Business understanding and Data understanding are very critical first couple of steps for any data science project. Read the information given below and also refer the data dictionary provided to get a good idea of the data and understanding.

Problem Statement

Home Credit Group expands unbanked population analytics from the perspective of Mortgage Backed Securities. Geographic Business Investment and Risk Credit Analytics.

The objective is to identify either spatially/geotemporal business in the mortgage loan. The mortgage bank would like to identify potential monetary/credit risk in the loan application. The company wants to understand the factors affecting the ability of a client to pay back the loan. All the details, from the time they are granting loans and it complete clients are settling mortgage loans to capture customer at a lower interest rate. The bank is strategizing to better market penetration and targeting new customers. A statistical model needs to be developed and implemented to predict the probability of a client defaulting on a loan. Also, the bank wants to know the factors dashboard which would generally give these interest from the agencies. You are hired by it consider the key factors and recommendations.

The dashboard must demonstrate relationships and trends for the key metrics as follows, number of loans, average interest loans, average interest rate, average credit score, family income to mortgage ratio comparison across different regions. The metrics are described in the data dictionary and are to be developed on these lines.

Business Objectives

Following are the Review Metrics that are to be home Home Credit, from 3rd of utilities, property loans

- Second Mortgage Households with a second mortgage statistics.
- Home Equity Loan Households with a home equity loan statistics.
- Debt Households with any type of debt statistics.
- Mortgage Costs Statistics regarding mortgage expenditure, home equity loans, utilities and property taxes.
- Home Owner Costs. Sum of utilities, property taxes statistics.
- Home Owner Costs per unit area the dashboard average monthly costs of utility Network.
- Gross Rent as Percent of Income Gross rent as the percent of income very interesting.
- High school Graduate High school graduate statistics.
- Population Demographics. Population demographic statistics.
- Age Demographics Age demographic statistics.
- Household Income Total income of people residing in the Household.
- Family Income Total income of people residing in the household.

Following are the Review Metrics that are to be home Home Credit, from 3rd of utilities, property loans

- Second Mortgage Households with a second mortgage statistics.
- Home Equity Loan Households with a home equity loan statistics.
- Debt Households with any type of debt statistics.
- Mortgage Costs Statistics regarding mortgage expenditure, home equity loans, utilities and property taxes.
- Home Owner Costs. Sum of utilities, property taxes statistics.
- Home Owner Costs per unit area the dashboard average monthly costs of utility Network.
- Gross Rent as Percent of Income Gross rent as the percent of income very interesting.
- High school Graduate High school graduate statistics.
- Population Demographics. Population demographic statistics.
- Age Demographics Age demographic statistics.
- Household Income Total income of people residing in the Household.
- Family Income Total income of people residing in the household.

```
!wget -q -O 'home_credit.csv' https://drive.google.com/uc?id=1Ldq7Ldq7Ldq7Ldq7Ldq7Ldq7Ldq7Ldq7L&export=download&confirm=t
!ls -lrtsh
!head -5 home_credit.csv
```

```
total 8K
4K -rw-r--r-- 1 root root 3.9K Sep 24 19:37 home_credit.csv
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,OWN_CAR_AGE,FLAG_MOBIL,FLAG_EMP_PHONE,FLAG_WORK_PHONE,FLAG_CONT_MOBILE,FLAG_PHONE,FLAG_EMAIL,OCCUPATION_TYPE,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_1,EXT_SOURCE_2,EXT_SOURCE_3,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,OBS_30_CNT,DEF_30_CNT,OBS_60_CNT,DEF_60_CNT,DAYS_DECISION,NAME_CONTRACT_STATUS,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_INSTALMENT,NAME_YIELD_GROUP,PRODUCT_COMBINATION,SK_ID_BUREAU,SK_ID_PREV,MONTHS_BALANCE
100002,1,Cash loans,M,N,Y,0,202500.0,406597.5,24700.5,351000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.018801,-9461,-637,-3648.0,-2120,,1,1,0,1,1,0,Laborers,1.0,2,2,WEDNESDAY,10,0,0,0,0,0,0,Business Entity Type 3,0.083037,0.262474,0.139376,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,reg oper account,block of flats,0.0149,Stone, brick,No,0.0,0.0,0.0,0.0,-873.0,Canceled,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,500.0,Furniture,6.0,middle,POS industry with interest,1.0,15705.0,100001
100003,0,Cash loans,F,N,N,0,270000.0,1293502.5,35698.5,1129500.0,Family,State servant,Higher education,Married,House / apartment,0.003541,-16765,-1188,-1186.0,-291,,1,1,0,1,1,0,Core staff,2.0,1,1,MONDAY,11,0,0,0,0,0,0,School,0.311267,0.622246,,0.0959,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0959,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0960,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-1491.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100002
100004,0,Revolving loans,M,Y,Y,0,67500.0,135000.0,6750.0,135000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.010032,-19046,-225,-4260.0,-2531,26.0,1,1,0,1,1,0,Laborers,1.0,2,2,WEDNESDAY,9,0,0,0,0,0,0,Business Entity Type 3,,0.555912,0.729567,0.0288,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0288,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0290,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-1026.0,Canceled,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100003
100006,0,Cash loans,F,N,Y,0,135000.0,312682.5,29686.5,297000.0,Unaccompanied,Working,Secondary / secondary special,Civil marriage,House / apartment,0.008019,-19005,-3039,-9833.0,-2437,,1,1,0,1,1,0,Sales staff,2.0,2,2,WEDNESDAY,17,0,0,0,0,0,0,Business Entity Type 3,,0.650442,,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,block of flats,,,No,0.0,0.0,0.0,0.0,-1400.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,6.0,middle,POS industry with interest,1.0,15705.0,100004
100007,0,Cash loans,M,N,Y,0,121500.0,513000.0,21865.5,513000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.028663,-19932,-3038,-4311.0,-3458,,1,1,0,1,1,0,Laborers,1.0,2,2,THURSDAY,11,0,0,0,0,0,0,Business Entity Type 3,0.322738,0.354021,0.621226,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-515.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,500.0,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100005
```

```
!wc home_credit.csv
!head home_credit.csv
```

```
  307511 3828753 683854148 home_credit.csv
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,OWN_CAR_AGE,FLAG_MOBIL,FLAG_EMP_PHONE,FLAG_WORK_PHONE,FLAG_CONT_MOBILE,FLAG_PHONE,FLAG_EMAIL,OCCUPATION_TYPE,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_1,EXT_SOURCE_2,EXT_SOURCE_3,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,OBS_30_CNT,DEF_30_CNT,OBS_60_CNT,DEF_60_CNT,DAYS_DECISION,NAME_CONTRACT_STATUS,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_INSTALMENT,NAME_YIELD_GROUP,PRODUCT_COMBINATION,SK_ID_BUREAU,SK_ID_PREV,MONTHS_BALANCE
100002,1,Cash loans,M,N,Y,0,202500.0,406597.5,24700.5,351000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.018801,-9461,-637,-3648.0,-2120,,1,1,0,1,1,0,Laborers,1.0,2,2,WEDNESDAY,10,0,0,0,0,0,0,Business Entity Type 3,0.083037,0.262474,0.139376,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,0.0247,0.0369,0.9722,0.6192,0.0143,0.0000,0.0690,0.0833,0.1250,0.0369,0.0202,0.0190,0.0000,0.0000,reg oper account,block of flats,0.0149,Stone, brick,No,0.0,0.0,0.0,0.0,-873.0,Canceled,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,500.0,Furniture,6.0,middle,POS industry with interest,1.0,15705.0,100001
100003,0,Cash loans,F,N,N,0,270000.0,1293502.5,35698.5,1129500.0,Family,State servant,Higher education,Married,House / apartment,0.003541,-16765,-1188,-1186.0,-291,,1,1,0,1,1,0,Core staff,2.0,1,1,MONDAY,11,0,0,0,0,0,0,School,0.311267,0.622246,,0.0959,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0959,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0960,0.0529,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-1491.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100004
100004,0,Revolving loans,M,Y,Y,0,67500.0,135000.0,6750.0,135000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.010032,-19046,-225,-4260.0,-2531,26.0,1,1,0,1,1,0,Laborers,1.0,2,2,WEDNESDAY,9,0,0,0,0,0,0,Business Entity Type 3,,0.555912,0.729567,0.0288,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0288,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0290,0.0287,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-1026.0,Canceled,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100005
100006,0,Cash loans,F,N,Y,0,135000.0,312682.5,29686.5,297000.0,Unaccompanied,Working,Secondary / secondary special,Civil marriage,House / apartment,0.008019,-19005,-3039,-9833.0,-2437,,1,1,0,1,1,0,Sales staff,2.0,2,2,WEDNESDAY,17,0,0,0,0,0,0,Business Entity Type 3,,0.650442,,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,,block of flats,,,No,0.0,0.0,0.0,0.0,-1400.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,,Furniture,6.0,middle,POS industry with interest,1.0,15705.0,100006
100007,0,Cash loans,M,N,Y,0,121500.0,513000.0,21865.5,513000.0,Unaccompanied,Working,Secondary / secondary special,Single / not married,House / apartment,0.028663,-19932,-3038,-4311.0,-3458,,1,1,0,1,1,0,Laborers,1.0,2,2,THURSDAY,11,0,0,0,0,0,0,Business Entity Type 3,0.322738,0.354021,0.621226,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,0.0495,0.0508,0.9851,0.7960,0.0605,0.0800,0.0417,0.0333,0.0013,0.0419,0.0767,0.0732,0.0000,0.0054,reg oper account,block of flats,0.0714,Stone, brick,No,0.0,0.0,0.0,0.0,-515.0,Approved,Cash through the bank,XAP,,Repeater,Mobile,POS,XNA,Stone,500.0,Furniture,12.0,middle,POS industry with interest,1.0,15705.0,100008
```

EDA (Exploratory Data Analysis)

```
!apt-get -qq install -sf libcairo2-dev libjpeg-dev libgif-dev
!pip -qq install -U cartopy
```

```
!pip -qq install -U pandas
!pip -qq install -U numpy
!pip -qq install -U matplotlib
!pip -qq install -U seaborn
```

```
!ls -ltrsh home_credit.csv
```

```
4K -rw-r--r-- 1 root root 3.9K Sep 24 19:37 home_credit.csv
```

```
import pandas as pd
pd.set_option('display.max_columns', 500)
df = pd.read_csv('home_credit.csv')
df.shape
```

```
(307511, 122)
```

```
df.sample(10)
```

| | 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 | OWN_CAR_AGE | FLAG_MOBIL | FLAG_EMP_PHONE | FLAG_WORK_PHONE | FLAG_CONT_MOBILE | FLAG_PHONE | FLAG_EMAIL | OCCUPATION_TYPE | 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_1 | EXT_SOURCE_2 | EXT_SOURCE_3 | 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 | OBS_30_CNT | DEF_30_CNT | OBS_60_CNT | DEF_60_CNT | DAYS_DECISION | NAME_CONTRACT_STATUS | 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_INSTALMENT | NAME_YIELD_GROUP | PRODUCT_COMBINATION | SK_ID_BUREAU | SK_ID_PREV | MONTHS_BALANCE |
|---|---|---

```
CREDIT_TERM_Cash loans    0.000000
CREDIT_TERM_Revolving loans    0.000000
Name: TARGET, dtype: float64
```

- Cash loans for population is a new variability by selecting appropriate class interval so that for no of observations don't exceed
- the frequency of repayers.

```
# Cash loans
df_train[df_train.CREDIT_TERM_Cash loans].TARGET.value_counts(normalize=True)*100
```

```
# Revolving loans
df_train[df_train.CREDIT_TERM_Revolving loans].TARGET.value_counts(normalize=True)*100
```

```
# Total cash - Cash, Total revolving - Revolving
df_train[df_train.CREDIT_TERM_Total cash].TARGET.value_counts(normalize=True)*100
df_train[df_train.CREDIT_TERM_Total revolving].TARGET.value_counts(normalize=True)*100
```

```
CREDIT_TERM_Total cash    100.0
Name: TARGET, dtype: float64
Total cash    100.0
Name: TARGET, dtype: float64
Total revolving    100.0
Name: TARGET, dtype: float64
```

```
df_temp = df[['Transaction', 'Customer', 'Product', 'Price', 'Quantity', 'Total', 'Payment', 'Date',
              'Day', 'Month', 'Year', 'Hour', 'Minute', 'Second', 'Day_Name', 'Month_Name',
              'Is_Weekend', 'Quarter', 'Month_Year', 'Hour_Bucket', 'Revenue_Bucket']]
df_temp.shape
```

```
(1,000,000, 21)
```

```
df_temp.duplicated().sum()
```

```
0
```

```
df_temp.isnull().sum()
```

```
Transaction    0
Customer       0
Product        0
Price          0
Quantity       0
Total          0
Payment        0
Date           0
Day            0
Month          0
Year           0
Hour           0
Minute         0
Second         0
Day_Name       0
Month_Name     0
Is_Weekend     0
Quarter        0
Month_Year     0
Hour_Bucket    0
Revenue_Bucket 0
dtype: int64
```

```
df_temp.dtypes.value_counts
```

```
int64    11
object    8
bool      2
dtype: int64
```

```
df_temp.nunique()
```

```
Transaction    1,000,000
Customer          10,000
Product               10
Price                 10
Quantity              10
Total                 91
Payment               10
Date                  30
Day                   30
Month                 12
Year                   1
Hour                  24
Minute                60
Second                60
Day_Name               7
Month_Name            12
Is_Weekend             2
Quarter                4
Month_Year            12
Hour_Bucket            4
Revenue_Bucket         3
dtype: int64
```

Nation Level

```
df_temp.describe(include=['object', 'bool'])
```

```
df_temp.describe(include=['int', 'float'])
```

```
df_temp[df_temp.duplicated()]
```

```
df_temp[df_temp.duplicated(keep=False)].sort_values('Transaction')
```

```
print("Minimum: %d" %df_temp['Transaction'].min())
print("Maximum: %d" %df_temp['Transaction'].max())
print("Mean: %f" %df_temp['Transaction'].mean())
print("First quartile: %f" %df_temp['Transaction'].quantile(0.25))
print("Median: %f" %df_temp['Transaction'].median())
print("Third quartile: %f" %df_temp['Transaction'].quantile(0.75))
```

```
Minimum: 1
Maximum: 1000000
Mean: 500000.500000
First quartile: 250000.750000
Median: 500000.500000
Third quartile: 749999.250000
```

State Level

```
df_temp[df_temp['Transaction'] == 1]
```

```
df_temp[df_temp['Transaction'] == 1000000]
```

```
df_temp.iloc[0, [0, 1]], df_temp.iloc[-1, [0, 1]]
```

```
(Transaction    1
Customer        1
dtype: int64,
Transaction    1000000
Customer         10000
dtype: int64)
```

```
for i in [1, 1000000]:
    print("%d" %i)
    print("Customer: %d" %df_temp[df_temp['Transaction'] == i]['Customer'].nunique())
    print("Product: %d" %df_temp[df_temp['Transaction'] == i]['Product'].nunique())
    print("Price: %d" %df_temp[df_temp['Transaction'] == i]['Price'].nunique())
    print("Quantity: %d" %df_temp[df_temp['Transaction'] == i]['Quantity'].nunique())
    print("Total: %d" %df_temp[df_temp['Transaction'] == i]['Total'].nunique())
    print("Payment: %d" %df_temp[df_temp['Transaction'] == i]['Payment'].nunique())
    print("Date: %d" %df_temp[df_temp['Transaction'] == i]['Date'].nunique())
    print("Day: %d" %df_temp[df_temp['Transaction'] == i]['Day'].nunique())
    print("Month: %d" %df_temp[df_temp['Transaction'] == i]['Month'].nunique())
    print("Year: %d" %df_temp[df_temp['Transaction'] == i]['Year'].nunique())
    print("Hour: %d" %df_temp[df_temp['Transaction'] == i]['Hour'].nunique())
    print("Minute: %d" %df_temp[df_temp['Transaction'] == i]['Minute'].nunique())
    print("Second: %d" %df_temp[df_temp['Transaction'] == i]['Second'].nunique())
    print("Day_Name: %d" %df_temp[df_temp['Transaction'] == i]['Day_Name'].nunique())
    print("Month_Name: %d" %df_temp[df_temp['Transaction'] == i]['Month_Name'].nunique())
    print("Is_Weekend: %d" %df_temp[df_temp['Transaction'] == i]['Is_Weekend'].nunique())
    print("Quarter: %d" %df_temp[df_temp['Transaction'] == i]['Quarter'].nunique())
    print("Month_Year: %d" %df_temp[df_temp['Transaction'] == i]['Month_Year'].nunique())
    print("Hour_Bucket: %d" %df_temp[df_temp['Transaction'] == i]['Hour_Bucket'].nunique())
    print("Revenue_Bucket: %d" %df_temp[df_temp['Transaction'] == i]['Revenue_Bucket'].nunique())
    print("\n")
```

```
1
Customer: 1
Product: 1
Price: 1
Quantity: 1
Total: 1
Payment: 1
Date: 1
Day: 1
Month: 1
Year: 1
Hour: 1
Minute: 1
Second: 1
Day_Name: 1
Month_Name: 1
Is_Weekend: 1
Quarter: 1
Month_Year: 1
Hour_Bucket: 1
Revenue_Bucket: 1


1000000
Customer: 10000
Product: 10
Price: 10
Quantity: 10
Total: 91
Payment: 10
Date: 30
Day: 30
Month: 12
Year: 1
Hour: 24
Minute: 60
Second: 60
Day_Name: 7
Month_Name: 12
Is_Weekend: 2
Quarter: 4
Month_Year: 12
Hour_Bucket: 4
Revenue_Bucket: 3
```

```
df_temp.groupby('Month_Year').count
```

```
<bound_method DataFrame.count of
                Transaction  Customer  Product  Price  Quantity  Total  \
Month_Year
2023-01-01            7000      7000     7000   7000      7000   7000
2023-02-01            7000      7000     7000   7000      7000   7000
2023-03-01            7000      7000     7000   7000      7000   7000
2023-04-01            7000      7000     7000   7000      7000   7000
2023-05-01            7000      7000     7000   7000      7000   7000
2023-06-01            7000      7000     7000   7000      7000   7000
2023-07-01            7000      7000     7000   7000      7000   7000
2023-08-01            7000      7000     7000   7000      7000   7000
2023-09-01            7000      7000     7000   7000      7000   7000
2023-10-01            7000      7000     7000   7000      7000   7000
2023-11-01            7000      7000     7000   7000      7000   7000
2023-12-01            7000      7000     7000   7000      7000   7000
            Payment  Date  Day  Month  Year  Hour  Minute  Second  Day_Name  \
Month_Year
2023-01-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-02-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-03-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-04-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-05-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-06-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-07-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-08-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-09-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-10-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-11-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
2023-12-01     7000  7000  7000   7000  7000  7000    7000    7000      7000
            Month_Name  Is_Weekend  Quarter  Month_Year  Hour_Bucket  \
Month_Year
2023-01-01        7000        7000     7000        7000         7000
2023-02-01        7000        7000     7000        7000         7000
2023-03-01        7000        7000     7000        7000         7000
2023-04-01        7000        7000     7000        7000         7000
2023-05-01        7000        7000     7000        7000         7000
2023-06-01        7000        7000     7000        7000         7000
2023-07-01        7000        7000     7000        7000         7000
2023-08-01        7000        7000     7000        7000         7000
2023-09-01        7000        7000     7000        7000         7000
2023-10-01        7000        7000     7000        7000         7000
2023-11-01        7000        7000     7000        7000         7000
2023-12-01        7000        7000     7000        7000         7000
            Revenue_Bucket
Month_Year
2023-01-01            7000
2023-02-01            7000
2023-03-01            7000
2023-04-01            7000
2023-05-01            7000
2023-06-01            7000
2023-07-01            7000
2023-08-01            7000
2023-09-01            7000
2023-10-01            7000
2023-11-01            7000
2023-12-01            7000>
```

```
df_temp.groupby('Month_Year').count
```

```
Transaction    Customer  Product  Price  Quantity  Total  Payment  Date  Day  Month  Year  Hour  Minute  Second  Day_Name  Month_Name  Is_Weekend  Quarter  Month_Year  Hour_Bucket  Revenue_Bucket
Month_Year
2023-01-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-02-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-03-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-04-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-05-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-06-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-07-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-08-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-09-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-10-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-11-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
2023-12-01     7000      7000     7000   7000      7000   7000     7000  7000 7000   7000  7000  7000    7000    7000    7000      7000        7000        7000     7000        7000         7000         7000
```

```
df[df.duplicated(subset=['Customer', 'Product', 'Price', 'Quantity', 'Total', 'Payment', 'Date',
                         'Day', 'Month', 'Year', 'Hour', 'Minute', 'Second', 'Day_Name',
                         'Month_Name', 'Is_Weekend', 'Quarter', 'Month_Year', 'Hour_Bucket',
                         'Revenue_Bucket'])].shape
```

```
(990,000, 21)
```

```
df[df.duplicated(subset=['Customer', 'Product', 'Price', 'Quantity', 'Total', 'Payment', 'Date',
                         'Day', 'Month', 'Year', 'Hour', 'Minute', 'Second', 'Day_Name',
                         'Month_Name', 'Is_Weekend', 'Quarter', 'Month_Year', 'Hour_Bucket',
                         'Revenue_Bucket'])].sample(5)
```

	Transaction	Customer	Product	Price	Quantity	Total	Payment	Date	Day	Month	Year	Hour	Minute	Second	Day_Name	Month_Name	Is_Weekend	Quarter	Month_Year	Hour_Bucket	Revenue_Bucket
1000000	1000000	10000	10	10	10	91	10	30	30	12	2023	23	59	59	Saturday	December	False	4	2023-12-01	4	3