```
import pandas as pd
df = pd.read_csv("Churn_Modelling (1).csv",sep = ",")

df
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0             1    15634602   Hargrave          619    France  Female
42.0
1             2    15647311       Hill          225     Spain  Female
41.0
2             3    15619304       Onio          629    France  Female
42.0
3             4    15701354       Boni          699    France  Female
39.0
4             5    15737888   Mitchell          850       NaN  Female
43.0
...         ...         ...        ...          ...       ...     ...
...
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
3            93826.63       0
4            79084.10       0
...               ...     ...
```

```
9995         96270.64       0
9996        101699.77       0
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

[10000 rows x 14 columns]
```

```
df.shape
```

```
(10000, 14)
```

```
df.head()
```

```
   RowNumber  CustomerId   Surname  CreditScore Geography  Gender
Age  \
0          1    15634602  Hargrave          619    France  Female
42.0
1          2    15647311      Hill          225     Spain  Female
41.0
2          3    15619304      Onio          629    France  Female
42.0
3          4    15701354      Boni          699    France  Female
39.0
4          5    15737888  Mitchell          850       NaN  Female
43.0

   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       2       0.00              1          1               1
1       1   83807.86              1          0               1
2       8  159660.80              3          1               0
3       1       0.00              2          0               0
4       2  125510.82              1          1               1

   EstimatedSalary  Exited
0        101348.88       1
1        112542.58       0
2        113931.57       1
3         93826.63       0
4         79084.10       0
```

```
df.tail()
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
```

```
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
9995         96270.64       0
9996        101699.77       0
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   RowNumber        10000 non-null  int64
 1   CustomerId       10000 non-null  int64
 2   Surname          10000 non-null  object
 3   CreditScore      10000 non-null  int64
 4   Geography        9999 non-null   object
 5   Gender           9999 non-null   object
 6   Age              9999 non-null   float64
 7   Tenure           10000 non-null  int64
 8   Balance          9999 non-null   float64
 9   NumOfProducts    10000 non-null  int64
 10  HasCrCard        10000 non-null  int64
 11  IsActiveMember   10000 non-null  int64
 12  EstimatedSalary  10000 non-null  float64
 13  Exited           10000 non-null  int64
dtypes: float64(3), int64(8), object(3)
memory usage: 1.1+ MB

df.describe() # statical information of columns containing the numeric
values we will get

         RowNumber    CustomerId  CreditScore          Age
Tenure  \
count  10000.00000  1.000000e+04  10000.00000  9999.000000
10000.000000
mean    5000.50000  1.569094e+07    650.48290    38.924892
```

```
5.012800
std     2886.89568  7.193619e+04     96.72014    10.486670
2.892174
min        1.00000  1.556570e+07    225.00000    18.000000
0.000000
25%     2500.75000  1.562853e+07    584.00000    32.000000
3.000000
50%     5000.50000  1.569074e+07    652.00000    37.000000
5.000000
75%     7500.25000  1.575323e+07    717.25000    44.000000
7.000000
max    10000.00000  1.581569e+07    850.00000    92.000000
10.000000

             Balance  NumOfProducts    HasCrCard  IsActiveMember  \
count    9999.000000   10000.000000  10000.00000    10000.000000
mean    76483.536070       1.530200      0.70550        0.515100
std     62400.186375       0.581654      0.45584        0.499797
min         0.000000       1.000000      0.00000        0.000000
25%         0.000000       1.000000      0.00000        0.000000
50%     97188.620000       1.000000      1.00000        1.000000
75%    127646.040000       2.000000      1.00000        1.000000
max    250898.090000       4.000000      1.00000        1.000000

       EstimatedSalary        Exited
count     10000.000000  10000.000000
mean     100090.239881      0.203700
std       57510.492818      0.402769
min          11.580000      0.000000
25%       51002.110000      0.000000
50%      100193.915000      0.000000
75%      149388.247500      0.000000
max      199992.480000      1.000000
```

finding null values

```
x = df.head()
y = x.isnull()
y
```

```
   RowNumber  CustomerId  Surname  CreditScore  Geography  Gender
Age  \
0      False       False    False        False      False   False
False
1      False       False    False        False      False   False
False
2      False       False    False        False      False   False
False
3      False       False    False        False      False   False
False
```

```
4      False       False    False        False       True   False
False

   Tenure  Balance  NumOfProducts  HasCrCard  IsActiveMember
EstimatedSalary  \
0   False    False          False      False           False
False
1   False    False          False      False           False
False
2   False    False          False      False           False
False
3   False    False          False      False           False
False
4   False    False          False      False           False
False

   Exited
0   False
1   False
2   False
3   False
4   False
```

```
y.sum() #head
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          1
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

```
df.isnull()
```

```
      RowNumber  CustomerId  Surname  CreditScore  Geography  Gender
Age  \
0         False       False    False        False      False   False
False
1         False       False    False        False      False   False
False
2         False       False    False        False      False   False
False
```

```
3         False       False    False        False      False   False
False
4         False       False    False        False       True   False
False
...         ...         ...      ...          ...        ...     ...
...
9995      False       False    False        False      False   False
False
9996      False       False    False        False      False   False
False
9997      False       False    False        False      False   False
False
9998      False       False    False        False      False   False
False
9999      False       False    False        False      False   False
False

      Tenure  Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0      False    False          False      False           False
1      False    False          False      False           False
2      False    False          False      False           False
3      False    False          False      False           False
4      False    False          False      False           False
...      ...      ...            ...        ...             ...
9995   False    False          False      False           False
9996   False    False          False      False           False
9997   False    False          False      False           False
9998   False    False          False      False           False
9999   False    False          False      False           False

      EstimatedSalary  Exited
0               False   False
1               False   False
2               False   False
3               False   False
4               False   False
...               ...     ...
9995            False   False
9996            False   False
9997            False   False
9998            False   False
9999            False   False

[10000 rows x 14 columns]

df.isnull().sum()

RowNumber          0
CustomerId         0
Surname            0
```

```
CreditScore        0
Geography          1
Gender             1
Age                1
Tenure             0
Balance            1
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

```
z = df.tail()
w = z.isnull()
w
```

```
      RowNumber  CustomerId  Surname  CreditScore  Geography  Gender
Age  \
9995      False       False    False        False      False   False
False
9996      False       False    False        False      False   False
False
9997      False       False    False        False      False   False
False
9998      False       False    False        False      False   False
False
9999      False       False    False        False      False   False
False

      Tenure  Balance  NumOfProducts  HasCrCard  IsActiveMember  \
9995   False    False          False      False           False
9996   False    False          False      False           False
9997   False    False          False      False           False
9998   False    False          False      False           False
9999   False    False          False      False           False

      EstimatedSalary  Exited
9995            False   False
9996            False   False
9997            False   False
9998            False   False
9999            False   False
```

```
w.sum() #tail
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
```

```
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

Handle missing values

```
df.Age = df.Age.fillna(df.Age.median())
```

```
df
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0             1    15634602   Hargrave          619    France  Female
42.0
1             2    15647311       Hill          225     Spain  Female
41.0
2             3    15619304       Onio          629    France  Female
42.0
3             4    15701354       Boni          699    France  Female
39.0
4             5    15737888   Mitchell          850       NaN  Female
43.0
...         ...         ...        ...          ...       ...     ...
...
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
```

```
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
3            93826.63       0
4            79084.10       0
...               ...     ...
9995         96270.64       0
9996        101699.77       0
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

[10000 rows x 14 columns]
```

```
df.Balance=df.Balance.fillna(df.Balance.median())
```

```
df
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0             1    15634602   Hargrave          619    France  Female
42.0
1             2    15647311       Hill          225     Spain  Female
41.0
2             3    15619304       Onio          629    France  Female
42.0
3             4    15701354       Boni          699    France  Female
39.0
4             5    15737888   Mitchell          850       NaN  Female
43.0
...         ...         ...        ...          ...       ...     ...
...
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
```

```
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
3            93826.63       0
4            79084.10       0
...               ...     ...
9995         96270.64       0
9996        101699.77       0
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

[10000 rows x 14 columns]
```

```
df.Geography = df.Geography.fillna(df.Geography.mode())
```

```
df
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0             1    15634602   Hargrave          619    France  Female
42.0
1             2    15647311       Hill          225     Spain  Female
41.0
2             3    15619304       Onio          629    France  Female
42.0
3             4    15701354       Boni          699    France  Female
39.0
4             5    15737888   Mitchell          850       NaN  Female
43.0
...         ...         ...        ...          ...       ...     ...
...
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
```

```
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
3            93826.63       0
4            79084.10       0
...               ...     ...
9995         96270.64       0
9996        101699.77       0
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

[10000 rows x 14 columns]

df.isnull().sum()

RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          1
Gender             1
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

```
df = df.fillna("unknown")
df
```

```
      RowNumber  CustomerId    Surname  CreditScore Geography  Gender
Age  \
0             1    15634602   Hargrave          619    France  Female
42.0
1             2    15647311       Hill          225     Spain  Female
41.0
2             3    15619304       Onio          629    France  Female
42.0
3             4    15701354       Boni          699    France  Female
39.0
4             5    15737888   Mitchell          850   unknown  Female
43.0
...         ...         ...        ...          ...       ...     ...
...
9995       9996    15606229   Obijiaku          771    France    Male
39.0
9996       9997    15569892  Johnstone          516    France    Male
35.0
9997       9998    15584532        Liu          709    France  Female
36.0
9998       9999    15682355  Sabbatini          772   Germany    Male
42.0
9999      10000    15628319     Walker          792    France  Female
28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
3          1       0.00              2          0               0
4          2  125510.82              1          1               1
...      ...        ...            ...        ...             ...
9995       5       0.00              2          1               0
9996      10   57369.61              1          1               1
9997       7       0.00              1          0               1
9998       3   75075.31              2          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
3            93826.63       0
4            79084.10       0
...               ...     ...
9995         96270.64       0
9996        101699.77       0
```

```
9997         42085.58       1
9998         92888.52       1
9999         38190.78       0

[10000 rows x 14 columns]
```

```
df.isnull().sum() # it is useful when we dealing with large values
```

```
RowNumber          0
CustomerId         0
Surname            0
CreditScore        0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
NumOfProducts      0
HasCrCard          0
IsActiveMember     0
EstimatedSalary    0
Exited             0
dtype: int64
```

```
df.columns
```

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore',
'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts',
'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited'],
      dtype='object')
```

Loc

```
df.loc[4:8]
```

```
   RowNumber  CustomerId   Surname  CreditScore Geography   Gender
Age  \
4          5    15737888  Mitchell          850   unknown   Female
43.0
5          6    15574012       Chu          645     Spain     Male
44.0
6          7    15592531  Bartlett          619    France     Male
50.0
7          8    15656148    Obinna          376   Germany  unknown
29.0
8          9    15792365        He          501    France     Male
44.0

   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
```

```
4       2  125510.82              1          1               1
5       8  113755.78              2          1               0
6       7       0.00              2          1               1
7       4  115046.74              4          1               0
8       4  142051.07              2          0               1

   EstimatedSalary  Exited
4         79084.10       0
5        149756.71       1
6         10062.80       0
7        119346.88       1
8         74940.50       0
```

```
df.loc[df.Age>40]
```

```
      RowNumber  CustomerId      Surname  CreditScore Geography
Gender   Age  \
0             1    15634602     Hargrave          619    France
Female  42.0
1             2    15647311         Hill          225     Spain
Female  41.0
2             3    15619304         Onio          629    France
Female  42.0
4             5    15737888     Mitchell          850   unknown
Female  43.0
5             6    15574012          Chu          645     Spain
Male  44.0
...         ...         ...          ...          ...       ...     ..
.   ...
9981       9982    15672754     Burbidge          498   Germany
Male  42.0
9982       9983    15768163      Griffin          655   Germany
Female  46.0
9986       9987    15581736     Bartlett          673   Germany
Male  47.0
9991       9992    15769959  Ajuluchukwu          597    France
Female  53.0
9998       9999    15682355    Sabbatini          772   Germany
Male  42.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
1          1   83807.86              1          0               1
2          8  159660.80              3          1               0
4          2  125510.82              1          1               1
5          8  113755.78              2          1               0
...      ...        ...            ...        ...             ...
9981       3  152039.70              1          1               1
9982       7  137145.12              1          1               0
9986       1  183579.54              2          0               1
```

```
9991       4   88381.21              1          1               0
9998       3   75075.31              2          1               0

      EstimatedSalary  Exited
0           101348.88       1
1           112542.58       0
2           113931.57       1
4            79084.10       0
5           149756.71       1
...               ...     ...
9981         53445.17       1
9982        115146.40       1
9986         34047.54       0
9991         69384.71       1
9998         92888.52       1

[3582 rows x 14 columns]
```

```
df.loc[0:3,['Age','CustomerId','RowNumber']]
```

```
    Age  CustomerId  RowNumber
0  42.0    15634602          1
1  41.0    15647311          2
2  42.0    15619304          3
3  39.0    15701354          4
```

```
df.loc[(df.HasCrCard == 1) & (df.Gender == 'Female')]
```

```
      RowNumber  CustomerId      Surname  CreditScore Geography
Gender   Age  \
0             1    15634602     Hargrave          619    France
Female  42.0
2             3    15619304         Onio          629    France
Female  42.0
4             5    15737888     Mitchell          850   unknown
Female  43.0
12           13    15632264          Kay          476    France
Female  34.0
14           15    15600882        Scott          635     Spain
Female  35.0
...         ...         ...          ...          ...       ...     ..
.   ...
9976       9977    15656062      Azikiwe          637    France
Female  33.0
9977       9978    15579969      Mancini          683    France
Female  32.0
9982       9983    15768163      Griffin          655   Germany
Female  46.0
9991       9992    15769959  Ajuluchukwu          597    France
Female  53.0
9999      10000    15628319       Walker          792    France
```

```
Female  28.0

      Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0          2       0.00              1          1               1
2          8  159660.80              3          1               0
4          2  125510.82              1          1               1
12        10       0.00              2          1               0
14         7       0.00              2          1               1
...      ...        ...            ...        ...             ...
9976       7  103377.81              1          1               0
9977       9       0.00              2          1               1
9982       7  137145.12              1          1               0
9991       4   88381.21              1          1               0
9999       4  130142.79              1          1               0

      EstimatedSalary  Exited
0           101348.88       1
2           113931.57       1
4            79084.10       0
12           26260.98       0
14           65951.65       0
...               ...     ...
9976         84419.78       0
9977         24991.92       0
9982        115146.40       1
9991         69384.71       1
9999         38190.78       0

[3191 rows x 14 columns]
```

```
u = df[['CustomerId','Surname','CreditScore']]
u.head()
```

```
   CustomerId   Surname  CreditScore
0    15634602  Hargrave          619
1    15647311      Hill          225
2    15619304      Onio          629
3    15701354      Boni          699
4    15737888  Mitchell          850
```

```
u = df[['CustomerId','Surname','CreditScore']].values
u
```

```
array([[15634602, 'Hargrave', 619],
       [15647311, 'Hill', 225],
       [15619304, 'Onio', 629],
       ...,
       [15584532, 'Liu', 709],
       [15682355, 'Sabbatini', 772],
       [15628319, 'Walker', 792]], dtype=object)
```

ILOC

**1.used to slice the dataset in the machine learning

**2.iloc is uses for indexed based

**3.while loc is uses for label

```
#dataset.iloc[rows,columns]
df.iloc[0,4]
```

```
'France'
```

```
x = df.iloc[0:4,2:5]
```

Input and Output data

```
x1 = df.iloc[:,-1:]
x1
```

```
      Exited
0          1
1          0
2          1
3          0
4          0
...      ...
9995       0
9996       0
9997       1
9998       1
9999       0

[10000 rows x 1 columns]
```

```
type(x1)
```

```
pandas.core.frame.DataFrame
```

```
x1 = df.iloc[:,-1]
x1
```

```
0       1
1       0
2       1
3       0
4       0
       ..
9995    0
9996    0
9997    1
9998    1
```

```
9999    0
Name: Exited, Length: 10000, dtype: int64
```

```
type(x1)
```

```
pandas.core.series.Series
```

label encoding

```
df.columns
```

```
Index(['RowNumber', 'CustomerId', 'Surname', 'CreditScore',
'Geography',
       'Gender', 'Age', 'Tenure', 'Balance', 'NumOfProducts',
'HasCrCard',
       'IsActiveMember', 'EstimatedSalary', 'Exited'],
      dtype='object')
```

```
df.Geography.value_counts()
```

```
France     5014
Germany    2509
Spain      2476
unknown       1
Name: Geography, dtype: int64
```

```
df.Age.value_counts()
```

```
37.0    479
38.0    477
35.0    474
36.0    456
34.0    447
       ...
92.0      2
82.0      1
88.0      1
85.0      1
83.0      1
Name: Age, Length: 70, dtype: int64
```

```
# converting of categorical data to numerical data
```

Method 1

```
from sklearn.preprocessing import LabelEncoder
from collections import Counter as count
```

```
s  = LabelEncoder()
print("Before lable encoder:",count(df['Geography']))
df['Geography']=s.fit_transform(df['Geography'])
```

```
Before lable encoder: Counter({'France': 5014, 'Germany': 2509,
'Spain': 2476, 'unknown': 1})

print('After label encoder:',count(df['Geography']))

After label encoder: Counter({0: 5014, 1: 2509, 2: 2476, 3: 1})
```

One hot Encoding

```
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
ct = ColumnTransformer([('on',OneHotEncoder(),[0])])

ct

ColumnTransformer(transformers=[('on', OneHotEncoder(), [0])])

df= ct.fit_transform(df)

df

<10000x10000 sparse matrix of type '<class 'numpy.float64'>'
     with 10000 stored elements in Compressed Sparse Row format>

df.shape

(10000, 10000)
```