Analysis:

The method should compute the formula:

$$futureInvestmentValue = investmentAmount \times (1 + monthlyInterestRate)^{numberOfYears \times 12}$$

This will give us the future investment value at a given interest rate (input) and investment amount (input) for a specified number of years (1-30). We then will print out a table of the future value from the years 1 to 30.

Design:

First, we create the method to compute the future investment value using the given formula rounded to the cents place. The formula calls for the monthly interest rate, and since the user inputs the annual interest rate, we need a method that does this conversion. In this method, we get rid of the percent sign, convert to a decimal, and divide by 12 (number of months in a year). In our main (test) program, we create a scanner object and receive the necessary input from the user. We save the monthly interest rate into a variable and set our specified number of years for the table to equal 30. We then calculate each year's investment value and save the values into an array. To print the table, we make a separate function (although not necessary for this particular problem, but will allow for easy manipulation of the number of rows in the table). This function prints a correctly formatted, left justified table with indexes for each row and each year's investment value.

Testing:

To test the program, we used the example input values in the problem: 1000 for the amount invested and 9% for the annual interest rate. We also tested the futureInvestmentValue() method by using the example in the problem:

futureInvestmentValue(1000, 0.05/12, 5). Furthermore, to test the getMonthlyInterest() method, we used a decimal value and a double digit percentage to cover all possible inputs. It also works with negative interest values and zero. There is an exception when the user inputs strings and symbols.