30/30 points for Program 3

Excellent work, overall. Very well thought-out and well-organized solution. Output was clearly painstakingly formatted.

```
/*****************************************************************************************
 * This program takes input data from a file called "loandata.txt"
 * displays the information to the console, and accumulates the information,
 * displaying totals at the end.
 *
 * Program by Michael Clinesmith
 * CST 183 Programming Assignment 3
 *****************************************************************************************/
import java.util.Scanner;
import javax.swing.JOptionPane;
import java.io.*;

public class LoanData
{

    public static void main(String[] args) throws IOException
    {
        final String FILE_NAME = "loandata.txt";                                            // loan data filename
        String name, creditRatingLabel;                                                     // loan data variables
        double principal, annualRate, monthlyRate, feeRate, monthlyPayment, payoff, feeDollars;
        int term, termMonths, creditRating;

        double totalPrincipal = 0, totalPayment = 0, totalPayoff = 0, totalFee = 0;     // accumulator variables
        int validRecords = 0;                                                               // counter variable
        String initialDisplay, recordDisplay, finalDisplay;                                 // display variable
        File loanData;                                                                      // input file variable

        // introductory message

        JOptionPane.showMessageDialog(null, "This program processes loan data located " +
                                    "in the file loandata.txt\n" +
                                    "and displays the information and totals to the console.");

        // open file

        loanData = new File(FILE_NAME);

        if(!loanData.exists())  // file not found
        {

            JOptionPane.showMessageDialog(null, "loandata.txt does not exist for processing.\n" +
                                    "The program will now end.");
            System.exit(0);
        }

        Scanner inputFile = new Scanner(loanData);

        //  display first heading
        initialDisplay =    String.format("%-12s","Customer") + "   " +
                            String.format("%-13s","Principal") + "   " +
                            String.format("%-5s","Rate") + "   " +
                            String.format("%-5s","Years") + "   " +
                            String.format("%-10s","Payment") + "   " +
                            String.format("%-13s","Payoff") + "   " +
                            String.format("%-10s","Fee") + "   " +
                            String.format("%-15s","Credit Rating");

        System.out.println(initialDisplay);

        //  input and process data
```

```
while(inputFile.hasNext())
{
    name = inputFile.next();                // input next record
    principal = inputFile.nextDouble();
    term = inputFile.nextInt();
    annualRate = inputFile.nextDouble();
    creditRating = inputFile.nextInt();

    if (creditRating < 580)                 // extra token for very poor credit ratings
    {
        feeRate = inputFile.nextDouble();
    }
    else
    {
        feeRate = 0.0;
    }
    validRecords++;                         // increase valid records count

    // process record

    termMonths = term * 12;
    monthlyRate = annualRate / 1200;        // also converts to a decimal value from percentage

                                            // monthly payment formula
    monthlyPayment = monthlyRate * principal / (1 - Math.pow(1 + monthlyRate, -termMonths));
    payoff = monthlyPayment * termMonths;
    feeDollars = feeRate / 100 * principal; // fee rate must be converted to a decimal value

    if (creditRating<300 || creditRating>850)   // display text rating based on credit range
    {
        creditRatingLabel = "Invalid Rating";
    }
    else if (creditRating<580)
    {
        creditRatingLabel = "Very Poor";
    }
    else if (creditRating<670)
    {
        creditRatingLabel = "Fair";
    }
    else if (creditRating<740)
    {
        creditRatingLabel = "Good";
    }
    else if (creditRating<800)
    {
        creditRatingLabel = "Very Good";
    }
    else
    {
        creditRatingLabel = "Exceptional";
    }

    // accumulate data
    totalPrincipal += principal;
    totalPayment += monthlyPayment;
    totalPayoff += payoff;
    totalFee += feeDollars;

    // display record
```

Nicely organized and structured code.

```
            recordDisplay =     String.format("%-12s",name) + "   " +
                                String.format("$%,12.2f", principal) + "   " +
                                String.format("%4.1f%%",annualRate) + "   " +
                                String.format("%5d", term) + "   " +
                                String.format("$%,9.2f", monthlyPayment) + "   " +
                                String.format("$%,12.2f", payoff) + "   " +
                                String.format("$%,9.2f", feeDollars) + "   " +
                                String.format("%-15s",creditRatingLabel);

            System.out.println(recordDisplay);

        }

        // display totals

            System.out.println("----------------------------------------------------------------------------------------");

        finalDisplay =      String.format("%-12s","Totals") + "   " +
                            String.format("$%,12.2f", totalPrincipal) + "   " +
                            String.format("%5s"," ") + "   " +
                            String.format("%5s", " ") + "   " +
                            String.format("$%,9.2f", totalPayment) + "   " +
                            String.format("$%,12.2f", totalPayoff) + "   " +
                            String.format("$%,9.2f", totalFee);


        System.out.println(finalDisplay);
        System.out.println(validRecords + " records processed.");

    }
}
```

Output looks excellent.  Great formatting.  Numbers spot-on with expected values.

```
Customer      Principal      Rate  Years  Payment      Payoff         Fee        Credit Rating
SMITH         $   20,000.00   4.5%     5  $    372.86  $   22,371.62  $     0.00 Very Good
JOHNSON       $   18,000.00   4.2%     4  $    408.04  $   19,585.72  $   360.00 Very Poor
WILLIAMS      $   50,000.00   5.6%     6  $    819.24  $   58,985.01  $     0.00 Fair
BROWN         $   12,000.00   3.3%     5  $    217.23  $   13,033.67  $     0.00 Fair
JONES         $    8,000.00   5.5%     3  $    241.57  $    8,696.42  $     0.00 Very Good
MILLER        $   16,000.00   5.9%     6  $    264.41  $   19,037.63  $   480.00 Very Poor
DAVIS         $   75,000.00   6.6%    12  $    755.39  $  108,775.66  $ 1,500.00 Very Poor
GARCIA        $   80,000.00   3.3%    15  $    564.08  $  101,534.60  $     0.00 Good
RODRIGUEZ     $   15,000.00   6.9%    10  $    173.39  $   20,806.87  $     0.00 Good
WILSON        $   32,000.00   4.2%     5  $    592.22  $   35,533.28  $     0.00 Very Good
MARTINEZ      $   77,000.00   2.9%    15  $    528.05  $   95,049.43  $     0.00 Very Good
ANDERSON      $   14,000.00   4.5%     6  $    222.24  $   16,001.02  $   140.00 Very Poor
TAYLOR        $    4,000.00   1.9%     4  $     86.61  $    4,157.09  $     0.00 Very Good
THOMAS        $    8,000.00   2.5%     6  $    119.77  $    8,623.31  $     0.00 Exceptional
HERNANDEZ     $   33,000.00   5.9%     7  $    480.50  $   40,362.16  $     0.00 Exceptional
MOORE         $    9,000.00   3.3%     5  $    162.92  $    9,775.25  $     0.00 Good
MARTIN        $   25,000.00   4.9%    10  $    263.94  $   31,673.22  $   500.00 Very Poor
JACKSON       $    5,000.00   1.9%     3  $    142.99  $    5,147.81  $     0.00 Exceptional
THOMPSON      $   66,000.00   2.5%    15  $    440.08  $   79,214.56  $     0.00 Very Good
WHITE         $   88,000.00   3.3%    30  $    385.40  $  138,744.21  $     0.00 Exceptional
-------------------------------------------------------------------------------------------
Totals        $  655,000.00               $ 7,240.92  $  837,108.55  $ 2,980.00
```

```
SOLUTION

Name          Principal  Rate   Years  Payment   Payoff       Fee    Credit Rating
SMITH          20000.00   4.5     5    372.86    22371.62    0.00    Very Good
JOHNSON        18000.00   4.2     4    408.04    19585.72  360.00    Very Poor
WILLIAMS       50000.00   5.6     6    819.24    58985.01    0.00    Fair
BROWN          12000.00   3.3     5    217.23    13033.67    0.00    Fair
JONES           8000.00   5.5     3    241.57     8696.42    0.00    Very Good
MILLER         16000.00   5.9     6    264.41    19037.63  480.00    Very Poor
DAVIS          75000.00   6.6    12    755.39   108775.66 1500.00    Very Poor
GARCIA         80000.00   3.3    15    564.08   101534.60    0.00    Good
RODRIGUEZ      15000.00   6.9    10    173.39    20806.87    0.00    Good
WILSON         32000.00   4.2     5    592.22    35533.28    0.00    Very Good
MARTINEZ       77000.00   2.9    15    528.05    95049.43    0.00    Very Good
ANDERSON       14000.00   4.5     6    222.24    16001.02  140.00    Very Poor
TAYLOR          4000.00   1.9     4     86.61     4157.09    0.00    Very Good
THOMAS          8000.00   2.5     6    119.77     8623.31    0.00    Exceptional
HERNANDEZ      33000.00   5.9     7    480.50    40362.16    0.00    Exceptional
MOORE           9000.00   3.3     5    162.92     9775.25    0.00    Good
MARTIN         25000.00   4.9    10    263.94    31673.22  500.00    Very Poor
JACKSON         5000.00   1.9     3    142.99     5147.81    0.00    Exceptional
THOMPSON       66000.00   2.5    15    440.08    79214.56    0.00    Very Good
WHITE          88000.00   3.3    30    385.40   138744.21    0.00    Exceptional
              655000.00               7240.92   837108.55 2980.00
```