

```
In [245... import pandas as pd
import numpy as np
import os
from datetime import datetime, timedelta
import datetime
```

Exploring and Cleaning the Data

```
In [246... os.getcwd()
```

```
Out[246]: 'C:\\Users\\tural\\OneDrive\\Desktop\\Study Materials\\Datasets'
```

```
In [247... os.chdir("C:\\Users\\tural\\OneDrive\\Desktop\\Study Materials\\Datasets")
df = pd.read_csv("application_record.csv")
df.head()
```

```
Out[247]:
```

	ID	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALTY	CNT_CHILDREN	AMT_INCOME_TOTAL
0	5008804	M	Y	Y	0	42750
1	5008805	M	Y	Y	0	42750
2	5008806	M	Y	Y	0	11250
3	5008808	F	N	Y	0	27000
4	5008809	F	N	Y	0	27000

```
In [248... #Lets first change the column names that we can work easily
df.columns = df.columns.str.lower()
df.columns = df.columns.str.replace('name_', '', regex=True)
df.columns = df.columns.str.replace('flag_', '', regex=True)
df.columns = df.columns.str.replace('amt_', '', regex=True)
df.columns = df.columns.str.replace('cnt_', '', regex=True)
df.columns = df.columns.str.replace('code_', '', regex=True)
df.rename(columns = {'days_birth': 'birthdate', 'days_employed': 'employed_since', 'mont
```

```
In [249... #Lets get some info on dataframe
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 438557 entries, 0 to 438556
Data columns (total 18 columns):
#   Column                Non-Null Count  Dtype
---  -
0   id                    438557 non-null  int64
1   gender                438557 non-null  object
2   own_car               438557 non-null  object
3   own_realty            438557 non-null  object
4   children              438557 non-null  int64
5   income_total          438557 non-null  float64
6   income_type           438557 non-null  object
7   education_type        438557 non-null  object
8   family_status         438557 non-null  object
9   housing_type          438557 non-null  object
10  birthdate             438557 non-null  int64
11  employed_since        438557 non-null  int64
12  mobil                 438557 non-null  int64
13  work_phone            438557 non-null  int64
14  phone                 438557 non-null  int64
15  email                 438557 non-null  int64
16  occupation_type       304354 non-null  object
17  fam_members           438557 non-null  float64
dtypes: float64(2), int64(8), object(8)
memory usage: 60.2+ MB
```

```
In [250...] #Now Lets check the number of rows and also the unique customers to see if we should c
df.id.nunique()
```

```
Out[250]: 438510
```

```
In [251...] #Apparently there are some duplicates, Lets clean them. As we have some different id k
df = df.drop_duplicates(subset='id', keep="last")
df = df.set_index('id')
df = df.drop_duplicates(keep='first')
```

```
In [252...] #Now we will create a function which will retrieve the birthday based on the given num
def birth(total_days):
    today = datetime.date.today()
    birthday = (today + timedelta(days=total_days)).strftime('%Y-%m-%d')
    return birthday

df['birthdate']=df['birthdate'].apply(Date_of_Birth)
```

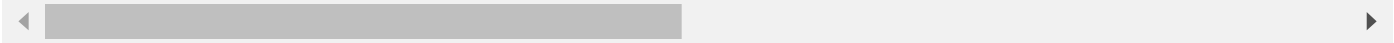
```
In [253...] #We will do the same for finding the employment dates
def employed(total_days):
    today = datetime.date.today()
    employed_date = (today + datetime.timedelta(days=total_days)).strftime('%Y-%m-%d')

df['employed_since']=df['employed_since'].apply(Date_of_Birth)
```

```
In [254...] #Lets convert the some of the floats into integers as we are sure that they cannot be
df['children'] = df['children'].astype(int)
df['fam_members'] = df['fam_members'].astype(int)
df.head()
```

Out[254]:

id	gender	own_car	own_realty	children	income_total	income_type	education_type	family_
5008804	M	Y	Y	0	427500.0	Working	Higher education	Civil m
5008806	M	Y	Y	0	112500.0	Working	Secondary / secondary special	M
5008808	F	N	Y	0	270000.0	Commercial associate	Secondary / secondary special	Singl n
5008812	F	N	Y	0	283500.0	Pensioner	Higher education	Sep
5008815	M	Y	Y	0	270000.0	Working	Higher education	M



Analysis and ML model is in progress...