Given Fraud.csv dataset has 63,62,620 rows and 11 columns / features.

Out of 11 columns,
5 columns ; amount, oldbalanceOrg, newbalanceOrg, oldbalanceDest, newbalanceDest are of float type(numeric data) that highly indicates that these are related to money. Therefore, we can not drop these columns.
3 columns; step, isFraud, isFlaggedFraud are of int type(numeric data).
3 columns; type, nameOrig, nameDest are categorical data.

Step column has value from 1 to 743, i.e., the dataset has only a month's dataset.
isFraud, isFlaggedFraud has value 0 and 1, 75% of the dataset is 0, i.e., at least 75% of the data is non-fraudulent. Thus, it creates a high chance of bias in the dataset.

There are no null values present in the dataset. There are no duplicated rows. We can proceed.

In heatmap,oldbalanceOrg and newbalanceOrg; oldbalanceDest and newbalanceDest are correlated, we can see that there is no or very low correlation of features with each other. Thus, we can say that the variables are independent of each other, have unique values which will help to take action on each columns separately.(no multicollinearity)

As see in the countplot and pie-chart, we can see data is highly imbalanced where non-fraud cases are 99.87% and fraud cases are just 0.13%. If we build model on this data, there are high chances of our predictions to be biased.
We only have 8213 cases of fraud out of 6362620 cases.

Here, we can see that CASH-OUT, PAYMENT have high occurrence in the type of transactions of dataset.
But, if we see the fraud cases, only CASH_OUT and TRANSFER are considered as fraudulent. Therefore, we will drop the rows other than these two data points.

Here, we can see that almost fraud takes place if type of transaction is CASH_OUT or TRANSFER.(50% of chances each)

The amount must be +ve. As we can see there are no transactions where the amount is negative. But where the amount is equal to 1, we can see that every transaction is fraud. Therefore we can predict that if the amount is 0, then the transaction is fraud.
Now, we can remove the 16 rows.

Here, the distribution of step(time) is uniform in case of fraudulent transactions, whereas the distribution of step(time) is specific in case of legit transactions.

We need to drop 'nameOrig' and 'nameDest' columns because they are unnecessary.

Now we have only one categorical variable 'type', only with 2 variables, thus we will convert them.
Precision shows how many of the flagged fraud cases are genuinely fraudulent.
Recall is a measure that tells us how well the model identifies actual fraudulent transactions among all the fraud cases.

Logistic Regression:
Logistic Regression's average recall score = 50%
It captured only half of the fraud cases.
Logistic Regression's precision = 90%
Here, we observe that precision is high and recall is low therefore this model has failed.

K-NN Model:
K-NN's average recall score = 69%
K-NN's precision = 91%

Decision Tree:
Decision Tree's average recall score = 89%
Decision Tree's precision = 89%

Random Forest:
Random Forest's average recall score = 89%
Random Forest's precision = 89%

Model	Precision	Recall	Accuracy
Logistic Regression	0.90	0.50	99.83%
K-NN	0.91	0.69	99.88%
Decision Tree	0.89	0.89	99.93%
Random Forest	0.89	0.89	99.937%

Conclusion:
The best model that can accurately classify transactions as either legitimate or fraudulent is Random Forest.