Importing Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Lasso
from sklearn import metrics
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

```
data = pd.read_csv('train_data.csv')
data.head()
```

ıcome	Employment status	Education level	Marital status	Dwelling	Age	Employment length	Has a mobile phone	Has a work phone	Ha ph
000.0	Working	Secondary / secondary special	Married	With parents	-16271	-3111	1.0	0.0	
000.0	Commercial associate	Higher education	Single / not married	House / apartment	-10130	-1651	1.0	0.0	
000.0	Commercial associate	Secondary / secondary special	Married	House / apartment	-12821	-5657	1.0	0.0	
000.0	Commercial associate	Higher education	Single / not married	House / apartment	-20929	-2046	1.0	0.0	
000.0	Working	Secondary / secondary special	Separated	House / apartment	-16207	-515	1.0	0.0	

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 23893 entries, 0 to 23892
Data columns (total 20 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   ID                   23893 non-null  int64
 1   Gender               23893 non-null  object
 2   Has a car            23893 non-null  object
 3   Has a property       23893 non-null  object
 4   Children count       23893 non-null  int64
 5   Income               23893 non-null  float64
 6   Employment status    23893 non-null  object
 7   Education level      23893 non-null  object
 8   Marital status       23893 non-null  object
 9   Dwelling             23893 non-null  object
 10  Age                  23893 non-null  int64
 11  Employment length    23893 non-null  object
 12  Has a mobile phone   23892 non-null  float64
 13  Has a work phone     23892 non-null  float64
 14  Has a phone          23892 non-null  float64
 15  Has an email         23892 non-null  float64
 16  Job title            16467 non-null  object
 17  Family member count  23892 non-null  float64
 18  Account age          23892 non-null  float64
 19  Is high risk         23892 non-null  float64
dtypes: float64(8), int64(3), object(9)
memory usage: 3.6+ MB
```

```
data.describe()
```

Children count	Income	Age	Has a mobile phone	Has a work phone	Has a phone	Ha
23893.000000	2.389300e+04	23893.000000	23892.0	23892.000000	23892.000000	23892.0
0.431298	1.866992e+05	-15979.815050	1.0	0.224217	0.294366	0.0
0.740487	1.002271e+05	4206.166773	0.0	0.417075	0.455767	0.2
0.000000	2.700000e+04	-25152.000000	1.0	0.000000	0.000000	0.0
0.000000	1.215000e+05	-19453.000000	1.0	0.000000	0.000000	0.0
0.000000	1.575000e+05	-15563.000000	1.0	0.000000	0.000000	0.0
1.000000	2.250000e+05	-12461.000000	1.0	0.000000	1.000000	0.0
19.000000	1.575000e+06	-7705.000000	1.0	1.000000	1.000000	1.0

Data Preprocesssing

```
print(data.isnull().sum())
```

```
ID                        0
Gender                    0
Has a car                 0
Has a property            0
Children count            0
Income                    0
Employment status         0
Education level           0
Marital status            0
Dwelling                  0
Age                       0
Employment length         0
Has a mobile phone        1
Has a work phone          1
Has a phone               1
Has an email              1
Job title              7426
Family member count       1
Account age               1
Is high risk              1
dtype: int64
```

```
data['Is high risk'].value_counts()
```

```
0.0    23487
1.0      405
Name: Is high risk, dtype: int64
```

```
data['Employment status'].value_counts()
```

```
Working                 12353
Commercial associate     5553
Pensioner                4042
State servant            1940
Student                     4
Name: Employment status, dtype: int64
```

```
data['Education level'].value_counts()
```

```
Secondary / secondary special    16249
Higher education                  6462
Incomplete higher                  921
Lower secondary                    238
Academic degree                     22
Name: Education level, dtype: int64
```

```
data['Dwelling'].value_counts()
```

```
House / apartment      21342
With parents            1158
Municipal apartment      741
Rented apartment         369
Office apartment         180
Co-op apartment          102
Name: Dwelling, dtype: int64
```

```
data['Marital status'].value_counts()
```

```
Married                 16416
Single / not married     3172
Civil marriage           1891
Separated                1388
Widow                    1025
Name: Marital status, dtype: int64
```

```
data.dropna(subset=['Has a mobile phone'],inplace = True)
```

```
print(data.isnull().sum())
```

```
ID                        0
Gender                    0
Has a car                 0
Has a property            0
Children count            0
Income                    0
Employment status         0
Education level           0
Marital status            0
Dwelling                  0
Age                       0
Employment length         0
Has a mobile phone        0
Has a work phone          0
Has a phone               0
Has an email              0
Job title              7425
Family member count       0
Account age               0
Is high risk              0
dtype: int64
```

```
data.replace({'Marital status':{'Married':1,'Single / not married':2,'Civil marriage':3,'Separated':4,'Widow':5}},inplace=True)
```

```
data.replace({'Employment status':{'Working':1,'Commercial associate':2,'Pensioner':3,'State servant':4,'Student':5}},inplace=True)
```

```
data.replace({'Education level':{'Secondary / secondary special':1,'Higher education':2,'Incomplete higher':3,'Lower secondary':4,'
```

```
data.replace({'Dwelling':{'House / apartment':1,'With parents':2,'Municipal apartment':3,'Rented apartment':4,'Office apartment':5,
```

```
data['Gender'].value_counts()
```

```
F    16025
M     7867
Name: Gender, dtype: int64
```

```
data.replace({'Gender':{'M':1,'F':2}},inplace=True)
```

```
data.replace({'Has a car':{'Y':1,'N':0}},inplace=True)
```

```
data.replace({'Has a property':{'Y':1,'N':0}},inplace=True)
```

```
data.head()
```

Income	Employment status	Education level	Marital status	Dwelling	Age	Employment length	Has a mobile phone	Has a work phone	Ha ph
35000.0	1	1	1	2	-16271	-3111	1.0	0.0	
35000.0	2	2	2	1	-10130	-1651	1.0	0.0	
30000.0	2	1	1	1	-12821	-5657	1.0	0.0	
50000.0	2	2	2	1	-20929	-2046	1.0	0.0	
70000.0	1	1	4	1	-16207	-515	1.0	0.0	

```
x=data.drop(['Is high risk','Job title'],axis=1)
```

```
print(x)
```

```
            ID  Gender  Has a car  Has a property  Children count    Income  \
0      5037048       1          1               1               0  135000.0
1      5044630       2          1               0               1  135000.0
2      5079079       2          0               1               2  180000.0
3      5112872       2          1               1               0  360000.0
4      5105858       2          0               0               0  270000.0
...        ...     ...        ...             ...             ...       ...
23887  5009286       2          0               0               0  130500.0
23888  5068395       2          0               1               0  202500.0
23889  5117420       2          1               0               0  112500.0
23890  5116955       2          0               1               1  112500.0
23891  5068037       2          0               0               0  135000.0

       Employment status  Education level  Marital status  Dwelling    Age  \
0                      1                1               1         2 -16271
1                      2                2               2         1 -10130
2                      2                1               1         1 -12821
3                      2                2               2         1 -20929
4                      1                1               4         1 -16207
...                  ...              ...             ...       ...    ...
23887                  1                1               3         1 -15140
23888                  3                2               2         1 -21344
23889                  1                1               1         1 -16215
23890                  2                1               1         1 -17243
23891                  2                1               1         1 -17614

      Employment length  Has a mobile phone  Has a work phone  Has a phone  \
0                 -3111                 1.0               0.0          0.0
1                 -1651                 1.0               0.0          0.0
2                 -5657                 1.0               0.0          0.0
3                 -2046                 1.0               0.0          0.0
4                  -515                 1.0               0.0          1.0
...                 ...                 ...               ...          ...
23887             -4816                 1.0               0.0          1.0
23888            365243                 1.0               0.0          1.0
23889             -3567                 1.0               0.0          0.0
23890             -2378                 1.0               1.0          0.0
23891             -4219                 1.0               0.0          0.0

       Has an email  Family member count  Account age
0               0.0                  2.0        -17.0
1               0.0                  2.0         -1.0
2               0.0                  4.0        -38.0
3               1.0                  1.0        -11.0
4               0.0                  1.0        -41.0
...             ...                  ...          ...
23887           0.0                  2.0        -48.0
23888           1.0                  1.0        -44.0
23889           0.0                  2.0         -6.0
23890           0.0                  3.0        -55.0
23891           0.0                  2.0        -37.0

[23892 rows x 18 columns]
```

```
y=data['Is high risk']
```

```
print(y)
```

```
0        0.0
1        0.0
2        0.0
3        0.0
4        0.0
        ...
23887    0.0
23888    0.0
23889    0.0
23890    0.0
23891    0.0
Name: Is high risk, Length: 23892, dtype: float64
```

Machine Learning Model(Logistics Regression)

Train_Test split

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.1,random_state=2)
```

```
[x_train.shape,x_train.shape, y_train.shape, y_test.shape]
```

```
[(21502, 18), (21502, 18), (21502,), (2390,)]
```

```
leg=LogisticRegression()
```

```
leg.fit(x_train,y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

MODEL EVALUATION

Accuracy Score

```
x_tr_d=leg.predict(x_train)
```

```
trdaac=accuracy_score(y_train, x_tr_d)
print("Accuracy on Training data is :",trdaac)
```

```
Accuracy on Training data is : 0.9829783275974328
```

```
x_test_d=leg.predict(x_test)
testdacc=accuracy_score(y_test, x_test_d)
print("Accuracy on Test data is;",testdacc)
```

```
Accuracy on Test data is; 0.9836820083682009
```

```
Prediction for Credit Card Approval
```

Prediction for Credit Card Approval

```
input_data=[5037048, 1, 1, 1, 0, 135000.0, 1, 1, 1, 2, -16271,-3111, 1.0, 0.0, 0.0, 0.0, 2.0, -17.0]
inpasnum=np.array(input_data)
inres=inpasnum.reshape(1,-1)
```

```
pred=leg.predict(inres)
pred
if(pred[0]==0):
  print("High risk for approving Credit Card")
else:
  print("Credit Card Approved")
```

```
High risk for approving Credit Card
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but LogisticRegr
  warnings.warn(
```