**QUESTION 1**

1. You are working as the IT support person for an electrician. The electrician has asked you to create a program that generates a report to show the month’s customers, the total number of customers, and the amount owed across all customers. Each customer record contains a customer number, customer name, number of kilowatt hours used (up to, but not including 1000.0) and the amount owed.

The amount owed is based on the number of kilowatt hours used based on the following rate schedule:

|  |  |
| --- | --- |
| **Number of Kilowatts Used** | **Cost** |
| 0.0 -199.9 | $0.11 per kilowatt |
| 200.0+ | $0.08 per kilowatt |

Your program should gather the input data from the user. Then your program should calculate and print the each customer’s record and amount owed on a weekly payroll report. All input data and calculated amounts should appear on the report. The total amount owed and total number of customers should appear at the end of the report. When there are no customers, an error message should appear instead of a report. You may assume the customer number and customer name have already been validated and will never included an empty value.

To arrive at a solution you have decided to follow the problem solving process and have created an IPO chart as follows:

|  |  |  |
| --- | --- | --- |
| **Input** | **Processing** | **Output** |
| Customer Record Details:                 customerNumber                 customerName                 kilowattsUsed | Print report heading Print report subheading Prompt for customerNumber Read customerNumber Prompt for customerName Read customerName Prompt for kilowattsUsed Read kilowattsUsed Validate kilowattsUsed Calculate amount owed Print customer record details Calculate total amount owed Print total number of customers  Print total amount owed  Determine whether to print report or error message | Report Heading Report Subheading Customer Record Details:                 customerNumber                 customerName                 kilowattsUsed                  amountOwed totalNumCustomers  totalOwed |

Then, you decide on what control structures might be needed to show how each of the processes will be performed to give you a high level of understanding of what you may want to put in your solution algorithm. You decide on the following:

* + A DOWHILE loop to control running through each customer since you do not know how many customers will be entered
  + A REPEAT UNTIL loop to control ensuring a valid kilowattsUsed value is entered to ensure the loop contents run at least once
  + An IF statement to determine if a valid kilowattsUsed value was entered
  + An IF statement to calculate the amount owed
  + An IF statement to determine if the report body should be printed or if an error message should be printed

Next, you create a solution algorithm, based on the IPO chart and high level control structures you identified.

PROCESS\_CUSTOMER\_RECORDS

1     SET minKilowatt = 0.0

2     SET maxKilowatt = 1000.0

3     SET lowRateKilowattMin = 200.0

4     SET lowRate = 0.08

5     SET highRate = 0.11

6     SET totalOwed = 0.0

7     SET numCustomers = 0

8     PRINT ‘Montly Billing Report’ heading

9     PRINT ‘Customer Number | Customer Name | Kilowatts Used | Amount Owed’ sub-heading

10    PROMPT for customerNumber

11    READ customerNumber

DOWHILE customerNumber <> ‘QUIT’

12          PROMPT for customerName

13          READ customerName

            REPEAT

14                PROMPT for kilowattsUsed

15                READ kilowattsUsed

16                IF kilowattsUsed < minKilowatt OR kilowattsUsed >= maxKilowatt THEN

                        PRINT ‘ERROR! Please enter a valid number of kilowatts’

                  ENDIF

            UNTIL kilowattsUsed >= minKilowatt AND kilowattsUsed < maxKilowatt

17          IF kilowattsUsed < lowRateKilowattMin THEN

                  SET amountOwed = kilowattsUsed \* highRate

            ELSE

                  SET amountOwed = kilowattsUsed \* lowRate

            ENDIF

18          totalOwed = totalOwed + amountOwed

19          numCustomers = numCustomers + 1

20          PRINT customerNumber, customerName, kilowattsUsed, amountOwed

21          PROMPT for customerNumber

22          READ customerNumber

ENDDO

23    IF numCustomers > 0 THEN

            PRINT numCustomers, totalOwed

ELSE

            PRINT ‘No customers entered’

ENDIF

END

Problem: Assuming you have completed your full desk check to confirm the algorithm is correct, convert the above solution algorithm into a Java solution. Keep in mind some of the nuances in converting pseudocode to Java code, such as the use of JOptionPane, the use of try/catch for data validation, adding constants for constant values, and how REPEAT UNTIL translates into a Java loop. Your Java solution should look near identical to the solution algorithm.

A ***sample*** completed report is as follows:

**Hints:**

* + Remember to use constants as appropriate, along with other good design practices you have learned in the course
  + Remember the use of try/catch to validate numeric input
  + Remember to use String.format to format a String for output

**Note:** For input/output, you must use the JOptionPane class.

**Your code must compile using the jGrasp IDE. If your code does not compile using jGrasp (for any reason), a grade of 0 will be earned. No exceptions!**

**Upload your completed .java file. Make sure you upload the .java file and not the .class file**