```java
   Java 1          Basic Mortgage calculator
import java.io.*;

public class MortgageNew
{

    public static void main(String[] args)throws IOException
    {

        //declare variables
        double loan1[],loan2[],loan3[];

        //call introduction method
        introduction();

        //call createLoan method
        loan1 = createLoan();
        loan2 = createLoan();
        loan3 = createLoan();

        //make term of loans into months
        loan1[2] = getNumberOfMonths(loan1[2]);
        loan2[2] = getNumberOfMonths(loan2[2]);
        loan3[2] = getNumberOfMonths(loan3[2]);

        //make the interest rate into monthly interest for each loan
        loan1[1] = getMonthlyInterestRate(loan1[3]);
        loan2[1] = getMonthlyInterestRate(loan2[3]);
        loan3[1] = getMonthlyInterestRate(loan3[3]);

        //calculate the monthly payment for each loan
        loan1[4] = payEachMonth(loan1[1],loan1[0],loan1[2]);
        loan2[4] = payEachMonth(loan2[1],loan2[0],loan2[2]);
        loan3[4] = payEachMonth(loan3[1],loan3[0],loan3[2]);

        //calculate the total loan payments for each loan
        loan1[6] = getLoanPayment(loan1[2],loan1[4]);
        loan2[6] = getLoanPayment(loan2[2],loan2[4]);
        loan3[6] = getLoanPayment(loan3[2],loan3[4]);

        //calculate the total interest paid on each loan
        loan1[5]= getTotalInterest(loan1[6],loan1[0]);
        loan2[5]= getTotalInterest(loan2[6],loan2[0]);
        loan3[5]= getTotalInterest(loan3[6],loan3[0]);

        //display table
        System.out.println("Loan number Loan Amount Rate Term Monthly Payment Total Interest Total Cost");
        System.out.printf("   Loan1     %.2f   %.0f%%   %.0f   %.2f       %.2f   %.2f\n",loan1[0],loan1[3],loan1[2],loan1[4],loan1[5],loan1[6]);
        System.out.printf("   Loan2     %.2f   %.0f%%   %.0f   %.2f       %.2f   %.2f\n",loan2[0],loan2[3],loan2[2],loan2[4],loan2[5],loan2[6]);
        System.out.printf("   Loan3     %.2f   %.0f%%   %.0f   %.2f       %.2f   %.2f\n",loan3[0],loan3[3],loan3[2],loan3[4],loan3[5],loan3[6]);

    }//end main

    /**     introduction method
     *      This method takes no parameters
     *      This method returns nothing
     *      This method is used to create an introduction screen (splash)
     *      It is it's own method so that it can be easy to change in later weeks when
     *      it is no longer a "basic mortgage calculator, and so I can easily keep track of the
     *      version number
     */

    public static void introduction()
    {
        System.out.println("\t\t   Basic Mortgage Calculator v4.0");
        System.out.println();
    }//end Introduction

    /**
     * This method takes no parameters, but instead creates an array that includes the loan amount,
     * term of the loan, and interest rate of the loan. This will be used to create
     * a table that shows information from all three loans.
     * @return loan[]
     * @throws IOException
     */

    public static double[] createLoan()throws IOException
    {
        //declare variables
        String totalLoan, percentInterest,numYears;
        double[] loan = new double[7];

        //constructor for user input reader
        BufferedReader dataIn = new BufferedReader(new
 InputStreamReader(System.in));

        // print prompts and get input
        System.out.print("\t\tPlease enter the total loan amount:
");
            totalLoan = dataIn.readLine();
            loan[0] = Double.parseDouble(totalLoan);

        System.out.print("\t\tPlease enter the interest rate in
percent: ");
            percentInterest = dataIn.readLine();
            loan[3] = Double.parseDouble(percentInterest);

        System.out.print("\t\tPlease Enter the term of the loan
in years: ");
            numYears = dataIn.readLine();
            loan[2] = Double.parseDouble(numYears);

        System.out.println();

        return loan;
    }//end createLoan

    /**
     * This method takes the term in years and makes it the term
 in months
     * @param numberOfYears
     * @return numberOfMonths
     */

    private static double getNumberOfMonths(double numberOfYears)
    {

        double numberOfMonths = (numberOfYears * 12);
        return numberOfMonths;
    }//end getNumberOfMonths

    /**
     * This method takes the annual interest rate and generates the monthly interest rate
     * @param yearlyInterest
     * @return monthlyInterest
     */

    private static double getMonthlyInterestRate(double yearlyInterestRate)
    {
        double monthlyInterestRate = (yearlyInterestRate / (12 * 100));
        return monthlyInterestRate;
    }//end getMonthlyInterestRate

    /**
     * This method uses the formula: M = P * ( J / (1 - (1 + J) ** -N))
     * where M is the monthlyPayment, P is the totalLoan, J is the monthlyInterestRate,
     * and N is the NumMonths. The formula should then result in
     * returning the total monthly payment.
     * @param monthlyInterstAmount
     * @param totalLoanAmount
     * @param lengthOfLoan
     * @return payPerMonth
     */

    private static double payEachMonth(double monthlyInterestAmount,double totalLoanAmount,double lengthOfLoan)
    {
        double payPerMonth = totalLoanAmount * ( monthlyInterestAmount / (1 - Math.pow(1 + monthlyInterestAmount, (-lengthOfLoan))));
        return payPerMonth;

    }//end payEachMonth

    /**
     * This method multiplies the term by the monthlyPayment to determine the total
     * paid on the loan.
     * @param term
     * @param monthlyPayment
     * @return
     */
    private static double getLoanPayment(double term ,double monthlyPayment)
    {
        double totalCost = (term * monthlyPayment);
        return totalCost;

    }//end getLoanPayment

    /**
     * This method takes the total cost of the loan and subtracts the original loan amount
     * in order to determine the amount of interest that was paid on the loan.
     * @param totalCost
     * @param loanAmount
     * @return
     */
    private static double getTotalInterest(double totalCost, double loanAmount)
    {
        double totalInterest = (totalCost - loanAmount);
        return totalInterest;

    }//end getTotalInterest

}//end class



```