|  |
| --- |
| **CS1101 Lab 10 – 1st Comprehensive Lab** |

**Released on: Monday October, 9**   
**Due on: Wednesday October, 25**  
**How:** submit on Piazza in folder lab10  
**What:** a zipped file containing a java file as a solution to your problem and a word file

**OBJECTIVE OF THIS LAB.** This comprehensive lab will enable you to apply, in a practical scenario, the topics introduced in the class so far: input/output, algorithms, variables, conditionals, loops, and methods.

|  |
| --- |
| You have been hired by Wells Fargo Bank to create a program that will serve as a Loan Evaluator. The loan evaluation consists of the following steps:  1/ Get customer information:   * Full name, * Age, * Occupation, * Work place/company, * Social Security Number (SSN), and * Monthly income.   The social security number is valid only if it is composed of 6 characters: three consonants and 3 digits. The first character should be a capital letter consonant, e.g., PW46y3 and T12tV9.  After getting personal information of customer, display "Welcome to our Bank John Doe", assuming the name of the customer in this case is John Doe.  Note: you need to validate the customer’s SSN. Here is how:  If a customer has a valid SSN, the program will keep running. However, a customer has three attempts to input a valid SSN. After three unsuccessful attempts, the program should display "Loan denied - invalid Social Security Number!" and end.  2/ Request the type of the loan from the user.  Get this information from the user and validate that the type of loan is “home", "car", or "land”.   * If the type is different from these three options, the loan will be denied. * If the loan is denied, the message "Loan denied – type of loan is not valid." should be displayed on the screen and the program should end.   Note: Create a method. You must create a method that:   * receives a string indicating the destination of the credit, for instance "home", and * returns a Boolean value indicating if the credit is denied (false) or not (true).   If the loan is denied, the “denied” message should be displayed by the main method and the program should end. If the loan is granted, then the program should proceed to the next step.  3/ Evaluate the ability of the customer to pay the loan.  For this task, the program will ask the user for his or her monthly expenses.   * If customer’s monthly expenses are less than or equal to 35% of the customer’s monthly income, the loan will be approved with a maximum amount of 10 times the customer’s monthly income. * If customer’s monthly expenses are more than 35% and less than or equal to 50% of the customer’s monthly income, the loan will be approved with a maximum amount of 5 times the customer’s monthly income. * If the loan is approved, a message should be displayed on the screen congratulating the customer, indicating the loan was approved and the amount. For more details, please refer to the test cases. * If the customer’s monthly expenses are more than 50% of the customer’s monthly income, the loan is denied. If the loan is denied, the following message should be displayed on the screen: "Loan denied – customer has high monthly expenses." See example below.   4/ Calculate the loan’s duration (i.e., the number of monthly payments) and the total interest amount that the customer will pay.  For this task, you have to ask the customer for the annual percentage rate (APR) and the monthly amount the customer will pay monthly towards the loan.  Note: Create a method for each of these.   * Create a method that takes as parameters the values of r, A, and P (as described below) and returns N, the number of monthly payments needed. * Create another method that takes as parameters the values of P, N, and A, and returns I, the total amount of interest the customer will pay. |

**HELP:**

Calculate the loan duration (in months) with the following formula:   
Number of payments (N) is calculated as:

Where:

* r is the monthly interest rate (which is = APR / 12);
* *A* is the loan amount (principal), calculated in step 3; and
* P is the monthly payment.

Example 1. Assume that the APR is 10%, the loan is $10,000, and the customer will make a monthly payment of $500.

Then r = (10/100)/12, A= 10,000, P = 500.

As a result: N = -log(1 - 0.00833\*10000/500) / log(1 + 0.008333) = 21.97 payments (rounded to two decimals). The system should display “Customer’s total payments are 21.97 months.”

Calculate total interest (*I)* with the following formula:

Where:

* P is the monthly payment;
* N is the number of payments (from the first formula, above); and
* A is the loan amount (principal).

Example 2. Assume that the monthly payment is $500, the number of payments is 21.97, and the loan amount is $10,000.

Then: I = 500\*21.97 - 10,000.00 = 985.00. The system will display: "Total interest the customer will pay is $985.00."

**TEST CASES**

Use the following test cases to trace and test your program. Note that TAs may request more test cases from you.

Test case 1: John Doe wants to get a loan. The following lines show the interaction with the Loan Evaluator:

Step 1: Getting customer's information:

Enter your name: John Doe

Enter your age: 35

Enter your occupation: Teacher

Enter the workplace: UTEP

Enter the Social Security Number(6 characters): T34o3v

Enter the monthly income: $5200.00

"Welcome to our Bank John Doe"

Step 2: Loan type ("home", "car", or "land"):

What is the purpose of the loan? home

Step 3: Budget and ability to pay the credit:

What are your monthly expenses? $1900.00

Step 4: Loan duration and total of interest to be paid:

Enter the Annual Percentage Rate (APR): 10%

Enter the monthly payment: 500.00

Step 5: Notification to user if the loan was approved or not:

Congratulations Mr. John Doe, your loan was approved.

The total amount to be loan is: 26,000.00.

Customer’s total payments are 68.44 months.

Total interest the customer will pay is $8220.85.

Test case 2: Mary Doe wants to get a loan. The following lines show the interaction with the Loan Evaluator:

Step 1: Getting customer's information:

Enter your name: Mary Doe

Enter your age: 28

Enter your occupation: Nurse

Enter the workplace: UTEP

Enter the Social Security Number(6 characters): S3q80k

Enter the monthly income: $7350.00

"Welcome to our Bank Mary Doe"

Step 2: Loan type ("home", "car", or "land"):

What is the purpose of the loan? Nursing equipment and supplies.

"Loan denied – type of loan is not valid."

Test case 3: Frank Johnson wants to get a loan. The following lines show the interaction with the Loan Evaluator:

Step 1: Getting customer's information:

Enter your name: Frank Johnson

Enter your age: 45

Enter your occupation: Electrical Engineer

Enter the workplace: UTEP

Enter the Social Security Number(6 characters): GVR097

Enter the monthly income: $8405.00

"Welcome to our Bank Frank Johnson"

Step 2: Loan type ("home", "car", or "land"):

What is the purpose of the loan? car

Step 3: Budget and ability to pay the credit:

What are your monthly expenses? $5790.00

"Loan denied – customer has high monthly expenses."

**WHAT DO YOU HAVE TO TURN IN?**

**Deliverable 1: Algorithm/Pseudocode**

Create a word document named “yourLastNameYourFirstNameLab10.docx”. In this document, you will write the pseudocode required to implement the functionality of Loan Evaluator (reminder: pseudocode is a set of instructions, not in java, that are seamlessly translatable in java or any other language).

Make sure your pseudocode includes instructions for the following:

* Asking and storing the information you need from the user, e.g., “What are your monthly expenses?”
* Implement a conditional when necessary, e.g., if the purpose of the loan is to buy a boat, the loan will be denied.
* Implement an repetition process when necessary, e.g., Enter your SSN.

**Tracing**

Using John Doe’s example described as a test case, trace the value of your variables. Remember, your pseudocode/code should work for different inputs, not just for John Doe.

**Deliverable 2: Loan Evaluator in Java.**

You will implement the pseudocode created in the Word document using Java. Your implementation should match your pseudocode. It is OK if after starting the implementation, you refine your pseudocode.

Your Java file should be named yourLastNameYourFirstNameLab10.java and the first lines of the code should be as follows:

/\* CS1101 – Intro to Computer Science Lab

Instructor: Dr. Ceberio

Comprehensive Lab 1

Submitted by: [YOUR NAME GOES HERE]

\*/

**Run your program.**

Run your program and make sure that it works as expected, e.g., when entering the input values you used in your tracing example, you should see the expected values.

**Testing**

Add screenshot of the results of your java file execution for different test cases to your word file.

**Note:**

If something was not clear about this comprehensive lab 1, please ask to your instructor or TA.

**HOW TO SUBMIT?**

On Piazza, under the folder lab10, submit a zipped file containing:

* Your Word document named “yourLastName-yourFirstName-Lab10.docx”.
* Your **JAVA file** called “yourLastNameYourFirstNameLab10.java”.

*Note: do not submit class files.*

**GRADING CRITERIA**

50% - Pseudocode (word document):

* 20% - Pseudocode of the main method
* 20% - Correct method signatures and pseudocode bodies when requested
* 10% - Tracing

50% Code (Java file):

* 10% - Appropriate variables definition and use of input/output operations
* 10% - Appropriate use of conditional and repetition structures
* 10% - Correct definition and use of methods
* 5% - Appropriate commenting
* 10% - Correct indentation
* 5% - Testing and screenshots

Late submission rules:

You will lose 15% for every 24 hours of lateness, for up to 72 hours. For example, if you submit 36 hours later your maximum percentage is 70%.

Refer to the course syllabus for policies on plagiarism.

**TIPS**

* Start early.
* Plan to work 5 to 7 additional hours outside the lab to complete this assignment and make sure your submission is on time.
* Ask clarifying questions to the instruction team if something is not clear.
* Plan to submit at least 30 minutes before the deadline.
* Double check that your files were uploaded correctly and can be downloaded and opened correctly.
* Keep it simple!