**Assignment: Summary Queries**

Brandon Trinkle

Arizona State University

IFT 300

Professor Ashish Gulati

11/4/2024

**Assignment: Summary Queries**

1. Write a SELECT statement that returns two columns from the Invoices table: VendorID and PaymentSum, where PaymentSum is the sum of the PaymentTotal column. Group the result set by VendorID.

/\*

\*\* Author: Brandon Trinkle

\*\* Course: IFT/300

\*\* SQL Server Version: Microsoft SQL Server 2012 (SP1)

\*\* OS : Windows

\*\* History

\*\* Date Created Comments

\*\* 11/12/2024

\*/

SELECT VendorID, SUM(PaymentTotal) AS PaymentSum

FROM Invoices

GROUP BY VendorID;

A computer screen shot of a computer screen

Description automatically generated

1. Write a SELECT statement that returns two columns: VendorName and PaymentSum, where PaymentSum is the sum of the PaymentTotal column. Group the result set by VendorName. Return only 10 rows, corresponding to the 10 vendors who’ve been paid the most.

/\*

\*\* Author: Brandon Trinkle

\*\* Course: IFT/300

\*\* SQL Server Version: Microsoft SQL Server 2012 (SP1)

\*\* OS : Windows

\*\* History

\*\* Date Created Comments

\*\* 11/12/2024

\*/

SELECT TOP 10 Vendors.VendorName, SUM(Invoices.PaymentTotal) AS PaymentSum

FROM Vendors

JOIN Invoices ON Vendors.VendorID = Invoices.VendorID

GROUP BY Vendors.VendorName

ORDER BY PaymentSum DESC;

A computer screen shot of a computer screen

Description automatically generated

3. Write a SELECT statement that returns three columns: VendorName, InvoiceCount, and InvoiceSum. InvoiceCount is the count of the number of invoices, and InvoiceSum is the sum of the InvoiceTotal column. Group the result set by vendor. Sort the result set so the vendor with the highest number of invoices appears first.

/\*

\*\* Author: Brandon Trinkle

\*\* Course: IFT/300

\*\* SQL Server Version: Microsoft SQL Server 2012 (SP1)

\*\* OS : Windows

\*\* History

\*\* Date Created Comments

\*\* 11/12/2024

\*/

SELECT Vendors.VendorName, COUNT(Invoices.InvoiceID) AS InvoiceCount, SUM(Invoices.InvoiceTotal) AS InvoiceSum

FROM Vendors

JOIN Invoices ON Vendors.VendorID = Invoices.VendorID

GROUP BY Vendors.VendorName

ORDER BY InvoiceCount DESC;

A computer screen shot of a computer screen

Description automatically generated

4. Write a SELECT statement that returns three columns: AccountDescription, LineItemCount, and LineItemSum. LineItemCount is the number of entries in the InvoiceLineItems table that have that AccountNo. LineItemSum is the sum of the InvoiceLineItemAmount column for that AccountNo. Filter the result set to include only those rows with LineItemCount greater than 1. Group the result set by account description, and sort it by descending LineItemCount.

/\*

\*\* Author: Brandon Trinkle

\*\* Course: IFT/300

\*\* SQL Server Version: Microsoft SQL Server 2012 (SP1)

\*\* OS : Windows

\*\* History

\*\* Date Created Comments

\*\* 11/12/2024

\*/

SELECT GLAccounts.AccountDescription,

COUNT(InvoiceLineItems.InvoiceID) AS LineItemCount,

SUM(InvoiceLineItems.InvoiceLineItemAmount) AS LineItemSum

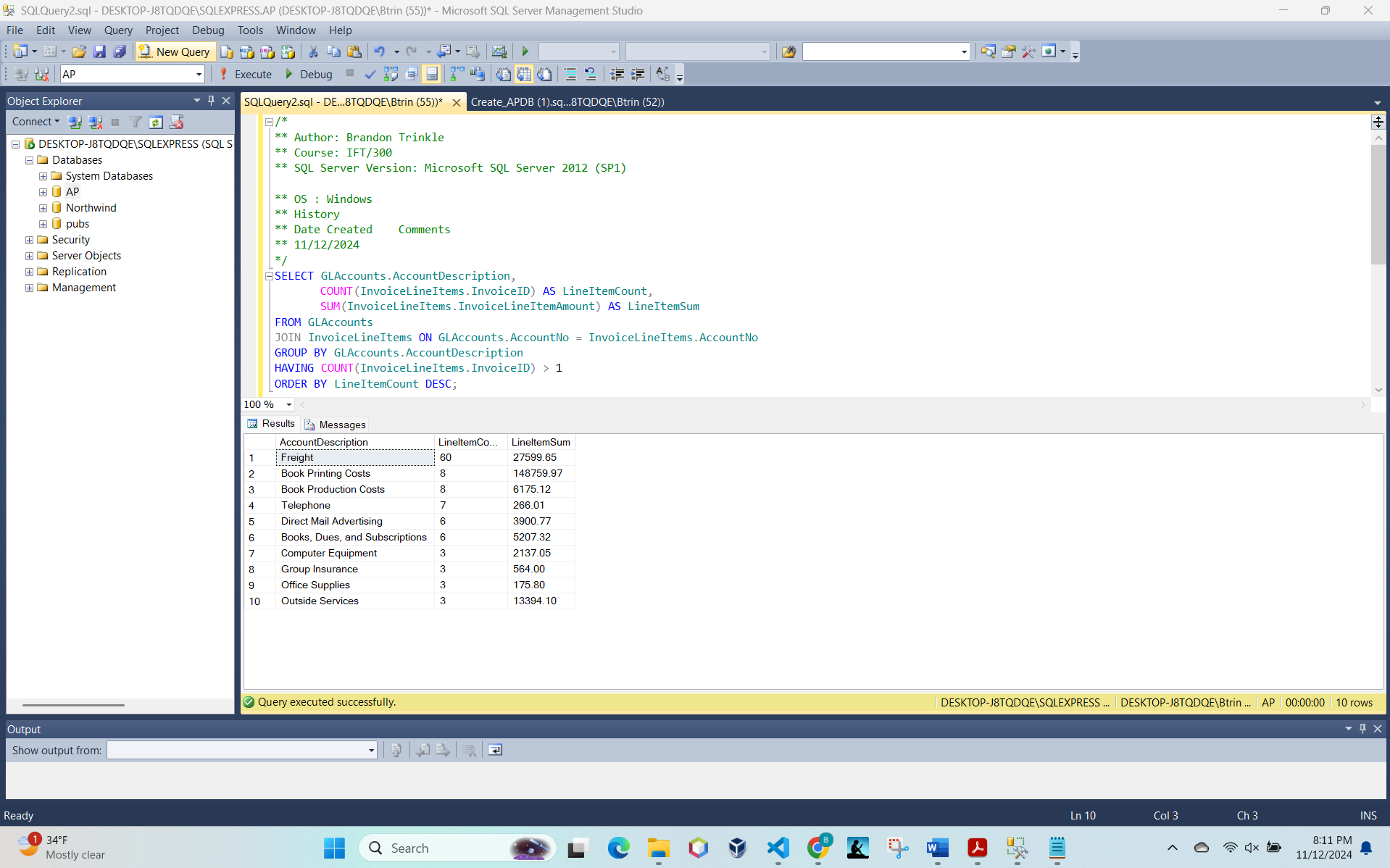
FROM GLAccounts

JOIN InvoiceLineItems ON GLAccounts.AccountNo = InvoiceLineItems.AccountNo

GROUP BY GLAccounts.AccountDescription

HAVING COUNT(InvoiceLineItems.InvoiceID) > 1

ORDER BY LineItemCount DESC;

****