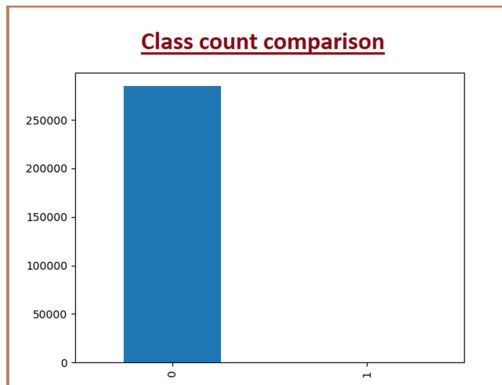
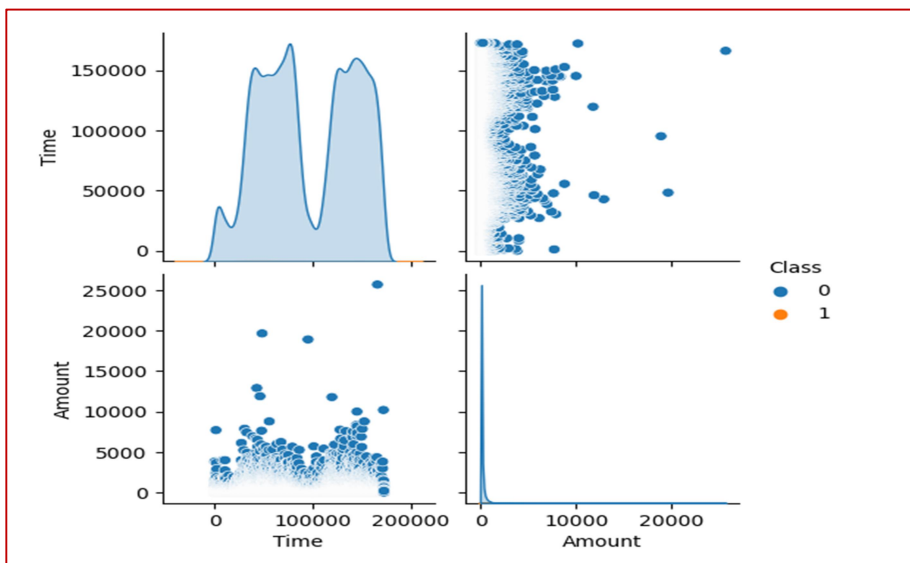


# Visualization of the supervised Learning project

## 1. Exploratory Data Analysis

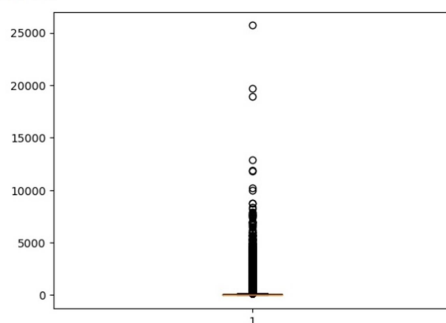


The count plot clearly say that the data set is imbalanced. So we can understand that we need to applying models on the dataset

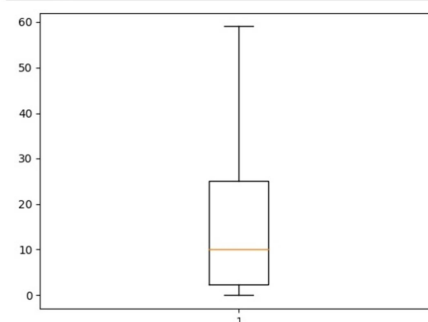


### Before & After Outliers treatment

```
In [8]: ##Check Outliers
plt.boxplot(df['Amount']) #no outlier
plt.show()
```



```
##Check Outliers
df=remove_outliers(df,"Amount")
plt.boxplot(df['Amount']) #no outlier
plt.show()
```



## 2. Accuracies of different supervised learning models

### Logistic Regression Accuracies on Test & Train Dataset

final_rus_test_log_vaules			VS			final_rus_train_log_vaules		
	Scores	Value					Scores	Value
0	Accuracy	93.296089				0	Accuracy	96.626506
1	Precision	96.296296				1	Precision	98.514851
2	Recall	89.655172				2	Recall	94.761905
3	F1-score	92.857143				3	F1-score	96.601942

### DecisionTree accuracies on Train & Test dataset

final_rus_test_dt_vaules			VS			final_rus_train_dt_vaules		
	Scores	Value					Scores	Value
0	Accuracy	87.150838				0	Accuracy	100.0
1	Precision	86.363636				1	Precision	100.0
2	Recall	87.356322				2	Recall	100.0
3	F1-score	86.857143				3	F1-score	100.0

### Random Forest accuracies on train & test

final_rus_test_rf_vaules			VS			final_rus_train_rf_vaules		
	Scores	Value					Scores	Value
0	Accuracy	92.737430				0	Accuracy	100.0
1	Precision	96.250000				1	Precision	100.0
2	Recall	88.505747				2	Recall	100.0
3	F1-score	92.215569				3	F1-score	100.0

### Final accuracy scores of all algorithms

final_result		
	Models	Accuracy
0	Logistic	93.296089
1	Decision Tree	87.150838
2	Random Forest	92.737430

### Accuracy score Logistic model after hyper parameter tuning

Accuracy: 0.9441340782122985 ←  
 ROC AUC Score: 0.9425287356321839  
 Classification Report:

	precision	recall	f1-score	support
0	0.90	1.00	0.95	92
1	1.00	0.89	0.94	87
accuracy			0.94	179
macro avg	0.95	0.94	0.94	179
weighted avg	0.95	0.94	0.94	179

Confusion Matrix:  
 [[92 0]  
 [10 77]]