SCORE CALCULATION ALGORITHM MODULE

a.) Payment History

def getscore(dataset):  
 #1. PAYMENT HISTORY -----------------------------------------------------| (feature 7)  
  
 # 1( Loan Amount , Credit balance) How many payments made  
 loan\_amount = dataset["Current Loan Amount"]  
 monthly\_debt = dataset["Monthly Debt"]  
  
 #1.a Monthly Percentage Paid from Overall Loan  
 cal= (monthly\_debt/loan\_amount) \* 100  
 # processing it to a logarithmic scoring value  
 calc\_value = (2 \*\* (-1/cal)) \* 90  
  
 #1.b Years of Credit History  
 years\_ch = dataset["Years of Credit History"]  
 years\_ch = (1.8 \*\* (-1/years\_ch)) \* 140  
  
 #FINAL SCORE FOR PAYMENT HISTORY  
 final\_minor1\_score=round(calc\_value + years\_ch, 2)  
  
 #END OF PAYMENT HISTORY--------------------------------------------------|

b.) Amounts Owed

#2. AMOUNTS OWED --------------------------------------------------------| (feature 8)  
  
 value1 = dataset["Current Loan Amount"]  
 value2 = dataset["Maximum Open Credit"]  
  
 #2.a Current Loan Amount  
 # current loan amount has the highest feature importance  
 loan = (1.8 \*\* (-1/value1)) \* 135  
  
 #2.b Credit Limit Percentage  
 limit = value1 / value2 \* 100  
 # Processing through arithmetic value  
 credlim = (round(2100000000/(20000000+(( limit ) \*\* 4)))) + 90  
  
 #FINAL SCORE FOR PAYMENT HISTORY  
 final\_minor2\_score = credlim + loan  
  
 #END OF AMOUNTS OWED ----------------------------------------------------|

c.) Length of Credit History

#3. LENGTH OF CREDIT HISTORY---------------------------------------------| (feature 9)  
  
 #Ignore delinquent above 84  
 deduct\_delinquent = dataset[dataset["Months since last delinquent"] <= 1000 ]  
 added = deduct\_delinquent.replace(0, 900)  
 deducted = round(15600/(160+(added["Months since last delinquent"] \*\* 2)))  
  
 #1.2 Years of credit history converted to points 1yr=2.67  
 years = years\_ch \* 2.67  
 # limiting score  
 years = (7 \*\* (1/-years)) \* 160.6   
  
 #FINAL SCORE FOR LENGTH OF CREDIT HISTORY  
 final\_minor3\_score = years - (deducted/2)  
  
 #END OF LENGTH OF CREDIT HISTORY-----------------------------------------| (feature 10)

d.) New Credit/Credit Mix

#4 NEW CREDIT / CREDIT MIX-----------------------------------------------|  
  
 final\_minor4\_score = (((5 \*\* (-2/dataset["Number of Open Accounts"])) \* 10)+20) \* 2  
  
 #END OF NEW CREDIT / CREDIT MIX------------------------------------------|

TOTAL SCORES

Credit\_score = final\_minor1\_score + final\_minor2\_score + final\_minor3\_score + final\_minor4\_score  
 format\_Credit\_score = round(Credit\_score)

SAVING SCORES TO CSV

dataset["payhis"] = final\_minor1\_score  
 dataset["amtowed"] = final\_minor2\_score  
 dataset["lenofcrhis"] = final\_minor3\_score  
 dataset["newcred"] = final\_minor4\_score  
 dataset["Score"] = format\_Credit\_score  
  
 return dataset