IS 2063: PA2 ASSIGNMENT

<table>
<tr><td>TOTAL POINTS: 100</td></tr>
<tr><td>DUE: CHECK BLACKBOARD BY 11:59 PM</td></tr>
<tr><td>
▪ READ CAREFULLY "ALL" THE INSTRUCTIONS!!!
▪ Do not submit code that doesn't compile, run, or generate correct output.
▪ No assistance given via email.
▪ Seek help during open tutoring or instructor office hours.
</td></tr>
</table>

CONTENTS: Press Alt, Left Arrow to return to table of contents.

UTSA HONOR CODE: As a UTSA student, you are **bound** by the honor code, so DO NOT cheat on <u>any</u> of your coursework. ***By submitting this assignment, you affirm that there has been no cheating or collusion in its completion;*** that the material you've used is from your textbook, professor, or Java tutors. Cheating can result in anyone, or combination, or all of the following: reduced or failing grade for the assignment, a signed statement of the infraction, reduced or failing grade for the course, reporting of student to the Department Chair, Dean's Office, and/or elevation to Student Conduct and Community Standards. It is not only cheating, but an infringement of copyright, if you post this assignment or your completed code to any website for broad consumption or distribution. The idea behind this assignment is not yours. The logic given to you for the assignment by the faculty is not yours. The code that is written embodies the idea.

*NOTE: **Make sure your PA1 is running with the correct logic from the plan and correct formatting (line advancing, spacing) and output <u>before</u> attempting PA2.** If you followed the PA1 plan, then the majority of the variables referenced in these instructions are the same; otherwise, you'll have to map them to yours. The underlying logic is basically the same.*

Print out your PA1 code (hardcopy) and the PA2 instructions. Read through the PA2 instructions thoroughly to get a good idea of the logic, the required control structures, the methods and any

changes that have to be made. Insert hand-written changes or cut your PA1 code with actual scissors and paste/tape the code in its proper order. The code in the main() from PA1 resides in the main() unless relocated to their respective methods, or in the case of some variables to the class level. After you've done this, open DrJava, copy your PA1 code into a .java file named according to the class name in the **UML** diagram, then start altering the code using the changes in the hardcopy of your PA1 code.

PREP WORK: Material from chapters 1-5 (including instructor PowerPoint slides, demo programs, and completed exercises).

GRADING: You'll be graded on *how well* you follow the program instructions and the ***accuracy* of your output as reflected in the prompts, the output specifications, and the sample output**. ***This includes spacing and line advancing.*** **Each line of output can be associated with multiple points in the code!** The instructions, prompts and output are what the user wants. You are **not** at liberty to change anything, but code to these requirements. You'll also be graded on the code given to you by your professor for this PA.

TEAM MEMBER RATING: Team members are to rate one another **only** if a member's performance is 3 or below out of a 5, which can affect a team member's grade. Refer to Blackboard for details.

PROGRAM INSTRUCTIONS:

1. **Work and submit this PA in a group/team. No plan is required.**

2. ***Commenting Your Program:***   Refer to the *Java Style Guide* posted in Blackboard for more commenting and formatting details. *Java doc comment box examples may have been given to you in demo programs coded during class or through a video posted by your professor.*

   a. In your program, YOU MUST insert a **program purpose** in the first comment box. The content of that first comment box was shown to you in the *Anatomy of a Java Program* lecture for chapter 1.

   b. Use Javadoc comment boxes beginning with /** and ending with */ for your comment boxes.

   c. Insert a Javadoc comment box above all your methods explaining what is going on with the code in the method.

   d. Line comment the import statements and the variables declared at the class level and/or in any method [including main()].

   e. Line comment close braces. The close brace for an else in an if-else is where you line comment for the structure. Line comment after the while in a do-while.

3. ***Formatting Rules***: **Always test** to validate your program is functioning properly with the correct output and spacing (line advances and spacing after punctuation). The %n can function differently when using separate printf() statements versus one printf().

4. ***Program Objective:*** Create a program that computes annual sales revenue to-date to determine the status of the company's sales revenue and whether a year-end bonus is in store for the employees. Sales revenue is captured by month within a quarter; therefore, the user can enter as many quarters as needed as long as the quarters are not less than 1 or greater than 4. If sales revenue is on target by 50% or more for a sales rep, an encouraging message is sent; otherwise,

a warning is sent. If sales revenue to date for the company is greater than or equal to 100% of projected annual sales, then employees qualify for a 2-5% year-end bonus; otherwise, no year-end bonus can be expected. Use printf() with format specifiers where needed.

5. **UML Diagram** for the class*: Data members are variables. The variables can be derived from the prompts and/or the output. The complete code for the PA is outlined in logical progression within this UML.*

Class Name	TeamMembersLastNamesYourSectionNoPA2
	(Example: AdamsShepherd001PA2)
	The last names are to be in *alphabetical* order.

Class Data Members	Fields are those variables used by more than one method. The minus sign means `private`.

| ```
-input:  Scanner
-monthNo:  String
-salesRep:  String
-quarter:  String
``` | ```
-quarterlySales:  double
-qtrCounter:  int
-noOfQtrs:  int
``` |

Method Data Members	+main() The plus sign means `public`.

| ```
salesRevenue, annualSales,
projectedSales,
percOfTargetCo,
percOfTargetRep:  double
``` | ```
qtrChoice, monthCounter,
noOfMonths, noSalesReps,
salesRepCtr:  int
``` |

+promptSalesRep()

```
index, spaceCount:  int
rep:  StringBuilder
```

+chooseQtr()

```
qtrChosen:  int
```

+promptSalesRevenue()

```
salesRev:  double
```

+printSalesRepRevReport()

```
dateTime:  Calendar
```

Methods	+main(args: String[]): static void

1. ***Prompt:***

What is the projected annual sales for Tandem?

Validate for a floating-point using a while validation test covered in class. Print the following error message and re-prompt if the input is not a double.

NOT a valid floating-point! Please re-enter the projected sales for Tandem: *<== Ignore Word-wrap. All on the same line.*

2. ***Prompt:***

How many sales reps work for Tandem?

Validate for an integer using a while validation test covered in class. Print the following error message and re-prompt if the input is not an integer.

NOT a valid integer! Please re-enter the number of sales reps for Tandem: *<== Ignore Word-wrap. All on the same line.*

3. Within the `do-while` loop:

   a. Reinitialize `qtrCounter` to 1.

   b. Zero out `quarterlySales`.

   c. Post-increment `salesRepCtr`.

   d. Call `promptSalesRep()`.

   c. Call `promptNoQtrs()`.

   d. Nested `while` loop controls the number of quarters.

      i. Initialize `monthCounter` to 1.

      ii. Call `chooseQtr()`.

      iii. Nested, nested `while` loop controls the number of months within a quarter.

         1) Call `determineMonthNo()`.

         2) Call `promptSalesRevenue()`.

         3) Use combined assignment operator to add `salesRevenue` to `quarterlySales`.

         4) Pre-increment `monthCounter`.

   d. Use combined assignment operator to add `quarterlySales` to `annualSales`.

   e. Pre-increment `qtrCounter`.

   f. Call `printSalesRepRevReport()`.

g. Calculate the `percOfTargetRep` which determines whether the sales rep is on target for their portion of the projected sales.

h. if the `noOfQtrs` is less than 4

    i. Use a **ternary operation** (see chapter 3 lesson 4) within a printf() to test if `percOfTargetRep` is greater than or equal to 50, if so print where the `substring()` grabs the first name **and uppercases the first letter**. Use the following message as the first value which will have no semicolon at the end.

String.format("%nKeep up the GOOD work, %s. "
  + "There is a possible year-end bonus!%n",   salesRep.substring(0,
    salesRep.indexOf(' ')).substring(0, 1).toUpperCase()
  + salesRep.substring(1, salesRep.indexOf(' ')))

    There is a space between the single quotes.

    ii. Otherwise, print:

    So far sales are lagging behind projections.

i. Calculate the `percOfTargetCo`.

j. Print:

CORPORATE SALES PERFORMANCE

k. Use a **ternary operation** within a printf() to test if `percOfTargetCo` is greater than or equal to 100, if it is print the message exactly as seen using the proper line advancing.

```
It's been a GOOD year so far.  There could be a year-end bonus of
about 2-5% if we can keep on top of our sales goals.  Thank you all
and please continue your excellent effort!
```

Otherwise, print the following all on the same line, no word wrap.

```
Sales are lagging projections.  A year-end bonus may not be
possible.
```

4. Exit.

5. You are to generate output from your program according to the instructions in the section.

`+promptSalesRep(salesRepCtr:  int):  static void`

1. Declare local variables.
2. ***Prompt:***

Enter the name of a sales rep:

OR

Enter the name of the next sales rep:

Use this ternary in a printf() to determine which prompt is used by inserting the words to put in %s based on the value in salesRepCtr.

"%nEnter the name of %s sales rep: ", salesRepCtr == 1 ? "a" : "the next"

3. Count the number of blank spaces using this for loop.

```
for(char blank : salesRep.toCharArray())
   {
      if(blank == ' ')
      {
         spaceCount++;

      }//END if there is a blank
   }//END for counting blanks in salesRep
```

4. Use the following code to create a `StringBuilder` object to properly capitalize the sales rep's name.

```
StringBuilder rep = new StringBuilder(salesRep);

rep.setCharAt(0, salesRep.toUpperCase().charAt(0));

index = salesRep.indexOf(' ');

for(int i = 1; i <= spaceCount; i++)
{
    rep.setCharAt(++index, salesRep.toUpperCase().charAt(index));
    index = salesRep.indexOf(' ', index);

}//END for i from 1 thru spaceCount

salesRep = rep.toString();
```

+promptNoQtrs(): static void

Prompt:

Enter the number of quarters worked (no less than 1 or greater than 4):

Validate for an integer using a while validation test covered in class. Print the following error message and re-prompt if the input is not an integer.

NOT a valid integer! Please re-enter the number of quarters worked (1-4):

+chooseQtr(): static int

1. Declare local variable.
2. ***Prompt:***

1. First Quarter

2. **Second Quarter**
3. **Third Quarter**
4. **Fourth Quarter**

Choose the quarter in which sales were earned:

OR

Choose the next quarter in which sales were earned:

Insert this code as the last concatenated line in the printf() for above prompt. The argument is a ternary that decides to put "next " or nothing in the %s. Note there is a space after the word next in the String literal. There is no space in-between the double quotes.

"%n%nChoose the %squarter in which sales were earned: "**,**
        qtrCounter > **1** **?** **"next "** **:** **""**

Validate for an integer using a while validation test covered in class. Print the above prompt and include the following line at the very top:

NOT a valid integer! Please re-enter the quarter.

After the validation, code the following ternary to setup for the next prompt.

quarter = (qtrChoice == **1****) ?** **"First Quarter"** **: (qtrChoice ==** **2****) ?**
"Second Quarter" **: (qtrChoice ==** **3****) ?** **"Third Quarter"** **:** **"Fourth Quarter"****;**

3. Code the return statement.

+determineMonthNo(monthCounter: int)

Use a **switch** to assign "1st", "2nd", "3rd" to monthNo based on monthCounter. No default option is needed.

+promptSalesRevenue(): static double

1. Declare local variable.
2. ***Prompt:***

Enter the sales revenue for the Xxx month of the Xxxxxx Xxxxxxx:

where Xxx holds **1st**, **2nd**, *or* **3rd** *and Xx Xxxxxx holds* **First Quarter**, **Second Quarter**, **Third Quarter**, **Fourth Quarter**.

Validate for a floating-point using a while validation test covered in class. Print the following error message and re-prompt if the input is not a double.

NOT a valid floating-point!

> **Please re-enter the sales revenue for the Xxx month of the Xxxxxx Xxxxxxx:** *<== Ignore Word-wrap. All on the same line.*
>
> 3. Code the return statement.
>
> ---
>
> **+printSalesRepRevReport(): static void**
>
> 1. Declare local variable.
> 2. ***Print*** the *final output*.

SALES REP OUTPUT SPECIFICATIONS: *The Zs and 9s represent the output as formatted numbers. Zs indicate zero-suppression of leading numbers. 9s represent numbers printed as digits from 0-9. Xs are text. Zs, 9s, Xs are values that will change, everything else are headers, titles, and labels. Use System.out.printf() and the appropriate format specifiers to properly format the output. The year is NOT to be hard coded into the header, instead, you will capture the system's date and format for the year only (refer to Appendix I). This is so the year corresponds to when the program is run. What is printed is for each sales rep.*

> ↙Header (Title). Triple line advance uses 2 %n's with 2nd and 3rd lines of the header using 1 %n.

TANDEM ENTERPRISES
SALES REVENUE FOR 9 QUARTER(S) OF 9999
SALES REP: Xxxxxxxxxxxx Xxxxxxxxxxxxxxxxxxxxx

Total Year-To-Date: $ZZZ,ZZZ,ZZ9.99 ←

> Total Label with year-to-date sales figure which tells you your output variable. When printed, the $ sign will not float to the leading digit.

***END SALES REP OUTPUT SPECIFICATION

FINAL OUTPUT SPECIFICATIONS: *Printed after the last sales rep.*

CORPORATE SALES PERFORMANCE

It's been a GOOD year so far. There could be a year-end bonus of about 2-5% if we can keep on top of our sales goals. Thank you all and please continue your excellent effort!

OR

Sales are lagging projections. A year-end bonus may not be possible.

END FINAL OUTPUT SPECIFICATIONS

SUBMISSIONS REQUIREMENT:

1. You will put your files in a group folder. This means **only <u>one</u> of the group members should submit the PA**. ***ALL* submissions** will be ignored if both group members post submissions. The *folder* and the *main() file* will reflect the *last names of both group members in alphabetical order*. E.g., **BunnyDuck001PA2**. Through line and method comments, group members will claim authorship and the work must be evenly distributed. Example:

```
    int age = 0;        //By Bugs Bunny:  STORES AN AGE


/**
 * By Daffy Duck
 * main() captures a first and last name, an age, and a zip code.
 * It multiplies the age and zip code to get a product.
 * Output will be made to the console screen and to a
 * GUI window.
 */
```

2. **WARNING:** One group member <u>CANNOT</u> and <u>MUST NOT</u> carry the load for these assignments. All group members must be actively engaged in developing and writing code. Deadlines set by the group for ultimate submission must be met. *No correspondence by a group member to another group member is to be ignored.* Your group member could be a possible contact for future employment or even your employer. ***All group members*** *are to agree to the submission,* **unless** *the group agrees to convey that authority to the submitter.*

3. **Upload Your Submission in This Order**

   a. **Word Document:** Copy your .java code into a Word document and save it with the same name as your program. Upload the document to Blackboard under this PA.

   b. **Zipping Folders: Your Java files "must be" in a folder.**

      1. Create a folder named for the program (excluding the file extension).

      2. Put your **.java, .class**, **.java~** files in the folder.

      3. To zip the folder, point to it then right click and

      4. **Filzip** if you have it **OR**

      5. Click **Send To** then click **Compressed (zipped) Folder**

      6. Upload your zipped folder to Blackboard under this PA.

4. **Uploading to Blackboard: Make sure your browser is properly configured for Blackboard (see syllabus).**

   a. Your submissions are to be uploaded to Blackboard through **Assignments** only.

   b. **Upload your files no later than the due date by 11:55 pm**; otherwise, you don't have time to recover from any problems and your assignment may not be accepted by Blackboard.

   c. Check to make sure your submission is uploaded.   Please **do not ask your instructor** to check whether your assignment has been uploaded. You can do this yourself. Or upload during a tutoring session when someone can help you.

   d. If you submit your assignment before the due date, want to make changes or upload additional files, you can **re-upload** your files.

5. **NO ASSIGNMENTS WILL BE ACCEPTED LATE OR VIA E-MAIL. DO NOT UPLOAD PROGRAMS THAT DON'T COMPILE OR DON'T PRODUCE CORRECT OUTPUT.**

SAMPLE OUTPUT: *This output is* **slightly different** *than PA1. It is always good to test your code using sample data to see if your program meets the output specifications.* ***Run your program using the data in the following sample output. Copy and paste the output into a traditional comment***

box /* */ at the end of your PA2.java file. *The comment box needs to be outside of the close brace for the class.* *Change your font in DrJava to Monospaced or Courier New so your output is not out of alignment.* <mark>Worth 5 points!</mark> *Your output will <u>not</u> print in bold.*

SALES NOT ON TRACK

What is the projected annual sales for Tandem? 5OO000

NOT a valid floating-point! Please re-enter the projected sales for Tandem: 500000

How many sales reps work for Tandem? @

NOT a valid integer! Please re-enter the number of sales reps for Tandem: 2

Enter the name of a sales rep: julian caesar   ← *Enter name all lowercase.*

Enter the number of quarters worked (no less than 1 or greater than 4): I   ← *Not a 1.*

NOT a valid integer! Please re-enter the number of quarters worked (1-4): 1

1. First Quarter
2. Second Quarter
3. Third Quarter
4. Fourth Quarter

Choose the quarter in which sales were earned: W

NOT a valid integer! Please re-enter the quarter.

1. First Quarter
2. Second Quarter
3. Third Quarter
4. Fourth Quarter

Choose the quarter in which sales were earned: 2

Enter the sales revenue for the 1st month of the Second Quarter: !00000

NOT a valid floating-point!

Please re-enter the sales revenue for the 1st month of the Second Quarter: 100000

Enter the sales revenue for the 2nd month of the Second Quarter: 50000

Enter the sales revenue for the 3rd month of the Second Quarter: 100000

TANDEM ENTERPRISES
SALES REVENUE FOR 1 QUARTER(S) OF 2022
SALES REP: Julian Caesar

Total Year-To-Date: $ 250,000.00

Keep up the GOOD work, Julian. There is a possible year-end bonus!

Enter the name of the next sales rep: Monique La Femme

Enter the number of quarters worked (no less than 1 or greater than 4): 2

1. **First Quarter**
2. **Second Quarter**
3. **Third Quarter**
4. **Fourth Quarter**

Choose the quarter in which sales were earned: 1

Enter the sales revenue for the 1st month of the First Quarter: 5000

Enter the sales revenue for the 2nd month of the First Quarter: 6000

Enter the sales revenue for the 3rd month of the First Quarter: 5000

1. **First Quarter**
2. **Second Quarter**
3. **Third Quarter**
4. **Fourth Quarter**

Choose the next quarter in which sales were earned: 2

Enter the sales revenue for the 1st month of the Second Quarter: 5000

Enter the sales revenue for the 2nd month of the Second Quarter: 6000

Enter the sales revenue for the 3rd month of the Second Quarter: 5000

TANDEM ENTERPRISES
SALES REVENUE FOR 2 QUARTER(S) OF 2022
SALES REP: Monique La Femme

Total Year-To-Date: $ 32,000.00

So far sales are lagging behind projections.

CORPORATE SALES PERFORMANCE

Sales are lagging projections. A year-end bonus may not be possible.

SALES MET OR EXCEEDED

What is the projected annual sales for Tandem? 500000

How many sales reps work for Tandem? 2

Enter the name of a sales rep: Julian Caesar

Enter the number of quarters worked (no less than 1 or greater than 4): 1

1. First Quarter
2. Second Quarter
3. Third Quarter
4. Fourth Quarter

Choose the quarter in which sales were earned: 2

Enter the sales revenue for the 1st month of the Second Quarter: 100000

Enter the sales revenue for the 2nd month of the Second Quarter: 50000

Enter the sales revenue for the 3rd month of the Second Quarter: 100000

TANDEM ENTERPRISES
SALES REVENUE FOR 1 QUARTER(S) OF 2022
SALES REP: Julian Caesar

Total Year-To-Date: $ 250,000.00

Keep up the GOOD work, Julian. There is a possible year-end bonus!

Enter the name of the next sales rep: Monique La Femme

Enter the number of quarters worked (no less than 1 or greater than 4): 2

1. First Quarter
2. Second Quarter
3. Third Quarter
4. Fourth Quarter

Choose the quarter in which sales were earned: 1

Enter the sales revenue for the 1st month of the First Quarter: 25000

Enter the sales revenue for the 2nd month of the First Quarter: 75000

Enter the sales revenue for the 3rd month of the First Quarter: 25000

1. First Quarter

2. **Second Quarter**
3. **Third Quarter**
4. **Fourth Quarter**

Choose the next quarter in which sales were earned: 2

Enter the sales revenue for the 1st month of the Second Quarter: 10000

Enter the sales revenue for the 2nd month of the Second Quarter: 25000

Enter the sales revenue for the 3rd month of the Second Quarter: 30000

TANDEM ENTERPRISES
SALES REVENUE FOR 2 QUARTER(S) OF 2022
SALES REP: Monique La Femme

Total Year-To-Date: $ 190,000.00

Keep up the GOOD work, Monique. There is a possible year-end bonus!

CORPORATE SALES PERFORMANCE

It's been a GOOD year so far. There could be a year-end bonus of about 2-5% if we can keep on top of our sales goals. Thank you all and please continue your excellent effort!