S.Thapa   *Final Project Pseudocode*
ICS4U
1/14/24

Class LoginWindow:
   Constructor:
      Initialize GUI for login/signup window
      Set up usernameField, passwordField, loginButton, signupButton, and
loggedInUsername variables
      Add event listeners for loginButton and signupButton

   authenticateUser(username, password):
      Return true if username is not empty and password length is at least 8 characters,
otherwise false

   signupUser(username, password):
      Return true if username is not empty and password length is at least 8 characters,
otherwise false

   openMortgageCalculator():
      Create an instance of MortgageCalculatorGUI with loggedInUsername
      Set it visible
      Close the current LoginWindow

Class MortgageCalculatorGUI:
   Constructor(loggedInUsername):
      Initialize GUI for the Mortgage Calculator
      Set up input fields, buttons, and labels
      Add event listeners for calculateButton and calculateLoanAmountAndTermButton
      Initialize amortizationSchedule as an empty list

   calculateMortgagePayment():
      Try:
         Parse input fields for loanAmount, annualInterestRate, loanTerm, downPayment,
and selectedPaymentPeriod
         Calculate monthlyPayment using provided values
         Update the resultLabel with the calculated monthlyPayment

Generate the amortizationSchedule based on the input data
Calculate and display the total interest paid

Catch NumberFormatException:
Display an error message

calculateLoanAmountAndTerm():
Try:
Parse input fields for desiredMonthlyPayment, desiredLoanTerm, annualInterestRate, and downPayment
Calculate loanAmount and loanTerm based on desired monthly payment and loan term
Update the resultLabel with the calculated loanAmount and loanTerm

Catch NumberFormatException:
Display an error message

calculateMonthlyPayment(loanAmount, annualInterestRate, loanTerm, downPayment, paymentPeriod):
Calculate and return the monthly payment based on input values and paymentPeriod

generateAmortizationSchedule(loanAmount, annualInterestRate, loanTerm, downPayment, paymentPeriod):
Generate and return the amortization schedule based on input values and paymentPeriod

calculateLoanAmount(desiredMonthlyPayment, desiredLoanTerm, annualInterestRate, downPayment, paymentPeriod):
Calculate and return the loan amount based on desired monthly payment and loan term

calculateLoanTerm(desiredMonthlyPayment, desiredLoanTerm, annualInterestRate, downPayment, paymentPeriod):
Calculate and return the loan term based on desired monthly payment and loan term

setVisible(visible):
Set the visibility of the mainFrame

Class Calculator:
    calculateTotalInterestPaid(amortizationSchedule):
        Calculate and return the total interest paid based on the amortization schedule

Main Function:
    Create an instance of LoginWindow and set it visible