```
import pandas as pd
import numpy as np


data = pd.read_csv("credit card.csv")
print(data.head())
```

```
   step      type    amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
0     1   PAYMENT   9839.64  C1231006815       170136.0       160296.36
1     1   PAYMENT   1864.28  C1666544295        21249.0        19384.72
2     1  TRANSFER    181.00  C1305486145          181.0            0.00
3     1  CASH_OUT    181.00   C840083671          181.0            0.00
4     1   PAYMENT  11668.14  C2048537720        41554.0        29885.86

      nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
0  M1979787155             0.0             0.0      0.0             0.0
1  M2044282225             0.0             0.0      0.0             0.0
2   C553264065             0.0             0.0      1.0             0.0
3    C38997010         21182.0             0.0      1.0             0.0
4  M1230701703             0.0             0.0      0.0             0.0
```

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

```
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    1
newbalanceDest    1
isFraud           1
isFlaggedFraud    1
dtype: int64
```

```
print(data.type.value_counts())
```

```
PAYMENT     19382
CASH_OUT    10689
CASH_IN      7632
TRANSFER     3974
DEBIT         594
Name: type, dtype: int64
```

```
type = data["type"].value_counts()
transactions = type.index
quantity = type.values

import plotly.express as px
figure = px.pie(data,
             values=quantity,
             names=transactions,hole = 0.5,
             title="Distribution of Transaction Type")
figure.show()
```

```
correlation = data.corr()
print(correlation["isFraud"].sort_values(ascending=False))
```

```
isFraud           1.000000
amount            0.058899
oldbalanceOrg    -0.004536
newbalanceDest   -0.008193
oldbalanceDest   -0.012463
newbalanceOrig   -0.015376
step             -0.050289
isFlaggedFraud         NaN
Name: isFraud, dtype: float64
<ipython-input-6-91bfb1e64f5b>:1: FutureWarning:

The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid
```

```
data["type"] = data["type"].map({"CASH_OUT": 1, "PAYMENT": 2,
                                 "CASH_IN": 3, "TRANSFER": 4,
                                 "DEBIT": 5})
data["isFraud"] = data["isFraud"].map({0: "No Fraud", 1: "Fraud"})
print(data.head())
```

```
   step  type    amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
0     1     2   9839.64  C1231006815       170136.0       160296.36
1     1     2   1864.28  C1666544295        21249.0        19384.72
2     1     4    181.00  C1305486145          181.0            0.00
3     1     1    181.00   C840083671          181.0            0.00
4     1     2  11668.14  C2048537720        41554.0        29885.86

      nameDest  oldbalanceDest  newbalanceDest   isFraud  isFlaggedFraud
0  M1979787155             0.0             0.0  No Fraud             0.0
1  M2044282225             0.0             0.0  No Fraud             0.0
2   C553264065             0.0             0.0     Fraud             0.0
3    C38997010         21182.0             0.0     Fraud             0.0
4  M1230701703             0.0             0.0  No Fraud             0.0
```

```
from sklearn.model_selection import train_test_split
x = np.array(data[["type", "amount", "oldbalanceOrg", "newbalanceOrig"]])
y = np.array(data[["isFraud"]])
```