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

In [2]:
```
# Objective:
# This case study aims to identify patterns which indicate if a client
# has difficulty paying their instalments which may be used for taking
# actions such as denying the loan, reducing the amount of loan, lending
# (too risky applicants) at a higher interest rate, etc. This will ensure
# that the consumers capable of repaying the loan are not rejected.
# Identification of such applicant's using EDA is the aim of this case study.
```

In [3]:
```
data = pd.read_csv("./data/application_data.csv")
```

Cleaning

In [4]:
```
data.head()
```

Out[4]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALT
0	100002	1	Cash loans	M	N	
1	100003	0	Cash loans	F	N	
2	100004	0	Revolving loans	M	Y	
3	100006	0	Cash loans	F	N	
4	100007	0	Cash loans	M	N	

5 rows × 122 columns

In [5]:
```
# Insight:
# There are huge amount of cloumns,
# the data could ne unbalanced.
# Should check data duplication,removing null rows.
```

In [6]:
```
data.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]:
```
# Insight: TARGET seems to be the target variable
```

In [8]:
```
data['TARGET'].value_counts(normalize=True)
```

Out[8]:
```
0    0.919271
1    0.080729
Name: TARGET, dtype: float64
```

In [9]:
```
# Insight: Confirmed - Imbalanced data
```

In [10]:
```
data.shape
```

Out[10]:
```
(307511, 122)
```

In [11]:
```
data.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 [12]:
```
data.info(verbose = True)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Data columns (total 122 columns):
 #    Column                        Dtype  
---   ------                        -----  
 0    SK_ID_CURR                    int64  
 1    TARGET                        int64  
 2    NAME_CONTRACT_TYPE            object 
 3    CODE_GENDER                   object 
 4    FLAG_OWN_CAR                  object 
 5    FLAG_OWN_REALTY               object 
 6    CNT_CHILDREN                  int64  
 7    AMT_INCOME_TOTAL              float64
 8    AMT_CREDIT                    float64
 9    AMT_ANNUITY                   float64
 10   AMT_GOODS_PRICE               float64
 11   NAME_TYPE_SUITE               object 
 12   NAME_INCOME_TYPE              object 
 13   NAME_EDUCATION_TYPE           object 
 14   NAME_FAMILY_STATUS            object 
 15   NAME_HOUSING_TYPE             object 
 16   REGION_POPULATION_RELATIVE    float64
 17   DAYS_BIRTH                    int64  
 18   DAYS_EMPLOYED                 int64  
 19   DAYS_REGISTRATION             float64
 20   DAYS_ID_PUBLISH               int64  
 21   OWN_CAR_AGE                   float64
 22   FLAG_MOBIL                    int64  
 23   FLAG_EMP_PHONE                int64  
 24   FLAG_WORK_PHONE               int64  
 25   FLAG_CONT_MOBILE              int64  
 26   FLAG_PHONE                    int64  
 27   FLAG_EMAIL                    int64  
 28   OCCUPATION_TYPE               object 
 29   CNT_FAM_MEMBERS               float64
 30   REGION_RATING_CLIENT          int64  
 31   REGION_RATING_CLIENT_W_CITY   int64  
 32   WEEKDAY_APPR_PROCESS_START    object 
 33   HOUR_APPR_PROCESS_START       int64  
 34   REG_REGION_NOT_LIVE_REGION    int64  
 35   REG_REGION_NOT_WORK_REGION    int64  
 36   LIVE_REGION_NOT_WORK_REGION   int64  
 37   REG_CITY_NOT_LIVE_CITY        int64  
 38   REG_CITY_NOT_WORK_CITY        int64  
 39   LIVE_CITY_NOT_WORK_CITY       int64  
 40   ORGANIZATION_TYPE             object 
 41   EXT_SOURCE_1                  float64
 42   EXT_SOURCE_2                  float64
 43   EXT_SOURCE_3                  float64
 44   APARTMENTS_AVG                float64
 45   BASEMENTAREA_AVG              float64
 46   YEARS_BEGINEXPLUATATION_AVG   float64
 47   YEARS_BUILD_AVG               float64
 48   COMMONAREA_AVG                float64
 49   ELEVATORS_AVG                 float64
 50   ENTRANCES_AVG                 float64
 51   FLOORSMAX_AVG                 float64
 52   FLOORSMIN_AVG                 float64
 53   LANDAREA_AVG                  float64
 54   LIVINGAPARTMENTS_AVG          float64
```

```
55   LIVINGAREA_AVG                float64
56   NONLIVINGAPARTMENTS_AVG       float64
57   NONLIVINGAREA_AVG             float64
58   APARTMENTS_MODE               float64
59   BASEMENTAREA_MODE             float64
60   YEARS_BEGINEXPLUATATION_MODE  float64
61   YEARS_BUILD_MODE              float64
62   COMMONAREA_MODE               float64
63   ELEVATORS_MODE                float64
64   ENTRANCES_MODE                float64
65   FLOORSMAX_MODE                float64
66   FLOORSMIN_MODE                float64
67   LANDAREA_MODE                 float64
68   LIVINGAPARTMENTS_MODE         float64
69   LIVINGAREA_MODE               float64
70   NONLIVINGAPARTMENTS_MODE      float64
71   NONLIVINGAREA_MODE            float64
72   APARTMENTS_MEDI               float64
73   BASEMENTAREA_MEDI             float64
74   YEARS_BEGINEXPLUATATION_MEDI  float64
75   YEARS_BUILD_MEDI              float64
76   COMMONAREA_MEDI               float64
77   ELEVATORS_MEDI                float64
78   ENTRANCES_MEDI                float64
79   FLOORSMAX_MEDI                float64
80   FLOORSMIN_MEDI                float64
81   LANDAREA_MEDI                 float64
82   LIVINGAPARTMENTS_MEDI         float64
83   LIVINGAREA_MEDI               float64
84   NONLIVINGAPARTMENTS_MEDI      float64
85   NONLIVINGAREA_MEDI            float64
86   FONDKAPREMONT_MODE            object
87   HOUSETYPE_MODE                object
88   TOTALAREA_MODE                float64
89   WALLSMATERIAL_MODE            object
90   EMERGENCYSTATE_MODE           object
91   OBS_30_CNT_SOCIAL_CIRCLE      float64
92   DEF_30_CNT_SOCIAL_CIRCLE      float64
93   OBS_60_CNT_SOCIAL_CIRCLE      float64
94   DEF_60_CNT_SOCIAL_CIRCLE      float64
95   DAYS_LAST_PHONE_CHANGE        float64
96   FLAG_DOCUMENT_2               int64
97   FLAG_DOCUMENT_3               int64
98   FLAG_DOCUMENT_4               int64
99   FLAG_DOCUMENT_5               int64
100  FLAG_DOCUMENT_6               int64
101  FLAG_DOCUMENT_7               int64
102  FLAG_DOCUMENT_8               int64
103  FLAG_DOCUMENT_9               int64
104  FLAG_DOCUMENT_10              int64
105  FLAG_DOCUMENT_11              int64
106  FLAG_DOCUMENT_12              int64
107  FLAG_DOCUMENT_13              int64
108  FLAG_DOCUMENT_14              int64
109  FLAG_DOCUMENT_15              int64
110  FLAG_DOCUMENT_16              int64
111  FLAG_DOCUMENT_17              int64
112  FLAG_DOCUMENT_18              int64
113  FLAG_DOCUMENT_19              int64
114  FLAG_DOCUMENT_20              int64
```

```
 115  FLAG_DOCUMENT_21              int64
 116  AMT_REQ_CREDIT_BUREAU_HOUR    float64
 117  AMT_REQ_CREDIT_BUREAU_DAY     float64
 118  AMT_REQ_CREDIT_BUREAU_WEEK    float64
 119  AMT_REQ_CREDIT_BUREAU_MON     float64
 120  AMT_REQ_CREDIT_BUREAU_QRT     float64
 121  AMT_REQ_CREDIT_BUREAU_YEAR    float64
dtypes: float64(65), int64(41), object(16)
memory usage: 286.2+ MB
```

In [13]:

```
data.dtypes.value_counts()
```

Out[13]:

```
float64    65
int64      41
object     16
dtype: int64
```

In [14]:

```
# Insight: 16 categorical variables and rest numerical features.
# we should convert numeric variables to categorical.
```

In [15]:

```
data.describe()
```

Out[15]:

	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNU
count	307511.000000	307511.000000	307511.000000	3.075110e+05	3.075110e+05	307499.000
mean	278180.518577	0.080729	0.417052	1.687979e+05	5.990260e+05	27108.573
std	102790.175348	0.272419	0.722121	2.371231e+05	4.024908e+05	14493.737
min	100002.000000	0.000000	0.000000	2.565000e+04	4.500000e+04	1615.500
25%	189145.500000	0.000000	0.000000	1.125000e+05	2.700000e+05	16524.000
50%	278202.000000	0.000000	0.000000	1.471500e+05	5.135310e+05	24903.000
75%	367142.500000	0.000000	1.000000	2.025000e+05	8.086500e+05	34596.000
max	456255.000000	1.000000	19.000000	1.170000e+08	4.050000e+06	258025.500

8 rows × 106 columns

In [16]:

```
# 2,application_data,TARGET,"Target variable
# (1 - client with payment difficulties: he/she had late payment more than X days on a
# 0 - all other cases)",
```

In [17]:

```
pd.set_option('display.max_rows', 122)
data.head(1).T
```

Out[17]:

	0
SK_ID_CURR	100002
TARGET	1
NAME_CONTRACT_TYPE	Cash loans
CODE_GENDER	M
FLAG_OWN_CAR	N
FLAG_OWN_REALTY	Y
CNT_CHILDREN	0
AMT_INCOME_TOTAL	202500.0
AMT_CREDIT	406597.5
AMT_ANNUITY	24700.5
AMT_GOODS_PRICE	351000.0
NAME_TYPE_SUITE	Unaccompanied
NAME_INCOME_TYPE	Working
NAME_EDUCATION_TYPE	Secondary / secondary special
NAME_FAMILY_STATUS	Single / not married
NAME_HOUSING_TYPE	House / apartment
REGION_POPULATION_RELATIVE	0.018801
DAYS_BIRTH	-9461
DAYS_EMPLOYED	-637
DAYS_REGISTRATION	-3648.0
DAYS_ID_PUBLISH	-2120
OWN_CAR_AGE	NaN
FLAG_MOBIL	1
FLAG_EMP_PHONE	1
FLAG_WORK_PHONE	0
FLAG_CONT_MOBILE	1
FLAG_PHONE	1
FLAG_EMAIL	0
OCCUPATION_TYPE	Laborers
CNT_FAM_MEMBERS	1.0
REGION_RATING_CLIENT	2
REGION_RATING_CLIENT_W_CITY	2
WEEKDAY_APPR_PROCESS_START	WEDNESDAY
HOUR_APPR_PROCESS_START	10

	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	Business Entity Type 3
EXT_SOURCE_1	0.083037
EXT_SOURCE_2	0.262949
EXT_SOURCE_3	0.139376
APARTMENTS_AVG	0.0247
BASEMENTAREA_AVG	0.0369
YEARS_BEGINEXPLUATATION_AVG	0.9722
YEARS_BUILD_AVG	0.6192
COMMONAREA_AVG	0.0143
ELEVATORS_AVG	0.0
ENTRANCES_AVG	0.069
FLOORSMAX_AVG	0.0833
FLOORSMIN_AVG	0.125
LANDAREA_AVG	0.0369
LIVINGAPARTMENTS_AVG	0.0202
LIVINGAREA_AVG	0.019
NONLIVINGAPARTMENTS_AVG	0.0
NONLIVINGAREA_AVG	0.0
APARTMENTS_MODE	0.0252
BASEMENTAREA_MODE	0.0383
YEARS_BEGINEXPLUATATION_MODE	0.9722
YEARS_BUILD_MODE	0.6341
COMMONAREA_MODE	0.0144
ELEVATORS_MODE	0.0
ENTRANCES_MODE	0.069
FLOORSMAX_MODE	0.0833
FLOORSMIN_MODE	0.125
LANDAREA_MODE	0.0377

	0
LIVINGAPARTMENTS_MODE	0.022
LIVINGAREA_MODE	0.0198
NONLIVINGAPARTMENTS_MODE	0.0
NONLIVINGAREA_MODE	0.0
APARTMENTS_MEDI	0.025
BASEMENTAREA_MEDI	0.0369
YEARS_BEGINEXPLUATATION_MEDI	0.9722
YEARS_BUILD_MEDI	0.6243
COMMONAREA_MEDI	0.0144
ELEVATORS_MEDI	0.0
ENTRANCES_MEDI	0.069
FLOORSMAX_MEDI	0.0833
FLOORSMIN_MEDI	0.125
LANDAREA_MEDI	0.0375
LIVINGAPARTMENTS_MEDI	0.0205
LIVINGAREA_MEDI	0.0193
NONLIVINGAPARTMENTS_MEDI	0.0
NONLIVINGAREA_MEDI	0.0
FONDKAPREMONT_MODE	reg oper account
HOUSETYPE_MODE	block of flats
TOTALAREA_MODE	0.0149
WALLSMATERIAL_MODE	Stone, brick
EMERGENCYSTATE_MODE	No
OBS_30_CNT_SOCIAL_CIRCLE	2.0
DEF_30_CNT_SOCIAL_CIRCLE	2.0
OBS_60_CNT_SOCIAL_CIRCLE	2.0
DEF_60_CNT_SOCIAL_CIRCLE	2.0
DAYS_LAST_PHONE_CHANGE	-1134.0
FLAG_DOCUMENT_2	0
FLAG_DOCUMENT_3	1
FLAG_DOCUMENT_4	0
FLAG_DOCUMENT_5	0
FLAG_DOCUMENT_6	0
FLAG_DOCUMENT_7	0

	0
FLAG_DOCUMENT_8	0
FLAG_DOCUMENT_9	0
FLAG_DOCUMENT_10	0
FLAG_DOCUMENT_11	0
FLAG_DOCUMENT_12	0
FLAG_DOCUMENT_13	0
FLAG_DOCUMENT_14	0
FLAG_DOCUMENT_15	0
FLAG_DOCUMENT_16	0
FLAG_DOCUMENT_17	0
FLAG_DOCUMENT_18	0
FLAG_DOCUMENT_19	0
FLAG_DOCUMENT_20	0
FLAG_DOCUMENT_21	0
AMT_REQ_CREDIT_BUREAU_HOUR	0.0
AMT_REQ_CREDIT_BUREAU_DAY	0.0
AMT_REQ_CREDIT_BUREAU_WEEK	0.0
AMT_REQ_CREDIT_BUREAU_MON	0.0
AMT_REQ_CREDIT_BUREAU_QRT	0.0
AMT_REQ_CREDIT_BUREAU_YEAR	1.0

In [18]:
```
# Based on this and description mentioned in "cloumns_description.csv" file,
# Many irrelevant columns are there, should remove them,
# many correlated columns are there, should remove them,
# Should check the null rows and remove them
```

In [19]:
```
data.describe(include="object")
```

Out[19]:

	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	NAME_TYPE_SU
count	307511	307511	307511	307511	306
unique	2	3	2	2	
top	Cash loans	F	N	Y	Unaccompai
freq	278232	202448	202924	213312	248

In [20]:
```
n_rows = data.shape[0]
null_df = (data.isnull().sum()/n_rows*100).sort_values(ascending= False)
```

In [21]:
```
null_df.head(122)
```

Out[21]:
```
COMMONAREA_MEDI                 69.872297
COMMONAREA_AVG                  69.872297
COMMONAREA_MODE                 69.872297
NONLIVINGAPARTMENTS_MODE        69.432963
NONLIVINGAPARTMENTS_AVG         69.432963
NONLIVINGAPARTMENTS_MEDI        69.432963
FONDKAPREMONT_MODE              68.386172
LIVINGAPARTMENTS_MODE           68.354953
LIVINGAPARTMENTS_AVG            68.354953
LIVINGAPARTMENTS_MEDI           68.354953
FLOORSMIN_AVG                   67.848630
FLOORSMIN_MODE                  67.848630
FLOORSMIN_MEDI                  67.848630
YEARS_BUILD_MEDI                66.497784
YEARS_BUILD_MODE                66.497784
YEARS_BUILD_AVG                 66.497784
OWN_CAR_AGE                     65.990810
LANDAREA_MEDI                   59.376738
LANDAREA_MODE                   59.376738
LANDAREA_AVG                    59.376738
BASEMENTAREA_MEDI               58.515956
BASEMENTAREA_AVG                58.515956
BASEMENTAREA_MODE               58.515956
EXT_SOURCE_1                    56.381073
NONLIVINGAREA_MODE              55.179164
NONLIVINGAREA_AVG               55.179164
NONLIVINGAREA_MEDI              55.179164
ELEVATORS_MEDI                  53.295980
ELEVATORS_AVG                   53.295980
ELEVATORS_MODE                  53.295980
WALLSMATERIAL_MODE              50.840783
APARTMENTS_MEDI                 50.749729
APARTMENTS_AVG                  50.749729
APARTMENTS_MODE                 50.749729
ENTRANCES_MEDI                  50.348768
ENTRANCES_AVG                   50.348768
ENTRANCES_MODE                  50.348768
LIVINGAREA_AVG                  50.193326
LIVINGAREA_MODE                 50.193326
LIVINGAREA_MEDI                 50.193326
HOUSETYPE_MODE                  50.176091
FLOORSMAX_MODE                  49.760822
FLOORSMAX_MEDI                  49.760822
FLOORSMAX_AVG                   49.760822
YEARS_BEGINEXPLUATATION_MODE    48.781019
YEARS_BEGINEXPLUATATION_MEDI    48.781019
YEARS_BEGINEXPLUATATION_AVG     48.781019
TOTALAREA_MODE                  48.268517
EMERGENCYSTATE_MODE             47.398304
OCCUPATION_TYPE                 31.345545
EXT_SOURCE_3                    19.825307
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_QRT       13.501631
AMT_REQ_CREDIT_BUREAU_YEAR      13.501631
NAME_TYPE_SUITE                  0.420148
OBS_30_CNT_SOCIAL_CIRCLE         0.332021
DEF_30_CNT_SOCIAL_CIRCLE         0.332021
```

```
OBS_60_CNT_SOCIAL_CIRCLE        0.332021
DEF_60_CNT_SOCIAL_CIRCLE        0.332021
EXT_SOURCE_2                    0.214626
AMT_GOODS_PRICE                 0.090403
AMT_ANNUITY                     0.003902
CNT_FAM_MEMBERS                 0.000650
DAYS_LAST_PHONE_CHANGE          0.000325
CNT_CHILDREN                    0.000000
FLAG_DOCUMENT_8                 0.000000
NAME_CONTRACT_TYPE              0.000000
CODE_GENDER                     0.000000
FLAG_OWN_CAR                    0.000000
FLAG_DOCUMENT_2                 0.000000
FLAG_DOCUMENT_3                 0.000000
FLAG_DOCUMENT_4                 0.000000
FLAG_DOCUMENT_5                 0.000000
FLAG_DOCUMENT_6                 0.000000
FLAG_DOCUMENT_7                 0.000000
FLAG_DOCUMENT_9                 0.000000
FLAG_DOCUMENT_21                0.000000
FLAG_DOCUMENT_10                0.000000
FLAG_DOCUMENT_11                0.000000
FLAG_OWN_REALTY                 0.000000
FLAG_DOCUMENT_13                0.000000
FLAG_DOCUMENT_14                0.000000
FLAG_DOCUMENT_15                0.000000
FLAG_DOCUMENT_16                0.000000
FLAG_DOCUMENT_17                0.000000
FLAG_DOCUMENT_18                0.000000
FLAG_DOCUMENT_19                0.000000
FLAG_DOCUMENT_20                0.000000
FLAG_DOCUMENT_12                0.000000
AMT_CREDIT                      0.000000
AMT_INCOME_TOTAL                0.000000
FLAG_PHONE                      0.000000
LIVE_CITY_NOT_WORK_CITY         0.000000
REG_CITY_NOT_WORK_CITY          0.000000
TARGET                          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
REGION_RATING_CLIENT            0.000000
FLAG_EMAIL                      0.000000
FLAG_CONT_MOBILE                0.000000
ORGANIZATION_TYPE               0.000000
FLAG_WORK_PHONE                 0.000000
FLAG_EMP_PHONE                  0.000000
FLAG_MOBIL                      0.000000
DAYS_ID_PUBLISH                 0.000000
DAYS_REGISTRATION               0.000000
DAYS_EMPLOYED                   0.000000
DAYS_BIRTH                      0.000000
REGION_POPULATION_RELATIVE      0.000000
NAME_HOUSING_TYPE               0.000000
NAME_FAMILY_STATUS              0.000000
NAME_EDUCATION_TYPE             0.000000
```

```
NAME_INCOME_TYPE                   0.000000
SK_ID_CURR                         0.000000
dtype: float64
```

In [22]:
```
null_df.tail(70)
```

Out[22]:
```
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_QRT      13.501631
AMT_REQ_CREDIT_BUREAU_YEAR     13.501631
NAME_TYPE_SUITE                 0.420148
OBS_30_CNT_SOCIAL_CIRCLE        0.332021
DEF_30_CNT_SOCIAL_CIRCLE        0.332021
OBS_60_CNT_SOCIAL_CIRCLE        0.332021
DEF_60_CNT_SOCIAL_CIRCLE        0.332021
EXT_SOURCE_2                    0.214626
AMT_GOODS_PRICE                 0.090403
AMT_ANNUITY                     0.003902
CNT_FAM_MEMBERS                 0.000650
DAYS_LAST_PHONE_CHANGE          0.000325
CNT_CHILDREN                    0.000000
FLAG_DOCUMENT_8                 0.000000
NAME_CONTRACT_TYPE              0.000000
CODE_GENDER                     0.000000
FLAG_OWN_CAR                    0.000000
FLAG_DOCUMENT_2                 0.000000
FLAG_DOCUMENT_3                 0.000000
FLAG_DOCUMENT_4                 0.000000
FLAG_DOCUMENT_5                 0.000000
FLAG_DOCUMENT_6                 0.000000
FLAG_DOCUMENT_7                 0.000000
FLAG_DOCUMENT_9                 0.000000
FLAG_DOCUMENT_21                0.000000
FLAG_DOCUMENT_10                0.000000
FLAG_DOCUMENT_11                0.000000
FLAG_OWN_REALTY                 0.000000
FLAG_DOCUMENT_13                0.000000
FLAG_DOCUMENT_14                0.000000
FLAG_DOCUMENT_15                0.000000
FLAG_DOCUMENT_16                0.000000
FLAG_DOCUMENT_17                0.000000
FLAG_DOCUMENT_18                0.000000
FLAG_DOCUMENT_19                0.000000
FLAG_DOCUMENT_20                0.000000
FLAG_DOCUMENT_12                0.000000
AMT_CREDIT                      0.000000
AMT_INCOME_TOTAL                0.000000
FLAG_PHONE                      0.000000
LIVE_CITY_NOT_WORK_CITY         0.000000
REG_CITY_NOT_WORK_CITY          0.000000
TARGET                          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
REGION_RATING_CLIENT            0.000000
FLAG_EMAIL                      0.000000
FLAG_CONT_MOBILE                0.000000
ORGANIZATION_TYPE               0.000000
FLAG_WORK_PHONE                 0.000000
FLAG_EMP_PHONE                  0.000000
FLAG_MOBIL                      0.000000
```

```
DAYS_ID_PUBLISH                0.000000
DAYS_REGISTRATION              0.000000
DAYS_EMPLOYED                  0.000000
DAYS_BIRTH                     0.000000
REGION_POPULATION_RELATIVE     0.000000
NAME_HOUSING_TYPE              0.000000
NAME_FAMILY_STATUS             0.000000
NAME_EDUCATION_TYPE            0.000000
NAME_INCOME_TYPE               0.000000
SK_ID_CURR                     0.000000
dtype: float64
```

In [23]:
```
# Insight: So there are more than 50 features out of 122 features,
# whose more than 31 percent values are null.
# So we should remove these features.
```

In [24]:
```
data_v2 = data.dropna(axis=1, thresh=n_rows*0.7)
```

In [25]:
```
data_v2.shape
```

Out[25]:
```
(307511, 72)
```

In [26]:
```
# Just checking if the dropped columns have any direct correlation with the target var
```

In [27]:
```
original_columns = data.columns.tolist()
dropped_columns = [col for col in original_columns if col not in data_v2.columns]
```

In [28]:
```
dropped_columns
```

Out[28]:
```
['OWN_CAR_AGE',
 'OCCUPATION_TYPE',
 'EXT_SOURCE_1',
 'APARTMENTS_AVG',
 'BASEMENTAREA_AVG',
 'YEARS_BEGINEXPLUATATION_AVG',
 'YEARS_BUILD_AVG',
 'COMMONAREA_AVG',
 'ELEVATORS_AVG',
 'ENTRANCES_AVG',
 'FLOORSMAX_AVG',
 'FLOORSMIN_AVG',
 'LANDAREA_AVG',
 'LIVINGAPARTMENTS_AVG',
 'LIVINGAREA_AVG',
 'NONLIVINGAPARTMENTS_AVG',
 'NONLIVINGAREA_AVG',
 'APARTMENTS_MODE',
 'BASEMENTAREA_MODE',
 'YEARS_BEGINEXPLUATATION_MODE',
 'YEARS_BUILD_MODE',
 'COMMONAREA_MODE',
 'ELEVATORS_MODE',
 'ENTRANCES_MODE',
 'FLOORSMAX_MODE',
 'FLOORSMIN_MODE',
 'LANDAREA_MODE',
 'LIVINGAPARTMENTS_MODE',
 'LIVINGAREA_MODE',
 'NONLIVINGAPARTMENTS_MODE',
 'NONLIVINGAREA_MODE',
 'APARTMENTS_MEDI',
 'BASEMENTAREA_MEDI',
 'YEARS_BEGINEXPLUATATION_MEDI',
 'YEARS_BUILD_MEDI',
 'COMMONAREA_MEDI',
 'ELEVATORS_MEDI',
 'ENTRANCES_MEDI',
 'FLOORSMAX_MEDI',
 'FLOORSMIN_MEDI',
 'LANDAREA_MEDI',
 'LIVINGAPARTMENTS_MEDI',
 'LIVINGAREA_MEDI',
 'NONLIVINGAPARTMENTS_MEDI',
 'NONLIVINGAREA_MEDI',
 'FONDKAPREMONT_MODE',
 'HOUSETYPE_MODE',
 'TOTALAREA_MODE',
 'WALLSMATERIAL_MODE',
 'EMERGENCYSTATE_MODE']
```

In [29]:
```
selected_columns = dropped_columns + ['TARGET']
data_selected = data[selected_columns]
```

In [30]:
```
correlation_matrix = data_selected.corr()
correlation_with_target = correlation_matrix['TARGET'].drop('TARGET')  # Drop the targ
```

In [31]:
```
correlation_with_target.sort_values(ascending=False)
```

Out[31]:
```
OWN_CAR_AGE                      0.037612
NONLIVINGAPARTMENTS_MODE        -0.001557
NONLIVINGAPARTMENTS_MEDI        -0.002757
NONLIVINGAPARTMENTS_AVG         -0.003176
YEARS_BEGINEXPLUATATION_MODE    -0.009036
YEARS_BEGINEXPLUATATION_AVG     -0.009728
YEARS_BEGINEXPLUATATION_MEDI    -0.009993
LANDAREA_MODE                   -0.010174
LANDAREA_AVG                    -0.010885
LANDAREA_MEDI                   -0.011256
NONLIVINGAREA_MODE              -0.012711
NONLIVINGAREA_MEDI              -0.013337
NONLIVINGAREA_AVG               -0.013578
COMMONAREA_MODE                 -0.016340
ENTRANCES_MODE                  -0.017387
COMMONAREA_AVG                  -0.018550
COMMONAREA_MEDI                 -0.018573
ENTRANCES_MEDI                  -0.019025
ENTRANCES_AVG                   -0.019172
BASEMENTAREA_MODE               -0.019952
YEARS_BUILD_MODE                -0.022068
BASEMENTAREA_MEDI               -0.022081
YEARS_BUILD_AVG                 -0.022149
YEARS_BUILD_MEDI                -0.022326
BASEMENTAREA_AVG                -0.022746
LIVINGAPARTMENTS_MODE           -0.023393
LIVINGAPARTMENTS_MEDI           -0.024621
LIVINGAPARTMENTS_AVG            -0.025031
APARTMENTS_MODE                 -0.027284
APARTMENTS_MEDI                 -0.029184
APARTMENTS_AVG                  -0.029498
LIVINGAREA_MODE                 -0.030685
ELEVATORS_MODE                  -0.032131
TOTALAREA_MODE                  -0.032596
FLOORSMIN_MODE                  -0.032698
LIVINGAREA_MEDI                 -0.032739
LIVINGAREA_AVG                  -0.032997
FLOORSMIN_MEDI                  -0.033394
FLOORSMIN_AVG                   -0.033614
ELEVATORS_MEDI                  -0.033863
ELEVATORS_AVG                   -0.034199
FLOORSMAX_MODE                  -0.043226
FLOORSMAX_MEDI                  -0.043768
FLOORSMAX_AVG                   -0.044003
EXT_SOURCE_1                    -0.155317
Name: TARGET, dtype: float64
```

In [32]:
```
# I guess OCCUPATION_TYPE seems an important feature which should be investigated.
```

In [33]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [34]:
```
data_v2['OCCUPATION_TYPE'] = data['OCCUPATION_TYPE']
```

In [35]:
```
data_v2.shape
```

Out[35]:
```
(307511, 73)
```

In [36]:
```
data_v2.columns
```

Out[36]:
```
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', 'AMT_GOODS_PRICE', 'NAME_TYPE_SUITE',
       'NAME_INCOME_TYPE', 'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS',
       'NAME_HOUSING_TYPE', 'REGION_POPULATION_RELATIVE', 'DAYS_BIRTH',
       'DAYS_EMPLOYED', 'DAYS_REGISTRATION', 'DAYS_ID_PUBLISH', 'FLAG_MOBIL',
       'FLAG_EMP_PHONE', 'FLAG_WORK_PHONE', 'FLAG_CONT_MOBILE', 'FLAG_PHONE',
       'FLAG_EMAIL', 'CNT_FAM_MEMBERS', 'REGION_RATING_CLIENT',
       'REGION_RATING_CLIENT_W_CITY', 'WEEKDAY_APPR_PROCESS_START',
       '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', 'ORGANIZATION_TYPE', 'EXT_SOURCE_2',
       'EXT_SOURCE_3', 'OBS_30_CNT_SOCIAL_CIRCLE', 'DEF_30_CNT_SOCIAL_CIRCLE',
       'OBS_60_CNT_SOCIAL_CIRCLE', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'DAYS_LAST_PHONE_CHANGE', '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',
       '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',
       'OCCUPATION_TYPE'],
      dtype='object')
```

In [37]:
```
def convert_to_years(x):
    return abs(x//365)
```

In [38]:
```
data_v2['YEARS_BIRTH']= data_v2['DAYS_BIRTH'].apply(convert_to_years)
data_v2.drop(['DAYS_BIRTH'],inplace=True,axis=1)
```

In [39]:
```
data_v2['YEARS_EMPLOYED']= data_v2['DAYS_EMPLOYED'].apply(convert_to_years)
data_v2.drop(['DAYS_EMPLOYED'],inplace=True,axis=1)
data_v2['YEARS_REGISTRATION']= data_v2['DAYS_REGISTRATION'].apply(convert_to_years)
data_v2.drop(['DAYS_REGISTRATION'],inplace=True,axis=1)
data_v2['YEARS_ID_PUBLISH']= data_v2['DAYS_ID_PUBLISH'].apply(convert_to_years)
data_v2.drop(['DAYS_ID_PUBLISH'],inplace=True,axis=1)
data_v2['YEARS_LAST_PHONE_CHANGE']= data_v2['DAYS_LAST_PHONE_CHANGE'].apply(convert_tc
data_v2.drop(['DAYS_LAST_PHONE_CHANGE'],inplace=True,axis=1)
```

In [40]:
```
data_v2.drop(['FLAG_MOBIL', 'FLAG_EMP_PHONE', 'FLAG_WORK_PHONE', 'FLAG_CONT_MOBILE',
       'FLAG_PHONE', 'FLAG_EMAIL','REGION_RATING_CLIENT','REGION_RATING_CLIENT_W_CITY'
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_2', 'FLAG_DOCUMENT_3','FLAG_DOCUM
       'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_8', 'FLAG_DOCUMENT_9','FLAG_DOCUMENT_10', 'FL
       'FLAG_DOCUMENT_13', 'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_15','FLAG_DOCUMENT_16',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'FLAG_DOCUMENT_21','WEEKDAY_APPR_PROCES
```

In [41]:
```
data_v2.drop(['REGION_POPULATION_RELATIVE','LIVE_REGION_NOT_WORK_REGION','REG_CITY_NOT
       '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'], axis= 1, inplace= True
```

In [42]:
```
data_v2.drop(['REG_REGION_NOT_LIVE_REGION',
       'REG_REGION_NOT_WORK_REGION','AMT_REQ_CREDIT_BUREAU_HOUR', 'AMT_REQ_CREDIT_BURE
```

```
       'AMT_REQ_CREDIT_BUREAU_WEEK', 'AMT_REQ_CREDIT_BUREAU_MON',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'AMT_REQ_CREDIT_BUREAU_YEAR', 'YEARS_ID_PUBLISH',
```

In [43]:
```
data_v2.columns
```

Out[43]:
```
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', 'AMT_GOODS_PRICE', 'NAME_INCOME_TYPE',
       'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE',
       'CNT_FAM_MEMBERS', 'ORGANIZATION_TYPE', 'OCCUPATION_TYPE',
       'YEARS_BIRTH', 'YEARS_EMPLOYED', 'YEARS_REGISTRATION'],
      dtype='object')
```

In [44]:
```
data_v2.describe()
```

Out[44]:

	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANNU
count	307511.000000	307511.000000	307511.000000	3.075110e+05	3.075110e+05	307499.000
mean	278180.518577	0.080729	0.417052	1.687979e+05	5.990260e+05	27108.573
std	102790.175348	0.272419	0.722121	2.371231e+05	4.024908e+05	14493.737
min	100002.000000	0.000000	0.000000	2.565000e+04	4.500000e+04	1615.500
25%	189145.500000	0.000000	0.000000	1.125000e+05	2.700000e+05	16524.000
50%	278202.000000	0.000000	0.000000	1.471500e+05	5.135310e+05	24903.000
75%	367142.500000	0.000000	1.000000	2.025000e+05	8.086500e+05	34596.000
max	456255.000000	1.000000	19.000000	1.170000e+08	4.050000e+06	258025.500

In [45]:
```
data_v2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Data columns (total 21 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   FLAG_OWN_CAR          307511 non-null  object 
 5   FLAG_OWN_REALTY       307511 non-null  object 
 6   CNT_CHILDREN          307511 non-null  int64  
 7   AMT_INCOME_TOTAL      307511 non-null  float64
 8   AMT_CREDIT            307511 non-null  float64
 9   AMT_ANNUITY           307499 non-null  float64
 10  AMT_GOODS_PRICE       307233 non-null  float64
 11  NAME_INCOME_TYPE      307511 non-null  object 
 12  NAME_EDUCATION_TYPE   307511 non-null  object 
 13  NAME_FAMILY_STATUS    307511 non-null  object 
 14  NAME_HOUSING_TYPE     307511 non-null  object 
 15  CNT_FAM_MEMBERS       307509 non-null  float64
 16  ORGANIZATION_TYPE     307511 non-null  object 
 17  OCCUPATION_TYPE       211120 non-null  object 
 18  YEARS_BIRTH           307511 non-null  int64  
 19  YEARS_EMPLOYED        307511 non-null  int64  
 20  YEARS_REGISTRATION    307511 non-null  float64
dtypes: float64(6), int64(5), object(10)
memory usage: 49.3+ MB
```

In [46]:
```
data_v2['CODE_GENDER'].value_counts()
```

Out[46]:
```
F      202448
M      105059
XNA         4
Name: CODE_GENDER, dtype: int64
```

In [47]:
```
# So we can impute "F" where we have "XNA"
```

In [48]:
```
data_v2.loc[data_v2['CODE_GENDER']=='XNA','CODE_GENDER']='F'
data_v2['CODE_GENDER'].value_counts()
```

Out[48]:
```
F    202452
M    105059
Name: CODE_GENDER, dtype: int64
```

In [49]:
```
data_v2['ORGANIZATION_TYPE'].value_counts()
```

Out[49]:

```
Business Entity Type 3    67992
XNA                       55374
Self-employed             38412
Other                     16683
Medicine                  11193
Business Entity Type 2    10553
Government                10404
School                     8893
Trade: type 7              7831
Kindergarten               6880
Construction               6721
Business Entity Type 1     5984
Transport: type 4          5398
Trade: type 3              3492
Industry: type 9           3368
Industry: type 3           3278
Security                   3247
Housing                    2958
Industry: type 11          2704
Military                   2634
Bank                       2507
Agriculture                2454
Police                     2341
Transport: type 2          2204
Postal                     2157
Security Ministries        1974
Trade: type 2              1900
Restaurant                 1811
Services                   1575
University                 1327
Industry: type 7           1307
Transport: type 3          1187
Industry: type 1           1039
Hotel                       966
Electricity                 950
Industry: type 4            877
Trade: type 6               631
Industry: type 5            599
Insurance                   597
Telecom                     577
Emergency                   560
Industry: type 2            458
Advertising                 429
Realtor                     396
Culture                     379
Industry: type 12           369
Trade: type 1               348
Mobile                      317
Legal Services              305
Cleaning                    260
Transport: type 1           201
Industry: type 6            112
Industry: type 10           109
Religion                     85
Industry: type 13            67
Trade: type 4                64
Trade: type 5                49
Industry: type 8             24
Name: ORGANIZATION_TYPE, dtype: int64
```

In [50]:
```
data_v2 = data_v2.replace('XNA',np.NaN)
# data_v2=data_v2.drop(data_v2.loc[data_v2['ORGANIZATION_TYPE']=='XNA'].index)
```

In [51]:
```
data_v2.shape
```

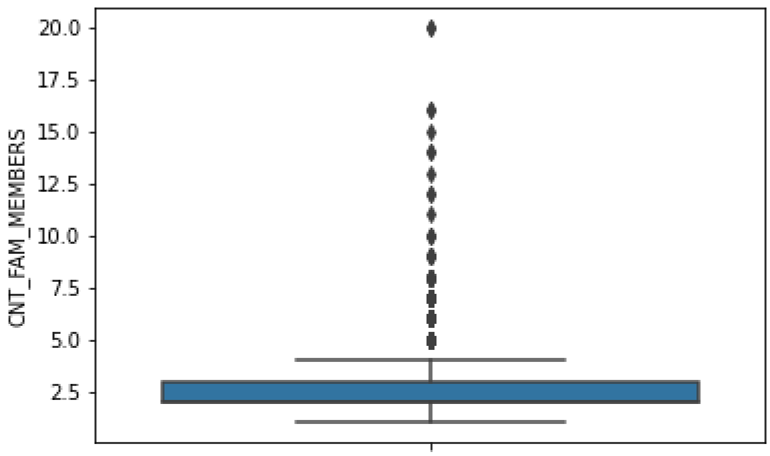
Out[51]:
```
(307511, 21)
```

In [52]:
```
data_v2.select_dtypes('object').columns
```

Out[52]:
```
Index(['NAME_CONTRACT_TYPE', 'CODE_GENDER', 'FLAG_OWN_CAR', 'FLAG_OWN_REALTY',
       'NAME_INCOME_TYPE', 'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS',
       'NAME_HOUSING_TYPE', 'ORGANIZATION_TYPE', 'OCCUPATION_TYPE'],
      dtype='object')
```

Univariate

In [53]:
```
sns.boxplot(y='CNT_FAM_MEMBERS',data=data_v2)
plt.show()
```



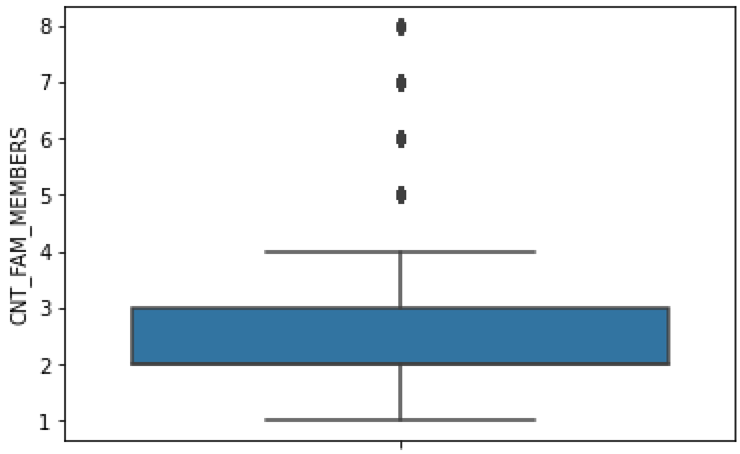
In [54]:
```
# Few outliers.
```

In [55]:
```
data_v2['CNT_FAM_MEMBERS'].value_counts()
```

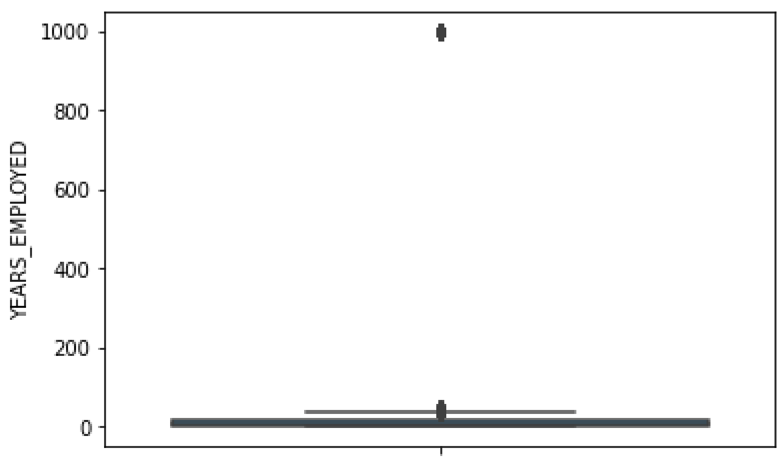
Out[55]:
```
2.0     158357
1.0      67847
3.0      52601
4.0      24697
5.0       3478
6.0        408
7.0         81
8.0         20
9.0          6
10.0         3
14.0         2
12.0         2
20.0         2
16.0         2
13.0         1
15.0         1
11.0         1
Name: CNT_FAM_MEMBERS, dtype: int64
```

In [56]:
```
data_v2 = data_v2[data_v2['CNT_FAM_MEMBERS'] <= 8]
```

In [57]:
```
sns.boxplot(y='CNT_FAM_MEMBERS',data=data_v2)
plt.show()
```



In [58]:
```
sns.boxplot(y= 'YEARS_EMPLOYED', data= data_v2)
plt.show()
```

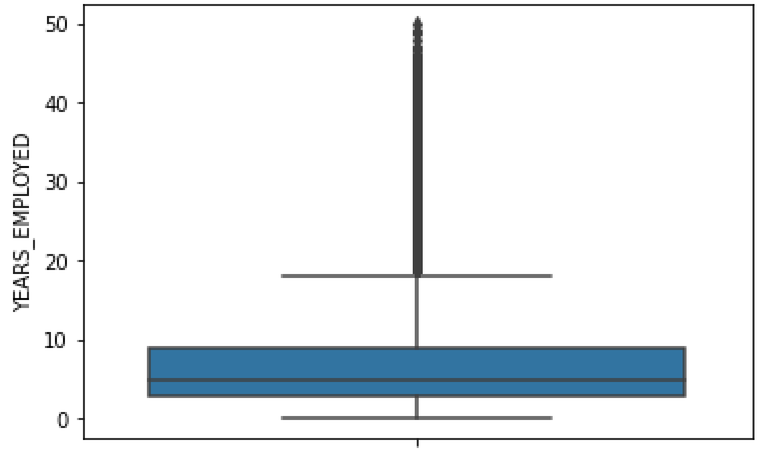


In [59]:
```
data_v2[data_v2.YEARS_EMPLOYED == 1000].NAME_INCOME_TYPE.value_counts()
```

Out[59]:
```
Pensioner     55351
Unemployed       22
Name: NAME_INCOME_TYPE, dtype: int64
```

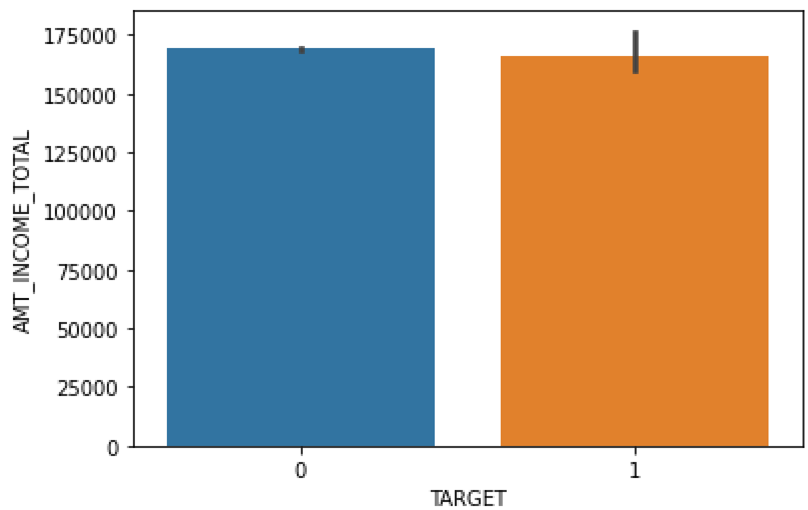
In [60]:
```
data_v2.loc[data_v2.YEARS_EMPLOYED == 1000, 'YEARS_EMPLOYED']= np.NAN
```

In [61]:
```
sns.boxplot(y= 'YEARS_EMPLOYED', data= data_v2)
plt.show()
```



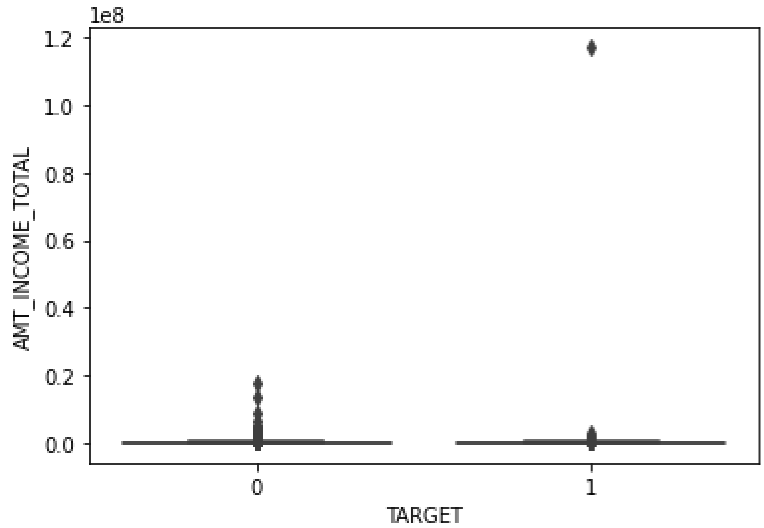
Bivariate

In [62]:
```
sns.barplot(x='TARGET', y='AMT_INCOME_TOTAL', data= data_v2)
plt.show()
```



In [63]:
```
sns.boxplot(x='TARGET', y='AMT_INCOME_TOTAL', data= data_v2)
```

Out[63]:
```
<AxesSubplot: xlabel='TARGET', ylabel='AMT_INCOME_TOTAL'>
```

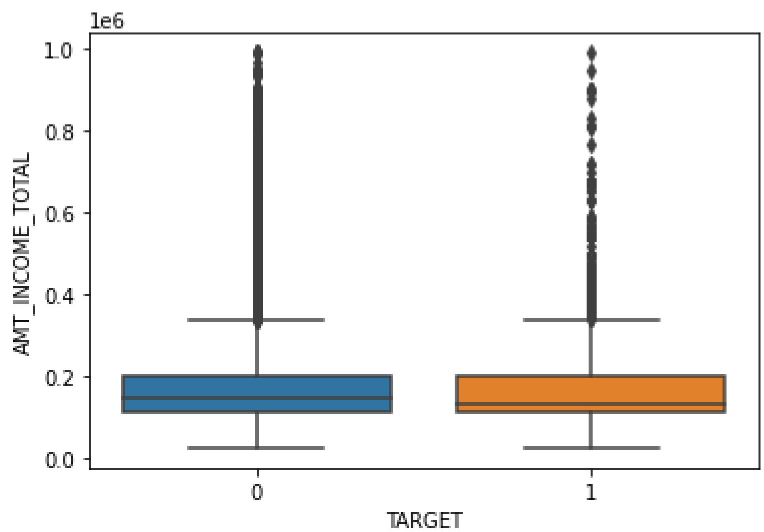


In [64]:
```
data_v2['AMT_INCOME_TOTAL'].max()
```
Out[64]:
```
117000000.0
```

In [65]:
```
data_v2 = data_v2[data_v2['AMT_INCOME_TOTAL'] < 999999]
```

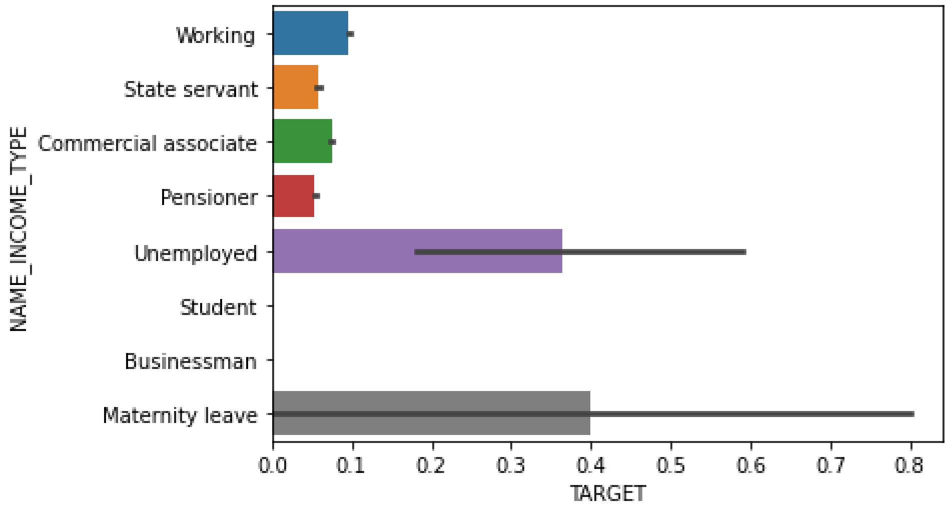
In [66]:
```
data_v2.shape
```
Out[66]:
```
(307239, 21)
```

In [67]:
```
sns.boxplot(x='TARGET', y='AMT_INCOME_TOTAL', data= data_v2)
```
Out[67]:
```
<AxesSubplot: xlabel='TARGET', ylabel='AMT_INCOME_TOTAL'>
```



In [68]:
```
# no success / no relationship found
```

In [69]:
```
sns.barplot(x= 'TARGET', y= 'NAME_INCOME_TYPE', data=data_v2)
plt.show()
```



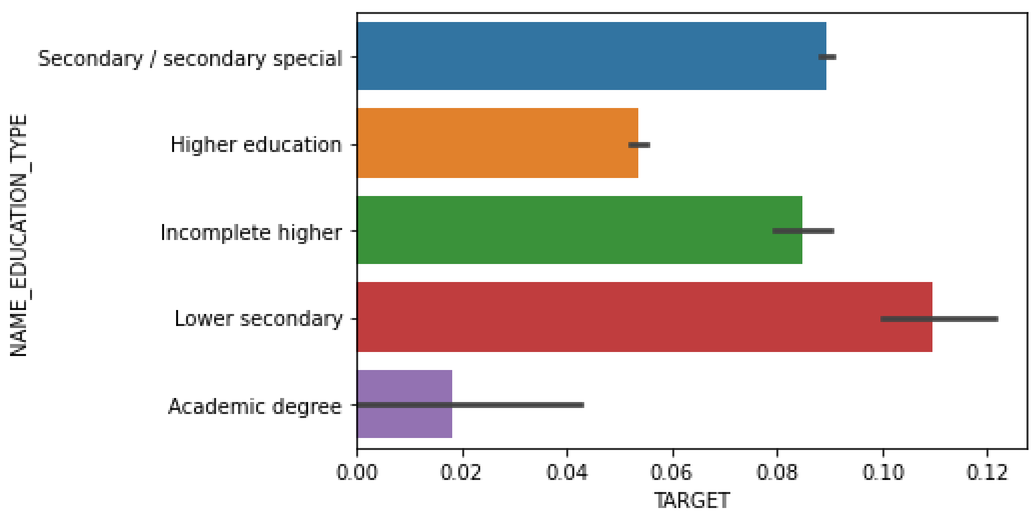
In [70]:
```
# Insight: Unemployed and Maternity Leave persons are defaulters.
```

In [71]:
```
pd.crosstab(data_v2['TARGET'], data_v2['NAME_EDUCATION_TYPE'])
```

Out[71]:

NAME_EDUCATION_TYPE	Academic degree	Higher education	Incomplete higher	Lower secondary	Secondary / secondary special
TARGET					
0	161	70679	9401	3396	198793
1	3	4003	871	417	19515

In [72]:
```
sns.barplot(x= 'TARGET', y= 'NAME_EDUCATION_TYPE', data=data_v2)
plt.show()
```



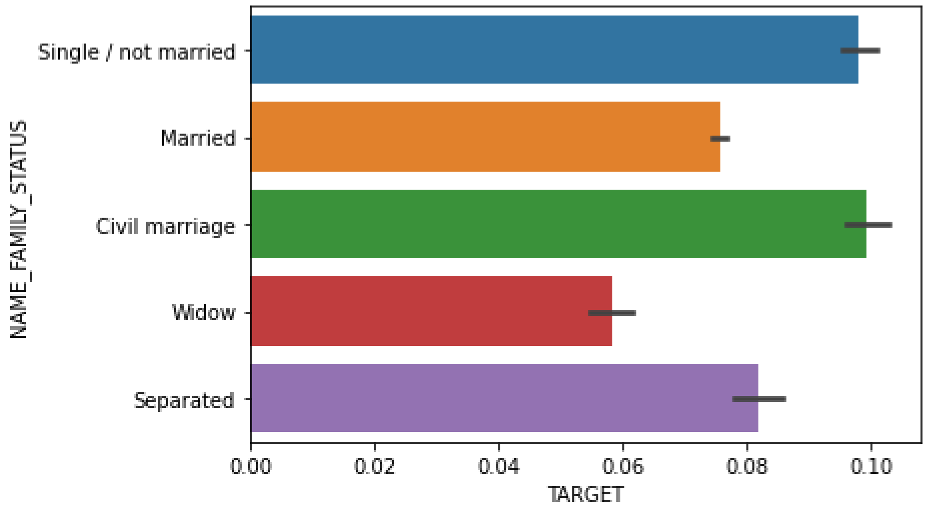
In [73]:
```
# Insight: Lower the education level, higher the chances of becoming defaulter
```

In [74]:
```
pd.crosstab(data_v2['TARGET'], data_v2['NAME_FAMILY_STATUS'])
```

Out[74]:

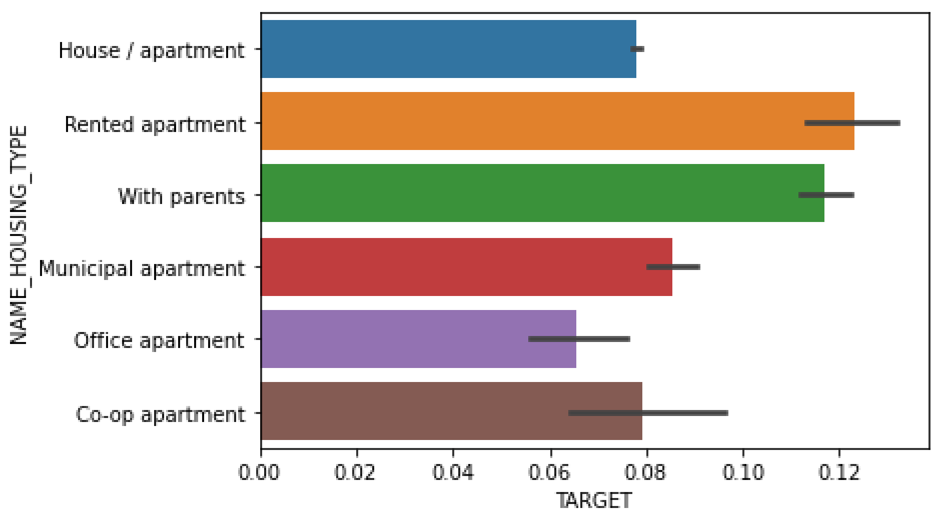
NAME_FAMILY_STATUS	Civil marriage	Married	Separated	Single / not married	Widow
TARGET					
0	26799	181400	18133	40953	15145
1	2957	14839	1620	4456	937

In [75]:
```
sns.barplot(x= 'TARGET', y= 'NAME_FAMILY_STATUS', data=data_v2)
plt.show()
```



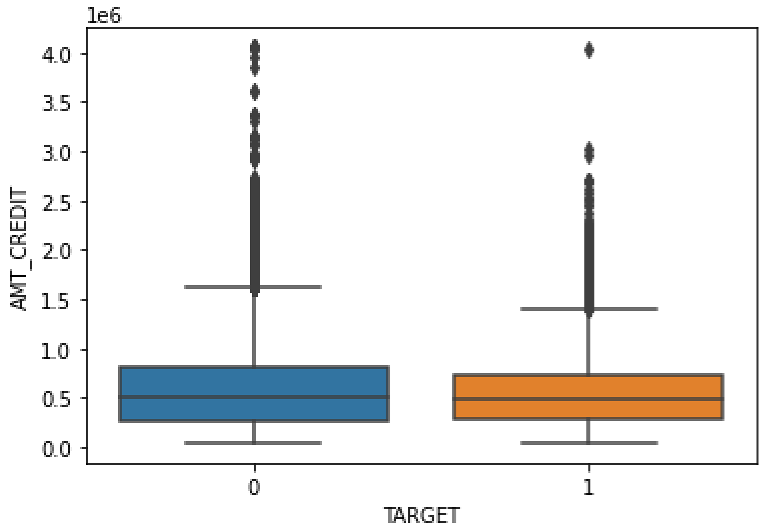
In [76]:
```
# Insight: if the marital status is single,
# there are highest chances of him/her to be defaulter
```

In [77]:
```
sns.barplot(x= 'TARGET', y= 'NAME_HOUSING_TYPE', data=data_v2)
plt.show()
```



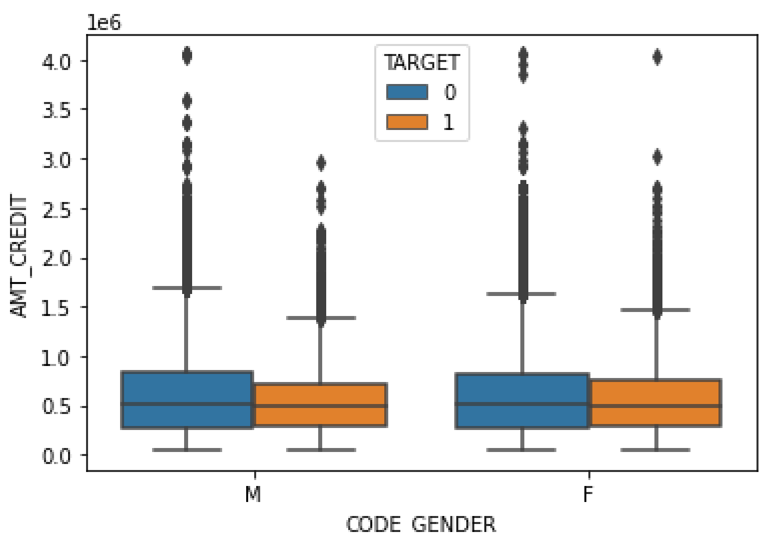
In [78]:
```
# Insight: Rented apppartments and With Parents category have highest probability of b
```

In [79]:
```
sns.boxplot(x= 'TARGET', y= 'AMT_CREDIT', data=data_v2)
plt.show()
```



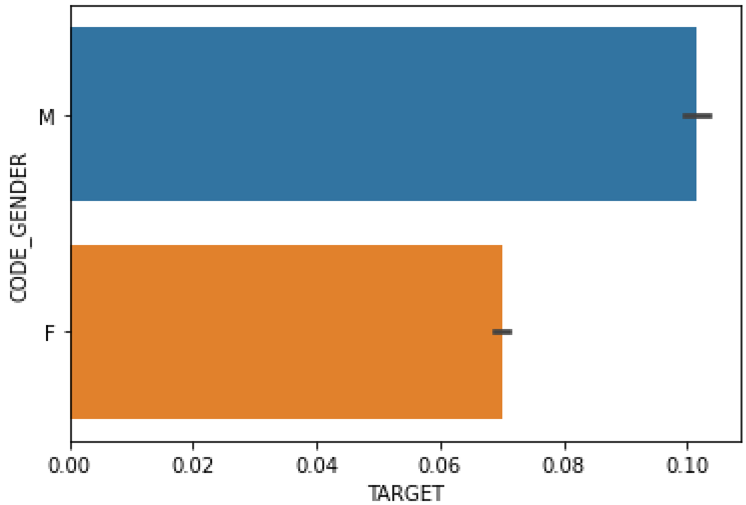
In [80]:
```
# Insight: There are outliers but we can say that higher the amount credited,
# there is no intuition that default probability is higher.
```

In [81]:
```
sns.boxplot(x='CODE_GENDER', y='AMT_CREDIT', data= data_v2, hue= 'TARGET')
plt.show()
```



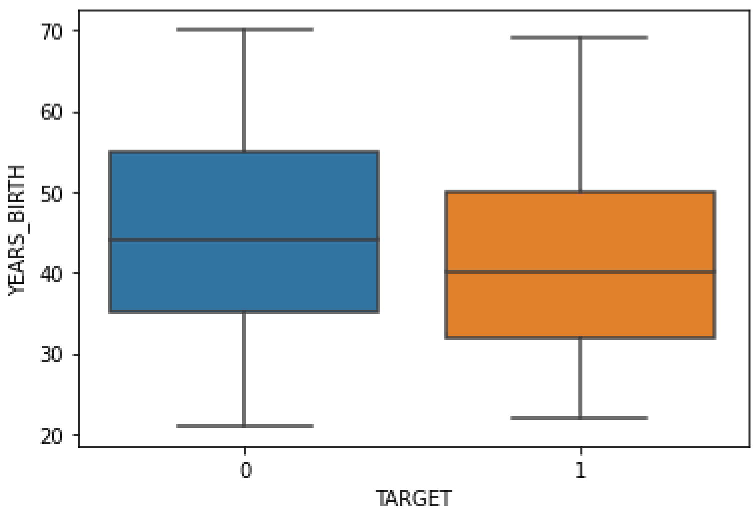
In [82]:
```
# Insight:
# Males have more chances of higher credit
# Both genders have similar probabilites of defaulting
```

In [83]:
```
sns.barplot(x='TARGET', y='CODE_GENDER', data=data_v2)
plt.show()
```



In [84]:
```
# Gender plays a vital role
# Males are more likely to default.
```

In [85]:
```
sns.boxplot(x='TARGET', y='YEARS_BIRTH', data=data_v2)
plt.show()
```



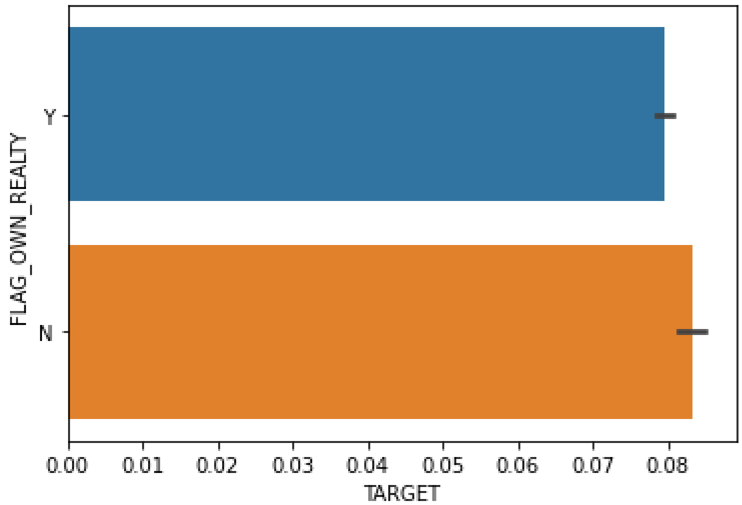
In [86]:
```
# Insight: Nothing much from age
# 30-40 is the age where maximum chances of being defaulter
# 20-30 is the age where minimum chances of being defaulter
```

In [87]:
```
pd.crosstab(data_v2['FLAG_OWN_REALTY'],data_v2['TARGET'])
```

Out[87]:

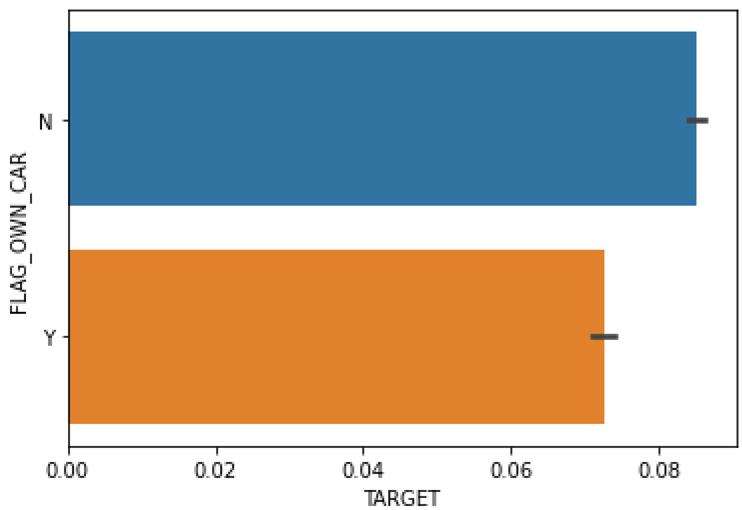
TARGET	0	1
FLAG_OWN_REALTY		
N	86260	7836
Y	196170	16973

In [88]:
```
sns.barplot(x= 'TARGET', y= 'FLAG_OWN_REALTY', data=data_v2)
plt.show()
```



In [89]:
```
# Insight: This is strange, owning a property has nothing to do with defaulting
# ideally if a person owns a property, then chances of defaulting should be significan
```

In [90]:
```
sns.barplot(x= 'TARGET', y= 'FLAG_OWN_CAR', data=data_v2)
plt.show()
```



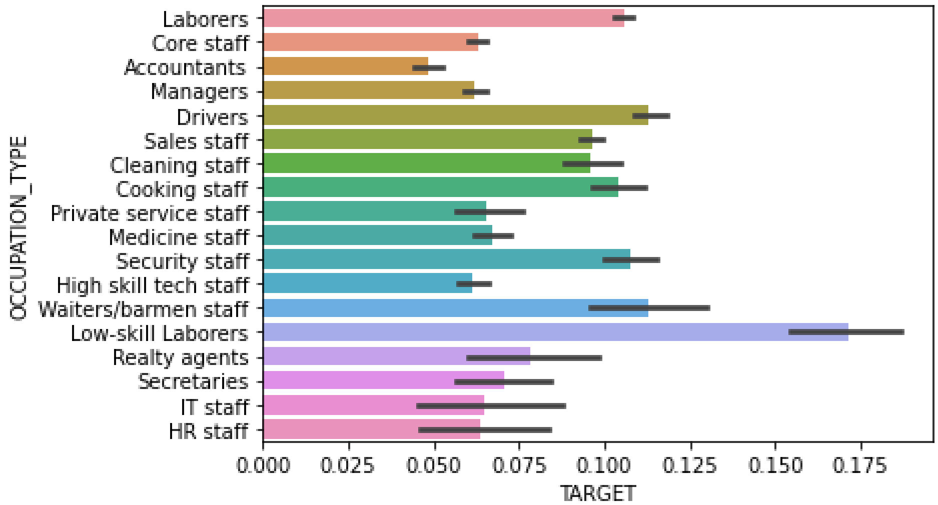
In [91]:
```
# Insight: those who don't own a car are more likely to default.
```

In [92]:
```
pd.crosstab(data_v2['TARGET'],data_v2['OCCUPATION_TYPE'],normalize='columns')
```

Out[92]:

OCCUPATION_TYPE	Accountants	Cleaning staff	Cooking staff	Core staff	Drivers	HR staff	High skill tech staff	IT staff
TARGET								
0	0.951623	0.90385	0.89556	0.936985	0.88681	0.936057	0.938456	0.935238
1	0.048377	0.09615	0.10444	0.063015	0.11319	0.063943	0.061544	0.064762

In [93]:
```
sns.barplot(x= 'TARGET', y= 'OCCUPATION_TYPE', data=data_v2)
plt.show()
```



In [94]:
```
# Insight: Low skill laborers, Security Staff, Waiters/Barmen staff, Drivers, Cooking
```

In [95]:
```
bins = [0,25000,50000,75000,100000,125000,150000,175000,200000,225000,250000,275000,30
slot = ['0-25000', '25000-50000','50000-75000','75000,100000','100000-125000', '125000
       '200000-225000','225000-250000','250000-275000','275000-300000','300000-325000'
       '375000-400000','400000-425000','425000-450000','450000-475000','475000-500000'

data_v2['AMT_INCOME_RANGE']=pd.cut(data_v2['AMT_INCOME_TOTAL'],bins,labels=slot)
```

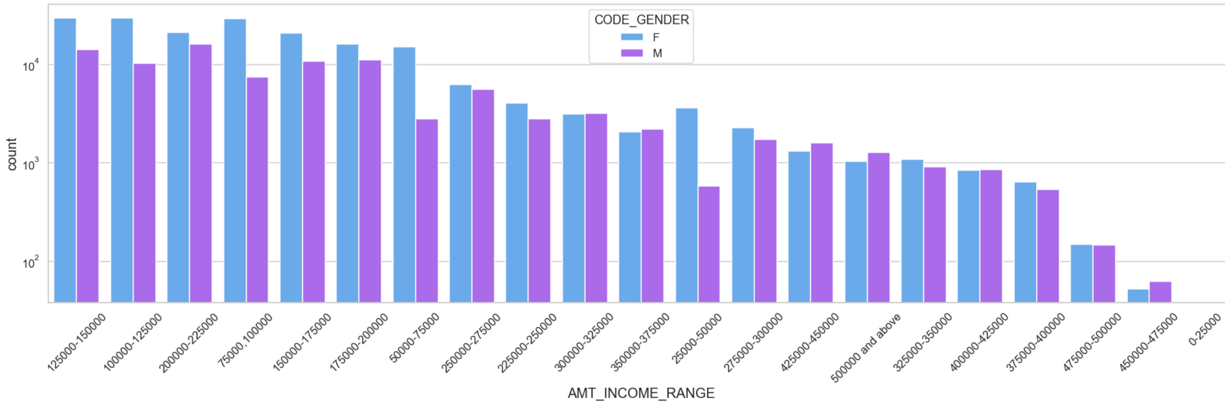
In [96]:
```
bins = [0,150000,200000,250000,300000,350000,400000,450000,500000,550000,600000,650000
slots = ['0-150000', '150000-200000','200000-250000', '250000-300000', '300000-350000'
        '450000-500000','500000-550000','550000-600000','600000-650000','650000-700000
        '800000-850000','850000-900000','900000 and above']

data_v2['AMT_CREDIT_RANGE']=pd.cut(data_v2['AMT_CREDIT'],bins=bins,labels=slots)
```

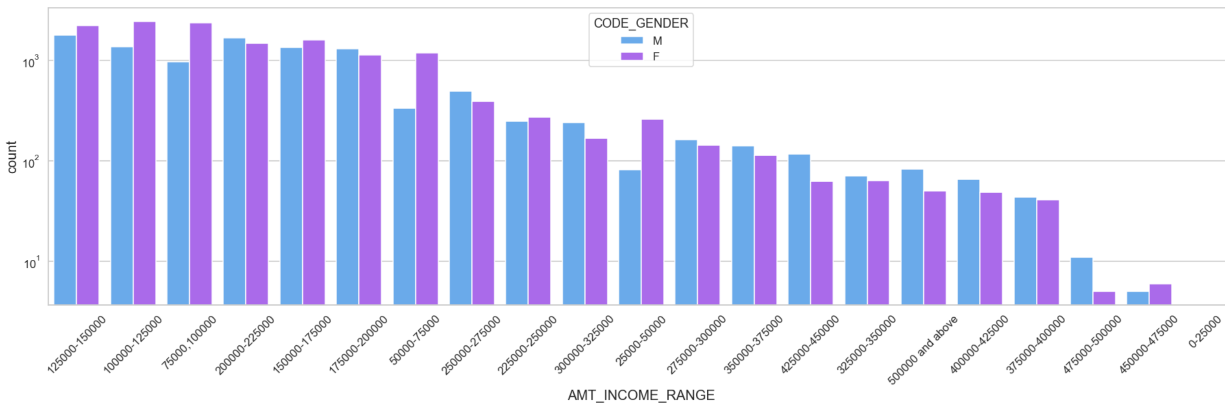
In [97]:
```
target0_df=data_v2.loc[data_v2["TARGET"]==0]
target1_df=data_v2.loc[data_v2["TARGET"]==1]
```

In [98]:
```
def uniplot(df,col,hue =None):

    sns.set_style('whitegrid')
    sns.set_context('talk')
    plt.rcParams["axes.labelsize"] = 20
    plt.rcParams['axes.titlesize'] = 22
    plt.rcParams['axes.titlepad'] = 30


    temp = pd.Series(data = hue)
    fig, ax = plt.subplots()
    width = len(df[col].unique()) + 7 + 4*len(temp.unique())
    fig.set_size_inches(width , 8)
    plt.xticks(rotation=45)
    plt.yscale('log')
    ax = sns.countplot(data = df, x= col, order=df[col].value_counts().index,hue = hue

    plt.show()
```

In [99]:
```
# Non Defaulters
uniplot(target0_df,col='AMT_INCOME_RANGE',hue='CODE_GENDER')
```

```
# Defaulters
uniplot(target1_df,col='AMT_INCOME_RANGE',hue='CODE_GENDER')
```



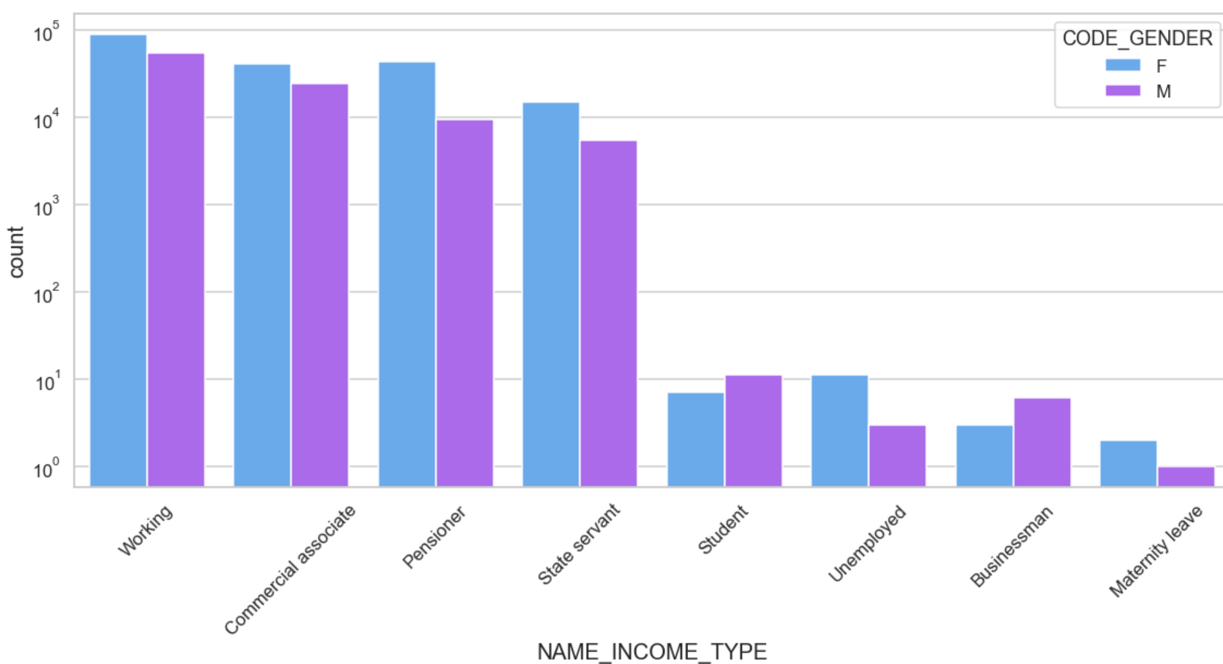


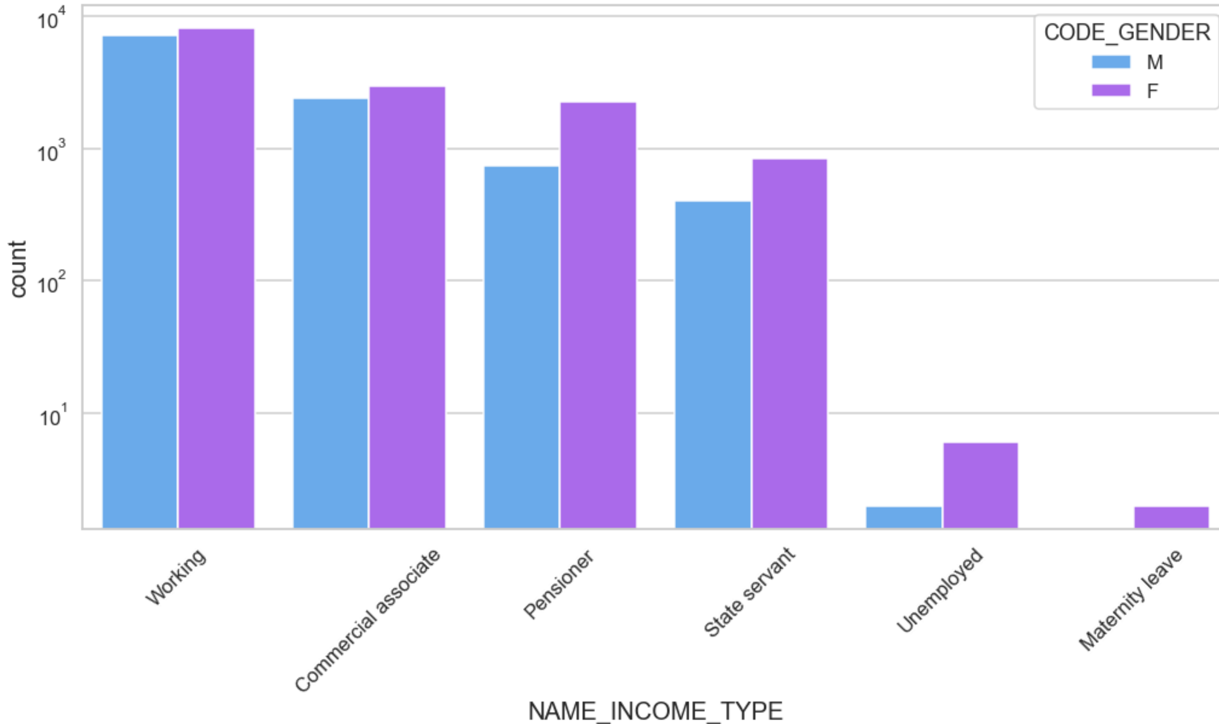
In [100…

```
# Insight: Female counts are higher than male.
# Income range from 100000 to 200000 is having more number of credits.
# This graph show that females are more than male in having credits for that range.
# Very less count for income range 400000 and above.
```

In [101…

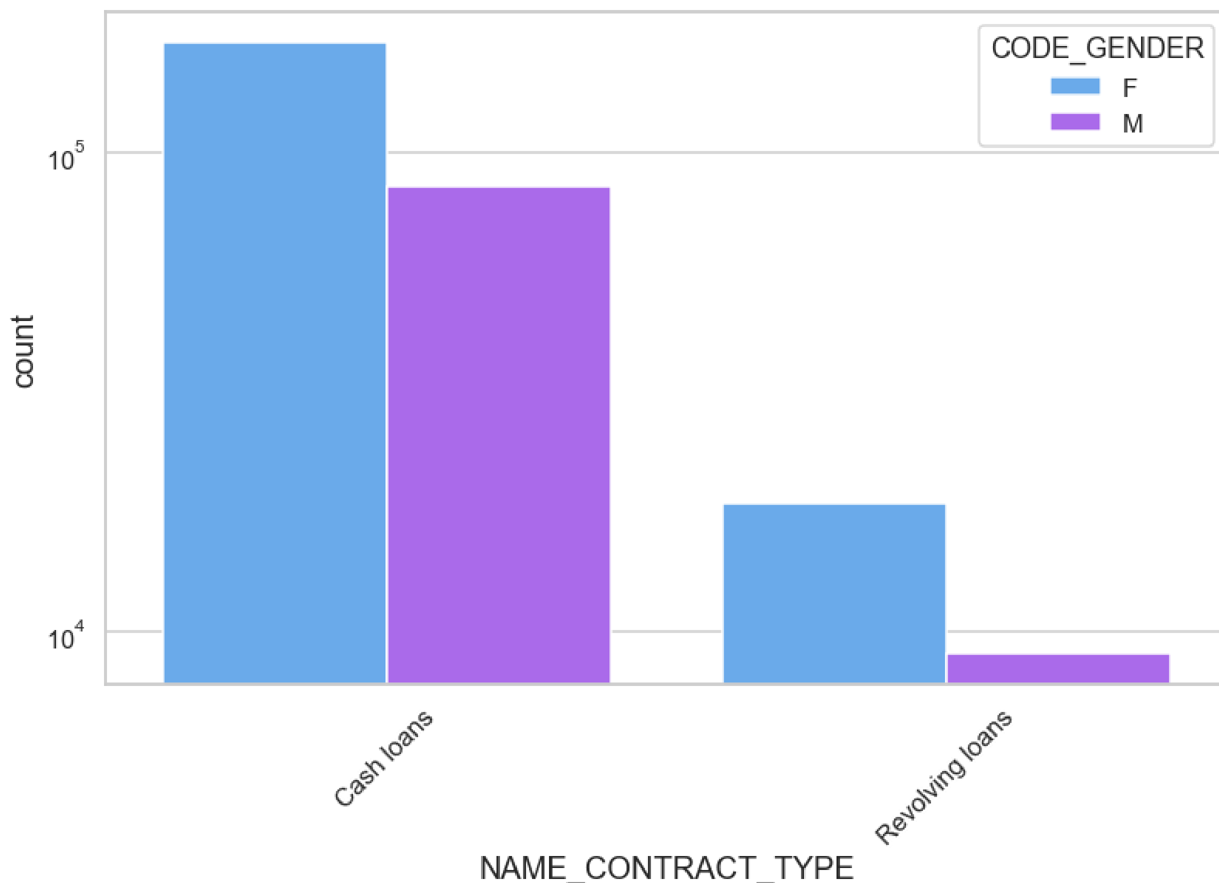
```
# Non Defaulters
uniplot(target0_df,col='NAME_INCOME_TYPE',hue='CODE_GENDER')
# Defaulters
uniplot(target1_df,col='NAME_INCOME_TYPE',hue='CODE_GENDER')
```

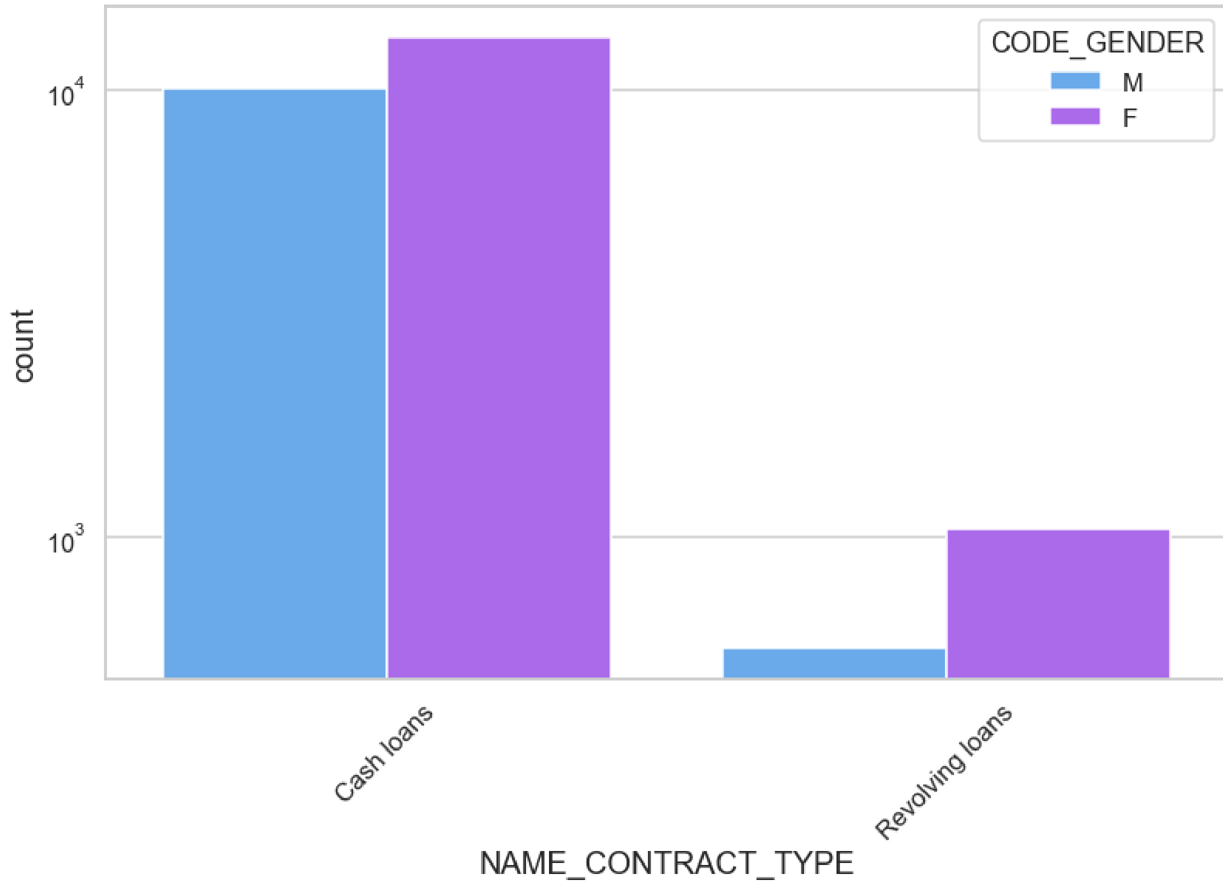




In [102…

```
# Non Defaulters
uniplot(target0_df,col='NAME_CONTRACT_TYPE',hue='CODE_GENDER')
# Defaulters
uniplot(target1_df,col='NAME_CONTRACT_TYPE',hue='CODE_GENDER')
```





In [103…

```
# Insight: ‘cash loans’ is having higher number of credits than ‘Revolving loans’ cont
# For this also Female is leading for applying credits.
```

In [104…

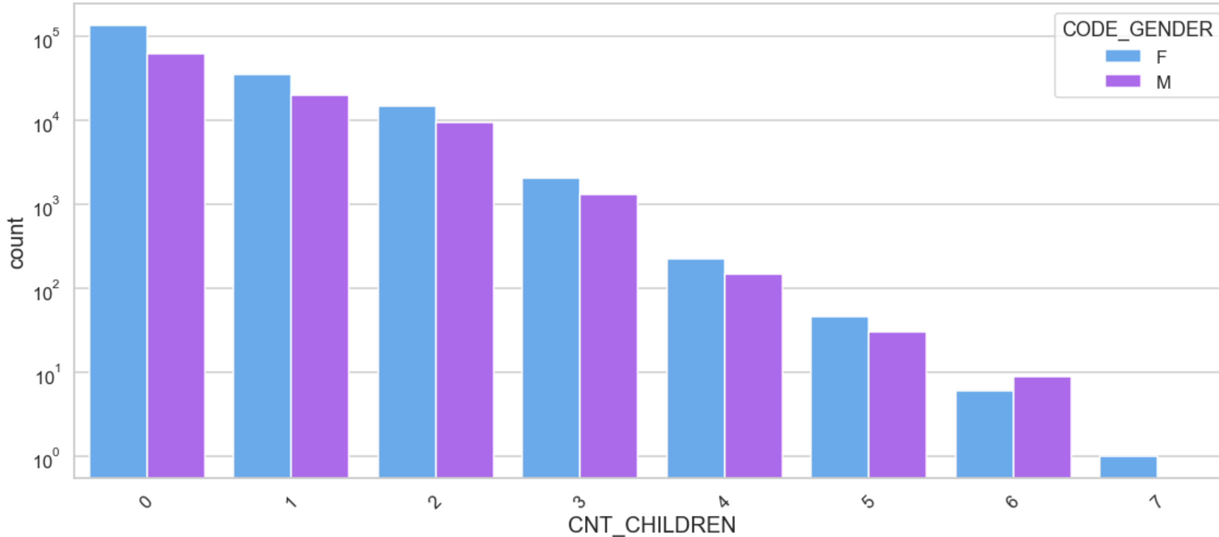
```
data_v2.CNT_CHILDREN.describe()
```

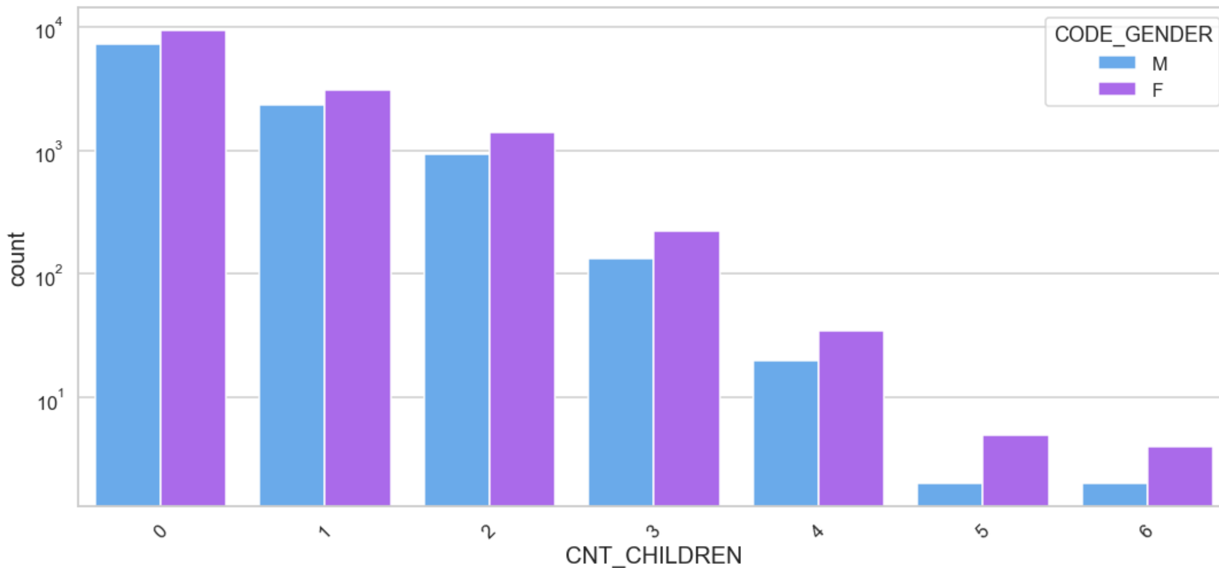
Out[104]:

```
count    307239.000000
mean          0.416318
std           0.716815
min           0.000000
25%           0.000000
50%           0.000000
75%           1.000000
max           7.000000
Name: CNT_CHILDREN, dtype: float64
```

In [105…

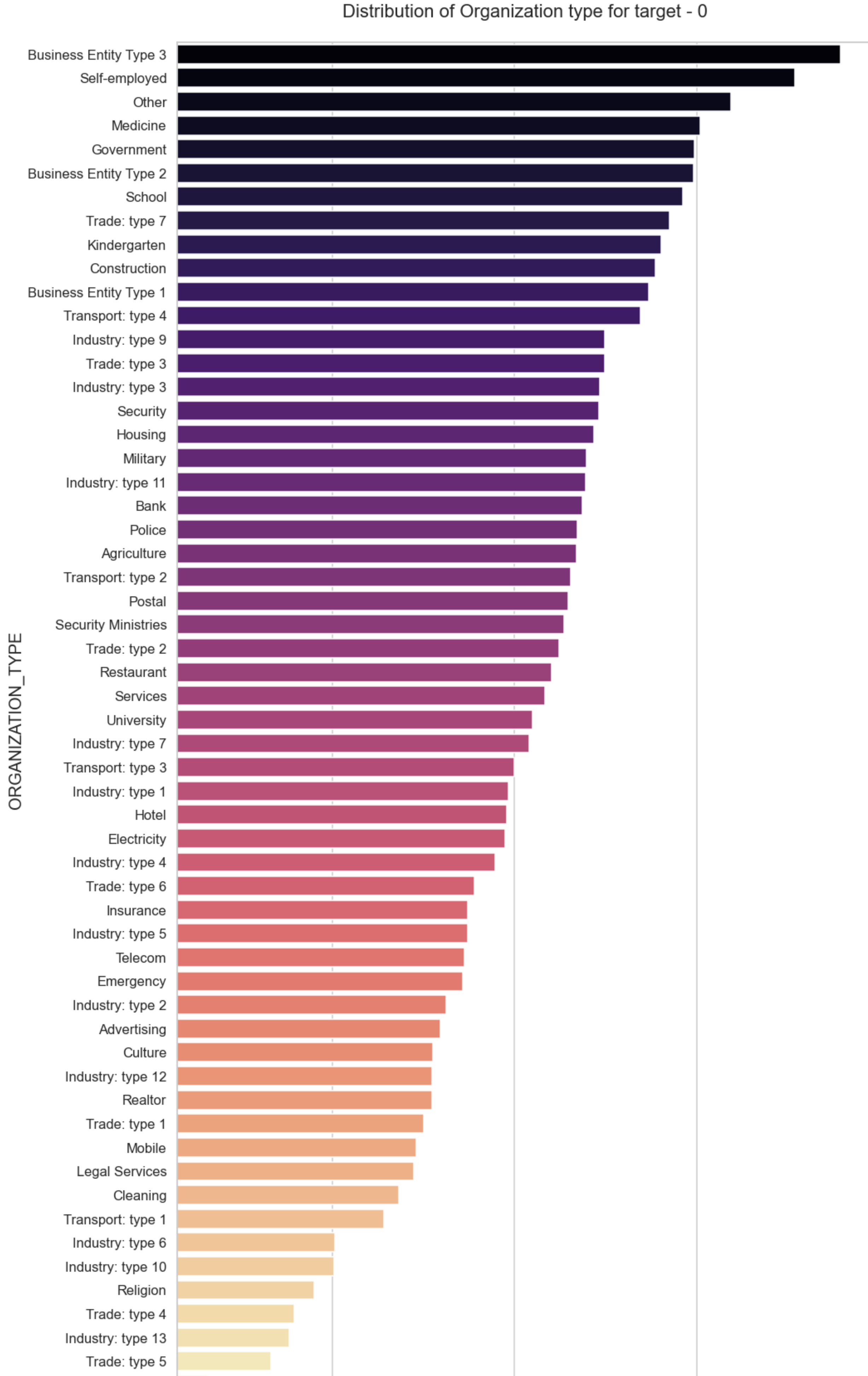
```
# Non Defaulters
uniplot(target0_df,col='CNT_CHILDREN',hue='CODE_GENDER')
# Defaulters
uniplot(target1_df,col='CNT_CHILDREN',hue='CODE_GENDER')
```





In [106…

```
sns.set_style('whitegrid')
sns.set_context('talk')
plt.figure(figsize=(15,30))
plt.rcParams["axes.labelsize"] = 20
plt.rcParams['axes.titlesize'] = 22
plt.rcParams['axes.titlepad'] = 30

plt.title("Distribution of Organization type for target - 0")

plt.xticks(rotation=90)
plt.xscale('log')

sns.countplot(data=target0_df,y='ORGANIZATION_TYPE',order=target0_df['ORGANIZATION_TYP

plt.show()
```


Distribution of Organization type for target - 0
ORGANIZATION_TYPE
Business Entity Type 3
Self-employed
Other
Medicine
Government
Business Entity Type 2
School
Trade: type 7
Kindergarten
Construction
Business Entity Type 1
Transport: type 4
Industry: type 9
Trade: type 3
Industry: type 3
Security
Housing
Military
Industry: type 11
Bank
Police
Agriculture
Transport: type 2
Postal
Security Ministries
Trade: type 2
Restaurant
Services
University
Industry: type 7
Transport: type 3
Industry: type 1
Hotel
Electricity
Industry: type 4
Trade: type 6
Insurance
Industry: type 5
Telecom
Emergency
Industry: type 2
Advertising
Culture
Industry: type 12
Realtor
Trade: type 1
Mobile
Legal Services
Cleaning
Transport: type 1
Industry: type 6
Industry: type 10
Religion
Trade: type 4
Industry: type 13
Trade: type 5

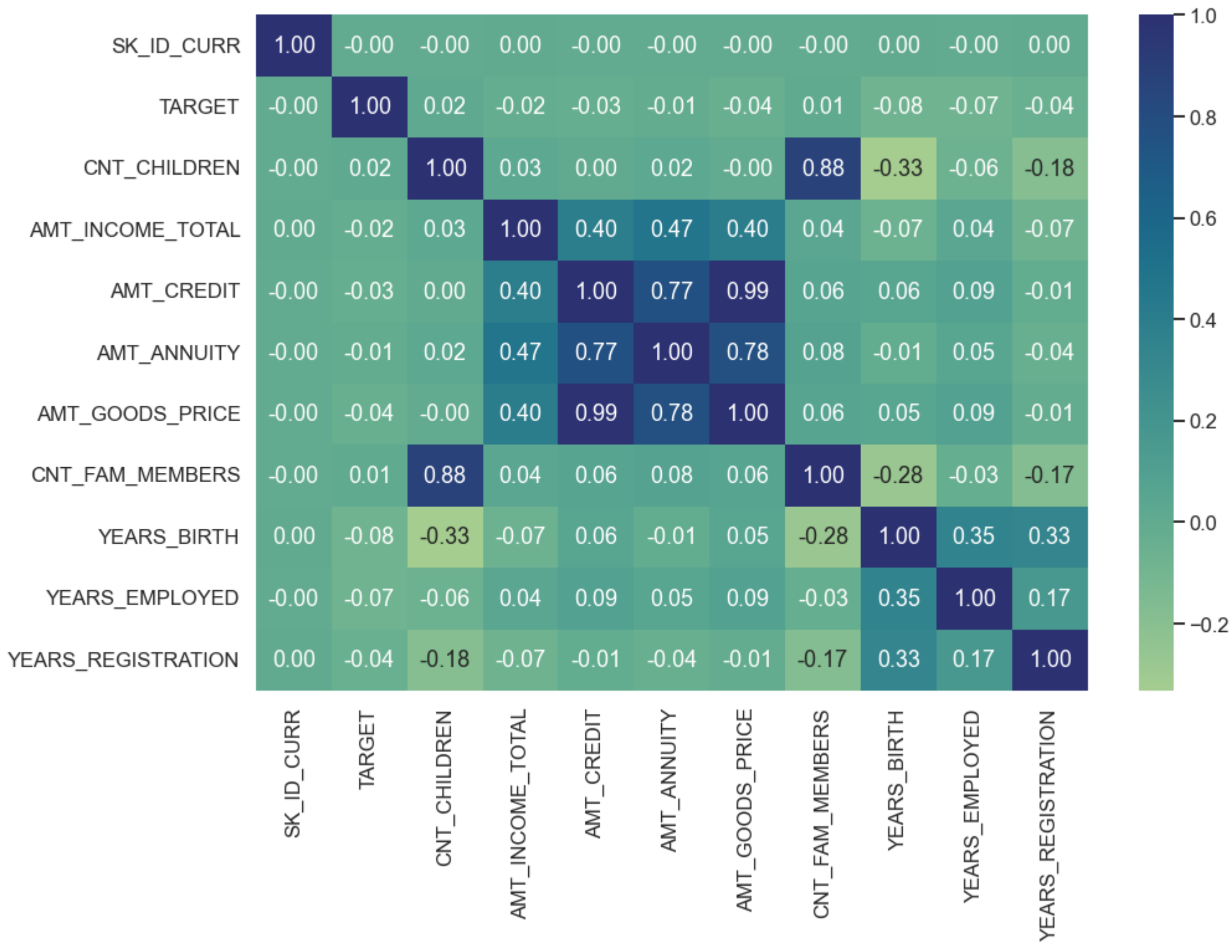
In [107…
```
# Insight: Clients which have applied for credits are from
# most of the organization type ‘Business entity Type 3’ ,
# ‘Self employed’, ‘Other’ , ‘Medicine’ and ‘Government’.
# Less clients are from Industry type 8,type 6, type 10,
# religion and trade type 5, type 4.
```

Multivariate

In [108…
```
numberic_cols= data_v2.select_dtypes(exclude=['object']).columns
```

In [109…
```
plt.subplots(figsize=(15,10))
sns.heatmap(data_v2[numberic_cols].corr(),annot=True, fmt = ".2f", cmap = "crest")
```

Out[109]:
```
<AxesSubplot: >
```



In [110…
```
# AMT_CREDIT and AMT_GOODS_PRICE have very high correlation.
```

In [111…
```
def top_10_corr(df):
    corr1 = data_v2.select_dtypes(exclude=['object']).corr()
    corr_df1 = corr1.where(np.triu(np.ones(corr1.shape), k=1).astype(np.bool))
    corr_df1 = corr_df1.unstack().reset_index().dropna(subset = [0])
    corr_df1.columns = ['VAR1', 'VAR2', 'Correlation_Value']
```

```
    corr_df1['Corr_abs'] = abs(corr_df1['Correlation_Value'])
    corr_df1.sort_values(by = "Corr_abs", ascending =False, inplace = True)
    return corr_df1.head(10)
```

In [112…

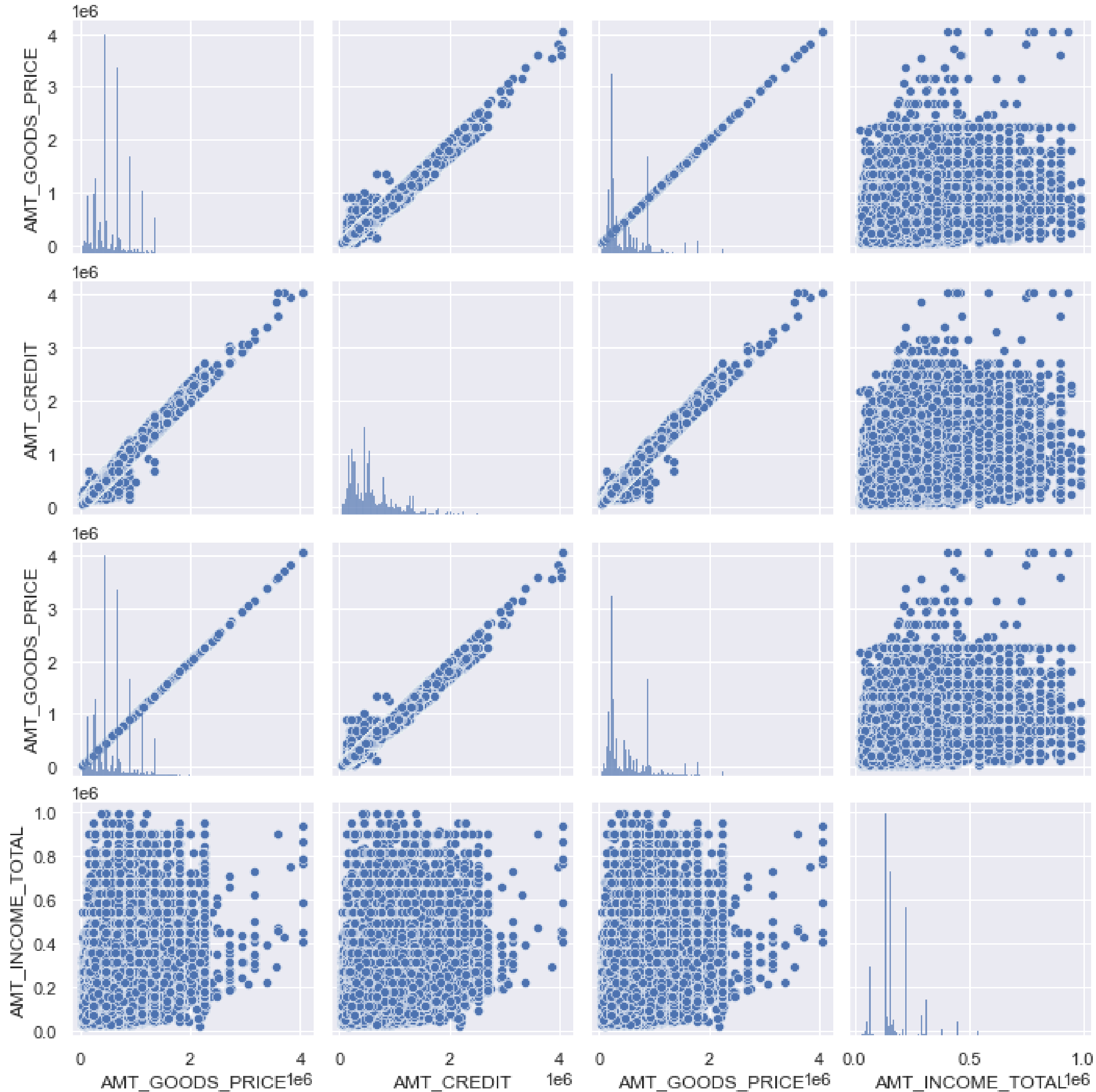
```
top_10_corr(data_v2)
```

Out[112]:

	VAR1	VAR2	Correlation_Value	Corr_abs
70	AMT_GOODS_PRICE	AMT_CREDIT	0.986945	0.986945
79	CNT_FAM_MEMBERS	CNT_CHILDREN	0.878022	0.878022
71	AMT_GOODS_PRICE	AMT_ANNUITY	0.775562	0.775562
59	AMT_ANNUITY	AMT_CREDIT	0.770861	0.770861
58	AMT_ANNUITY	AMT_INCOME_TOTAL	0.473435	0.473435
69	AMT_GOODS_PRICE	AMT_INCOME_TOTAL	0.403872	0.403872
47	AMT_CREDIT	AMT_INCOME_TOTAL	0.397650	0.397650
107	YEARS_EMPLOYED	YEARS_BIRTH	0.351608	0.351608
90	YEARS_BIRTH	CNT_CHILDREN	-0.333090	0.333090
118	YEARS_REGISTRATION	YEARS_BIRTH	0.331782	0.331782

In [113…

```
sns.set(rc={'figure.figsize':(10, 8)})
sns.pairplot(data_v2, vars = ['AMT_GOODS_PRICE','AMT_CREDIT','AMT_GOODS_PRICE', 'AMT_I
plt.show()
```



In [114…
```
# Insight: The features Income, credit amount and good price have high correlation wit
#  The larger the applicant's income, the larger the credit amount, similar to the val
```

Merging two datasets

In [115…
```
previous_data = pd.read_csv("./data/previous_application.csv")
```

In [116…
```
previous_data.head()
```

Out[116]:

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDI
0	2030495	271877	Consumer loans	1730.430	17145.0	17145
1	2802425	108129	Cash loans	25188.615	607500.0	679671
2	2523466	122040	Cash loans	15060.735	112500.0	136444
3	2819243	176158	Cash loans	47041.335	450000.0	470790
4	1784265	202054	Cash loans	31924.395	337500.0	404055

5 rows × 37 columns

In [117…

```
previous_data.columns
```

Out[117]:

```
Index(['SK_ID_PREV', 'SK_ID_CURR', 'NAME_CONTRACT_TYPE', 'AMT_ANNUITY',
       'AMT_APPLICATION', 'AMT_CREDIT', 'AMT_DOWN_PAYMENT', 'AMT_GOODS_PRICE',
       'WEEKDAY_APPR_PROCESS_START', 'HOUR_APPR_PROCESS_START',
       'FLAG_LAST_APPL_PER_CONTRACT', 'NFLAG_LAST_APPL_IN_DAY',
       'RATE_DOWN_PAYMENT', 'RATE_INTEREST_PRIMARY',
       'RATE_INTEREST_PRIVILEGED', '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 [118…

```
previous_data.shape
```

Out[118]:

```
(1670214, 37)
```

In [119…

```
previous_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1670214 entries, 0 to 1670213
Data columns (total 37 columns):
 #   Column                       Non-Null Count    Dtype  
---  ------                       --------------    -----  
 0   SK_ID_PREV                   1670214 non-null  int64  
 1   SK_ID_CURR                   1670214 non-null  int64  
 2   NAME_CONTRACT_TYPE           1670214 non-null  object 
 3   AMT_ANNUITY                  1297979 non-null  float64
 4   AMT_APPLICATION              1670214 non-null  float64
 5   AMT_CREDIT                   1670213 non-null  float64
 6   AMT_DOWN_PAYMENT             774370 non-null   float64
 7   AMT_GOODS_PRICE              1284699 non-null  float64
 8   WEEKDAY_APPR_PROCESS_START   1670214 non-null  object 
 9   HOUR_APPR_PROCESS_START      1670214 non-null  int64  
 10  FLAG_LAST_APPL_PER_CONTRACT  1670214 non-null  object 
 11  NFLAG_LAST_APPL_IN_DAY       1670214 non-null  int64  
 12  RATE_DOWN_PAYMENT            774370 non-null   float64
 13  RATE_INTEREST_PRIMARY        5951 non-null     float64
 14  RATE_INTEREST_PRIVILEGED     5951 non-null     float64
 15  NAME_CASH_LOAN_PURPOSE       1670214 non-null  object 
 16  NAME_CONTRACT_STATUS         1670214 non-null  object 
 17  DAYS_DECISION                1670214 non-null  int64  
 18  NAME_PAYMENT_TYPE            1670214 non-null  object 
 19  CODE_REJECT_REASON           1670214 non-null  object 
 20  NAME_TYPE_SUITE              849809 non-null   object 
 21  NAME_CLIENT_TYPE             1670214 non-null  object 
 22  NAME_GOODS_CATEGORY          1670214 non-null  object 
 23  NAME_PORTFOLIO               1670214 non-null  object 
 24  NAME_PRODUCT_TYPE            1670214 non-null  object 
 25  CHANNEL_TYPE                 1670214 non-null  object 
 26  SELLERPLACE_AREA             1670214 non-null  int64  
 27  NAME_SELLER_INDUSTRY         1670214 non-null  object 
 28  CNT_PAYMENT                  1297984 non-null  float64
 29  NAME_YIELD_GROUP             1670214 non-null  object 
 30  PRODUCT_COMBINATION          1669868 non-null  object 
 31  DAYS_FIRST_DRAWING           997149 non-null   float64
 32  DAYS_FIRST_DUE               997149 non-null   float64
 33  DAYS_LAST_DUE_1ST_VERSION    997149 non-null   float64
 34  DAYS_LAST_DUE                997149 non-null   float64
 35  DAYS_TERMINATION             997149 non-null   float64
 36  NFLAG_INSURED_ON_APPROVAL    997149 non-null   float64
dtypes: float64(15), int64(6), object(16)
memory usage: 471.5+ MB
```

In [120…

```
previous_data.dtypes.value_counts()
```

Out[120]:

```
object     16
float64    15
int64       6
dtype: int64
```

In [121…

```
previous_data.describe()
```

Out[121]:

	SK_ID_PREV	SK_ID_CURR	AMT_ANNUITY	AMT_APPLICATION	AMT_CREDIT	AMT_DOWN_PA
count	1.670214e+06	1.670214e+06	1.297979e+06	1.670214e+06	1.670213e+06	7.743
mean	1.923089e+06	2.783572e+05	1.595512e+04	1.752339e+05	1.961140e+05	6.697
std	5.325980e+05	1.028148e+05	1.478214e+04	2.927798e+05	3.185746e+05	2.092
min	1.000001e+06	1.000010e+05	0.000000e+00	0.000000e+00	0.000000e+00	-9.000
25%	1.461857e+06	1.893290e+05	6.321780e+03	1.872000e+04	2.416050e+04	0.000
50%	1.923110e+06	2.787145e+05	1.125000e+04	7.104600e+04	8.054100e+04	1.638
75%	2.384280e+06	3.675140e+05	2.065842e+04	1.803600e+05	2.164185e+05	7.740
max	2.845382e+06	4.562550e+05	4.180581e+05	6.905160e+06	6.905160e+06	3.060

8 rows × 21 columns

In [122…

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

Out[122]:

	NAME_CONTRACT_TYPE	WEEKDAY_APPR_PROCESS_START	FLAG_LAST_APPL_PER_CONTRACT	N
count	1670214	1670214	1670214	
unique	4	7	2	
top	Cash loans	TUESDAY	Y	
freq	747553	255118	1661739	

In [123…

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

In [124…

```
null_df.head(40)
```

Out[124]:
```
RATE_INTEREST_PRIVILEGED      99.643698
RATE_INTEREST_PRIMARY         99.643698
AMT_DOWN_PAYMENT              53.636480
RATE_DOWN_PAYMENT             53.636480
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
NAME_YIELD_GROUP               0.000000
NAME_PORTFOLIO                 0.000000
NAME_SELLER_INDUSTRY           0.000000
SELLERPLACE_AREA               0.000000
CHANNEL_TYPE                   0.000000
NAME_PRODUCT_TYPE              0.000000
SK_ID_PREV                     0.000000
NAME_GOODS_CATEGORY            0.000000
NAME_CLIENT_TYPE               0.000000
CODE_REJECT_REASON             0.000000
SK_ID_CURR                     0.000000
DAYS_DECISION                  0.000000
NAME_CONTRACT_STATUS           0.000000
NAME_CASH_LOAN_PURPOSE         0.000000
NFLAG_LAST_APPL_IN_DAY         0.000000
FLAG_LAST_APPL_PER_CONTRACT    0.000000
HOUR_APPR_PROCESS_START        0.000000
WEEKDAY_APPR_PROCESS_START     0.000000
AMT_APPLICATION                0.000000
NAME_CONTRACT_TYPE             0.000000
NAME_PAYMENT_TYPE              0.000000
dtype: float64
```

In [125…
```
previous_data_v2 = previous_data.dropna(axis=1, thresh=n_rows*0.8)
```

In [126…
```
len(previous_data_v2.columns)
```

Out[126]:
```
23
```

In [127…
```
n_rows = previous_data_v2.shape[0]
null_df = (previous_data_v2.isnull().sum()/n_rows*100).sort_values(ascending= False)
```

In [128…
```
null_df.head()
```

Out[128]:
```
PRODUCT_COMBINATION     0.020716
AMT_CREDIT              0.000060
NAME_PAYMENT_TYPE       0.000000
NAME_YIELD_GROUP        0.000000
NAME_SELLER_INDUSTRY    0.000000
dtype: float64
```

In [129…

```
# Merge the Application dataset with previous appliaction dataset
combine_data =pd.merge(left=data_v2,right=previous_data_v2,how='inner',on='SK_ID_CURR'
```

In [130…

```
combine_data.columns
```

Out[130]:

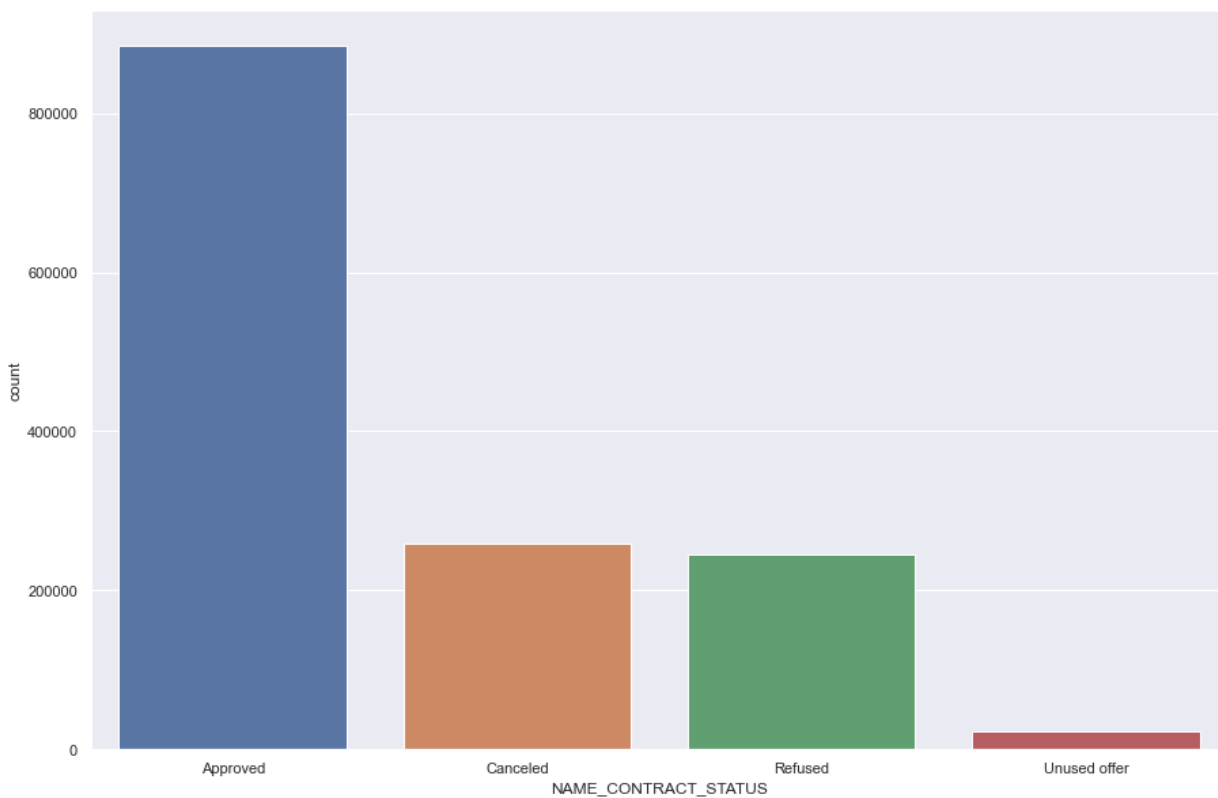
```
Index(['SK_ID_CURR', 'TARGET', 'NAME_CONTRACT_TYPE_x', 'CODE_GENDER',
       'FLAG_OWN_CAR', 'FLAG_OWN_REALTY', 'CNT_CHILDREN', 'AMT_INCOME_TOTAL',
       'AMT_CREDIT_x', 'AMT_ANNUITY', 'AMT_GOODS_PRICE', 'NAME_INCOME_TYPE',
       'NAME_EDUCATION_TYPE', 'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE',
       'CNT_FAM_MEMBERS', 'ORGANIZATION_TYPE', 'OCCUPATION_TYPE',
       'YEARS_BIRTH', 'YEARS_EMPLOYED', 'YEARS_REGISTRATION',
       'AMT_INCOME_RANGE', 'AMT_CREDIT_RANGE', 'SK_ID_PREV',
       'NAME_CONTRACT_TYPE_y', 'AMT_APPLICATION', 'AMT_CREDIT_y',
       'WEEKDAY_APPR_PROCESS_START', 'HOUR_APPR_PROCESS_START',
       'FLAG_LAST_APPL_PER_CONTRACT', 'NFLAG_LAST_APPL_IN_DAY',
       'NAME_CASH_LOAN_PURPOSE', 'NAME_CONTRACT_STATUS', 'DAYS_DECISION',
       'NAME_PAYMENT_TYPE', 'CODE_REJECT_REASON', 'NAME_CLIENT_TYPE',
       'NAME_GOODS_CATEGORY', 'NAME_PORTFOLIO', 'NAME_PRODUCT_TYPE',
       'CHANNEL_TYPE', 'SELLERPLACE_AREA', 'NAME_SELLER_INDUSTRY',
       'NAME_YIELD_GROUP', 'PRODUCT_COMBINATION'],
      dtype='object')
```

In [131…

```
# combine_data.drop(['SK_ID_CURR','WEEKDAY_APPR_PROCESS_START','HOUR_APPR_PROCESS_STAR
```

In [132…

```
plt.subplots(figsize=(15,10))
sns.countplot(x = combine_data['NAME_CONTRACT_STATUS'])
plt.show()
```

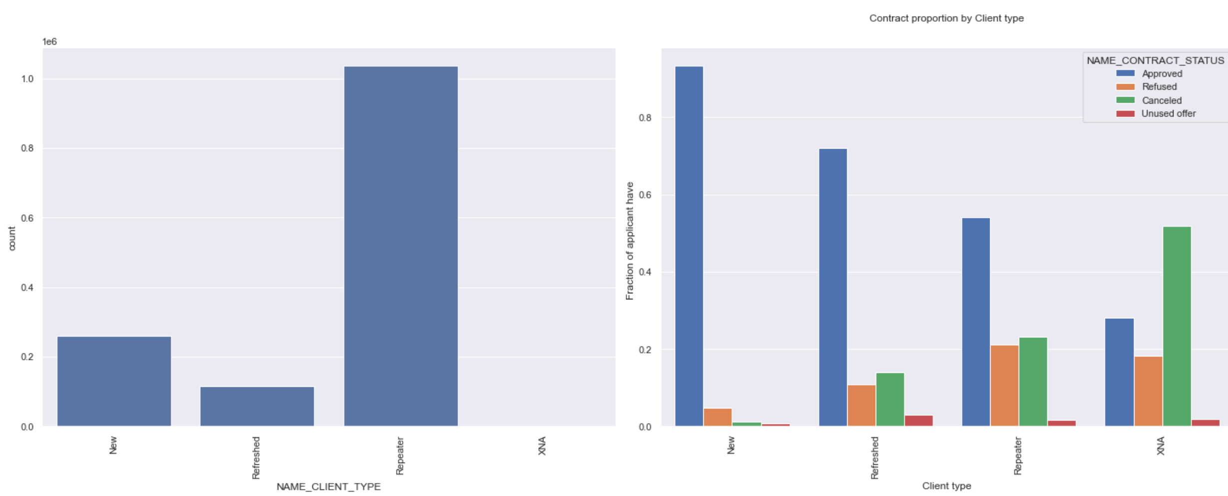


In [133…

```
def plot_var_2(col_name, full_name, continuous):
    fig, (ax1, ax2) = plt.subplots(1, 2, sharex=False, figsize=(20,8))
    if continuous:
        sns.distplot( combine_data[col_name],bins = 40, kde=False, ax=ax1)
    else:
```

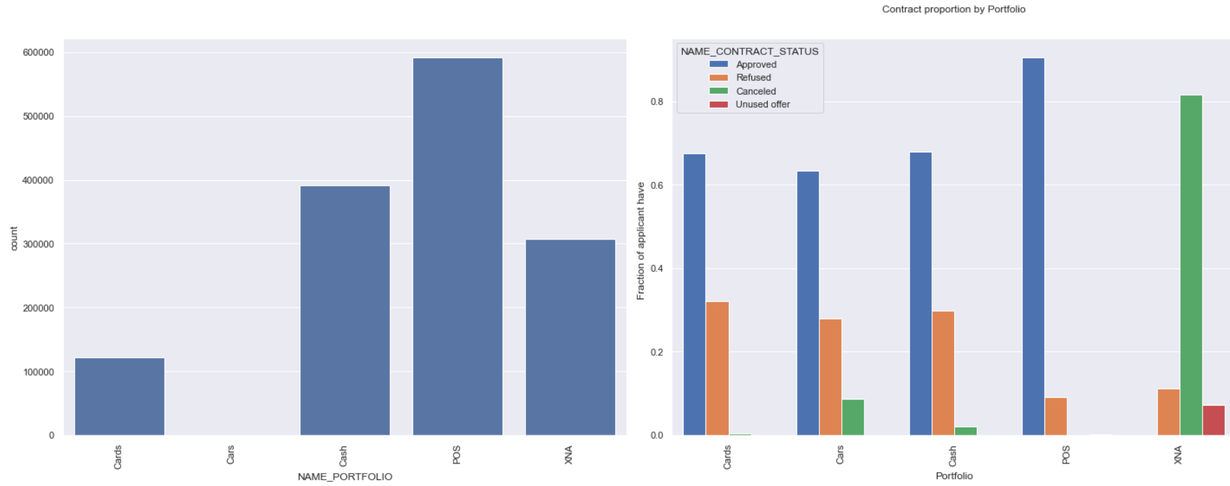
```
        sns.countplot(x = combine_data[col_name], order=sorted(combine_data[col_name].
    ax1.set_xticklabels(sorted(combine_data[col_name].unique()), rotation = 90)


    if continuous:
        sns.boxplot(y='NAME_CONTRACT_STATUS', x= col_name, data=combine_data, ax=ax2)
        ax2.set_ylabel('')
        ax2.set_title(full_name + ' by Contract status')
    else:
        contrac_status_rates = combine_data.groupby(col_name,as_index = False)['NAME_C
        sns.barplot(x=contrac_status_rates[col_name], y=contrac_status_rates['proporti
        ax2.set_ylabel('Fraction of applicant have  ')
        ax2.set_title('Contract proportion by ' + full_name)
        ax2.set_xlabel(full_name)
        plt.xticks(rotation=90)

    if continuous:
        facet = sns.FacetGrid(combine_data, hue = 'NAME_CONTRACT_STATUS', height=3, as
        facet.map(sns.kdeplot, col_name, shade=True)
        facet.add_legend()

    plt.tight_layout()
```

In [134…
```
plot_var_2('NAME_CLIENT_TYPE','Client type',continuous= False)
```



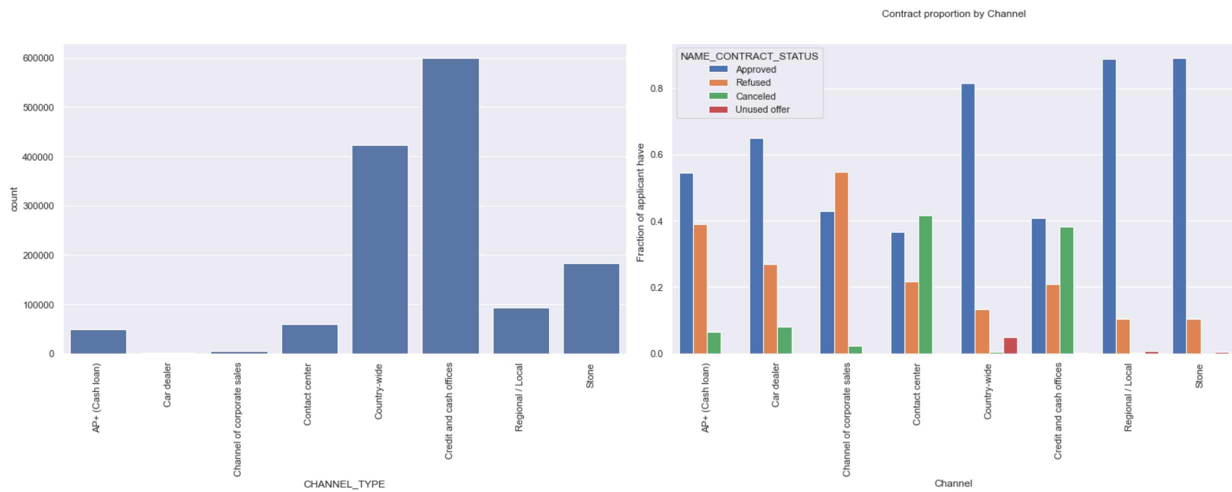
In [135…
```
# Insight: Repeater is the type of customer with the highest rejection and cancellatic
```

In [136…
```
plot_var_2('NAME_PORTFOLIO','Portfolio',continuous= False)
```



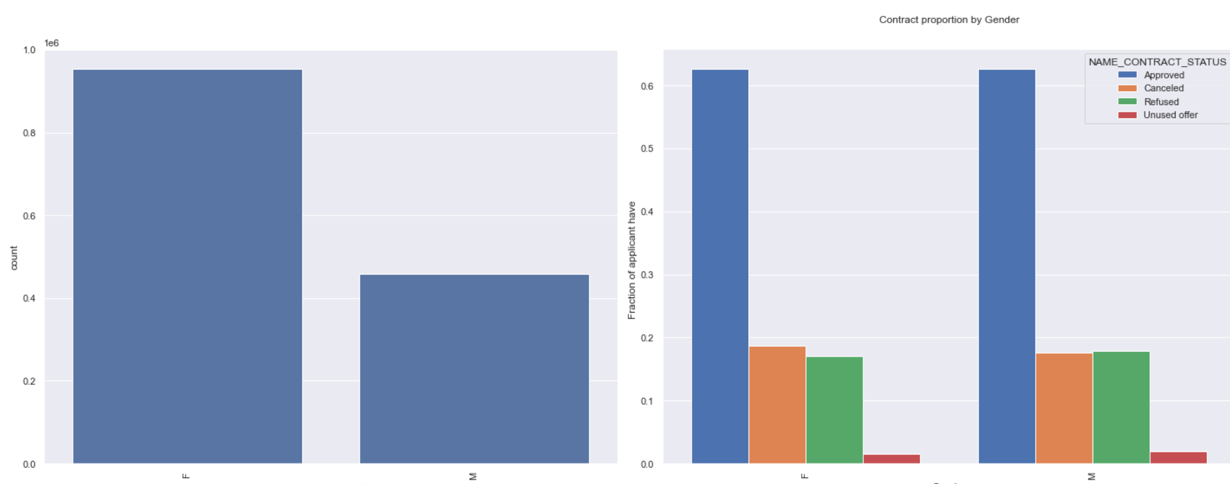
In [137…
```
# Insight: Applicant without Portfolio information will usually get loan canceled
```

In [138…
```
plot_var_2('CHANNEL_TYPE','Channel',continuous= False)
```



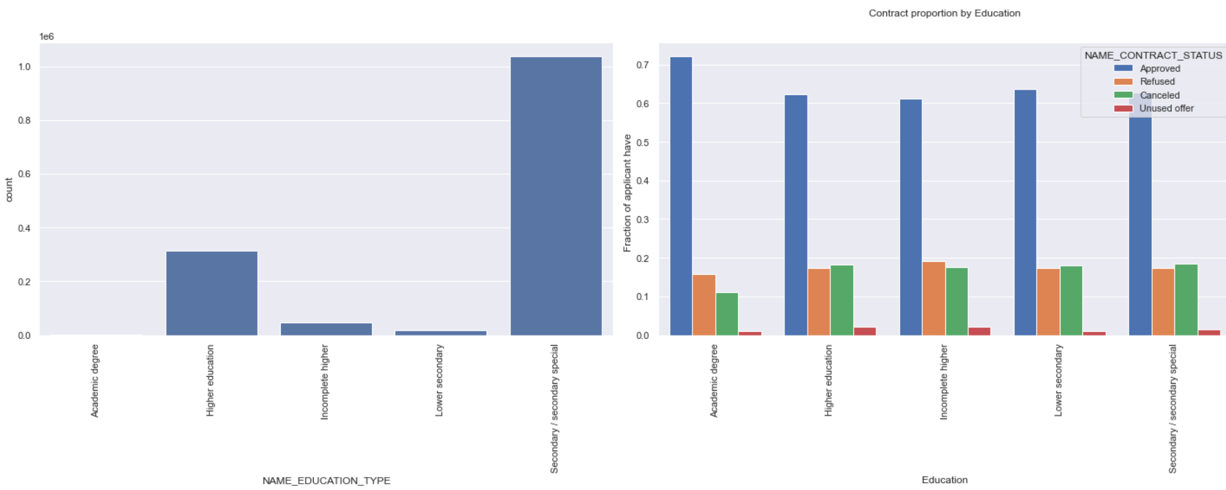
In [139…
```
# Insight: Regional / Local and Stone channel has a higher loan approval rate than oth
```

In [140…
```
plot_var_2('CODE_GENDER','Gender',continuous= False)
```



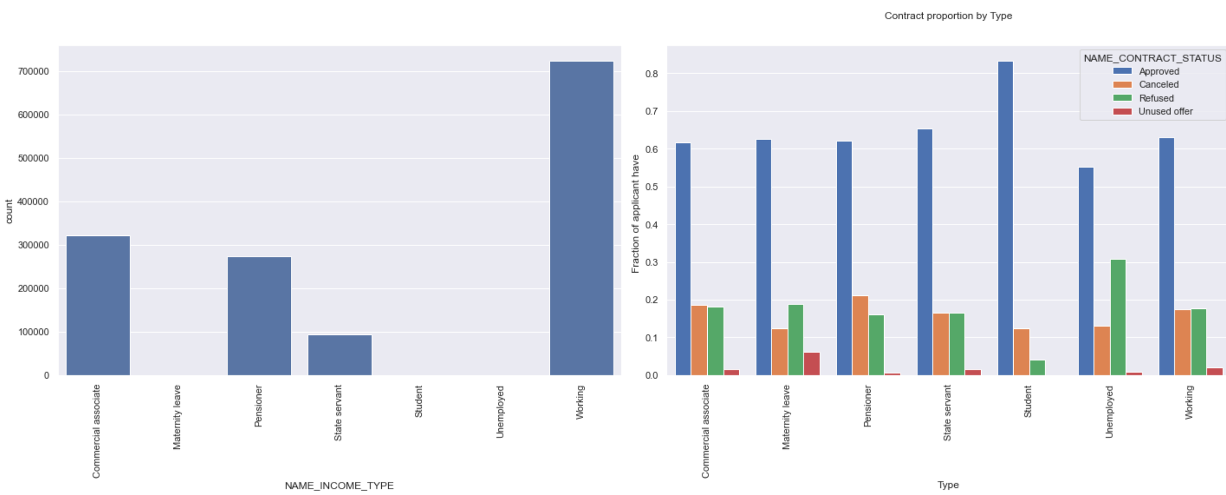
In [141…
```
# Insight: Here we can see that Female is getting more Refused more approved more canc
# more unused but in case of male it is having average in every category.
```

In [142…
```
plot_var_2('NAME_EDUCATION_TYPE','Education',continuous= False)
```



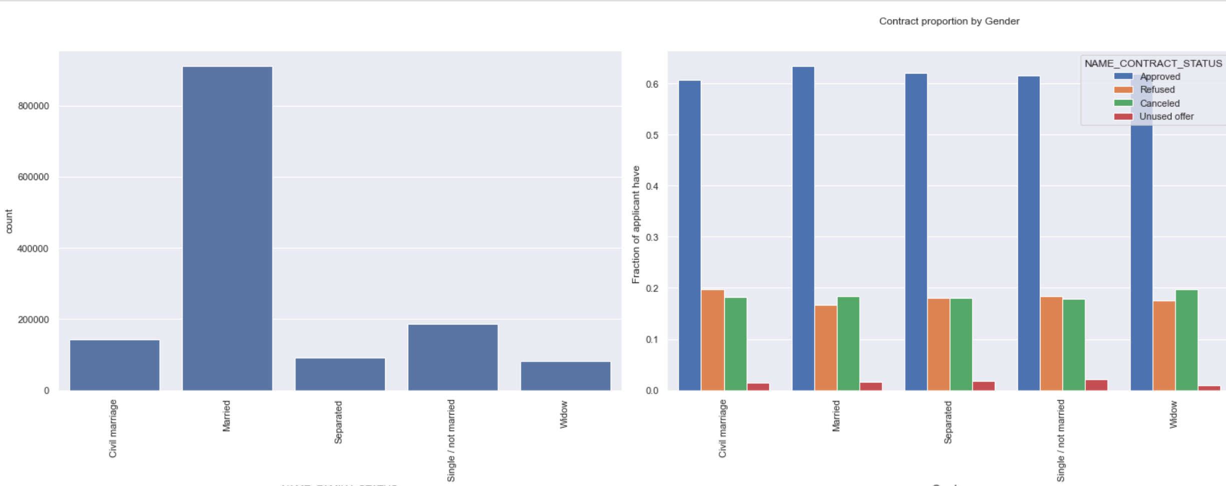
In [143…
```
# Insight: Here we can see that Secondary/ Secondary special is more effective in ever
```

In [144…
```
plot_var_2('NAME_INCOME_TYPE','Type',continuous= False)
```



In [145…
```
# Insight: Here we can see that the working type people are applying more loans as com
```

In [146…
```
plot_var_2('NAME_FAMILY_STATUS','Gender',continuous= False)
```

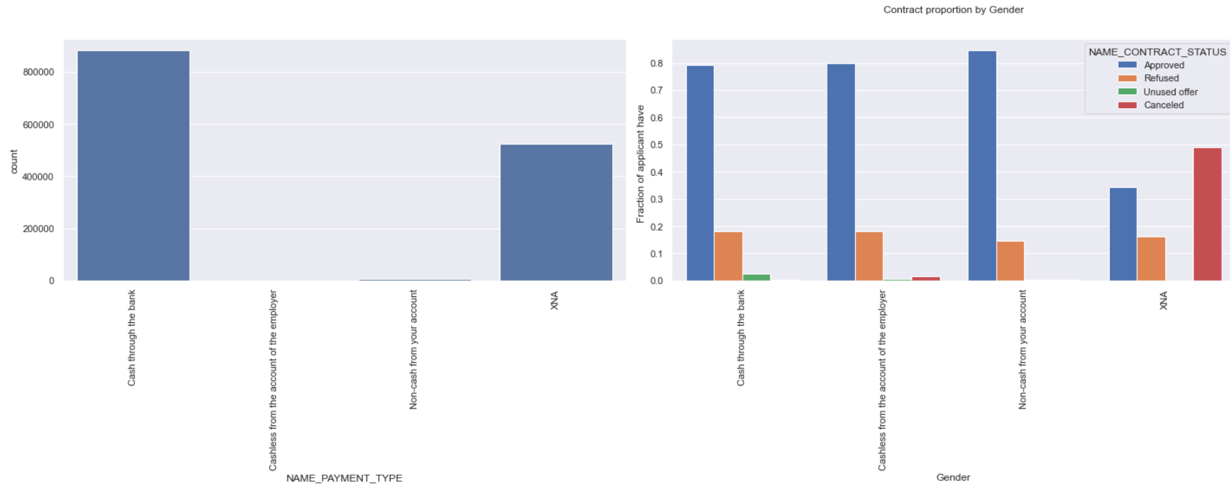


In [147…

```
# Here we can see that the Married people are applying and taking loans more than the
```

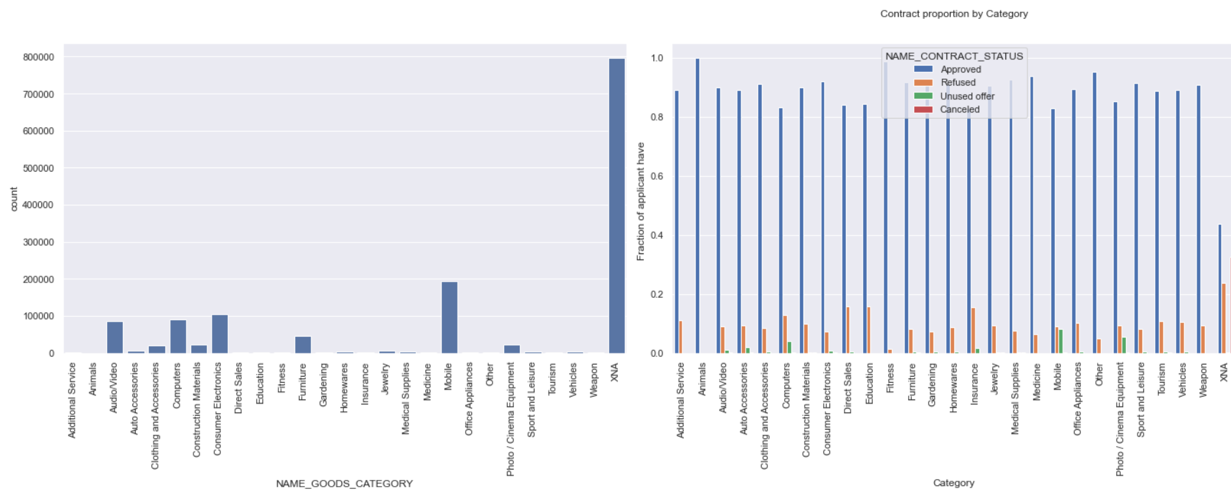
In [148…

```
plot_var_2('NAME_PAYMENT_TYPE','Gender',continuous= False)
```



In [149…

```
plot_var_2('NAME_GOODS_CATEGORY','Category',continuous= False)
```



In [ ]: