INPC MACHINE LEARNING

In [11]:
```
from sklearn.datasets import load_boston
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor

#import boston_valuation as val

%matplotlib inline
```

In [12]:
```
inpc_data=pd.read_csv("final_inpc.csv",index_col=0)
```

In [13]:
```
inpc_data.head(5)
```

Out[13]:

	person_id	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
0	1	M	NaN	NaN	NaN
1	10	F	NaN	NaN	NaN
2	100	F	NaN	NaN	NaN
3	1000	F	61.046	NaN	NaN
4	10000	M	46.767	NaN	NaN

In [14]:
```
inpc_data.tail(5)
```

Out[14]:

	person_id	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
1060971	999995	F	NaN	NaN	NaN
1060972	999996	M	NaN	NaN	NaN
1060973	999997	M	NaN	NaN	NaN
1060974	999998	M	NaN	NaN	NaN
1060975	999999	F	NaN	NaN	NaN

In [19]:
```
columns = ['person_id']
inpc_data.drop(columns, inplace=True, axis=1)
```

In [20]:
```
inpc_data.shape
```

Out[20]:
```
(1060976, 4)
```

In [21]:
```
inpc_data.count()
```

Out[21]:
```
gender             1060976
Age_T2D_First       301398
Age_AD_First         10580
T2D_OR_AD_FIRST       8044
dtype: int64
```

In [22]:
```
pd.isnull(inpc_data)
```

Out[22]:

	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
0	False	True	True	True
1	False	True	True	True
2	False	True	True	True
3	False	False	True	True
4	False	False	True	True
...	...	...	...	...
1060971	False	True	True	True
1060972	False	True	True	True
1060973	False	True	True	True
1060974	False	True	True	True
1060975	False	True	True	True

1060976 rows × 4 columns

In [23]:
```
inpc_data.isnull().sum()
```

Out[23]:
```
gender                   0
Age_T2D_First       759578
Age_AD_First       1050396
T2D_OR_AD_FIRST    1052932
dtype: int64
```

In [24]:
```
inpc_data=inpc_data.fillna(" ")
```

In [25]:
```
inpc_data.isnull().sum()
```

Out[25]:
```
gender             0
Age_T2D_First      0
Age_AD_First       0
T2D_OR_AD_FIRST    0
dtype: int64
```

In [28]:
```
inpc_data.count()
```

Out[28]:
```
gender             1060976
Age_T2D_First      1060976
Age_AD_First       1060976
T2D_OR_AD_FIRST    1060976
dtype: int64
```

In [30]:
```
inpc_data.head(18)
```

Out[30]:

	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
0	M			
1	F			
2	F			
3	F	61.046		
4	M	46.767		
5	F	60.879		
6	M			
7	F			
8	F			
9	M			
10	M			
11	F			
12	F			
13	F			
14	F			
15	M			
16	F			
17	M			

In [31]:
```
inpc_data.tail(18)
```

Out[31]:

	gender	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
1060958	M			
1060959	F	70.581		
1060960	M			
1060961	M			
1060962	M			
1060963	M			
1060964	M			
1060965	M			
1060966	M			
1060967	F			
1060968	F			
1060969	M			
1060970	F			
1060971	F			
1060972	M			
1060973	M			
1060974	M			
1060975	F			

In [32]:
```
inpc_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1060976 entries, 0 to 1060975
Data columns (total 4 columns):
 #   Column           Non-Null Count    Dtype 
---  ------           --------------    ----- 
 0   gender           1060976 non-null  object
 1   Age_T2D_First    1060976 non-null  object
 2   Age_AD_First     1060976 non-null  object
 3   T2D_OR_AD_FIRST  1060976 non-null  object
dtypes: object(4)
memory usage: 40.5+ MB
```

In [33]:
```
# Import label encoder
from sklearn import preprocessing

# label_encoder object knows how to understand word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in columns
inpc_data['gender']= label_encoder.fit_transform(inpc_data['gender'])

inpc_data['gender'].unique()
```

Out[33]:
```
array([1, 0, 2])
```

In [34]:
```
inpc_data.columns
```

Out[34]:
```
Index(['gender', 'Age_T2D_First', 'Age_AD_First', 'T2D_OR_AD_FIRST'], dtype
='object')
```

In [35]:
```
inpc_data["Age_T2D_First"] = pd.to_numeric(inpc_data.Age_T2D_First,errors='co
```

In [36]:
```
inpc_data["Age_AD_First"] = pd.to_numeric(inpc_data.Age_AD_First,errors='coer
```

In [37]:
```
inpc_data["T2D_OR_AD_FIRST"] = pd.to_numeric(inpc_data.T2D_OR_AD_FIRST,errors
```

In [38]:
```
inpc_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1060976 entries, 0 to 1060975
Data columns (total 4 columns):
 #   Column           Non-Null Count    Dtype
---  ------           --------------    -----
 0   gender           1060976 non-null  int32
 1   Age_T2D_First    301398 non-null   float64
 2   Age_AD_First     10580 non-null    float64
 3   T2D_OR_AD_FIRST  8044 non-null     float64
dtypes: float64(3), int32(1)
memory usage: 36.4 MB
```

In [43]:
```
inpc_data1=inpc_data[['Age_T2D_First', 'Age_AD_First', 'T2D_OR_AD_FIRST']]
```

In [44]:
```
inpc_data1.describe()
```

Out[44]:

	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
count	301398.000000	10580.000000	8044.000000
mean	62.378507	50.676195	-0.291610
std	13.971256	16.835661	1.034345
min	1.241000	5.303000	-4.312000
25%	53.293000	38.123250	-0.890000
50%	63.260000	51.387500	-0.126000
75%	72.257000	63.203500	0.167000
max	101.175000	98.531000	4.370000

In [45]:
```
inpc_data1.corr() # Pearson Correlation Coefficients
```

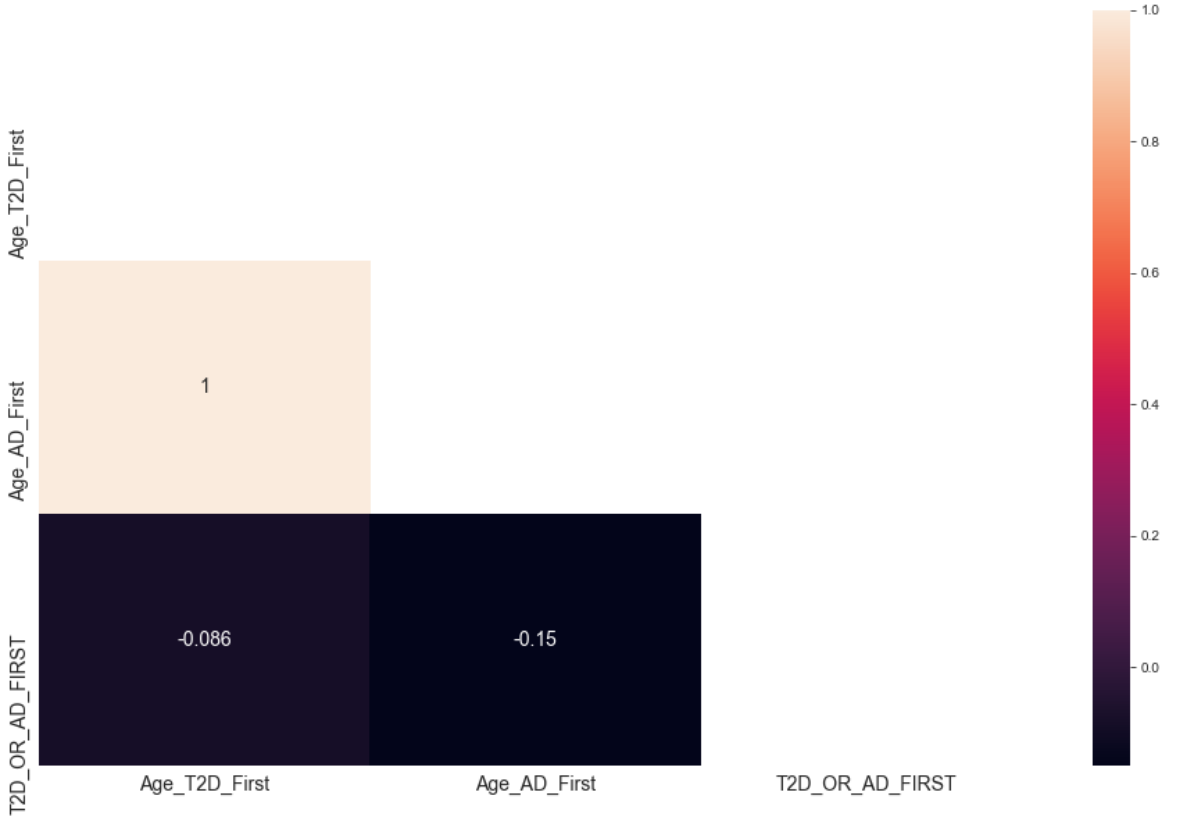
Out[45]:

	Age_T2D_First	Age_AD_First	T2D_OR_AD_FIRST
Age_T2D_First	1.000000	0.997952	-0.086182
Age_AD_First	0.997952	1.000000	-0.149735
T2D_OR_AD_FIRST	-0.086182	-0.149735	1.000000

In [46]:
```
mask = np.zeros_like(inpc_data1.corr())
triangle_indices = np.triu_indices_from(mask)
mask[triangle_indices] = True
mask
```

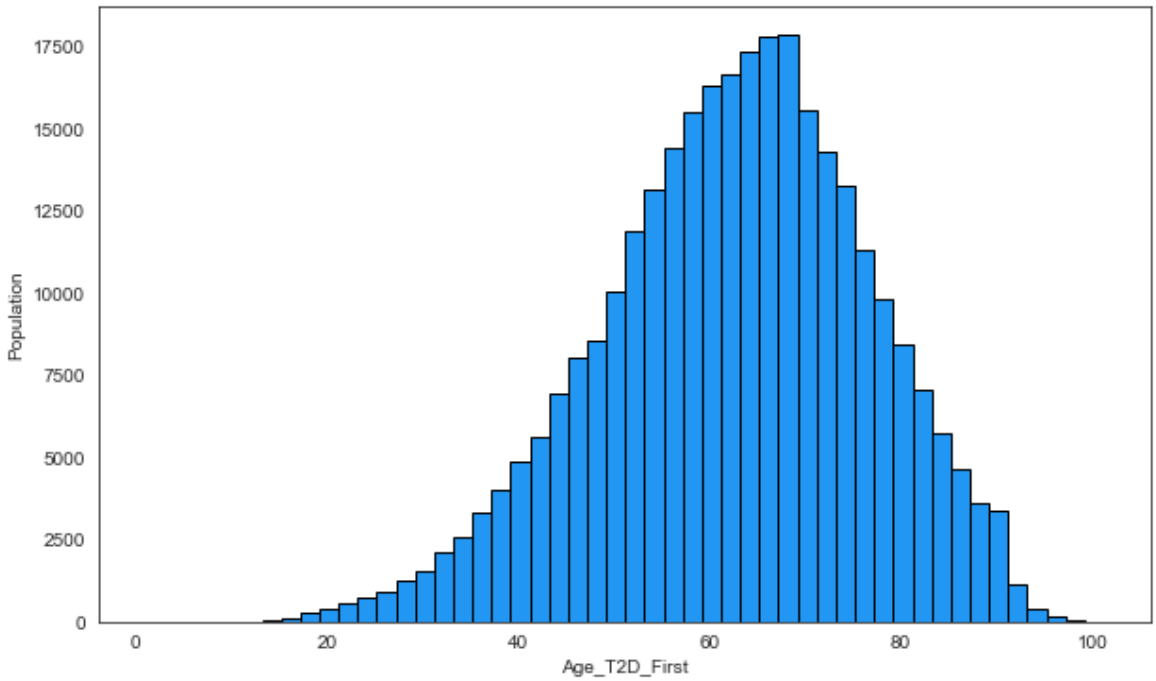
Out[46]:
```
array([[1., 1., 1.],
       [0., 1., 1.],
       [0., 0., 1.]])
```

In [71]:
```
plt.figure(figsize=(16,10))
sns.heatmap(inpc_data1.corr(), mask=mask, annot=True, annot_kws={"size": 14})
sns.set_style('white')
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
plt.show()
```



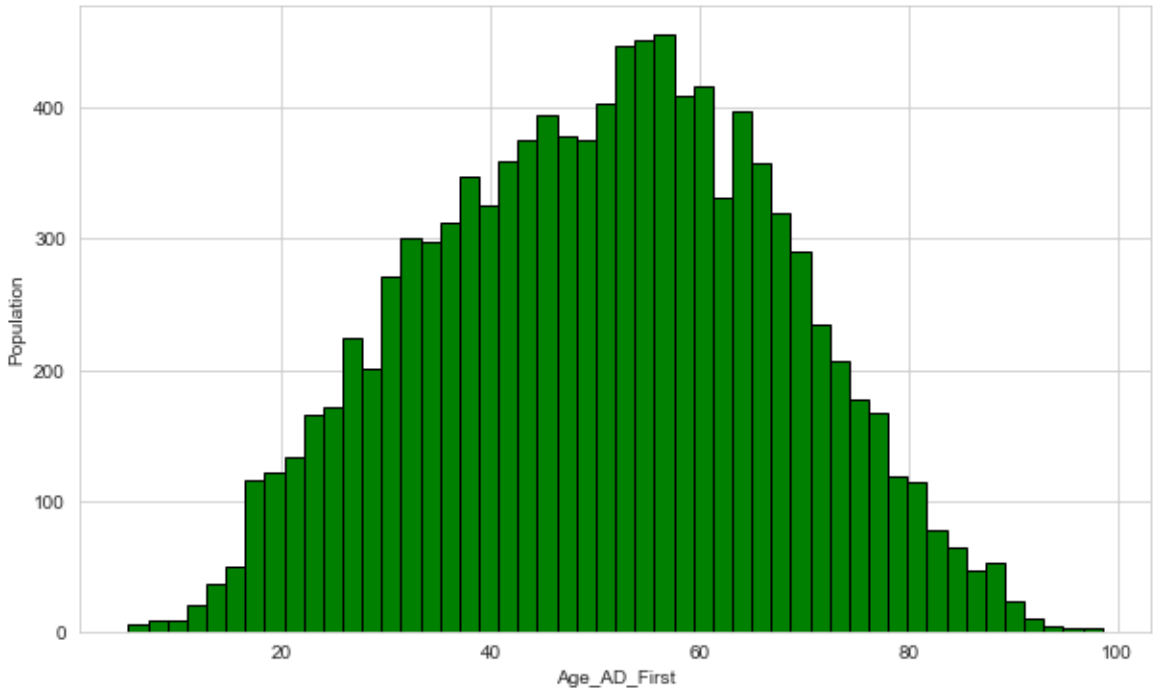
Visualising Data - Histograms, Distributions and Bar Charts

In [72]:
```
plt.figure(figsize=(10, 6))
plt.hist(inpc_data['Age_T2D_First'], bins=50, ec='black', color='#2196f3')
plt.xlabel('Age_T2D_First')
plt.ylabel('Population')
plt.show()
```



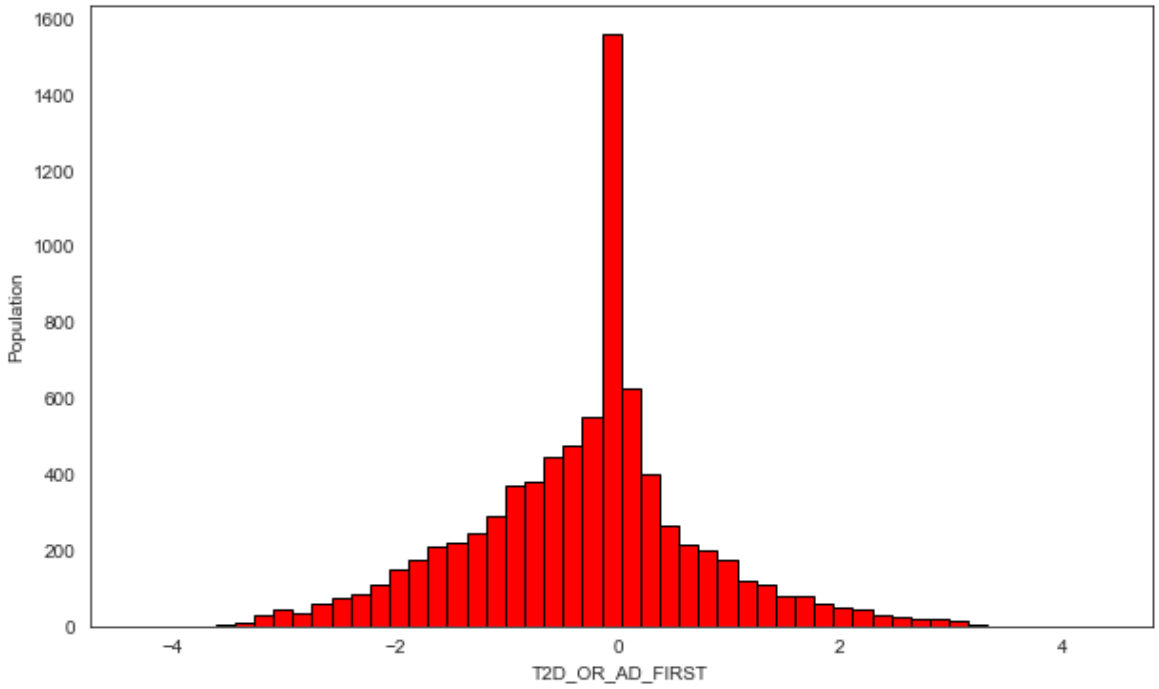
In [70]:
```
plt.figure(figsize=(10, 6))
plt.hist(inpc_data['Age_AD_First'], bins=50, ec='black', color='green')
plt.xlabel('Age_AD_First')
plt.ylabel('Population')
plt.show()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\numpy\lib\histograms.py:839: Run
timeWarning: invalid value encountered in greater_equal
  keep = (tmp_a >= first_edge)
C:\Users\mayam\anaconda3\lib\site-packages\numpy\lib\histograms.py:840: Run
timeWarning: invalid value encountered in less_equal
  keep &= (tmp_a <= last_edge)
```

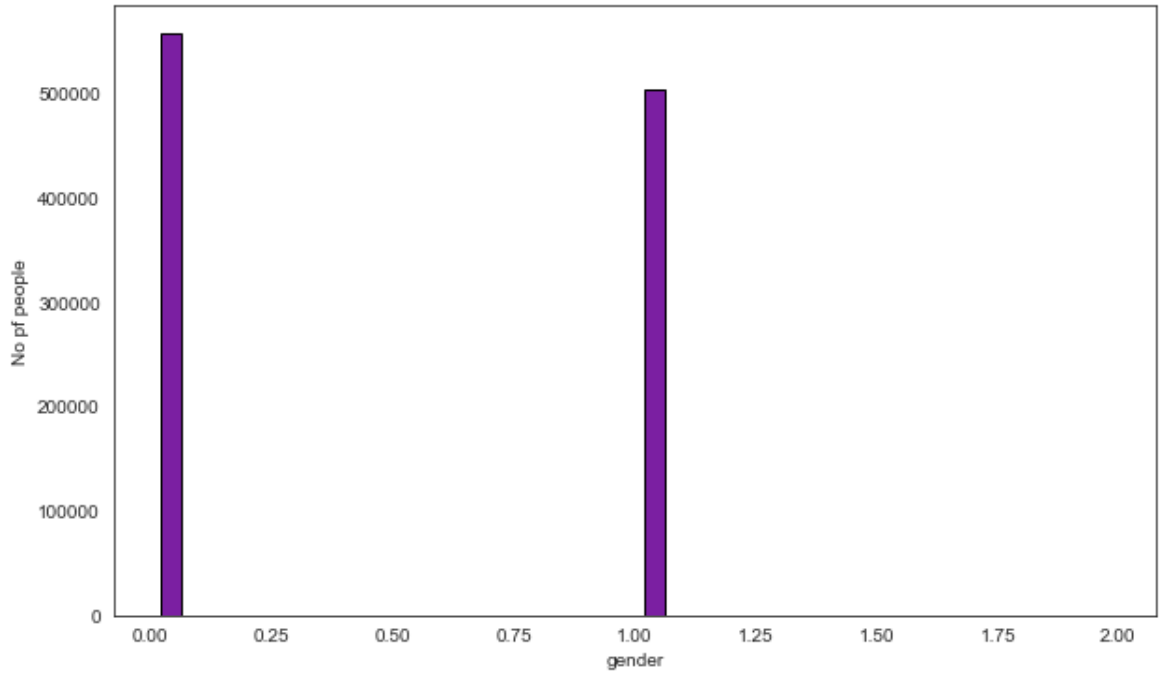


In [73]:

```
plt.figure(figsize=(10, 6))
plt.hist(inpc_data['T2D_OR_AD_FIRST'], bins=50, ec='black', color='red')
plt.xlabel('T2D_OR_AD_FIRST')
plt.ylabel('Population')
plt.show()
```



In [74]:
```
plt.figure(figsize=(10, 6))
plt.hist(inpc_data['gender'], bins=24, ec='black', color='#7b1fa2', rwidth=0.
plt.xlabel('gender')
plt.ylabel('No pf people')
plt.show()
```



In [52]:
```
inpc_data['Age_T2D_First'].describe()
```

Out[52]:
```
count    301398.000000
mean         62.378507
std          13.971256
min           1.241000
25%          53.293000
50%          63.260000
75%          72.257000
max         101.175000
Name: Age_T2D_First, dtype: float64
```

In [53]:
```
inpc_data['Age_AD_First'].describe()
```

Out[53]:
```
count    10580.000000
mean        50.676195
std         16.835661
min          5.303000
25%         38.123250
50%         51.387500
75%         63.203500
max         98.531000
Name: Age_AD_First, dtype: float64
```

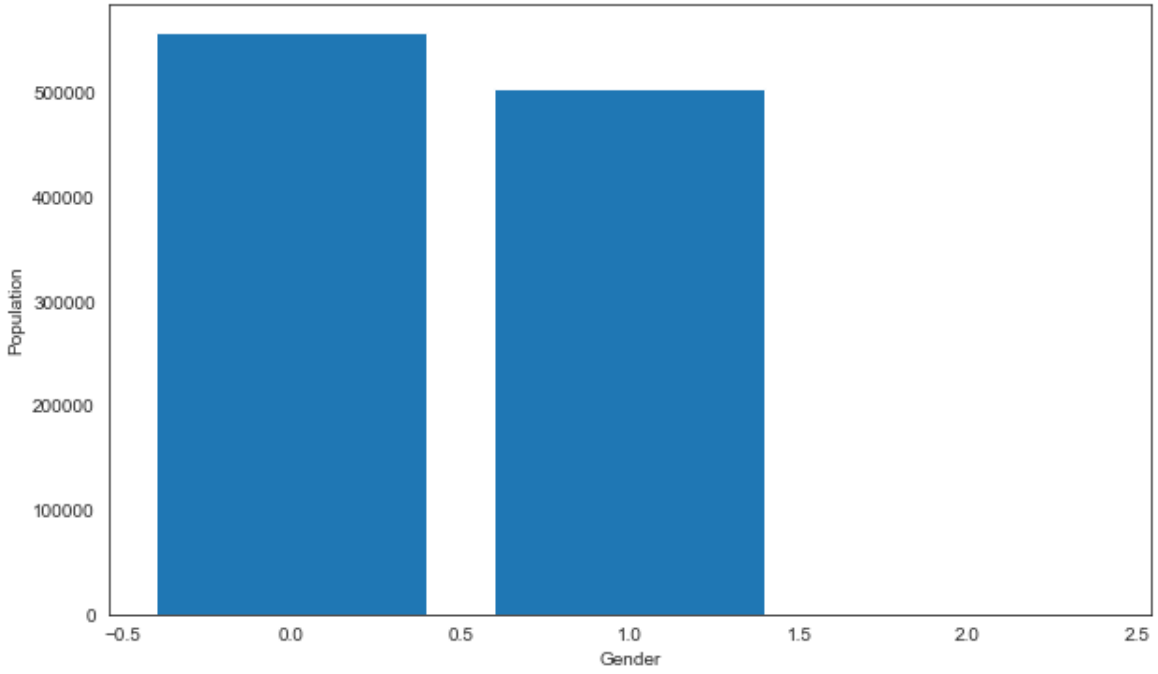
In [54]:
```
inpc_data['T2D_OR_AD_FIRST'].describe()
```

Out[54]:
```
count    8044.000000
mean       -0.291610
std         1.034345
min        -4.312000
25%        -0.890000
50%        -0.126000
75%         0.167000
max         4.370000
Name: T2D_OR_AD_FIRST, dtype: float64
```

In [56]:
```
inpc_data['gender'].value_counts()
```

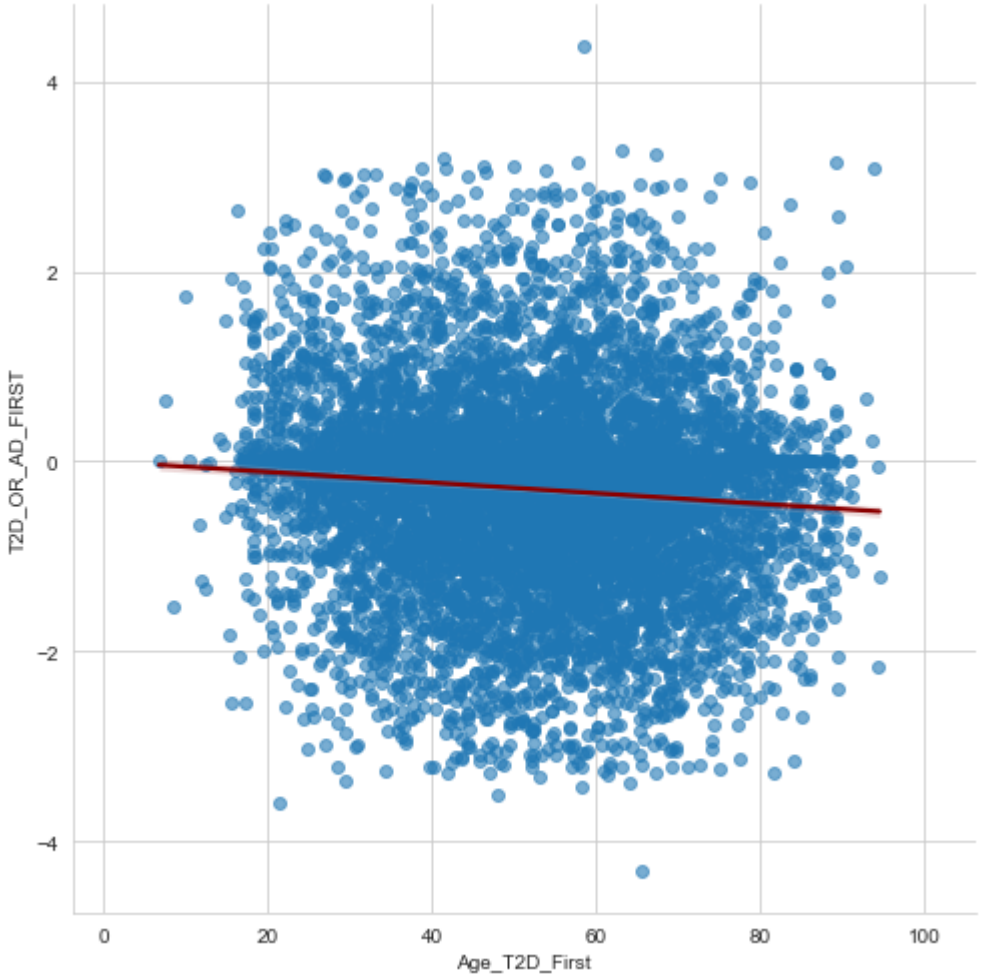
Out[56]:
```
0    557014
1    503739
2       223
Name: gender, dtype: int64
```

In [75]:
```
frequency = inpc_data['gender'].value_counts()
#type(frequency)
#frequency.index
#frequency.axes[0]
plt.figure(figsize=(10, 6))
plt.xlabel('Gender')
plt.ylabel('Population')
plt.bar(frequency.index, height=frequency)
plt.show()
```



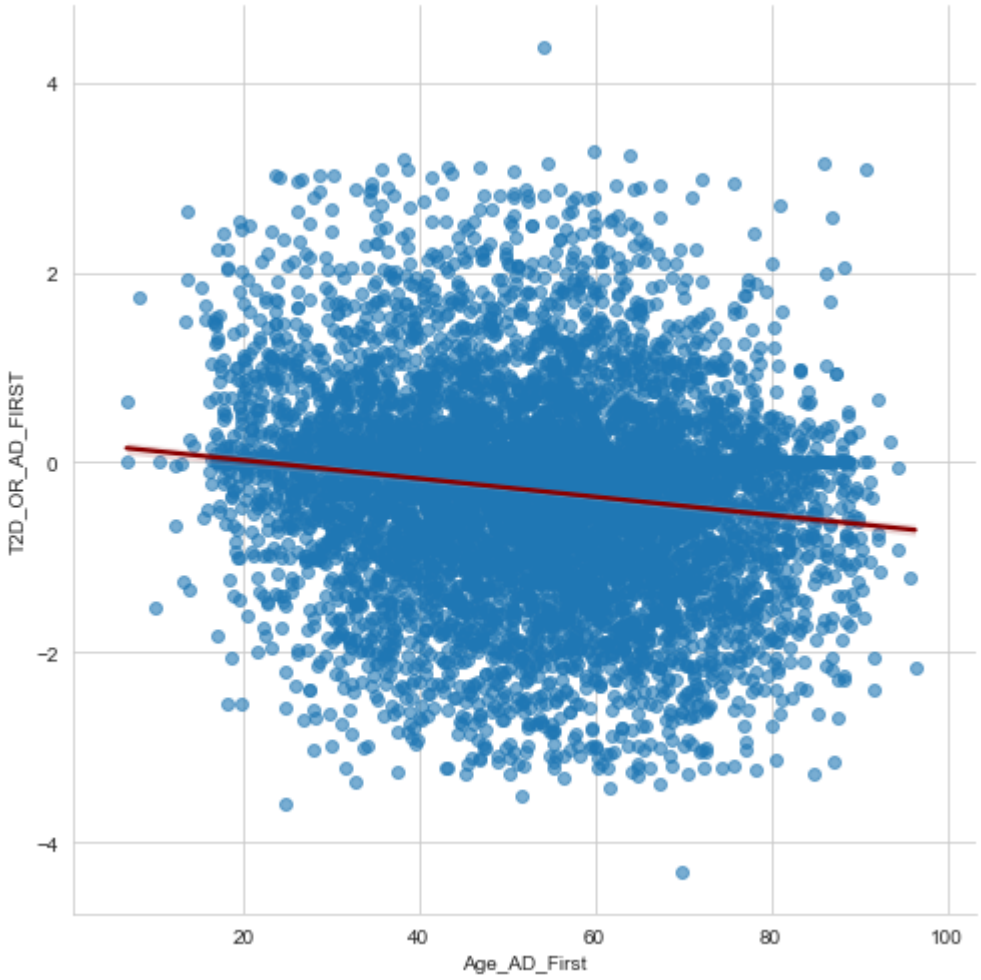
In [81]:
```
_style('whitegrid')
lot(x='Age_T2D_First', y='T2D_OR_AD_FIRST', data=inpc_data, size=7, scatter_kw()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\seaborn\regression.py:574: UserWarning: The `size` parameter has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



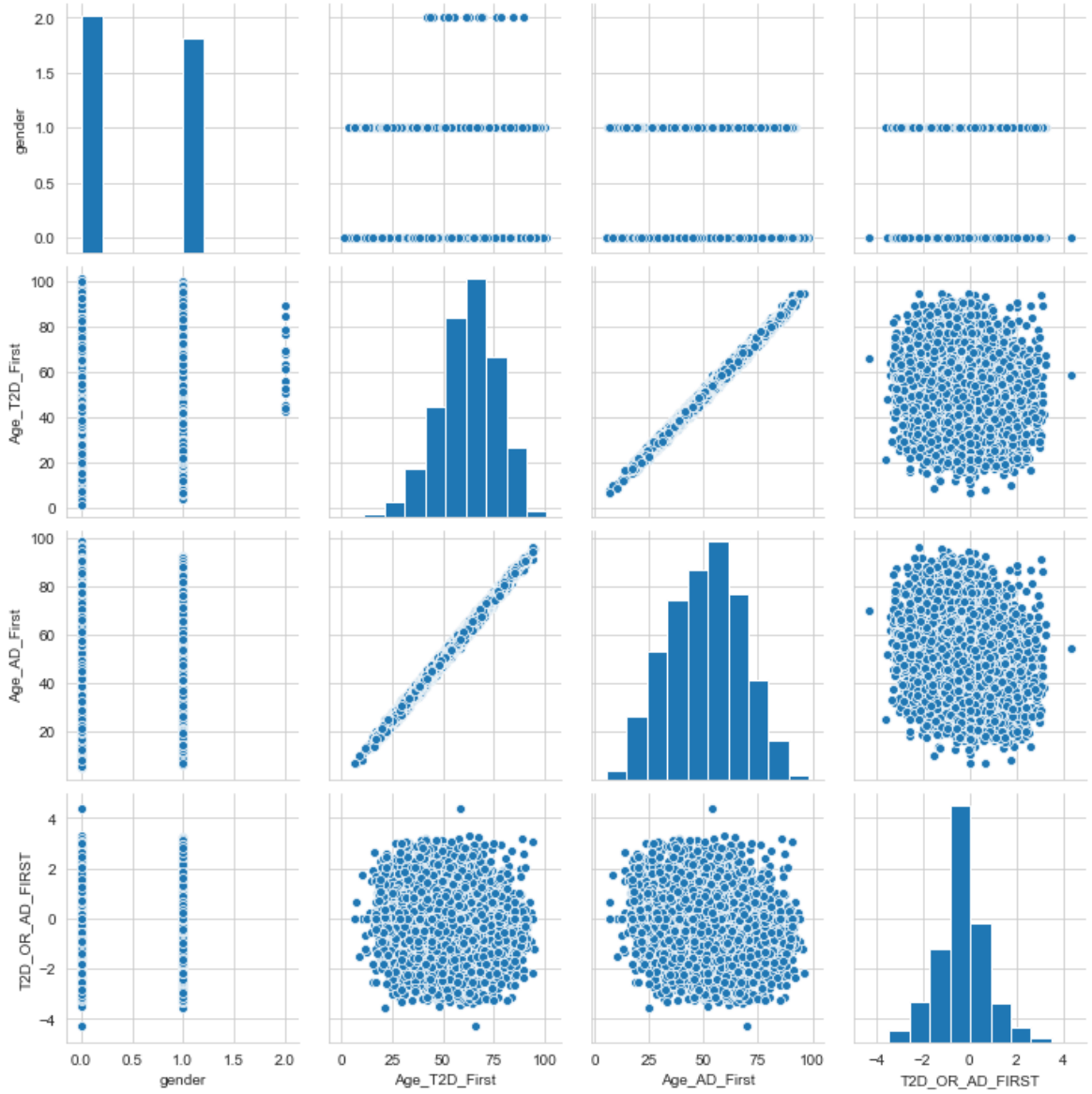
In [80]:
```
t_style('whitegrid')
plot(x='Age_AD_First', y='T2D_OR_AD_FIRST', data=inpc_data, size=7, scatter_k
ow()
```

```
C:\Users\mayam\anaconda3\lib\site-packages\seaborn\regression.py:574: UserW
arning: The `size` parameter has been renamed to `height`; please update yo
ur code.
  warnings.warn(msg, UserWarning)
```



In [79]:
```
%%time

sns.pairplot(inpc_data)
plt.show()
```




```
Wall time: 11.9 s
```