|  |  |  |  |
| --- | --- | --- | --- |
| **Instructor** | ***Katherine Papademas*** | **Due Date** | **9/8/2016** |

**PROJECT Financial Interest - Using Methods with Java 50 points**

**Objective** To type, compile and execute a program that computes simple interest.

***PROJECT DESCRIPTION***

This project has you running a program that uses methods ( modular programming ) to compute simple interest.

***Information About This Project***

According to the Mathematics of Finance, simple interest is defined as interest, which is based upon the original principal only. The formula to compute simple interest is given by

Interest Earned = Principal × Rate × Time

or algebraically,

*I* = *P* · *R* · *T*

where *I* is the Interest earnings, *P* is the Principal or the amount invested, *R* is the annual interest Rate and *T* is the Time that the money is kept on deposit, expressed in terms of years

Note: that if *R* is given as an annual interest rate, then the time *T* must be expressed in years.

Under the rules of simple interest, the amount in the account *A* after a principal *P* has been invested for a period of *T* years is given as:

*A* = *P* + *I*

Since *I* = *P* · *R* · *T* then the above formula can also be expressed as:

*A* = *P* + *P* · *R* · *T*

or in the familiar factored form:

*A* = *P* ( 1 + *P* · *T* )

This last formula is useful to determine one of the variables *A* , *P* , *R* or *T* when the remaining three variables are known.

***Steps To Complete This Project***

**STEP 1**  **Open Eclipse, JCreator or Similar Java Development Environment**

Open Eclipse or similar Java text editor. Here are the steps to create a Java project for this assignment using Eclipse for Windows.

**Using Eclipse**

From the Eclipse main menu, click File , point to New and point to and click on Java Project . When the **New Java Project** dialog box opens, select and fill the settings as follows:

Set the project name : **Lab 02**

Set the workspace location : ( use the default location )

Click Finish to close the **New Java Project** dialog box.

Now ensure that the Eclipse **Project Explorer** window is open by selecting the main menu item Window , point to Show View and click Project Explorer .

Navigate to the **Project Explorer**.

**PROJECT Financial Interest - Using Methods with Java**

Double - click the project name ( **Lab 02** ) to reveal the **src** ( source ) folder.

Right - click the **src** folder and select New from the menu and point to and click on Class . When the **New Java Class** dialog box opens, name the source file as: **Simple**

Into the **Code** window, shown below, copy the program code shown in **Figure 1** ,which follows, exactly as it appears, except substitute your own name in place of Sammy Student. Notice that the class name in the source code must match the Java file name.

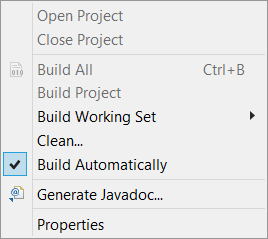
**STEP 2**  **Dissect the Given Program Code**

Examine the source code and comments given in **Figure 1** , which follows. The program code consists of a class containing global variables and a class method named **getData()** , which is used to prompt the user and receive the required data from the user. **getData()** is used to update the global variables. The **main()** method is used to call the **getData()** method and to perform the required computation of the program, namely to compute the interest earnings, given a principal, rate and time values.

**STEP 3 Build, Compile and Run the Program**

Let us now test the initial source code given in **Figure 1** .

From the Eclipse Project menu, ensure that the option Build Automatically is selected. This will indicate that Eclipse will automatically build your project when you run it.



Now select Run on the menu bar, point to Run As and then point to and click

on Java Application .

If you do not have any errors, proceed to the next step, otherwise read the error messages and make any necessary corrections by comparing your screen code to

the original code shown within **Figure 1** . Then run your program again.

**PROJECT Financial Interest - Using Methods with Java**

**Figure 1 Source Code for the Simple Interest Program**

|  |
| --- |
| **// Program to calculate Simple Interest.**  **// Programmer: Sammy Student, File Name: Simple.java**  **// the packages**  **import java.text.DecimalFormat;**  **import javax.swing.JOptionPane;**  **// the class definition**  **public class Simple**  **{**  **// the global variables are declared**  **static double earnings = 0, principal = 0, rate = 0, time = 0;**  **// introduce a DecimalFormat object**  **static DecimalFormat twoPlace = new DecimalFormat("0.00");**    **// the method to obtain data**  **public static void getData()**  **{**  **// local variables declared and assigned initial values**  **String firstNum = "", secondNum = "", thirdNum = "";**  **// local variables updated via prompt boxes**  **firstNum = JOptionPane.showInputDialog("Enter the Principal");**  **secondNum = JOptionPane.showInputDialog("Enter Rate, as a %");**  **thirdNum = JOptionPane.showInputDialog("Enter Time, in years");**    **// update the global variables**  **principal = Double.parseDouble(firstNum);**  **rate = Double.parseDouble(secondNum);**  **time = Double.parseDouble(thirdNum);**  **}**    **// the main() method**  **public static void main(String args[])**  **{**  **// call the getData() method**  **getData();**    **// update a global variable**  **earnings = principal \* rate / 100 \* time;**  **// display the result in a message box**  **JOptionPane.showMessageDialog(null, "interest earnings: $" +**  **twoPlace.format(earnings), "Result", JOptionPane.PLAIN\_MESSAGE);**    **System.exit(0);**  **}**  **}** |

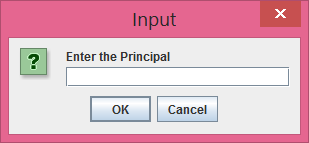
**PROJECT Financial Interest - Using Methods with Java**

**STEP 4**  **Test Your Program**

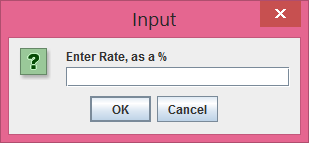
With your program running, test it by using the input information given below.

|  |  |  |
| --- | --- | --- |
| Principal | $ 30,000.00 | **input** |
| Rate | 6.5 % |  |
| Time | 5 years |  |
| Interest Earnings | $ 9,750.00 | **output** |

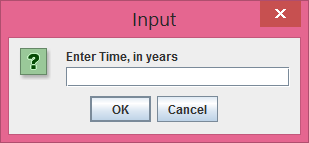
First enter the amount of the principal.



Then enter the rate, as a percent.

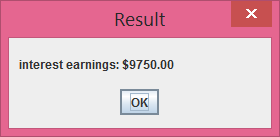


Finally, enter the time of deposit or term of the loan.



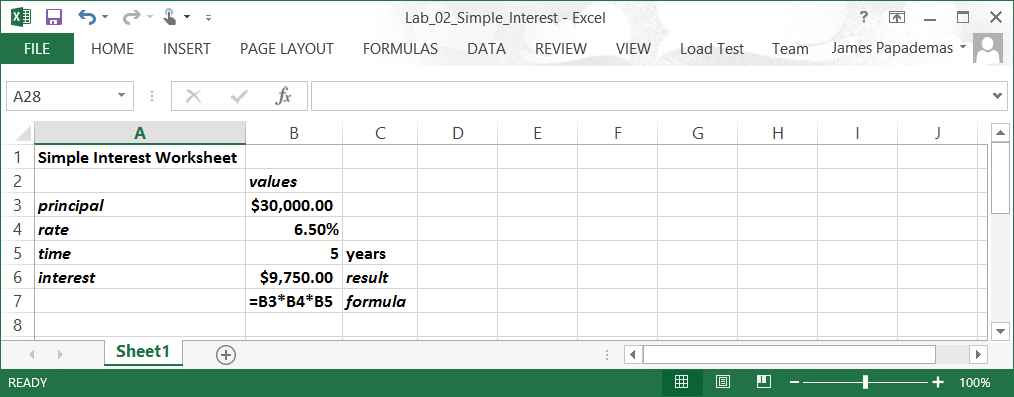
**PROJECT Financial Interest - Using Methods with Java**

Your program should then display the correct output, as was given above.



**STEP 5 Verify Your Output**

You can construct a worksheet, using MS Excel or equivalent spreadsheet software, to verify the operation of the initial source code for this project.



**STEP 6**  **Modify Your Program**

With your program working correctly, modify the code such that not only the interest earnings will be displayed but also the accumulated amount in the account after the time period has passed. Similar to the interest earnings, display the output in a **JOptionPane** . The accumulated amount is to be computed via a new Java method.

Modify your program according to these requirement specifications:

• Within the class **Simple** definition, but outside any other method declaration or definition, create a new **public** **static** method and name it as:

**calcEarnings()**

The above method is to be declared with a **void** return type.

Move the following block of code, from the **main()** method, such that it appears within the **calcEarnings()** method.

**// update a global variable**

**earnings = principal \* rate / 100 \* time;**

**PROJECT Financial Interest - Using Methods with Java**

Have this method called within the **main()** method.

• Within the class **Simple** definition, but outside any other method declaration or definition, create a new **public** **static** method and name it as:

**calcAccumulated()**

The above method is to be declared with a **void** return type.

To use this method, first declare a global variable that will be assigned the accumulated amount, i.e. the sum of the interest earnings and the principal.

Have this method called within the **main()** method.

• Within the class **Simple** definition, but outside any other method declaration or definition, create a new **public** **static** method and name it as:

**showAccumulated()**

The above method is to be declared with a **void** return type.

To use this method, first declare a global class variable that will be assigned the accumulated amount, i.e. the sum of the interest earnings and the principal.

Move the following block of code, from the **main()** method, such that it appears within the **showAccumulated()** method. Modify the statement accordingly such that it will perform the task of its method.

**// display the result in a message box**

**JOptionPane.showMessageDialog(null, "interest earnings: $" + twoPlace.format(earnings), "Result", JOptionPane.PLAIN\_MESSAGE);**

Have this method called within the **main()** method.

• Your **main()** method should now just be comprised of four method calls - for the **getData()** method and the above three new methods.

**STEP 4**  **Run the Modified Program**

Next, run your modified program for each of the scenarios listed below. Take individual screen snapshots for each of the scenarios and paste them into an MS Word document. To take screen snapshots and paste them into MS Word, use the instructions which follow.

With your output screen open still open and visible in your program run mode, press and hold your keyboard Alt key and then press your Print Screen key. Then, leaving your program open, paste the screen snapshot into a word processor such as MS Word by opening MS Word and, in a new document, pressing and holding your Ctrl key and then pressing your V key.

**PROJECT Financial Interest - Using Methods with Java**

|  |  |  |  |
| --- | --- | --- | --- |
| **(a)** | Principal | $ 10,000.00 |  |
|  | Rate | 3.0 % |  |
|  | Time | 10 years |  |
|  | Interest Earnings | $ |  |
|  | Amount in Account | $ |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **(b)** | Principal | $ 500.00 |  |
|  | Rate | 3.25 % |  |
|  | Time | 1 year |  |
|  | Interest Earnings | $ |  |
|  | Amount in Account | $ |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **(c)** | Principal | $ 1,500.00 |  |
|  | Rate | 1.75 % |  |
|  | Time | 5 years |  |
|  | Interest Earnings | $ |  |
|  | Amount in Account | $ |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **(d)** | Principal | $ 9,200.00 |  |
|  | Rate | 0.72 % |  |
|  | Time | 8 years |  |
|  | Interest Earnings | $ |  |
|  | Amount in Account | $ |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **(e)** | Principal | $ 800.00 |  |
|  | Rate | 3.00 % |  |
|  | Time | 9 months |  |
|  | Interest Earnings | $ |  |
|  | Amount in Account | $ |  |

**STEP 3 Submit Your Program Code**

Submit your screen snapshots document of your runs for credit in MS Word. Include your program source code within this document as well.