CIS 5355
Assignment # 2
(10 points)
Due: **BY 11:59 pm on Friday, February 4, 2022**
To be done by each student individually

The instructor expects individual effort on this assignment. This assignment is NOT a group project. Collaboration of any type is not sanctioned and will be treated per the Academic Dishonesty policy as stated in the course syllabus. Academic Dishonesty also includes use of non-sanctioned external resources such as assignments and class material of anyone from previous semester(s) at Texas State or from other universities. Each submission will be closely examined for plagiarism. Any form of academic dishonesty will be penalized to the fullest extent.

A. Introduction:

You will use the SaleCo database located on the CISSQL.MATRIX.TXSTATE.EDU\CIS3325 server to formulate and then execute queries in response to the questions listed in the assignment specifications section below:

B. Submission Requirements:

1. As a part of this assignment, you will submit **a single pdf document** that will include your formulation of the query in SQL code followed by the output result from the SQL Server database in response to that query. Name the pdf document FirstNameLastNameAssignment2.pdf e.g., **ShawnDanteAssignment2.pdf.**
2. The SQL code for each query should be preceded by a comment. The comment should the name the query as 'Question # XXX', where XXX is the question number of the question that is being answered by this query e.g., for question 1, the query would be named 'Query 1'.
3. Inclusion of the verbiage of the question itself in the comment is helpful but not required. If provided, the instructor can reference specific parts of the question verbiage if the SQL code is incorrect or errors are found in the SQL formulation.
4. If you are using MS-word or other text editor, make sure to convert the final text document to a PDF format. ***PDF is preferred as it facilitates online grading and writing comments.***
5. Expected output is provided in these specifications where queries involved may be complex. Not all queries have expected output. In those cases, you will need to verify the correctness of your output manually. Your output should match in terms of layout, formatting, and column aliases.

C. **Formulate SQL queries to run against the SaleCo database located on the university-hosted SQL Server DBMS. Requirements specific to this assignment are listed below:**

1. **Assignment Specific Requirements:**
    1. Use of column aliases, rather than the actual column or field names, FOR ALL attributes is REQUIRED, unless otherwise specified or shown in expected output.
    2. There are a total of 10 problems. Half are provided for you to practice on while the other half is required to be submitted as a part of this assignment. Problems marked with 'Problems to be submitted as Assignment 2' indicate the problems for which SQL formulation and resultant output need to be submitted as a part of this assignment. Please do not submit answers to practice problems. They will not be graded.
    3. When the output contains currency values or date values, they must be formatted as instructed or as shown in expected output.
    4. Output layout, column aliases, order of listed attributes, formatting of values for those problems that are required to be submitted as Assignment 2 must match exactly with the expected output shown. Any deviation from these will be subject to deduction of grading points.
    5. Every one of the assigned problems in this assignment deals with one and only one entity in each SELECT statement. None of them requires tables to be joined. Queries based on the joined tables or SELECT statements based on multiple tables will automatically receive a grade of zero. Therefore, think through the problem. The assignment will be graded on both the correct output as well as efficiency of your SQL statements. A Select statement that can give correct result based on one entity per Select statement is much more efficient than the one that is based on multiple tables per Select statement.

2. **Problems to practice on (these are not to be submitted):**
    1. Display the count of all invoices issued by SaleCo. Only the count need to be displayed.
    2. Write a query to count the number of customers with a current balance of **at least** $500.00. Display only the count of the number of customers that meet this criterion.
    3. Write a query to display the total purchase per invoice. The total purchase per invoice is the sum of product purchases in line number that corresponds to the invoice. Product purchase amount per line item in an invoice is calculated as (Line Quantity * Line Price). The total purchase amount per invoice is the sum of amount purchased in line items of the invoice. For example, if three products were purchased on a given invoice, then the sum of purchase amount paid for each of these products is the invoice total. Your output should display Invoice number and Invoice total for that invoice. The invoice total must be formatted for display as USD. Sort the results by invoice number in increasing order. The output should look like:

INV_NUMBER	Invoice Total
1001	$24.94
1002	$9.98
1003	$153.85
1004	$34.87
1005	$70.44
1006	$397.83
1007	$34.97
1008	$399.15

4. Write a query to display a list of customers who did not make purchases during the invoicing period. A customer is considered not making a purchase during the invoicing period if there are no invoices associated with his/her customer code. This means that their customer code will not appear in the Invoice entity. Display customer code and outstanding balance. The balance must be shown in currency format. List should be in order of increasing balance. HINT: this does not involve joining tables. A query using a join will automatically receive a zero grade for the question.

CUS_CODE	Outstanding Balance
10010	$0.00
10019	$0.00
10016	$221.19
10013	$536.75
10017	$768.93

5. Write a query to display customer balance summary for all customers who have NOT made purchases during the current invoicing period. The summary should display total, minimum, maximum, and average outstanding balance. This order should be maintained. Additionally, values must be displayed in USD currency format.

Total Balance	Minimum Balance	Maximum Balance	Average Balance
$1,526.87	$0.00	$768.93	$305.37

3. Problems to submit as Assignment # 2

1. Prepare a list of customers with current balance of **either exactly** $0.00 **or greater than** $500.00. Only customers located in phone area code 615 should be included in this list. Display customer code, customer's full name (see definition of full name later), current balance and phone area code. Customer's full name should be displayed in 'Firstname MI. Lastname' format if there is a middle initial (e.g. 'Shawn K. Dante') in one column. If the customer does not have a middle initial, the full name should be displayed in 'Firstname Lastname' format (e.g. 'Shawn Dante') in one column. Customer's current balance must be displayed in US currency format (USD with 2 decimal places as $xx.xx). The list should be presented in ascending order by the amount of customer's current balance. Note the use of column aliases.

Cust. Number	Customer Name	Outstanding Balance	cus_areacode
10010	Alfred A. Ramas	$0.00	615
10014	Myron Orlando	$0.00	615
10019	Olette K. Smith	$0.00	615
10013	Paul F. Olowski	$536.75	615
10017	George Williams	$768.93	615

2. The German subsidiary of SaleCo has requested the total value of products in inventory. The product prices are stored in USD currency but should be converted to German currency of 'Euro' using the most recent conversion rate found on web sites such as x-rates.com. The converted total value should be displayed in Euro currency format. The example shows the value in German currency (Euro) using the conversion rate as of Jan. 31, 2022. You will need to find the conversion rate as of the date of your assignment submission.

Inventory Value
1.342.522,28 €

3. Write a query to display the average price of all products supplied by each vendor. A product is considered to be supplied by a vendor if the vendor code in the Product entity is not null. Only Items supplied by a vendor should be included in the list. In addition, only products whose average price is more than $50.00 should be included in the list. . The list should display vendor code and the average price of all products supplied by that vendor. The list should be in decreasing order by average price. Average price must be displayed in US currency format.

	Vendor	avg price
1	24288	$155.59
2	25595	$89.63

4. Display a list of customers who have made a purchase during the current invoicing period. To determine if a customer has made a purchase during the current invoicing period, look at the Invoice entity. Customers with their cus_code listed in the Invoice entity are considered having made at least one purchase during the current invoicing period. For this list of customers, display the customer code and invoice date. If a customer has made multiple purchases during the invoicing period, that customer must only appear once in this list. In other words, the list should not contain any duplicate customer codes. Further, the invoice date must be formatted to display as 'mm/dd/yyyy' and only invoices written after January 16, 2018 should be displayed.

	Customer Code	Invoice Date
1	10011	1/17/2018
2	10014	1/17/2018
3	10015	1/17/2018
4	10018	1/17/2018

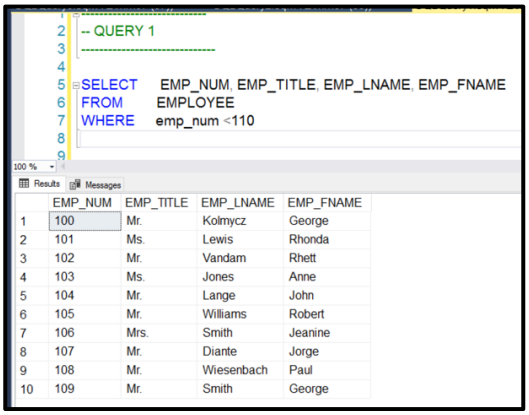
5. Suppose invoices are due no more than 90-days past the date the invoices were issued or posted. Write a query to display invoice number, customer code, posted invoice date and date invoice is due. The dates should be displayed in 'mm/dd/yyyy' format. The list should display only invoices associated with vendors 10011, 10014, and 10018 and only invoices issued or posted before January 17, 2018. Arrange the list in normal numeric order by invoice number.

	Invoice #	Customer #	Invoice Date	Invoice Due
1	1001	10014	1/16/2018	4/16/2018
2	1002	10011	1/16/2018	4/16/2018

D. <u>Instructions for creating a PDF Document:</u>

1. Ensure that your SQL query text and corresponding output results are fully displayed in the SSMS editor window.
2. Capture a screenshot of only the portion of the editor window showing the SQL query text and its associated output result. On Mac, press command+shift+4 keys and then select the portion that you would like to have a screenshot of.
3. Open a blank word document, insert your full name and assignment # at the top of the first page.
4. Following the name and assignment #, paste the screenshot so that it is fully visible and readable at 100% viewing. If you have to enlarge the view, you need to adjust the size of the pasted screenshot.
5. Repeat steps 1, 2, and 4 for each query. Restrict one query and its output to a page.
6. Save the word document as ***YourFirstLastName*Assignment2.docx** to your disk.
7. Export this word document to a PDF format (Use Save as and then select 'PDF' in file format to save. Save the pdf to your disk as ***YourFirstLastName*Assignment2.pdf.**
8. Upload the pdf file from step 7 to the Assignment section on Canvas.

Sample of PDF document to be submitted (or something similar)



```
-- QUERY 1
----------------------------

SELECT   EMP_NUM, EMP_TITLE, EMP_LNAME, EMP_FNAME
FROM     EMPLOYEE
WHERE    emp_num <110
```

Results | Messages

	EMP_NUM	EMP_TITLE	EMP_LNAME	EMP_FNAME
1	100	Mr.	Kolmycz	George
2	101	Ms.	Lewis	Rhonda
3	102	Mr.	Vandam	Rhett
4	103	Ms.	Jones	Anne
5	104	Mr.	Lange	John
6	105	Mr.	Williams	Robert
7	106	Mrs.	Smith	Jeanine
8	107	Mr.	Diante	Jorge
9	108	Mr.	Wiesenbach	Paul
10	109	Mr.	Smith	George