The Monthly Loan Repayment Scheduler

# Files:

# Purpose:

A program that would calculate the interest payments and print out a schedule of interest and total payment.

# Specification:

The Monthly Loan Repayment Scheduler is a command line application that would allow users to input their amount of the loan, the interest and the length of the loan and outputting the loan schedule. This application helps people to look at loans at a variety of interest rates, their monthly payments and the total payments.

# User Guide:

Provided is a executable jar file.

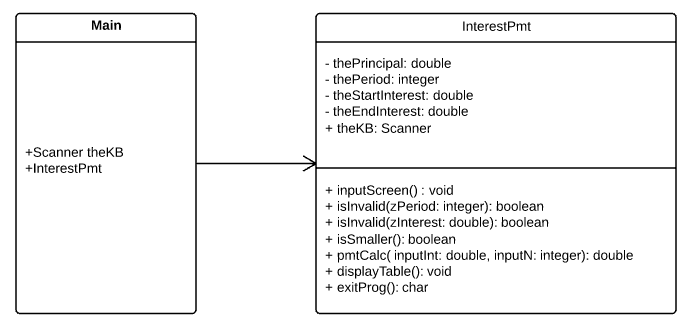
Open up the command line. Enter the below command

java -jar Assignment1Q1\_Bankloan.jar

# Structure:

The project has a simple structure, it has a client and a class where the functionality of the application is programmed.

# Design



# Algorithm

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Authors: Rebecca Lim

Date: 27/08/2017

Purpose: The bank loan application

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

Begin

Class InterestPmt

declare private double thePrincipal

declare private integer double thePeriod

declare private double theStartInterest

declare private double theEndInterest

public function double getThePrincipal

return thePrincipal

end function

public function void setThePrincipal(double thePrincipal)

this.thePrincipal = thePrincipal

end function

public function integer getThePeriod

return thePeriod

end function

public function void setThePeriod(int thePeriod)

this.thePeriod = thePeriod

end function

public function double getTheStartInterest

return theStartInterest

end function

public function void setTheStartInterest(double theStartInterest)

this.theStartInterest = theStartInterest

end function

public function double getTheEndInterest

return theEndInterest

end function

public function void setTheEndInterest

return theEndInterest

end function

public function void studentInfo

print "Name: Rebecca Lim"

print "Student Number: 33111264"

print "Enrolment: Internal"

print "Tutor: Mark Abernethy"

print "Tutorial: Thursday, 10.30am"

end function

public function void inputScreen

declare boolean theFlag = false

do

print "Please enter the principal amount: "

read thePrincipal

if(isInvalid(thePrincipal))

print "Please enter a number greater than 0"

end if

while (isInvalid(thePrincipal))

do

print "Please enter the rate of the starting interest:"

read theStartInterest

if(isInvalid(theStartInterest))

print "Please enter a number greater than 0"

end if

while (isInvalid(theStartInterest))

do

print "Please enter the rate of the ending interest"

read theEndInterest

if(this.isSmaller())

print "Please enter an interest greater than the starting interest"

theFlag = true

else

theFlag = false

end if

if(this.isInvalid(theEndInterest))

print "Please enter a number greater than 0"

theFlag = true

end if

while(theFlag)

do

print "Please enter the number of years"

read thePeriod

if(isInvalid(thePeriod))

print "Please enter a number greater than 0"

end if

while (isInvalid(thePeriod))

end function

public function boolean isInvalid(int zPeriod)

declare boolean theFlag = false

if(zPeriod <= 0)

theFlag = true

end if

return theFlag

end function

public function boolean isInvalid(double zAmount)

declare boolean theFlag = false

if(zAmount <= 0)

theFlag = true

end if

return theFlag

end function

public function boolean isSmaller

declare boolean theFlag = false

if(theStartInterest > theEndInterest)

theFlag = true

end if

return theFlag

end function

public function double pmtCalc(double inputInt, int inputN)

declare double calcPmt

calcPmt = (intputInt \* thePrincipal) / (1-((1+inputInt)^

(-1\*inputN)))

return calcPmt

end function

public function void displaytable

declare double thePmt

declare double totPmt

print "Loan Amount:", thePrincipal

print "Number of years: ", thePeriod

print " %20s %20s %20s \n", "Interest Rate", "Monthly Payment", "Total Payment"

while(theStartInterest < = theEndInterest)

declare double r = (theStartInterest / 12)/100

declare integer n = thePeriod \* 12

thePmt = pmtCalc(r,n)

totPmt = thePmt\*n

print "%20.2f %20.2f %20.2f \n", theStartInterest, thePmt, totPmt

theStartInterest = theStartInterest + 0.25

end while

end function

end Class

class Assignment1Q1\_BankLoan

declare character theFlag = 'y'

new InterestPmt()

do

thePmt.inputScreen()

thePmt.displayTable()

theFlag = thePmt.exitProg()

while(theFlag == 'y')

end class

end

# Limitations

The limitation of this application is that it is unable to accept alphabetic characters on input. Using a try-catch it is able to catch the exception, however the use of the try-catch is out of the scope of this assignment.