Credit EDA Assignment

CANDIDATE: ANURAG THAWAIT

Steps Of Analysis:

1. Read the given Datasets
2. Find the basic information as shape, describe, info, null values etc.
3. Identify the target variable
4. Identify affecting data points: Consider by checking 4-5 top columns which are relatable to the problem solution.
    a. Data Cleaning
    b. Managing Null values
5. Relate the data points to Target variable by plotting visualizations
    a. Managing Outliers
    b. Write outcome
6. Review Visualizations for conclusions and requirements
7. Conclusion

Step 1: Read the given Datasets

Importing libraries and rules

```
In [1]: import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
        import warnings
        warnings.filterwarnings('ignore')
        ## for whole data to be seen
        pd.set_option('display.max_rows', None)
        pd.set_option('display.max_columns', None)
```

Read Data

```
In [*]: prev= pd.read_csv('previous_application.csv')
```

```
In [*]: appl= pd.read_csv('application_data.csv')
```

Step 2: Find the basic information as shape, describe, info, null values etc.

In [4]:
```
appl.head()
```
Out[4]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN
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	

In [5]:
```
appl.describe()
```
Out[5]:

	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_AI
count	307511.000000	307511.000000	307511.000000	3.075110e+05	3.075110e+05	307499
mean	278180.518577	0.080729	0.417052	1.687979e+05	5.990260e+05	27108
std	102790.175348	0.272419	0.722121	2.371231e+05	4.024908e+05	14493
min	100002.000000	0.000000	0.000000	2.565000e+04	4.500000e+04	1615
25%	189145.500000	0.000000	0.000000	1.125000e+05	2.700000e+05	16524
50%	278202.000000	0.000000	0.000000	1.471500e+05	5.135310e+05	24903
75%	367142.500000	0.000000	1.000000	2.025000e+05	8.086500e+05	34596
max	456255.000000	1.000000	19.000000	1.170000e+08	4.050000e+06	258025

In [7]:
```
appl.info('all')
```
```
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 [8]:
```
appl.dtypes
```
Out[8]:
```
SK_ID_CURR               int64
TARGET                   int64
NAME_CONTRACT_TYPE      object
CODE_GENDER             object
FLAG_OWN_CAR            object
FLAG_OWN_REALTY         object
CNT_CHILDREN             int64
AMT_INCOME_TOTAL       float64
AMT_CREDIT             float64
AMT_ANNUITY            float64
AMT_GOODS_PRICE        float64
```

Step 3: Identify the target variable

Business Understanding

The loan providing companies find it hard to give loans to the people due to their insufficient or non-existent credit history. Because of that, some consumers use it to their advantage by becoming a defaulter. Suppose you work for a consumer finance company which specialises in lending various types of loans to urban customers. You have to use EDA to analyse the patterns present in the data. This will ensure that the applicants capable of repaying the loan are not rejected.

When the company receives a loan application, the company has to decide for loan approval based on the applicant's profile. Two types of risks are associated with the bank's decision:

```
If the applicant is likely to repay the loan, then not approving the loan results in a loss of business to the company

If the applicant is not likely to repay the loan, i.e. he/she is likely to default, then approving the loan may lead to a financial loss for the company.
```

The data given below contains the information about the loan application at the time of applying for the loan. It contains two types of scenarios:

```
The client with payment difficulties: he/she had late payment more than X days on at least one of the first Y instalments of the loan in our sample,

All other cases: All other cases when the payment is paid on time.
```

When a client applies for a loan, there are four types of decisions that could be taken by the client/company):

```
Approved: The Company has approved loan Application

Cancelled: The client cancelled the application sometime during approval. Either the client changed her/his mind about the loan or in some cases due to a higher risk of the client, he received worse pricing which he did not want.

Refused: The company had rejected the loan (because the client does not meet their requirements etc.).

Unused offer:  Loan has been cancelled by the client but at different stages of the process.
```

Step 4: Identify affecting data points: Consider by checking 4-5 top columns which are relatable to the problem solution.

a. Data Cleaning
b. Managing Null values

Data Cleaning

Managing Null values

In [10]:
```
#percentage of missing values in each column
appl.isnull().sum()/len(appl)*100
```

Out[10]:
```
SK_ID_CURR                    0.000000
TARGET                        0.000000
NAME_CONTRACT_TYPE            0.000000
CODE_GENDER                   0.000000
FLAG_OWN_CAR                  0.000000
FLAG_OWN_REALTY               0.000000
CNT_CHILDREN                  0.000000
AMT_INCOME_TOTAL              0.000000
AMT_CREDIT                    0.000000
AMT_ANNUITY                   0.003902
AMT_GOODS_PRICE               0.090403
NAME_TYPE_SUITE               0.420148
NAME_INCOME_TYPE              0.000000
NAME_EDUCATION_TYPE           0.000000
NAME_FAMILY_STATUS            0.000000
NAME_HOUSING_TYPE             0.000000
REGION_POPULATION_RELATIVE    0.000000
DAYS_BIRTH                    0.000000
DAYS_EMPLOYED                 0.000000
```

In [11]:
```
#coloums having greater than 45% null value

nullvalues=appl.isnull().sum()/len(appl)*100
nullvalues=nullvalues[nullvalues.values>45.0]
print(nullvalues)
```

In [13]:
```
#Removing all columns having more than 45% null values
nullvalues = list(nullvalues[nullvalues.values>=45.0].index)
appl.drop(labels=nullvalues,axis=1,inplace=True)
print(len(nullvalues))
```

```
49
```

In [14]:
```
#shape of the dataframe after removing columns
appl.shape
```

Out[14]:
```
(307511, 73)
```

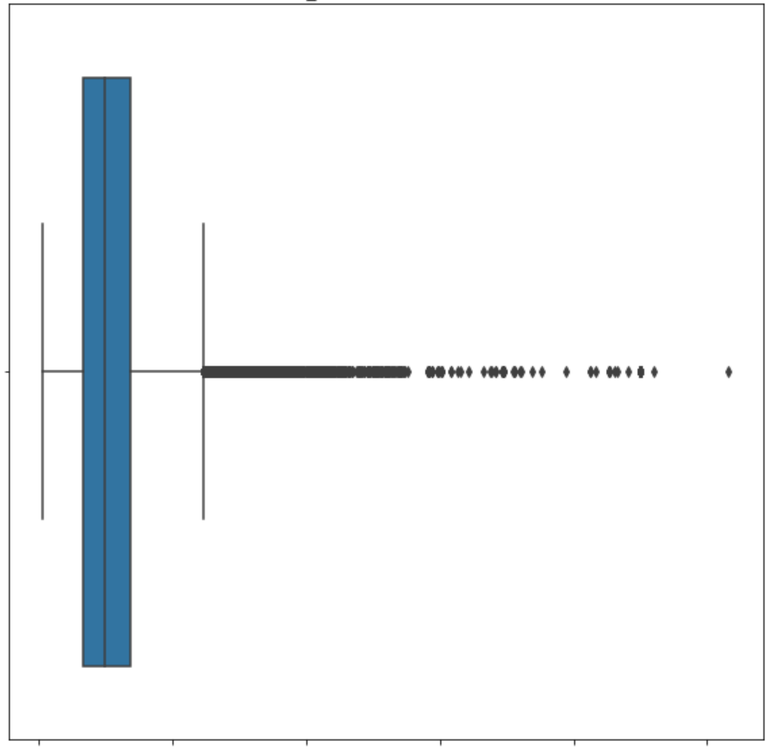
In [15]:
```
# columns having smaller value of null percentage
appl.isnull().sum()/len(appl)*100
```

Out[15]:
```
SK_ID_CURR            0.000000
TARGET                0.000000
NAME_CONTRACT_TYPE    0.000000
CODE_GENDER           0.000000
FLAG_OWN_CAR          0.000000
FLAG_OWN_REALTY       0.000000
CNT_CHILDREN          0.000000
AMT_INCOME_TOTAL      0.000000
AMT_CREDIT            0.000000
AMT_ANNUITY           0.003902
AMT_GOODS_PRICE       0.090403
NAME_TYPE_SUITE       0.420148
NAME_INCOME_TYPE      0.000000
```

Step 5: Relate the data points to Target variable by plotting visualizations

a. Managing Outliers
b. Write outcome

In [18]:
```
#plotting the values of AMT_ANNUITY column using box plot to detect outliers
plt.figure(figsize=(10,10))
sns.boxplot(appl.AMT_ANNUITY)
plt.title("AMT_ANNUTY data visual",fontsize=14)
plt.show()
```




```
50%       24903.000000
75%       34596.000000
max      258025.500000
Name: AMT_ANNUITY, dtype: float64
```

Mean: 27108, Median: 24903, There are sever outliners and the difference between max and min is quite severe. So w
those null values.

In [20]:
```
#count of missing value for AMT_ANNUITY column
appl.AMT_ANNUITY.isnull().sum()
```

Out[20]: 12

In [21]:
```
# Filling missing values in column AMT_ANNUITY with median
fillings1=appl['AMT_ANNUITY'].median()
appl['AMT_ANNUITY'].fillna(value = fillings1, inplace =True)
```

In [22]:
```
#count of missing value for AMT_ANNUITY column
appl.AMT_ANNUITY.isnull().sum()
```

Out[22]: 0

In [23]:
```
# Checking the columns having less null percentage
appl.isnull().sum()/len(appl)*100
```

Out[23]:
```
SK_ID_CURR                0.000000
TARGET                    0.000000
NAME_CONTRACT_TYPE        0.000000
CODE_GENDER               0.000000
FLAG_OWN_CAR              0.000000
FLAG_OWN_REALTY           0.000000
CNT_CHILDREN              0.000000
```

3. Analysis of Code gender

In [30]:
```
#count of each gender M/F
appl['CODE_GENDER'].value_counts()
```

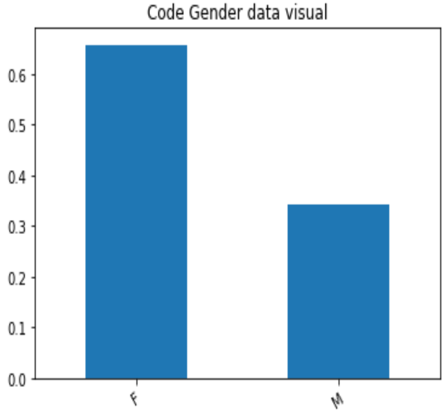
Out[30]:
```
F      202448
M      105059
XNA         4
Name: CODE_GENDER, dtype: int64
```

Since Female(F) is having the majority and only 4 rows are having XNA values. So, using F as mode to replace that data.

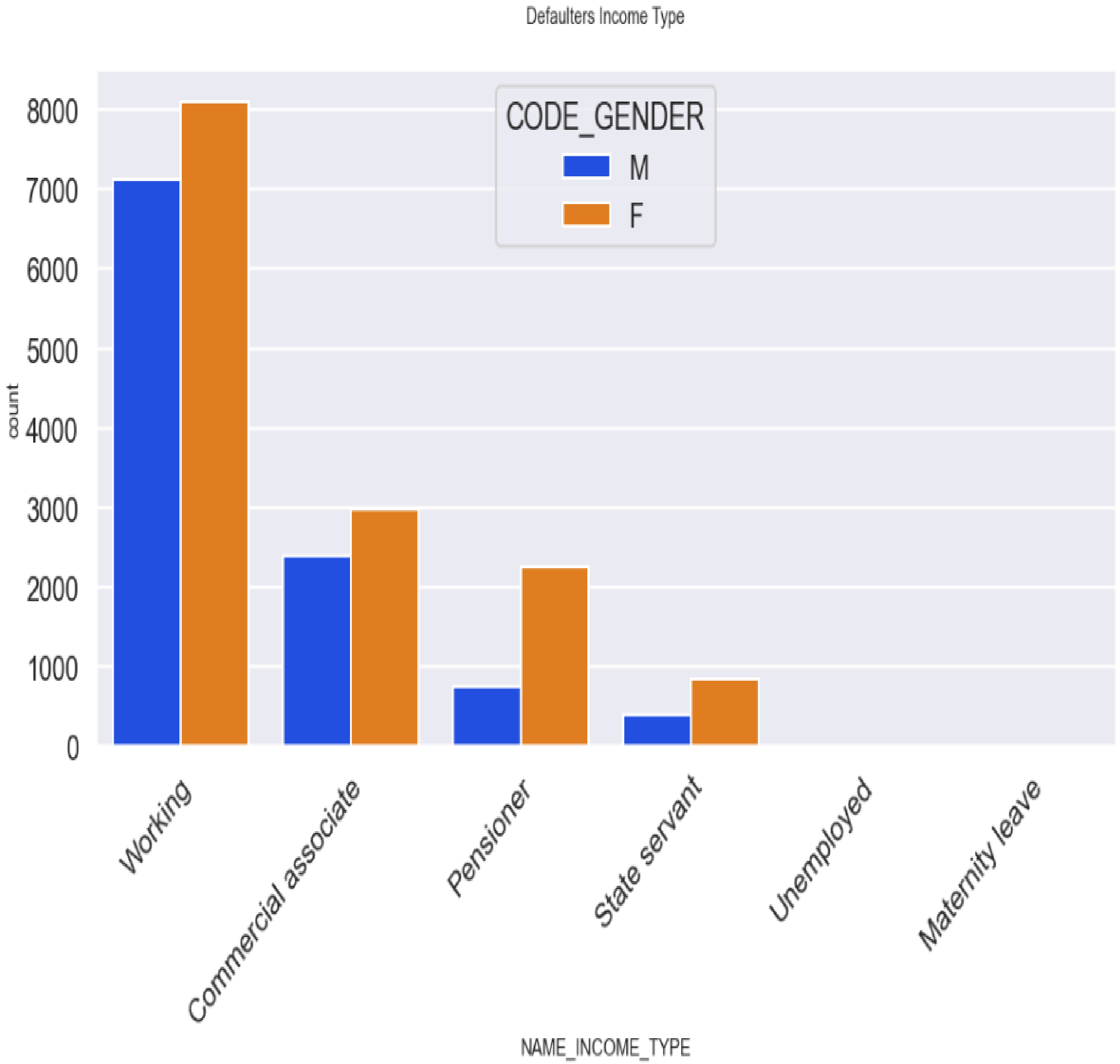
In [31]:
```
#replace XNA with F and checking count of each gender M/F
appl.loc[appl['CODE_GENDER']=='XNA','CODE_GENDER']='F'
appl['CODE_GENDER'].value_counts()
```

Out[31]:
```
F    202452
M    105059
Name: CODE_GENDER, dtype: int64
```

In [32]:
```
#plot the bar graph of CODE_GENDER
appl['CODE_GENDER'].value_counts(normalize=True).plot.bar(title='Code Gender data visual')
plt.xticks(rotation=35)
plt.show()
```

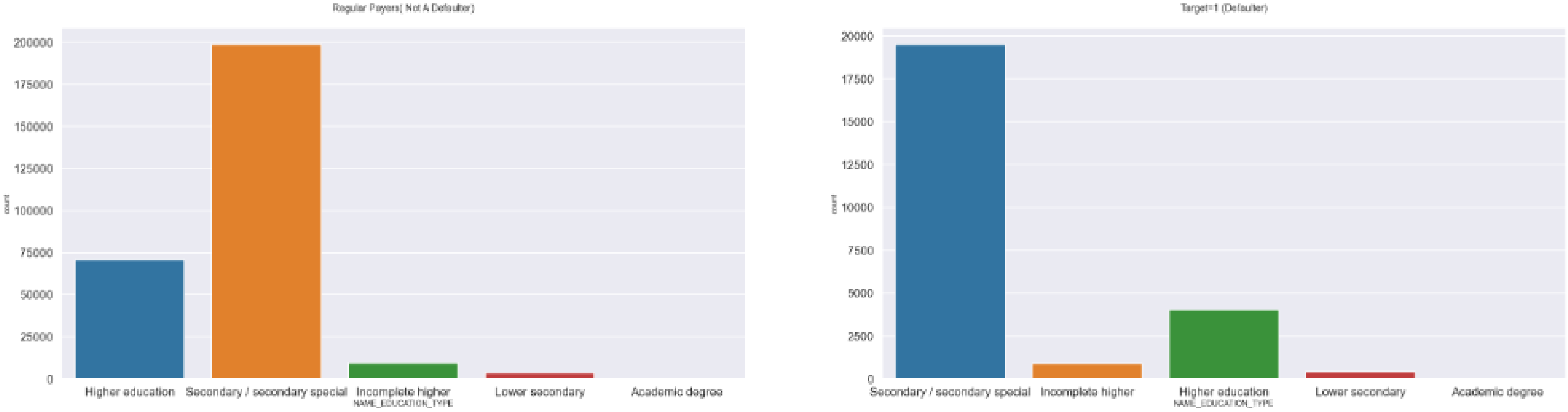


In [50]:
```
# Plotting for Income Type for defaulters
plotting(defaulters,col='NAME_INCOME_TYPE',title='Defaulters Income Type',hue='CODE_GENDER')
```



Step 6: Review Visualizations for conclusions and requirements

In [52]:
```
# Plotting for NAME_EDUCATION_TYPE for target0 and target1
fig, ax=plt.subplots(1,2,figsize=(50,12))
sns.countplot(payers['NAME_EDUCATION_TYPE'], ax=ax[0]).set_title('Regular Payers( Not A Defaulter)')
sns.countplot(defaulters['NAME_EDUCATION_TYPE'], ax=ax[1]).set_title('Target=1 (Defaulter)')
fig.show()
```



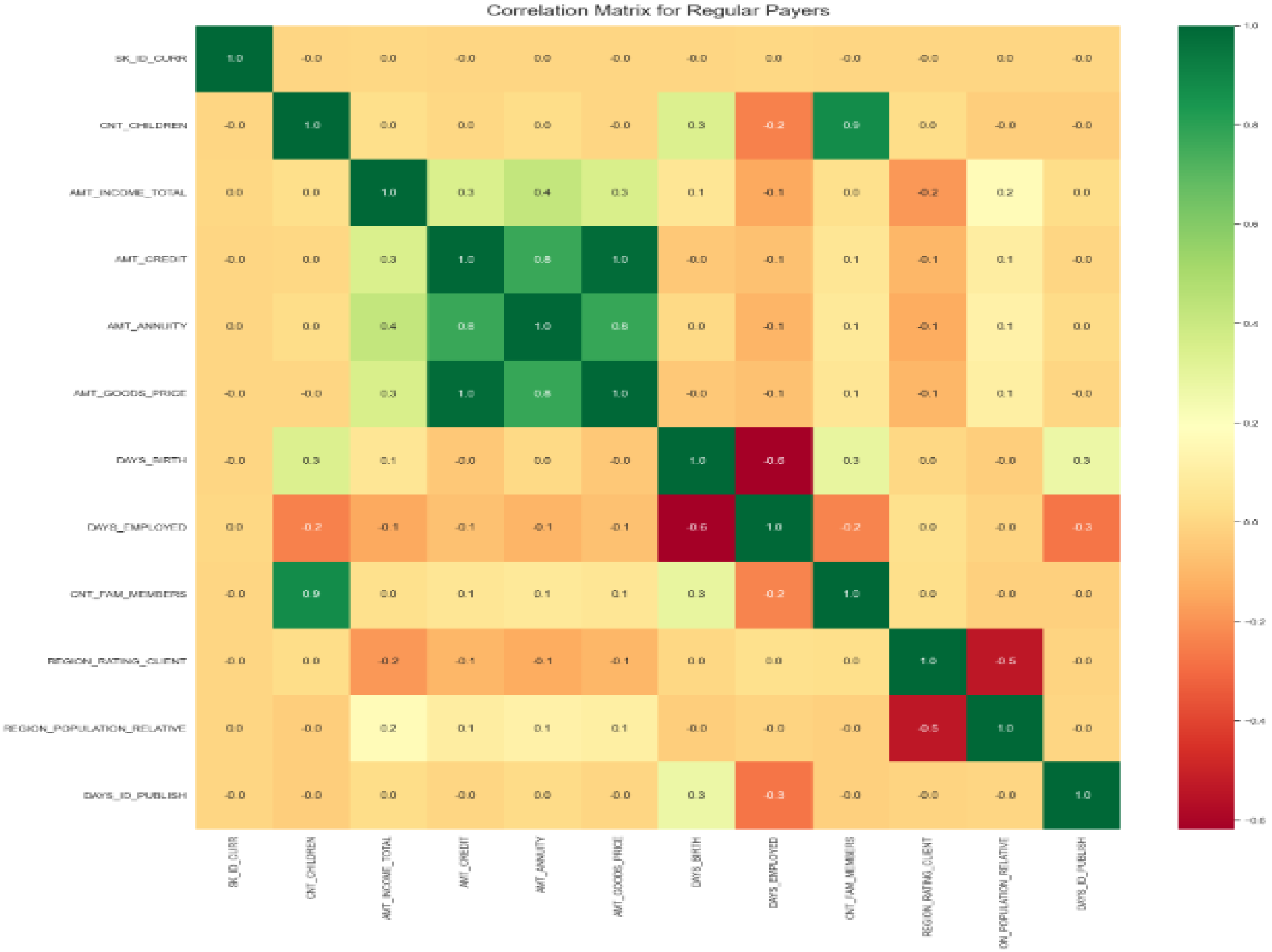
Conclusion:

people with secondary education has defaulted the most.

Analysing correlation for numerical columns for both Payers and Defaulters

In [57]:
```
#Plotting Correlation matrix for Regular Payer application data
d=payers[['SK_ID_CURR','CNT_CHILDREN','AMT_INCOME_TOTAL','AMT_CREDIT','AMT_ANNUITY',
                              'AMT_GOODS_PRICE','DAYS_BIRTH','DAYS_EMPLOYED','CNT_FAM_MEMBERS','REGION_RATING_CLIENT',
                              'REGION_POPULATION_RELATIVE','DAYS_ID_PUBLISH']]

plt.figure(figsize=(30,30))
sns.heatmap(d.corr(), fmt='.1f', cmap="RdYlGn", annot=True)
plt.title("Correlation Matrix for Regular Payers",fontsize=30, pad=20 )
plt.show()
```



Step 7: Conclusion

Write all the conclusions from all the graphs and summarize that into main points, as done in the workbook.

- The Heatmap shows points of highest and lowest values in correlation
- The Bar graph shows direct comparison
- The Pie chart may be useful in a scenario of distribution but for clearer picture, bars charts are used.

THANK YOU