Name: Sadman Sakib
ID:1620676042

Problem 1

Correlation Matrix :

```
In [17]: import numpy as np

         # Create correlation matrix
         corr_matrix = combined_df.corr().abs()

         # Select upper triangle of correlation matrix
         upper = corr_matrix.where(np.triu(np.ones(corr_matrix.shape), k=1).astype(np.bool))

         # Find features with correlation greater than 0.8
         columns = [column for column in upper.columns if any(upper[column] > 0.9)]

         columns.append('Risk')
         df = combined_df[columns]
```

I haven't used any graph in this problem

Classifier

Here I have used "Decision Tree Classifier" . I have used Bagging. Bagging is used when our goal is to reduce the variance of a decision tree. Here idea is to create several subsets of data from training sample chosen randomly with replacement. Now, each collection of subset data is used to train their decision trees

Table:

Model	Supervised Model
Training Accuracy	0.9681
Testing Accuracy	0.9580

Name: Sadman Sakib
ID:1620676042

ROC: