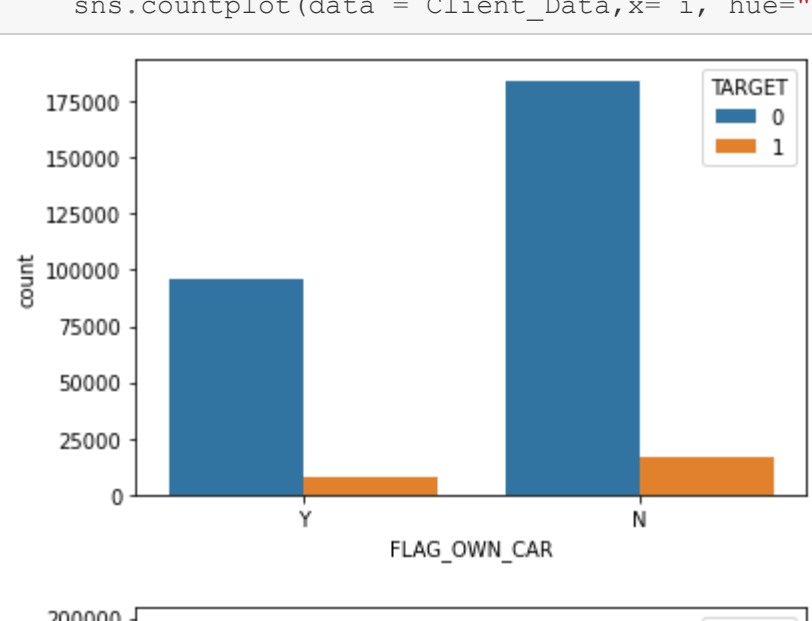
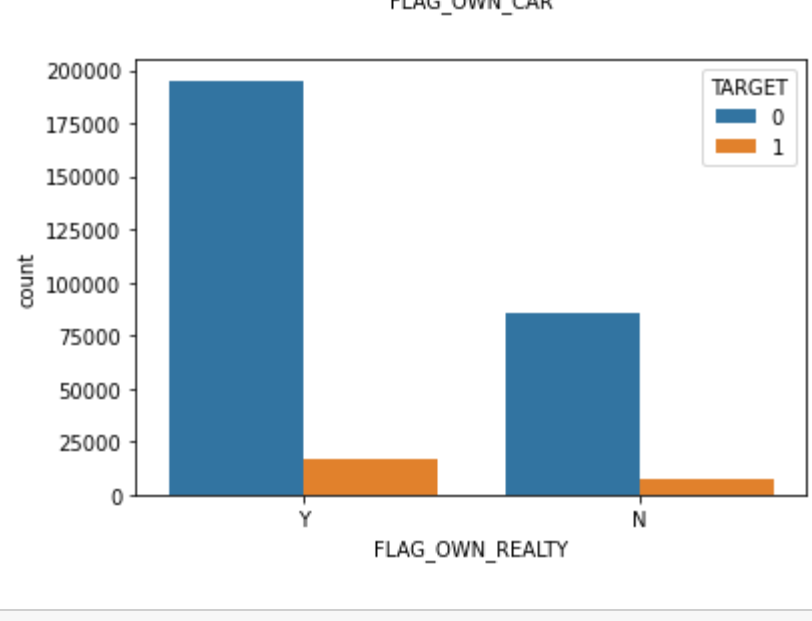
In [32]:
```
for i in Client_Data[["FLAG_OWN_CAR","FLAG_OWN_REALTY"]]:
    plt.figure(i)
    sns.countplot(data = Client_Data,x= i, hue="TARGET",order=("Y","N"))
```





In [33]:
```
for i in Client_Data[["FLAG_OWN_CAR","FLAG_OWN_REALTY"]]:
    defaulters1 = Client_Data[(Client_Data[i]=="Y") & (Client_Data["TARGET"] ==1)]
    defaulters2 = Client_Data[(Client_Data[i]=="N") & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,"(H/C)",":  ",round((len(defaulters1)/Total)*100,3))
    print(i,"(NH/NC)",":  ",round((len(defaulters2)/Total)*100,3))
```

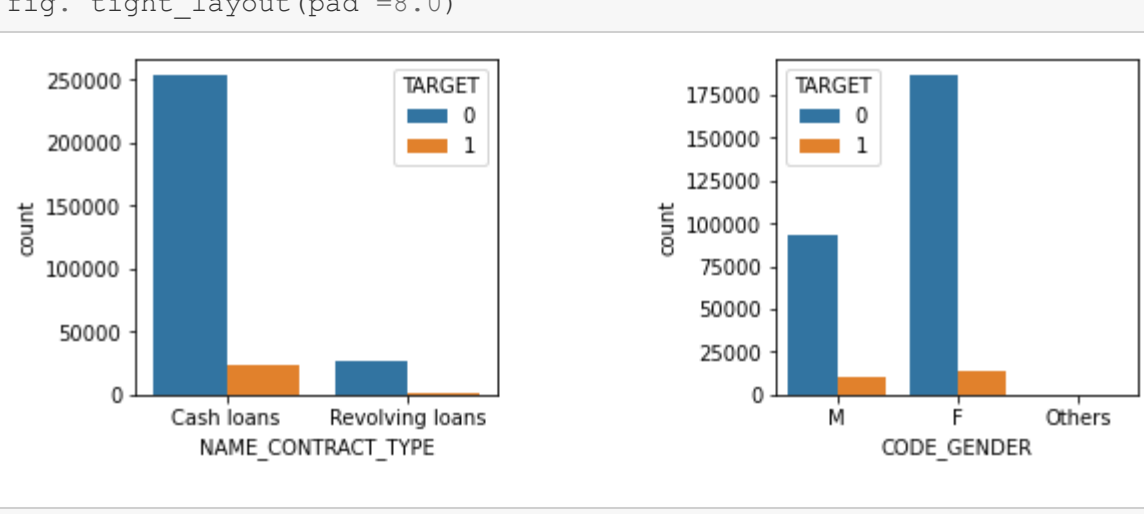
```
FLAG_OWN_CAR (H/C) :   30.532
FLAG_OWN_CAR (NH/NC) :   69.468
FLAG_OWN_REALTY (H/C) :   68.452
FLAG_OWN_REALTY (NH/NC) :   31.548
```

- 70% of Defaulters dont have own car but had Own house which is a Risk factor

Family Details

In [34]:
```
Fam_Details = Client_Data[["NAME_CONTRACT_TYPE","CODE_GENDER","CNT_FAM_MEMBERS","CNT_CHILDREN","NAME_IN
COME_TYPE","NAME_EDUCATION_TYPE","NAME_FAMILY_STATUS","NAME_TYPE_SUITE","NAME_HOUSING_TYPE"]]
```

In [35]:
```
fig,axes =plt.subplots(1,2,figsize=(10,5),)
sns.countplot(data =Client_Data,x="NAME_CONTRACT_TYPE",hue="TARGET",ax=axes[0])
sns.countplot(x="CODE_GENDER",hue="TARGET",data =Client_Data,ax=axes[1])
fig. tight_layout(pad =8.0)
```

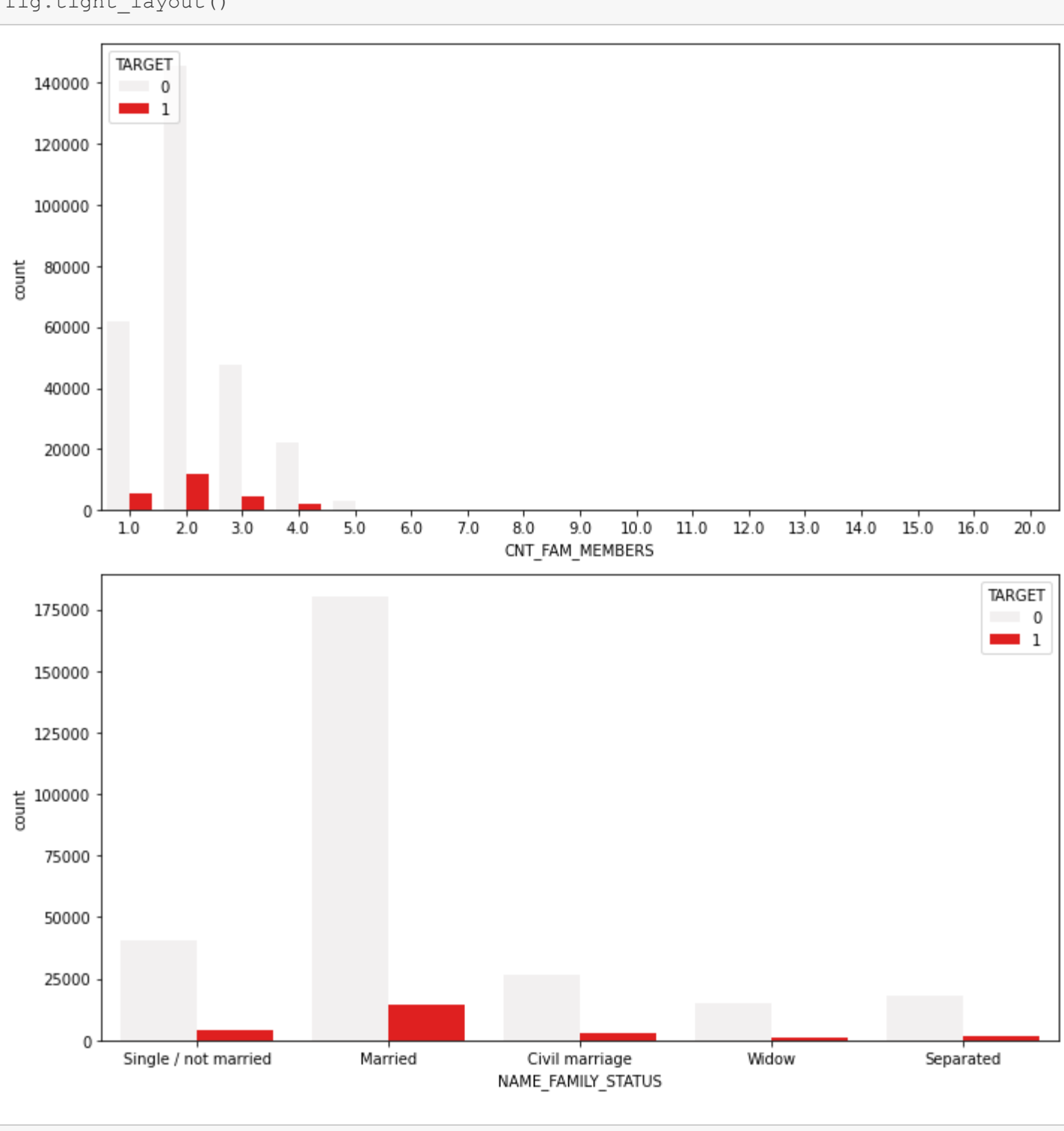


In [36]:
```
for i in Fam_Details["NAME_CONTRACT_TYPE"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_CONTRACT_TYPE"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",(len(defaulters1)/Total)*100)
```

```
Cash loans :   93.7454486608949
Revolving loans :   6.254551339105105
```

- Highest Applicants and Defaulters are Female and of Cash loans
- But when compared to total Male Applicants, the deafulters were quite high in Male

In [37]:
```
fig,axes =plt.subplots(2,1,figsize=(10,10))
sns.countplot(data =Client_Data,x="CNT_FAM_MEMBERS",hue="TARGET",ax=axes[0],color="red")
sns.countplot(data =Client_Data,x="NAME_FAMILY_STATUS",hue="TARGET",ax=axes[1],color="red")
fig.tight_layout()
```



In [38]:
```
for i in Fam_Details["NAME_FAMILY_STATUS"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_FAMILY_STATUS"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

```
Single / not married :   17.967
Married :   59.815
Civil marriage :   11.922
Widow :   3.779
Separated :   6.518
```

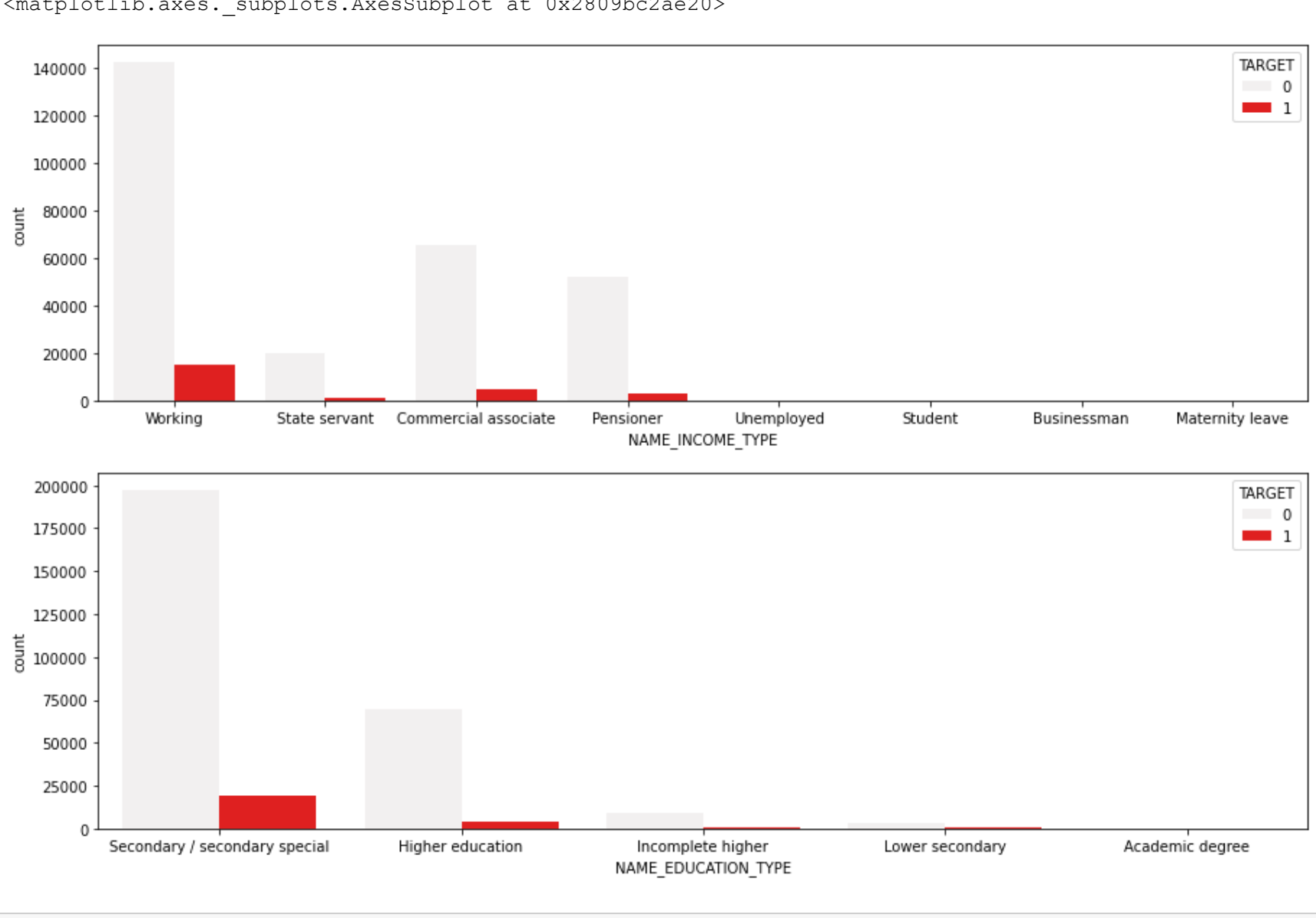
In [39]:
```
for i in Fam_Details["CNT_FAM_MEMBERS"].unique():
    defaulters1 = Fam_Details[(Fam_Details["CNT_FAM_MEMBERS"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

```
1.0 :   22.87
2.0 :   48.357
3.0 :   18.557
4.0 :   8.609
5.0 :   1.323
6.0 :   0.223
9.0 :   0.0
7.0 :   0.024
8.0 :   0.024
10.0 :   0.004
13.0 :   0.004
14.0 :   0.0
12.0 :   0.0
20.0 :   0.0
15.0 :   0.0
16.0 :   0.0
11.0 :   0.004
```

- Eventhough highest Defaulters lie around Married, but risk factor is high with ~5 - ~10% Single,Civil marriages and Seperated
- Highest Defaulters with Family members less than 5

In [40]:
```
fig,axes =plt.subplots(2,1,figsize=(15,10))
sns.countplot(data =Client_Data,x="NAME_INCOME_TYPE",hue="TARGET",ax=axes[0],color="red")
sns.countplot(data =Client_Data,x="NAME_EDUCATION_TYPE",hue="TARGET",ax=axes[1],color="red")
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2809bc2ae20>
```



In [41]:
```
for i in Fam_Details["NAME_EDUCATION_TYPE"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_EDUCATION_TYPE"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

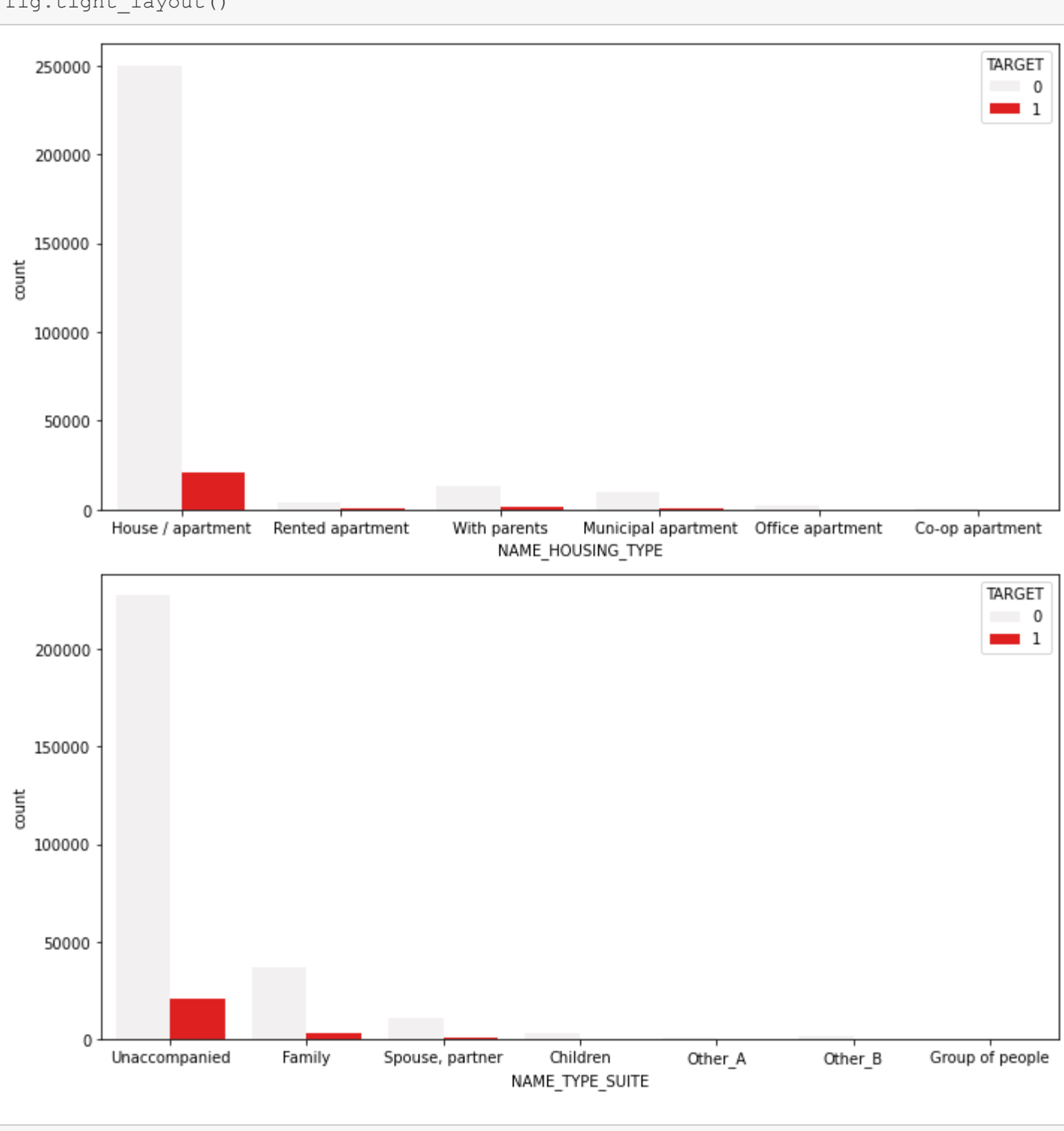
```
Secondary / secondary special :   78.684
Higher education :   16.118
Incomplete higher :   3.508
Lower secondary :   1.679
Academic degree :   0.012
```

In [42]:
```
for i in Fam_Details["NAME_INCOME_TYPE"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_INCOME_TYPE"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

```
Working :   61.372
State servant :   5.021
Commercial associate :   21.567
Pensioner :   11.999
Unemployed :   0.032
Student :   0.0
Businessman :   0.0
Maternity leave :   0.008
```

- Highest Defaulters were Working and Commercial associate, but the risk to be assesed on 11% Pensioner based on their asset value.

In [43]:
```
fig,axes =plt.subplots(2,1,figsize=(10,10))
sns.countplot(data =Client_Data,x="NAME_HOUSING_TYPE",hue="TARGET",ax=axes[0],color="red")
sns.countplot(data =Client_Data,x="NAME_TYPE_SUITE",hue="TARGET",ax=axes[1],color="red")
fig.tight_layout()
```



In [44]:
```
for i in Fam_Details["NAME_HOUSING_TYPE"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_HOUSING_TYPE"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

```
House / apartment :   85.691
Rented apartment :   2.415
With parents :   7.007
Municipal apartment :   3.843
Office apartment :   0.688
Co-op apartment :   0.356
```

In [45]:
```
for i in Fam_Details["NAME_TYPE_SUITE"].unique():
    defaulters1 = Fam_Details[(Fam_Details["NAME_TYPE_SUITE"] == i) & (Client_Data["TARGET"] ==1)]
    Total = len(Client_Data[Client_Data["TARGET"] ==1])
    print(i,":  ",round((len(defaulters1)/Total)*100,3))
```

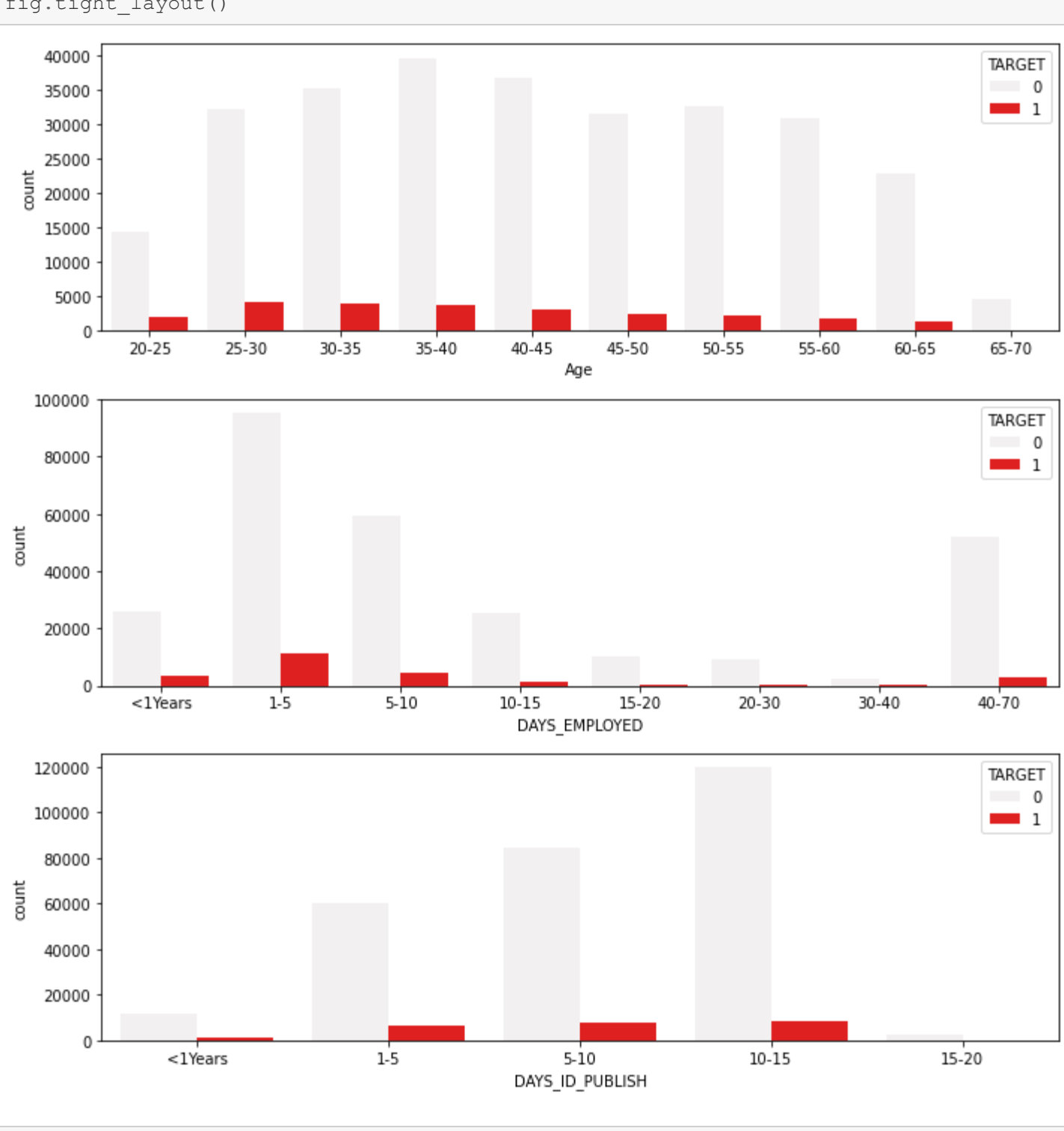
```
Unaccompanied :   82.13
Family :   12.169
Spouse, partner :   3.621
Children :   0.975
Other_A :   0.307
Other_B :   0.704
Group of people :   0.093
```

- 82% Defaulters were Unaccompanied while submiting loan Application, but cannot be considered as primary risk factor
- 2% Defaulters live in Rented Apartments a concern factor.

In [46]:
```
Client_Data1=Client_Data.copy()
```

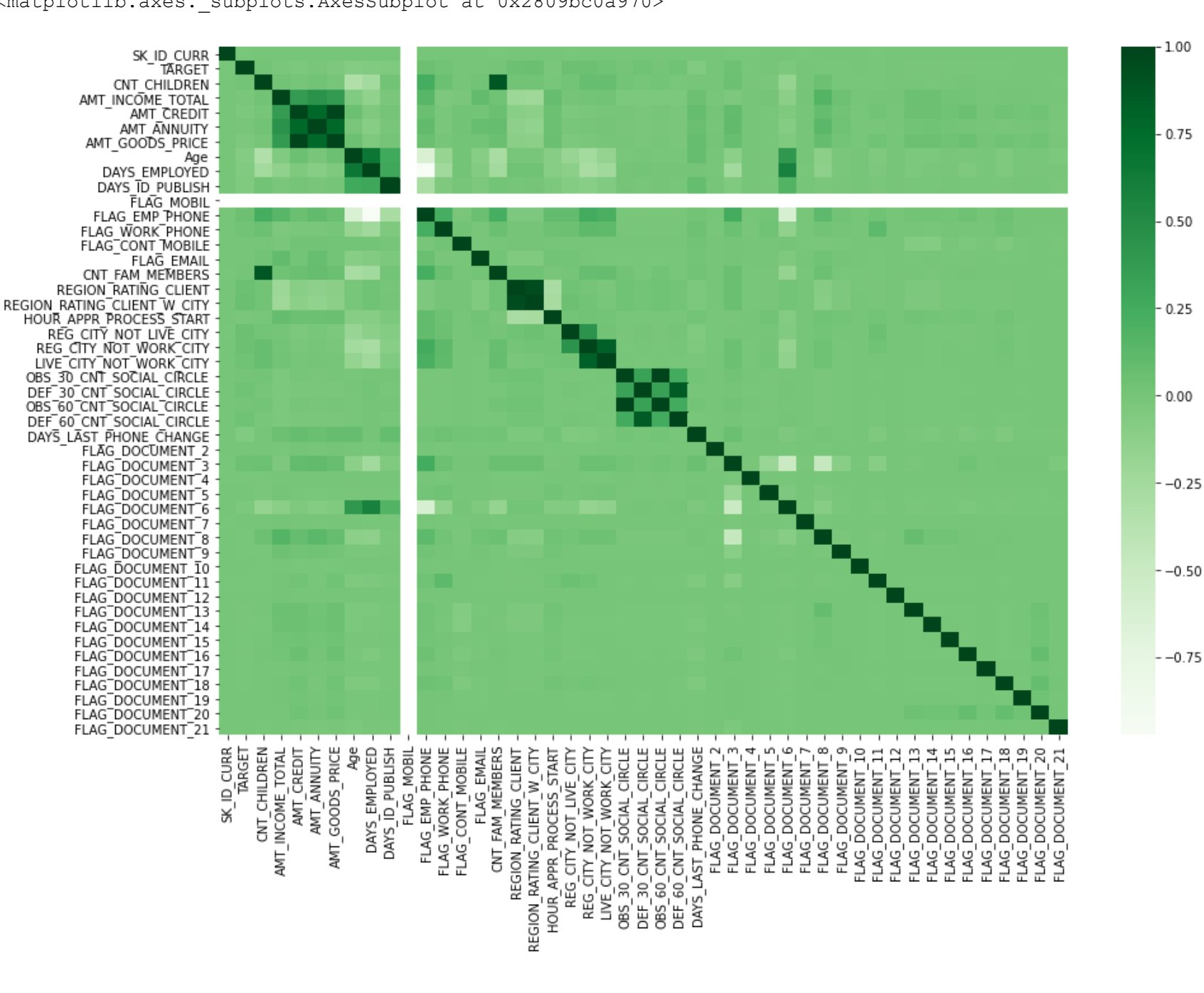
In [47]:
```
Client_Data1["Age"] = pd.cut(x= Client_Data1["Age"],bins = [20,25,30,35,40,45,50,55,60,65,70],labels =
['20-25','25-30','30-35','35-40','40-45','45-50','50-55','55-60','60-65','65-70'])
Client_Data1["DAYS_EMPLOYED"] = pd.cut(x= Client_Data1["DAYS_EMPLOYED"],bins = [0.1,1.0,5.0,10.0,15.0,2
0.0,30.0,40.0,70.0],labels = ['<1Years','1-5','5-10','10-15','15-20','20-30','30-40','40-70'])
Client_Data1["DAYS_ID_PUBLISH"] = pd.cut(x= Client_Data1["DAYS_ID_PUBLISH"],bins = [0.1,1.0,5.0,10.0,1
5.0,20.0],labels = ['<1Years','1-5','5-10','10-15','15-20'])
```

In [48]:
```
fig,axes =plt.subplots(3,1,figsize=(10,10))
sns.countplot(data =Client_Data1,x="Age",hue="TARGET",ax=axes[0],color="red")
sns.countplot(data =Client_Data1,x="DAYS_EMPLOYED",hue="TARGET",ax=axes[1],color="red")
sns.countplot(data =Client_Data1,x="DAYS_ID_PUBLISH",hue="TARGET",ax=axes[2],color="red")
fig.tight_layout()
```



In [49]:
```
plt.figure(figsize=(15,10))
sns.heatmap(Client_Data.corr(), cmap="Greens")
```

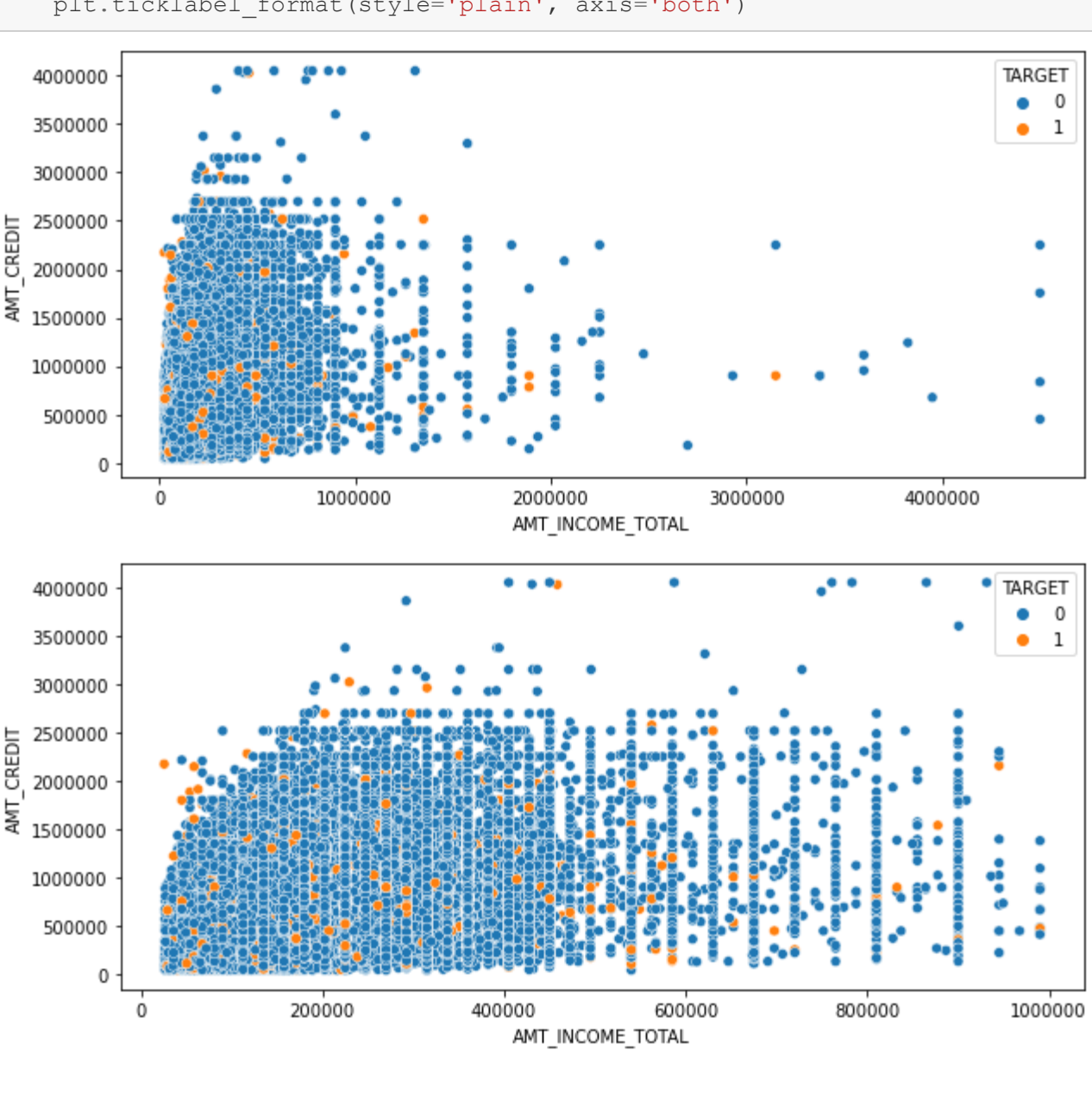
Out[49]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2809bc0a970>
```



- Amt_Goods Price is highly correlated with Amt_Credit.
- From our Document Status plot, we didnt got much insights, but we could see correlation exists between Age,Days_Employed with Flag document6.

Bi-Variate Analysis

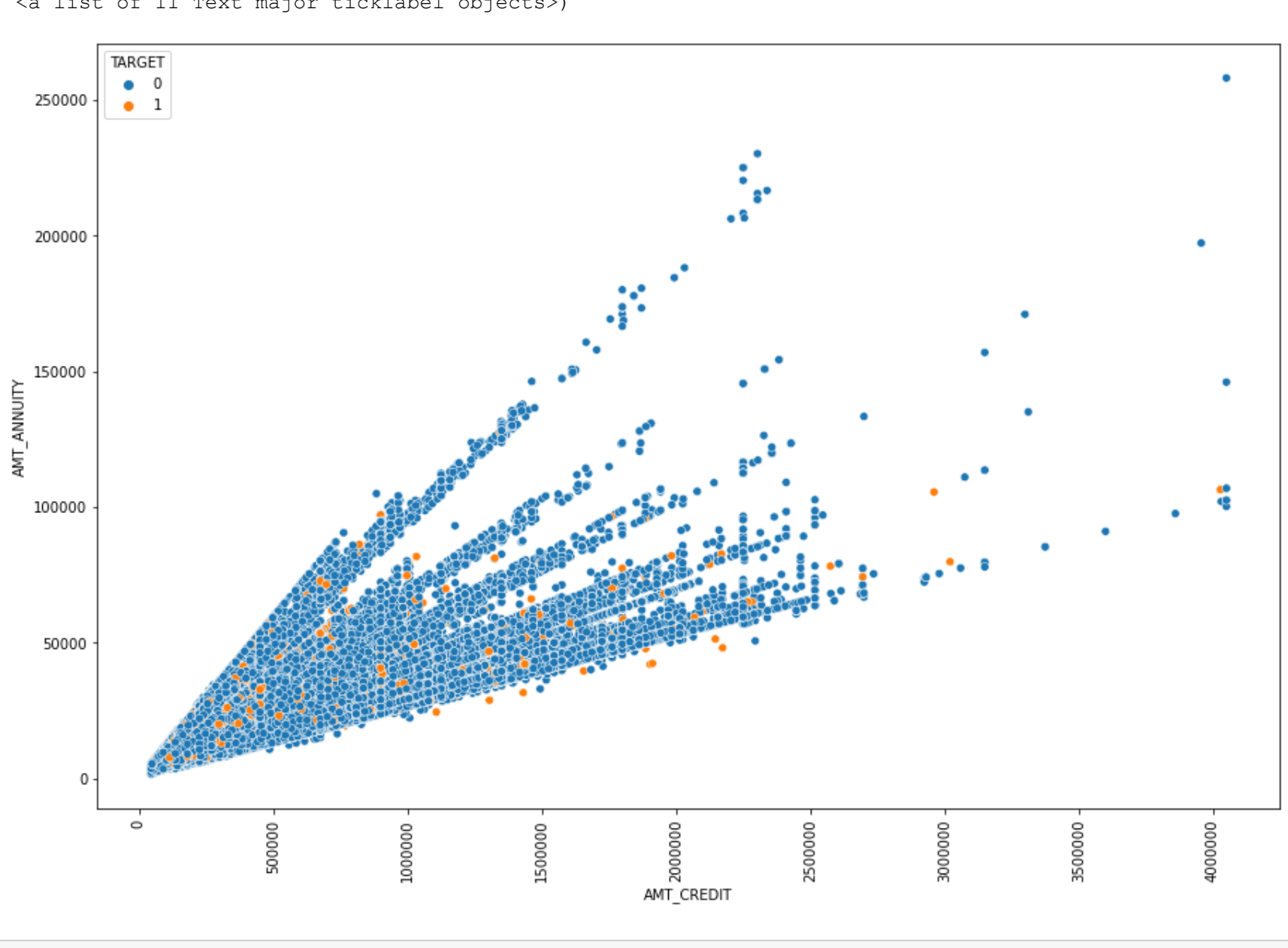
In [50]:
```
fig,ax =plt.subplots(2,1,figsize=(10,10))
sns.scatterplot(data=Client_Data, x="AMT_INCOME_TOTAL", y="AMT_CREDIT",hue ="TARGET",ax=ax[0])
Inc =Client_Data.loc[Client_Data["AMT_INCOME_TOTAL"]<1000000,["AMT_INCOME_TOTAL","TARGET","AMT_CREDIT"
]]
sns.scatterplot(data=Inc, x="AMT_INCOME_TOTAL", y="AMT_CREDIT",hue ="TARGET",ax=ax[1])

for ax in fig.axes:
    plt.sca(ax)
    plt.ticklabel_format(style='plain', axis='both')
```



- Majority of Applicants has Income <1Millions and Credited loan amount is < 2.5 Millions
- More Defaulters fall under Income range of 0.4 Millions

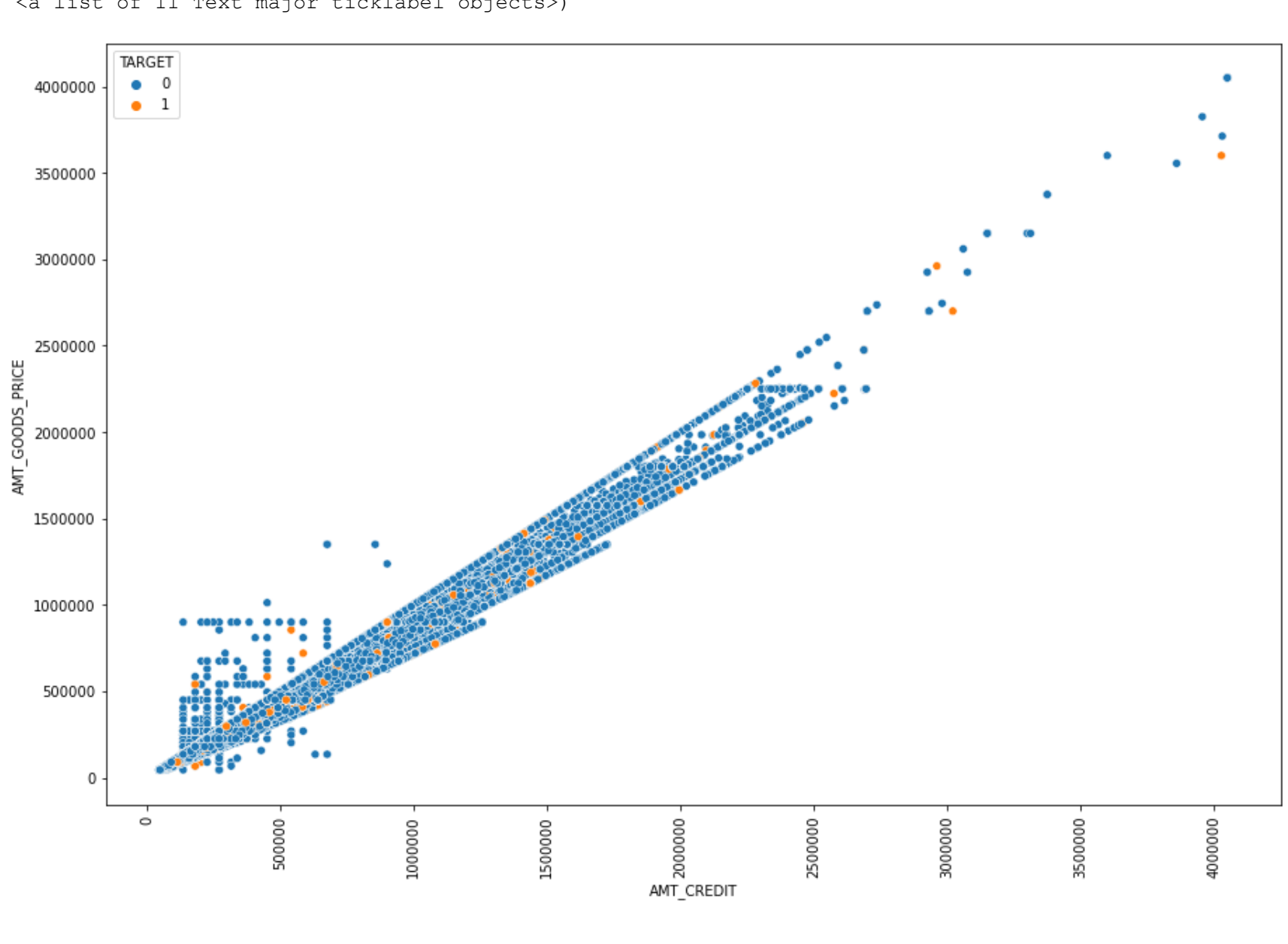
In [51]:
```
plt.figure(figsize=(15,10),)
sns.scatterplot(data=Client_Data, x="AMT_CREDIT", y="AMT_ANNUITY",hue ="TARGET")
plt.ticklabel_format(style='plain', axis='both')
plt.xticks(rotation =90)
```

Out[51]:
```
(array([-500000.,       0.,  500000., 1000000., 1500000., 2000000.,
        2500000., 3000000., 3500000., 4000000., 4500000.]),
 <a list of 11 Text major ticklabel objects>)
```



In [52]:
```
plt.figure(figsize=(15,10),)
sns.scatterplot(data=Client_Data, x="AMT_CREDIT", y="AMT_GOODS_PRICE",hue ="TARGET")
plt.ticklabel_format(style='plain', axis='both')
plt.xticks(rotation =90)
```

Out[52]:
```
(array([-500000.,       0.,  500000., 1000000., 1500000., 2000000.,
        2500000., 3000000., 3500000., 4000000., 4500000.]),
 <a list of 11 Text major ticklabel objects>)
```



Conclusion:

8% were Defaulters out of 3Lakh Applicants.

Data_Discrepancies:

- AMT_INCOME column has income ranging from 6-120 Millions, but the credit amount 5% which is concern factor.
- Few Records in Days Employeed has crossed max 70 years, they were imputed with 70.

Findings:

- Maximum loans sanctioned were Cash loans.
- Highest Applicants are female, but deafuters were high in Male.
- Risk is less with applicants who are having either car or Own House.
- Applicants in Rented houses and Pensioners are of high risk.
- Risk is high with Single parent applicants
- Maximum Defaulters lie in Age group of 25-30.
- Maximum Defaulters has employee experince < 5years, but the risk lies with <1 year Experience Employees.
- Most Applicants and defaulters fall under Income range of 40K and 2.5 Lakhs.

In [53]:
```
Client_Data.to_csv("Client_data.csv",index=False)
```

In [ ]: