Assignment #3: Object-Oriented Programming

ISDS 558, Prof. Robin Burke
Spring 2002
Due: 3/19/2002

Objective:
Construct an interface, implementing classes, a collection class and a testing application.

What to do:
In this assignment, you will develop four class modules.

1. ILoan interface. This class module has no associated implementation, but defines the API that other classes obey. Here is its API:
   Methods:
   Init (Principal As Currency, Rate As Double, Periods As Integer)
   Pay ( )
   Properties:
   Payment
   LoanID
   StartingBalance
   Balance
   BalancePaid
   InterestPaid
   Rate
   CurrentPeriod
   IsPaid
   Periods
   LoanType
   Note that all properties except for LoanID are read-only. That is, they should have Get property procedures associated with them, but not Let.

2. CLoanFixed class module. This module will need to implement ILoan and provide code for all of the methods and properties given in the ILoan interface to implement an amortized loan. Assume that each call to Pay results in a payment of the full payment amount. Objects of this class should always return LoanType = 0

3. CLoanAdjustable class module. This module will be much like CLoanFixed, but it will implement an adjustable rate loan. Objects of this type should always return LoanType = 1. In addition to the functionality in CLoanFixed, this object will have the following additional API:
   Methods:
   InitTerms (LifetimeCap As Double, PeriodicCap As Double)
   Properties:
   LifetimeCap
   PeriodicCap
   ReachedLifetimeCap
   ReachedPeriodicCap
   In addition, the Rate property of the CLoanAdjustable is a read/write property.

4. CLoans collection. This class module will implement a collection of loans. Follow the implementation described on pages 268-275 of the book. Use the LoanID property as the unique key.

5. Build an application that demonstrates your objects. The application will have two forms. The LoanList form will contain a grid control listing all of the loans that have been defined. The contents of the grid will come from a CLoans collection object. This form should have a button to add new loans to the collection and a button to make one payment on all of the loans. The LoanEntry form will contain a form in which a new loan can be defined. Use a radio button to select between fixed and adjustable and enable/disable fields for LifetimeCap and PeriodicCap as appropriate.
What to turn in:
Before class time, place your project and all associated files (including the database) in a folder titled as in the following example “RBurke_Hwk3”. (Use your first initial and last name, of course.) Put this folder in the course Drop Box on Doctor (\Doctor\Assignments\Burke\isd558\HWK3\). If you make an error and want to resubmit, that is OK, but you won’t be able to delete, modify or overwrite any folders you have already submitted. Submit a second time, adding a letter to the end of the folder name (e.g. “RBurke_Hwk3A”). I will grade the most recent version submitted (up to class time).

Assessment
This assignment will be assessed on the completeness of the solution to the problem, on the clarity of the user interface and on the quality of the coding style. Partial solutions will be given partial credit.

Hints and Notes:
• In an adjustable loan, the lifetime cap is the amount that the interest rate can rise over the lifetime of the loan. So the maximum rate on the loan is (startingRate + LifetimeCap). If a caller attempts to raise the interest rate beyond this cap, the property procedure should set the rate to the maximum rate.
• The periodic cap on an adjustable loan is the amount that the interest rate can rise over a single payment period. If a caller attempts to raise the interest rate beyond this amount, it should be set to the maximum. This means you have to keep track of what the interest rate was after each call to Pay, so you’ll know how high it can go.
• If a loan is paid off, Pay should do nothing. No negative balances!

Extra Credit:
• Allow editing of the interest rate of adjustable loans (but no other properties) in the grid control.