED	DAYS_BIRTH	0.582185
527	REG_REGION_NOT_WORK_REGION	REG_REGION_NOT_LIVE_REGION	0.497937

In [134]:
```
defaulters_corr_unstack.dropna(subset=['corr']).sort_values(by=['corr'],ascending=False).hea
```

Out[134]:

	var1	var2	corr
757	OBS_60_CNT_SOCIAL_CIRCLE	OBS_30_CNT_SOCIAL_CIRCLE	0.998269
163	AMT_GOODS_PRICE	AMT_CREDIT	0.982783
428	REGION_RATING_CLIENT_W_CITY	REGION_RATING_CLIENT	0.956637
353	CNT_FAM_MEMBERS	CNT_CHILDREN	0.885484
790	DEF_60_CNT_SOCIAL_CIRCLE	DEF_30_CNT_SOCIAL_CIRCLE	0.868994
560	LIVE_REGION_NOT_WORK_REGION	REG_REGION_NOT_WORK_REGION	0.847885
659	LIVE_CITY_NOT_WORK_CITY	REG_CITY_NOT_WORK_CITY	0.778540
164	AMT_GOODS_PRICE	AMT_ANNUITY	0.752295
131	AMT_ANNUITY	AMT_CREDIT	0.752195
263	DAYS_EMPLOYED	DAYS_BIRTH	0.582185

In [135]:
```
repayers_corr=repayers.corr()

repayers_corr_unstack=repayers_corr.where(np.triu(np.ones(defaulters_corr.shape),k=1).astype

repayers_corr_unstack.head()

repayers_corr_unstack['corr']=abs(repayers_corr_unstack['corr'])
repayers_corr_unstack.dropna(subset=['corr']).sort_values(by=['corr'],ascending=False).head(
```

Out[135]:

	var1	var2	corr
757	OBS_60_CNT_SOCIAL_CIRCLE	OBS_30_CNT_SOCIAL_CIRCLE	0.998508
163	AMT_GOODS_PRICE	AMT_CREDIT	0.987022
428	REGION_RATING_CLIENT_W_CITY	REGION_RATING_CLIENT	0.950149
353	CNT_FAM_MEMBERS	CNT_CHILDREN	0.878571
560	LIVE_REGION_NOT_WORK_REGION	REG_REGION_NOT_WORK_REGION	0.861861
790	DEF_60_CNT_SOCIAL_CIRCLE	DEF_30_CNT_SOCIAL_CIRCLE	0.859332
659	LIVE_CITY_NOT_WORK_CITY	REG_CITY_NOT_WORK_CITY	0.830381
164	AMT_GOODS_PRICE	AMT_ANNUITY	0.776421
131	AMT_ANNUITY	AMT_CREDIT	0.771297
263	DAYS_EMPLOYED	DAYS_BIRTH	0.626114

In [136]:
```
num_data.head()
```

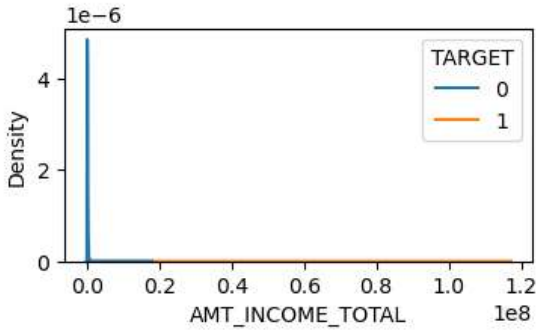
Out[136]:

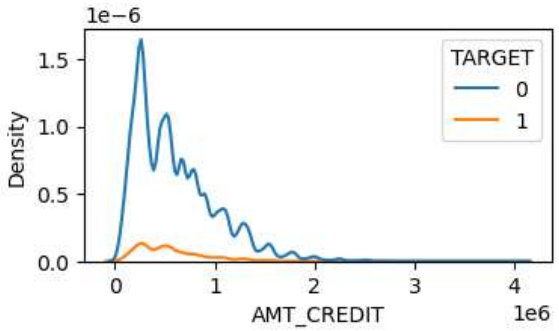
	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNUITY	AMT_GOODS_PRI
0	100002	1	0	202500.0	406597.5	24700.5	351000
1	100003	0	0	270000.0	1293502.5	35698.5	1129500
2	100004	0	0	67500.0	135000.0	6750.0	135000
3	100006	0	0	135000.0	312682.5	29686.5	297000
4	100007	0	0	121500.0	513000.0	21865.5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

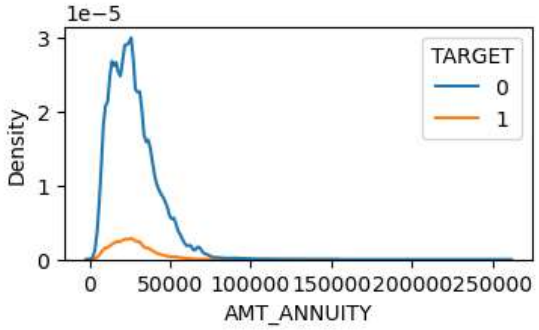
In [137]:
```
amt_var=['AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY','AMT_GOODS_PRICE']
```

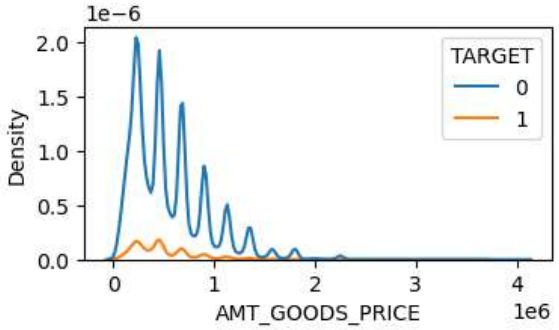
In [138]:
```
#sns.kdeplot(data=num_data, x='AMT_CREDIT',hue='TARGET')
```

In [139]:
```
#Univariate

plt.figure(figsize=(10,5))


for i, col in enumerate(amt_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)
```



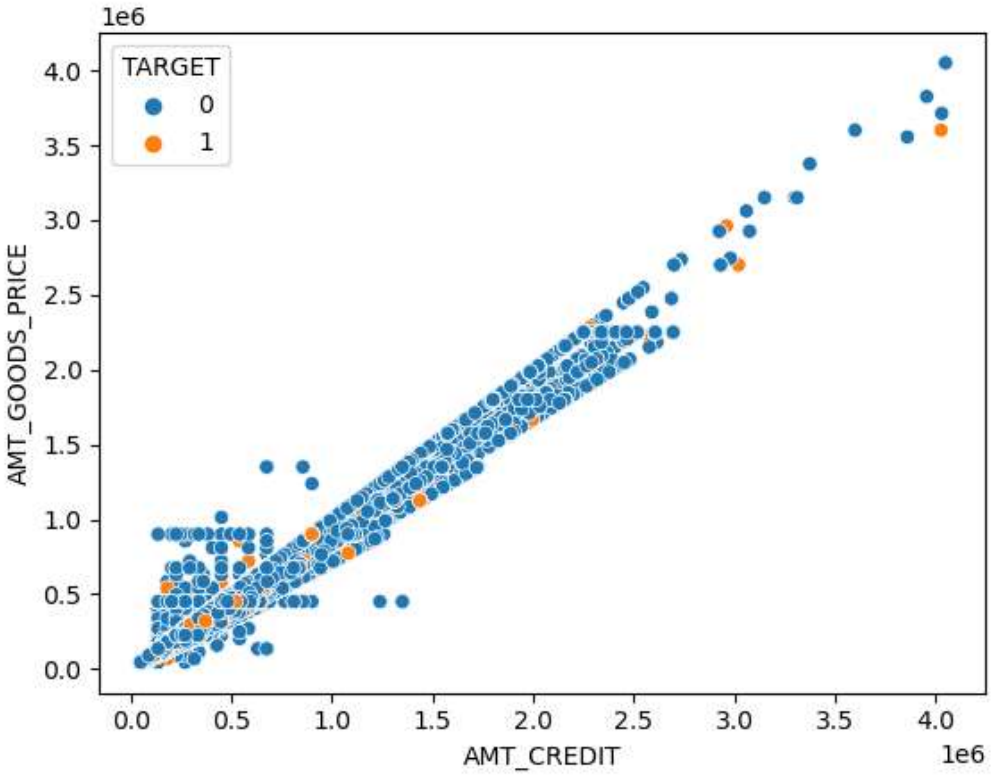






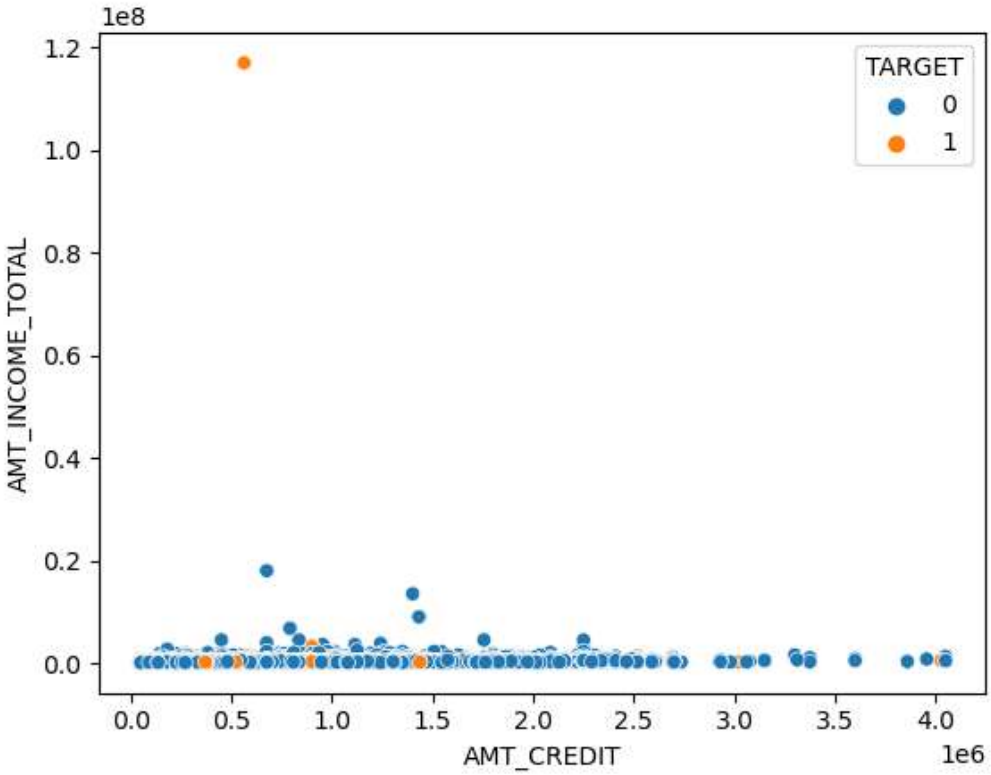
In [140]:
```
#Bivariet
sns.scatterplot(data=num_data,x='AMT_CREDIT',y='AMT_GOODS_PRICE',hue='TARGET')
```

Out[140]:
```
<Axes: xlabel='AMT_CREDIT', ylabel='AMT_GOODS_PRICE'>
```



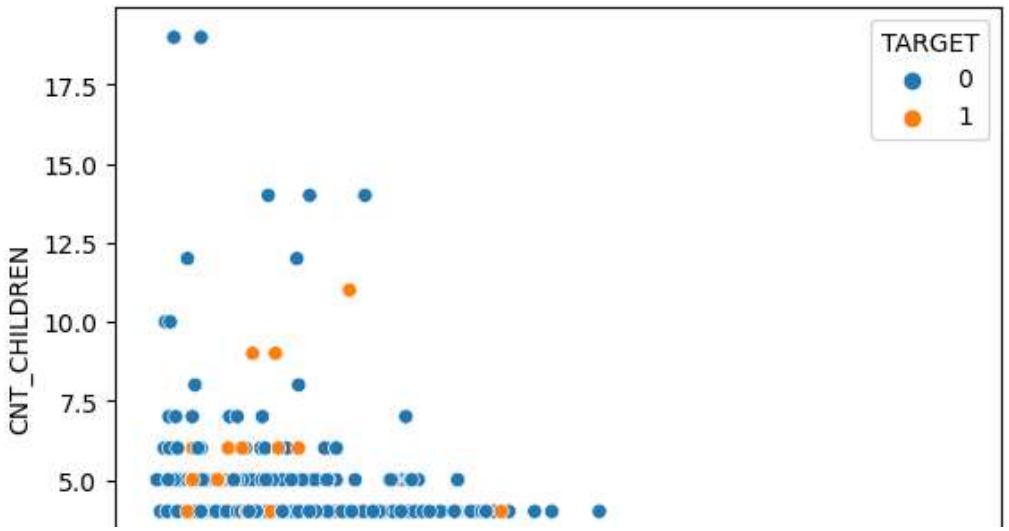
In [141]:
```
sns.scatterplot(data=num_data,x='AMT_CREDIT',y='AMT_INCOME_TOTAL',hue='TARGET')
```

Out[141]:
```
<Axes: xlabel='AMT_CREDIT', ylabel='AMT_INCOME_TOTAL'>
```



In [142]:
```
sns.scatterplot(data=num_data,x='AMT_CREDIT',y='CNT_CHILDREN',hue='TARGET')
```

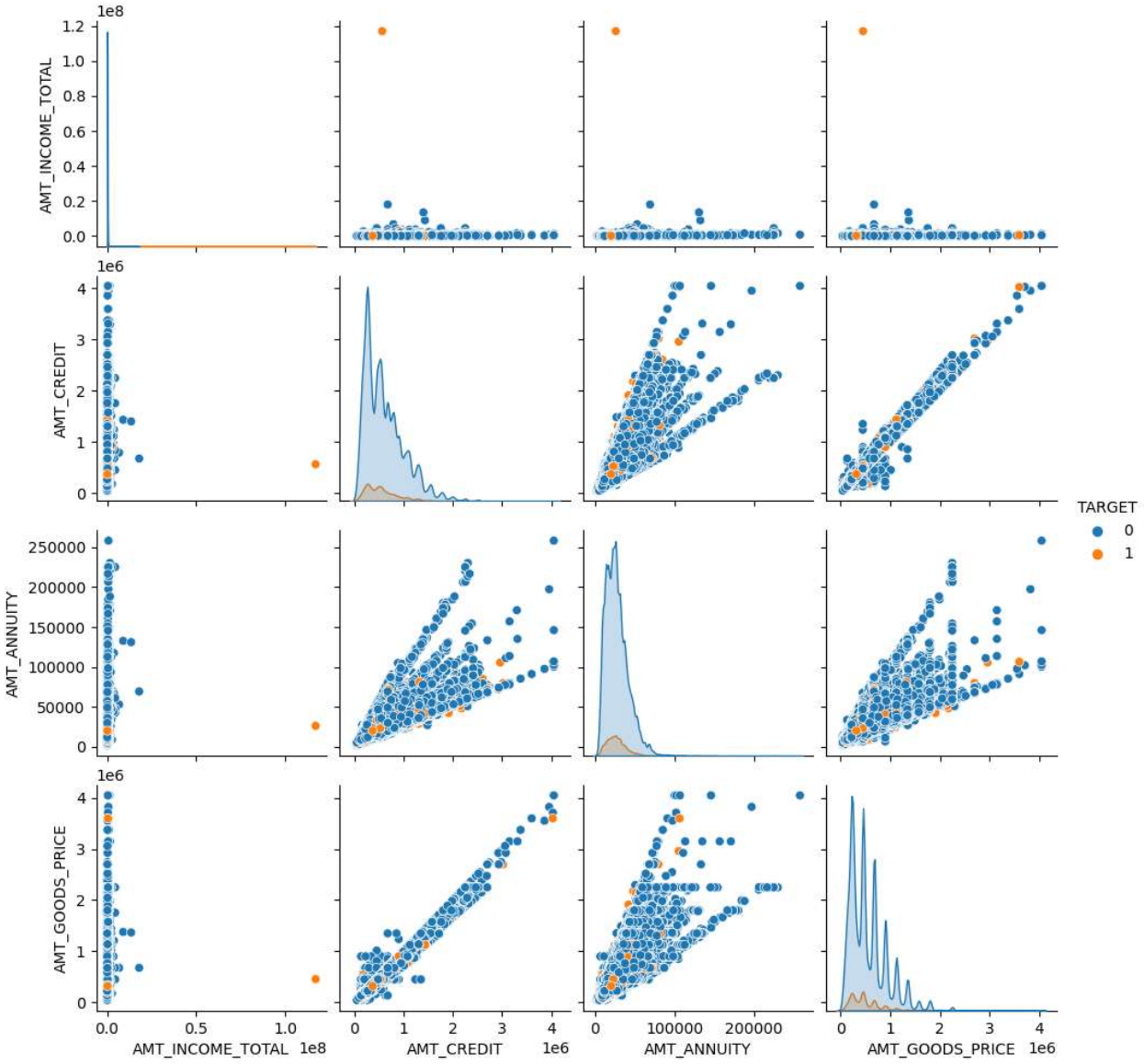
Out[142]:
```
<Axes: xlabel='AMT_CREDIT', ylabel='CNT_CHILDREN'>
```



In [143]:
```
amt_var=num_data[['AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY','AMT_GOODS_PRICE','TARGET']]
```

In [144]:
```
sns.pairplot(data=amt_var,hue='TARGET')
```

Out[144]:
```
<seaborn.axisgrid.PairGrid at 0x2c182c67190>
```



In [145]:
```
null_count=pd.DataFrame(prev_app.isnull().sum().sort_values(ascending=False)/prev_app.shape[
null_count.head()
```

Out[145]:

	var	count_pct
0	RATE_INTEREST_PRIVILEGED	99.643698
1	RATE_INTEREST_PRIMARY	99.643698
2	AMT_DOWN_PAYMENT	53.636480
3	RATE_DOWN_PAYMENT	53.636480
4	NAME_TYPE_SUITE	49.119754

In [146]:
```
var_msng_ge_40 = list(null_count[null_count['count_pct']>=40]['var'])
var_msng_ge_40
```

Out[146]:
```
['RATE_INTEREST_PRIVILEGED',
 'RATE_INTEREST_PRIMARY',
 'AMT_DOWN_PAYMENT',
 'RATE_DOWN_PAYMENT',
 'NAME_TYPE_SUITE',
 'NFLAG_INSURED_ON_APPROVAL',
 'DAYS_TERMINATION',
 'DAYS_LAST_DUE',
 'DAYS_LAST_DUE_1ST_VERSION',
 'DAYS_FIRST_DUE',
 'DAYS_FIRST_DRAWING']
```

In [147]:
```
nva_cols=var_msng_ge_40+['WEEKDAY_APPR_PROCESS_START','HOUR_APPR_PROCESS_START','FLAG_LAST_A
len(nva_cols)
```

Out[147]: 15

In [148]:
```
len(prev_app.columns)
```

Out[148]: 37

In [149]:
```
prev_app_nva_col_rmvd=prev_app.drop(labels=nva_cols,axis=1)
len(prev_app_nva_col_rmvd.columns)
```

Out[149]: 22

In [150]:
```
prev_app_nva_col_rmvd.head()
```

Out[150]:

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_G
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 [151]:
```
prev_app_nva_col_rmvd.isnull().sum().sort_values(ascending=False)/prev_app_nva_col_rmvd.shape
```

Out[151]:
```
AMT_GOODS_PRICE           23.081773
AMT_ANNUITY               22.286665
CNT_PAYMENT               22.286366
PRODUCT_COMBINATION        0.020716
AMT_CREDIT                 0.000060
NAME_GOODS_CATEGORY        0.000000
NAME_YIELD_GROUP           0.000000
NAME_SELLER_INDUSTRY       0.000000
SELLERPLACE_AREA           0.000000
CHANNEL_TYPE               0.000000
NAME_PRODUCT_TYPE          0.000000
NAME_PORTFOLIO             0.000000
SK_ID_PREV                 0.000000
NAME_CLIENT_TYPE           0.000000
SK_ID_CURR                 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
CODE_REJECT_REASON         0.000000
dtype: float64
```

In [152]:
```
prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].agg(func=['mean','median'])
```

Out[152]:
```
mean      227847.279283
median    112320.000000
Name: AMT_GOODS_PRICE, dtype: float64
```

In [153]:
```
prev_app_nva_col_rmvd['AMT_GOODS_PRICE_MEDIAN']=prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].fil
```

In [154]:
```
prev_app_nva_col_rmvd['AMT_GOODS_PRICE_MEAN']=prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].fillna
prev_app_nva_col_rmvd['AMT_GOODS_PRICE_MODE']=prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].fillna
```

In [155]:
```
prev_app_nva_col_rmvd.groupby(['AMT_GOODS_PRICE_MEDIAN']).size().sort_values(ascending=False
```

Out[155]:
```
AMT_GOODS_PRICE_MEDIAN
112320.000     385545
45000.000       47831
225000.000      43549
135000.000      40666
450000.000      38926
90000.000       29367
180000.000      24736
270000.000      20567
675000.000      20235
67500.000       16857
900000.000      15572
112500.000      14373
315000.000      11920
229500.000      10756
337500.000      10537
360000.000      10286
157500.000      10220
1350000.000      8434
```

In [156]:
```
prev_app_nva_col_rmvd.groupby(['AMT_GOODS_PRICE_MEAN']).size().sort_values(ascending=False)
```

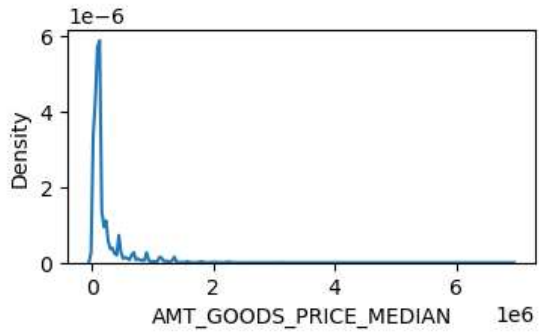
Out[156]:
```
AMT_GOODS_PRICE_MEAN
2.278473e+05    385515
4.500000e+04     47831
2.250000e+05     43549
1.350000e+05     40666
4.500000e+05     38926
9.000000e+04     29367
1.800000e+05     24736
2.700000e+05     20567
6.750000e+05     20235
6.750000e+04     16857
9.000000e+05     15572
1.125000e+05     14373
3.150000e+05     11920
2.295000e+05     10756
3.375000e+05     10537
3.600000e+05     10286
1.575000e+05     10220
1.350000e+06      8434
```

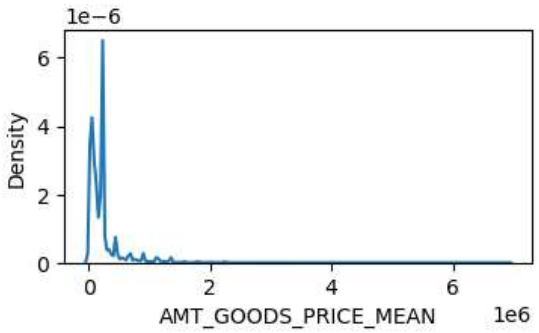
In [157]:
```
prev_app_nva_col_rmvd.groupby(['AMT_GOODS_PRICE_MODE']).size().sort_values(ascending=False)
```

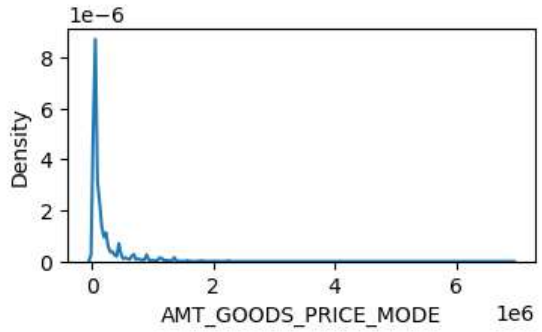
Out[157]:
```
AMT_GOODS_PRICE_MODE
45000.000      433346
225000.000      43549
135000.000      40666
450000.000      38926
90000.000       29367
180000.000      24736
270000.000      20567
675000.000      20235
67500.000       16857
900000.000      15572
112500.000      14373
315000.000      11920
229500.000      10756
337500.000      10537
360000.000      10286
157500.000      10220
1350000.000      8434
202500.000       7937
```

In [158]:
```
gp_cols=['AMT_GOODS_PRICE_MEDIAN','AMT_GOODS_PRICE_MEAN','AMT_GOODS_PRICE_MODE']
```

In [159]:
```
plt.figure(figsize=(10,5))

for i,col in enumerate(gp_cols):
    plt.subplot(2,2,i+1)
    sns.kdeplot(data=prev_app_nva_col_rmvd,x=col)
    plt.subplots_adjust(wspace=0.5,hspace=0.5)
```







In [160]:
```
prev_app_nva_col_rmvd['AMT_GOODS_PRICE']=prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].fillna(prev
```

In [161]:
```
prev_app_nva_col_rmvd['AMT_GOODS_PRICE'].isnull().sum()
```

Out[161]:
```
0
```

In [162]:
```
prev_app_nva_col_rmvd['AMT_ANNUITY'].agg(func=['mean','median','max'])
```

Out[162]:
```
mean       15955.120659
median     11250.000000
max       418058.145000
Name: AMT_ANNUITY, dtype: float64
```

In [163]:
```
prev_app_nva_col_rmvd['AMT_ANNUITY']=prev_app_nva_col_rmvd['AMT_ANNUITY'].fillna(prev_app_nva
```

In [164]:
```
prev_app_nva_col_rmvd['AMT_ANNUITY'].isnull().sum()
```

Out[164]:
```
0
```

In [165]:
```
prev_app_nva_col_rmvd['PRODUCT_COMBINATION'].head()
```

Out[165]:
```
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 [166]:
```
prev_app_nva_col_rmvd['PRODUCT_COMBINATION']=prev_app_nva_col_rmvd['PRODUCT_COMBINATION'].fi
```

In [167]:
```
prev_app_nva_col_rmvd['CNT_PAYMENT'].agg(func=['mean','median','max'])
```

Out[167]:
```
mean      16.054082
median    12.000000
max       84.000000
Name: CNT_PAYMENT, dtype: float64
```

In [168]:
```
prev_app_nva_col_rmvd[prev_app_nva_col_rmvd['CNT_PAYMENT'].isnull()].groupby(['NAME_CONTRACT_
```

Out[168]:
```
NAME_CONTRACT_STATUS
Canceled        305805
Refused          40897
Unused offer     25524
Approved             4
dtype: int64
```

In [169]:
```
#we cant fill with mean median mode becuase it bad impact on data as canceled data is too muc
prev_app_nva_col_rmvd['CNT_PAYMENT']=prev_app_nva_col_rmvd['CNT_PAYMENT'].fillna(0)
```

In [170]:
```
prev_app_nva_col_rmvd.isnull().sum().sort_values(ascending=False)
```

Out[170]:
```
AMT_CREDIT                1
SK_ID_PREV                0
NAME_GOODS_CATEGORY       0
AMT_GOODS_PRICE_MEAN      0
AMT_GOODS_PRICE_MEDIAN    0
PRODUCT_COMBINATION       0
NAME_YIELD_GROUP          0
CNT_PAYMENT               0
NAME_SELLER_INDUSTRY      0
SELLERPLACE_AREA          0
CHANNEL_TYPE              0
NAME_PRODUCT_TYPE         0
NAME_PORTFOLIO            0
NAME_CLIENT_TYPE          0
SK_ID_CURR                0
CODE_REJECT_REASON        0
NAME_PAYMENT_TYPE         0
DAYS_DECISION             0
NAME_CONTRACT_STATUS      0
NAME_CASH_LOAN_PURPOSE    0
AMT_GOODS_PRICE           0
AMT_APPLICATION           0
AMT_ANNUITY               0
NAME_CONTRACT_TYPE        0
AMT_GOODS_PRICE_MODE      0
dtype: int64
```

In [171]:
```
prev_app_nva_col_rmvd=prev_app_nva_col_rmvd.drop(labels=['AMT_GOODS_PRICE_MEDIAN','AMT_GOODS_
```

In [172]:
```
prev_app_nva_col_rmvd.isnull().sum().sort_values(ascending=False)
```

Out[172]:
```
AMT_CREDIT                1
SK_ID_PREV                0
NAME_CLIENT_TYPE          0
NAME_YIELD_GROUP          0
CNT_PAYMENT               0
NAME_SELLER_INDUSTRY      0
SELLERPLACE_AREA          0
CHANNEL_TYPE              0
NAME_PRODUCT_TYPE         0
NAME_PORTFOLIO            0
NAME_GOODS_CATEGORY       0
CODE_REJECT_REASON        0
SK_ID_CURR                0
NAME_PAYMENT_TYPE         0
DAYS_DECISION             0
NAME_CONTRACT_STATUS      0
NAME_CASH_LOAN_PURPOSE    0
AMT_GOODS_PRICE           0
AMT_APPLICATION           0
AMT_ANNUITY               0
NAME_CONTRACT_TYPE        0
PRODUCT_COMBINATION       0
dtype: int64
```

In [173]:
```
len(prev_app_nva_col_rmvd.columns)
```

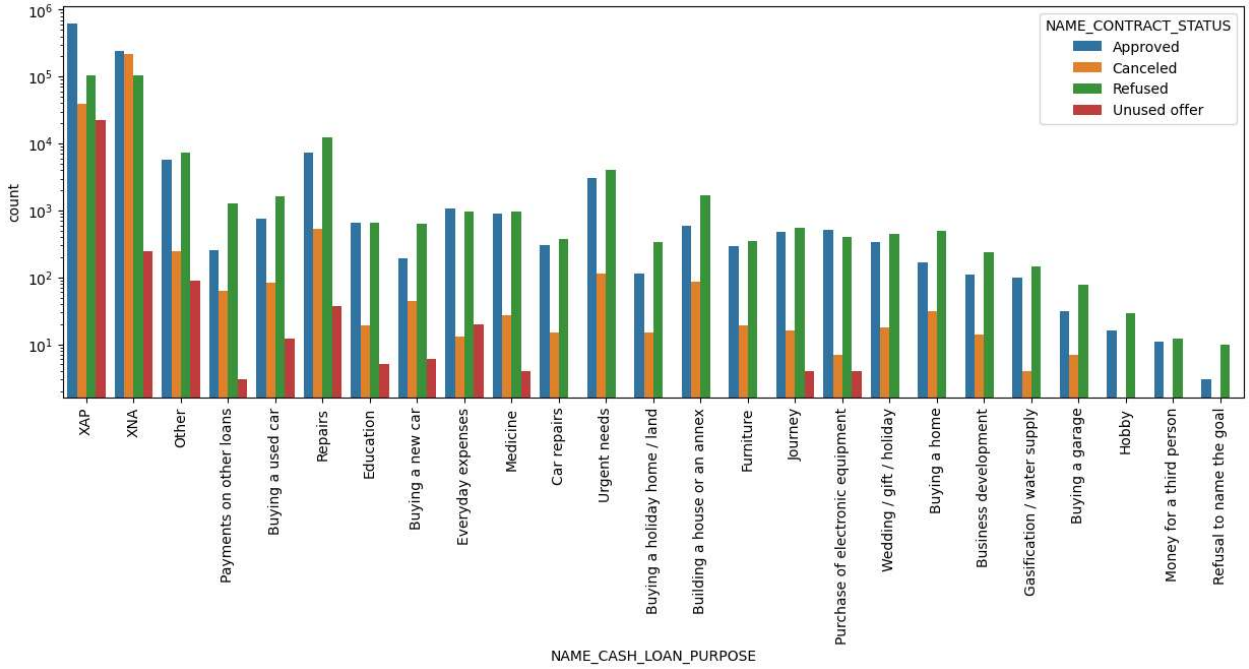
Out[173]:
```
22
```

In [174]:
```
merged_df=pd.merge(app_score_col_rmvd,prev_app_nva_col_rmvd, how='inner',on='SK_ID_CURR')
plt.figuremerged_df.head()
```

Out[174]:

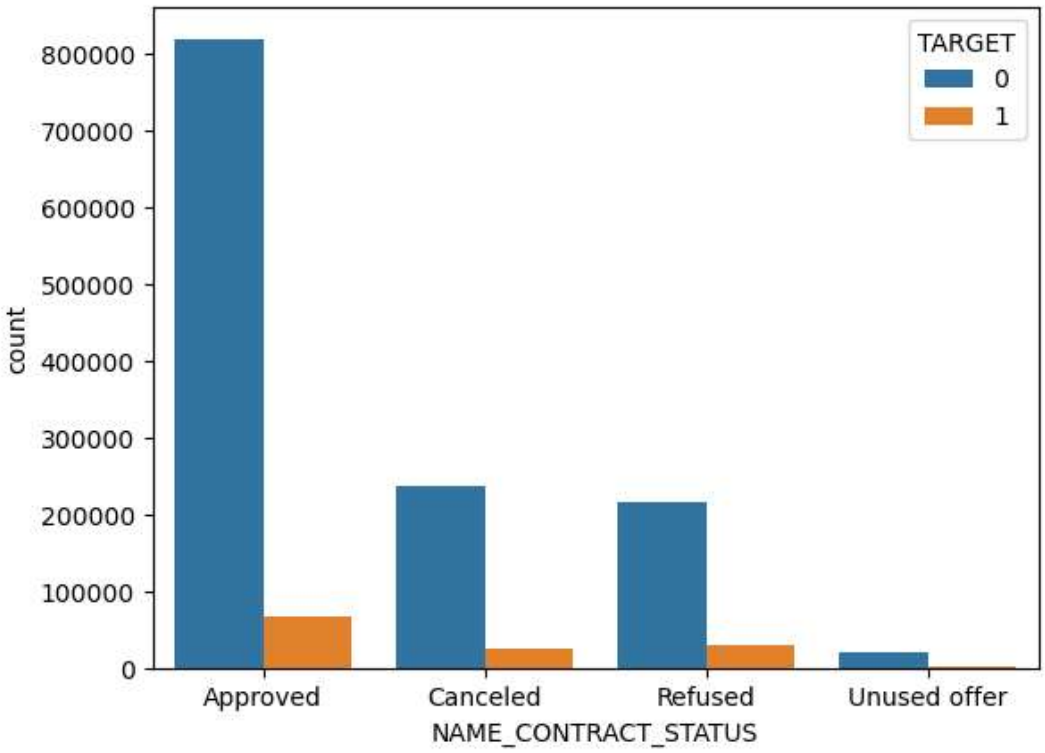
...T_INCOME_TOTAL	AMT_CREDIT_x	AMT_ANNUITY_x	AMT_GOODS_PRICE_x	NAME_TYPE_SUITE	NAME_INCOME_T
202500.0	406597.5	24700.5	351000.0	Unaccompanied	Wor
270000.0	1293502.5	35698.5	1129500.0	Family	State ser
270000.0	1293502.5	35698.5	1129500.0	Family	State ser
270000.0	1293502.5	35698.5	1129500.0	Family	State ser
67500.0	135000.0	6750.0	135000.0	Unaccompanied	Wor

In [175]:
```
plt.figure(figsize=(15,5))

sns.countplot(data=merged_df,x='NAME_CASH_LOAN_PURPOSE',hue='NAME_CONTRACT_STATUS')
plt.xticks(rotation=90)
plt.yscale('log')
```



In [181]:
```
sns.countplot(data=merged_df,x='NAME_CONTRACT_STATUS',hue='TARGET')
```

Out[181]: <Axes: xlabel='NAME_CONTRACT_STATUS', ylabel='count'>



In [185]:
```
merged_agg=merged_df.groupby(['NAME_CONTRACT_STATUS','TARGET']).size().reset_index().rename(
merged_agg
```

Out[185]:

	NAME_CONTRACT_STATUS	TARGET	counts
0	Approved	0	818856
1	Approved	1	67243
2	Canceled	0	235641
3	Canceled	1	23800
4	Refused	0	215952
5	Refused	1	29438
6	Unused offer	0	20892
7	Unused offer	1	1879

In [190]:
```
sum_df=merged_agg.groupby(['NAME_CONTRACT_STATUS'])['counts'].sum().reset_index()
sum_df
```

Out[190]:

	NAME_CONTRACT_STATUS	counts
0	Approved	886099
1	Canceled	259441
2	Refused	245390
3	Unused offer	22771

In [192]:
```
merged_agg_2=pd.merge(merged_agg,sum_df,how='left',on='NAME_CONTRACT_STATUS')
merged_agg_2
```

Out[192]:

	NAME_CONTRACT_STATUS	TARGET	counts_x	counts_y
0	Approved	0	818856	886099
1	Approved	1	67243	886099
2	Canceled	0	235641	259441
3	Canceled	1	23800	259441
4	Refused	0	215952	245390
5	Refused	1	29438	245390
6	Unused offer	0	20892	22771
7	Unused offer	1	1879	22771

In [193]:
```
merged_agg_2['PCT']=round(merged_agg_2['counts_x']/merged_agg_2['counts_y']*100, 2)
```

In [194]:
```
merged_agg_2
```

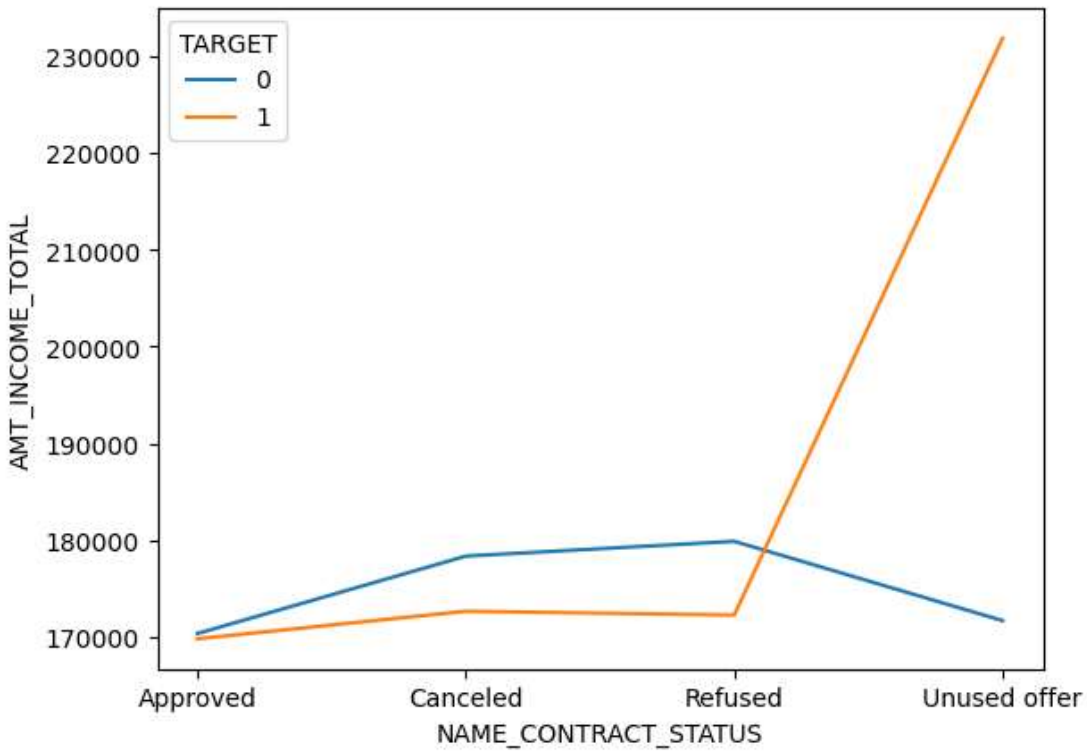
Out[194]:

	NAME_CONTRACT_STATUS	TARGET	counts_x	counts_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
7	Unused offer	1	1879	22771	8.25

In [197]:
```
sns.lineplot(data=merged_df,x='NAME_CONTRACT_STATUS',y='AMT_INCOME_TOTAL',ci=None,hue='TARGE
```

```
C:\Users\mshiv\AppData\Local\Temp\ipykernel_9548\563267390.py:1: FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

  sns.lineplot(data=merged_df,x='NAME_CONTRACT_STATUS',y='AMT_INCOME_TOTAL',ci=None,hue='TA
RGET')
```

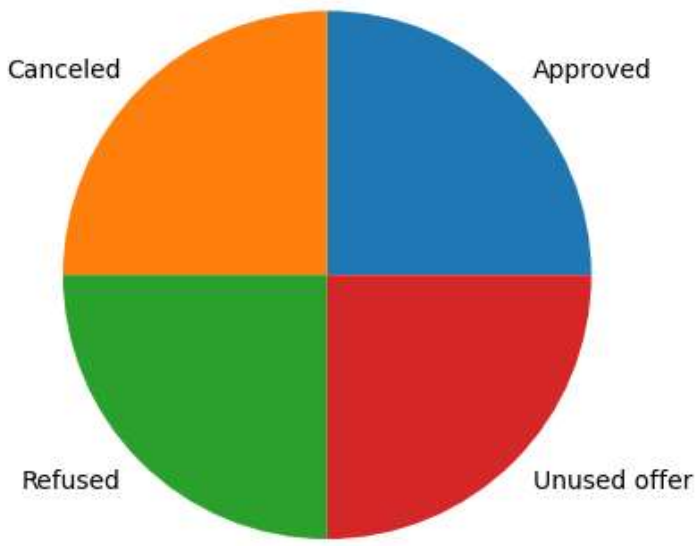
Out[197]:
```
<Axes: xlabel='NAME_CONTRACT_STATUS', ylabel='AMT_INCOME_TOTAL'>
```



In [200]:
```
chrt_pie=merged_agg_2.groupby('NAME_CONTRACT_STATUS')['NAME_CONTRACT_STATUS'].count()
chrt_pie
```

Out[200]:
```
NAME_CONTRACT_STATUS
Approved        2
Canceled        2
Refused         2
Unused offer    2
Name: NAME_CONTRACT_STATUS, dtype: int64
```

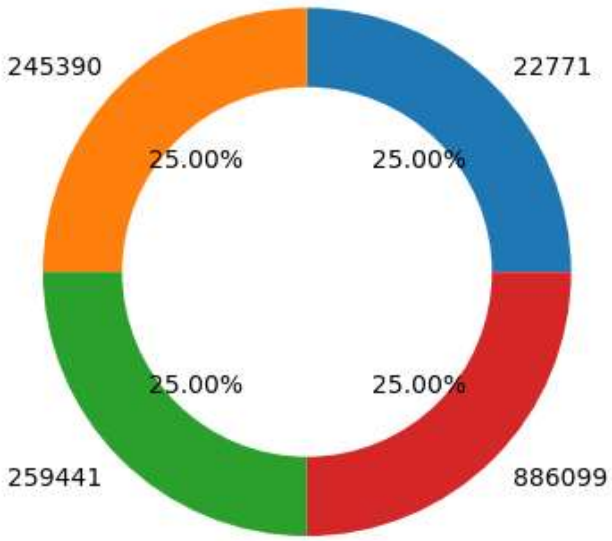
In [205]:
```
plt.pie(chrt_pie,labels=chrt_pie.index)
plt.show()
```



In [204]:
```
chrt_dnt=merged_agg_2.groupby('counts_y')['counts_y'].count()
chrt_dnt
```

Out[204]:
```
counts_y
22771     2
245390    2
259441    2
886099    2
Name: counts_y, dtype: int64
```

In [206]:
```
plt.pie(chrt_dnt, labels=chrt_dnt.index, autopct='%.2f%%', wedgeprops=dict(width=0.3))
plt.show()
```



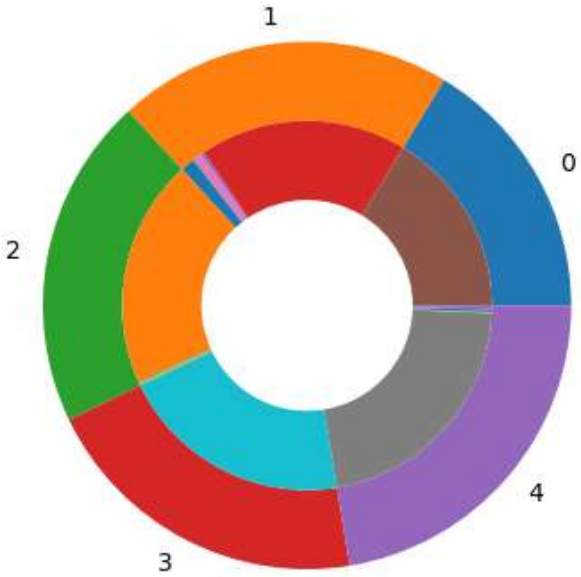
In [209]:
```
chrt_dnt_2=pd.crosstab(merged_df.AMT_GOODS_PRICE_x,merged_df.NAME_INCOME_TYPE)
```

In [215]:
```
chrt_dnt_plting=chrt_dnt_2.head().reset_index()
chrt_dnt_plting
```

Out[215]:

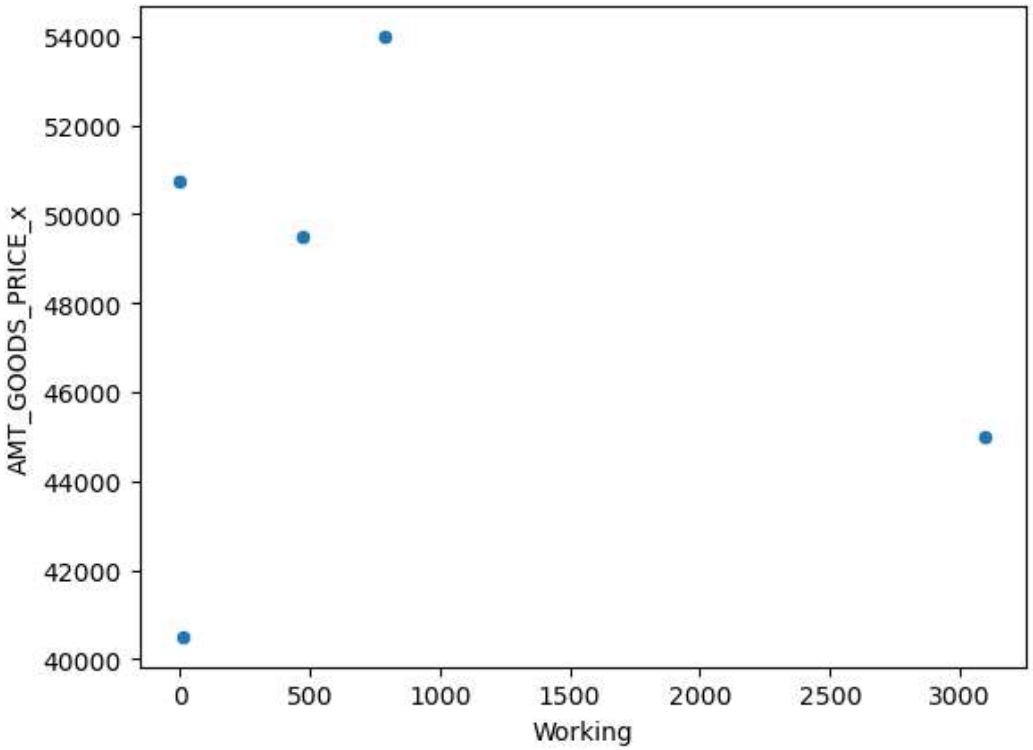
NAME_INCOME_TYPE	AMT_GOODS_PRICE_x	Commercial associate	Maternity leave	Pensioner	State servant	Student	Unemployed	W
0	40500.0	0	0	0	0	0	0	
1	45000.0	731	0	1770	299	0	0	
2	49500.0	238	0	165	67	0	0	
3	50751.0	0	0	12	0	0	0	
4	54000.0	220	0	574	59	0	0	

In [216]:
```
plt.pie(chrt_dnt_plting.sum(axis=1), labels=chrt_dnt_plting.index,
       radius=1,wedgeprops=dict(width=0.3))
plt.pie(chrt_dnt_plting.values.flatten(), radius=0.7,wedgeprops=dict(width=0.3))
plt.show()
```



In [218]:
```
sns.scatterplot(x='Working',y='AMT_GOODS_PRICE_x',data=chrt_dnt_plting)
```

Out[218]: <Axes: xlabel='Working', ylabel='AMT_GOODS_PRICE_x'>



In [ ]: