# Exercise Instructions.

1. Type your name and the date into the space provided.
2. Use the SQL Server Management Studio to complete this lab.
3. Complete each of the exercises in this lab per the directions provided below.
4. Upload and submit before the due date.

# Exercises

1. Write a select that produces a list that shows the 4-digit year of each year when Eagle's Assembly employees were born and the number of Assembly employees born in that year.

Hints:

1. Any employee with a job title of Assembly is an Assembly employee. Use a WHERE clause to limit the rows you select.
2. You probably want to use the Year, and count functions when solving this exercise.
3. You will need to use a GROUP BY since the count function is an aggregate function.

Paste below the **code** you wrote **and the run results** you obtained for this exercise:

SELECT YEAR(BirthDate) AS YearBorn, COUNT(\*) AS NumberBorn

FROM Employee

WHERE JobTitle = 'Assembly'

GROUP BY YEAR(BirthDate)

ORDER BY YearBorn ASC;

YearBorn NumberBorn

1971 1

1973 2

1974 2

1975 2

1978 2

1980 1

1981 2

1982 2

1984 1

1987 1

1991 1

1992 1

------------------------------------------

2. Write a select that produces a list that shows each month and the number of Eagle Engineer employees born each month. Format your results like the example shown.

Hints:

1. The database is subject to insertions and deletions so your run results may differ from the sample.
2. An engineer is an employee with a job title that contains the word engineer.

Month EngineersBorn

------------------------------ -------------

June 2

May 1

November 1

September 1

(4 rows affected)

Paste below the **code** you wrote **and the run results** you obtained for this exercise:

SELECT DATENAME(month, DATEADD(month, MONTH(Birthdate), 0)) AS Month, COUNT(\*) AS EngineersBorn

FROM Employee

WHERE JobTitle LIKE '%Engineer%'

GROUP BY MONTH(Birthdate)

ORDER BY EngineersBorn DESC;

Month EngineersBorn

--------------------------- ---------------

July 2

October 1

December 1

April 1

June 1

(5 rows affected)

------------------------------------------

3. Write a select that list each engineer’s name, and how many years the engineer has worked for the Eagle Corporation.

Paste below the **code** you wrote **and the run results** you obtained for this exercise:

SELECT (LastName + ', ' + FirstName) AS Name,

DATEDIFF(year, HireDate, GETDATE()) AS Years

FROM Employee

ORDER BY Name ASC;

Name Years

Abbott, Michael 7

Albregts, Nicholas 6

Alvarez, Melissa 7

Blume, Meghan 6

Boden, John 2

Brose, Jack 7

Bush, Rita 7

Butler, Ronald 7

Cheswick, Sherman 4

Cochran, Steve 7

Deagen, Kathryn 7

Deppe, David 6

Eckelman, Paul 5

Edaton, Michael 1

German, Gary 7

Gustavel, Kristen 7

Hess, Steve 7

Hettinger, Gregory 7

Hickman, Steven 7

Jones, Charles 6

Keck, David 7

Krasner, Jason 7

Lilley, Edna 7

Manaugh, Jim 7

Moore, Kristey 7

Nairn, Michelle 7

Ortman, Austin 7

Osman, Jamie 6

Platt, Joseph 7

Ray, Sunny 2

Reece, Phil 6

Rodgers, Laura 7

Roland, Allison 7

Rosner, Joanne 5

Shlick, James 3

Stahley, Ryan 7

Stevenson, Gabrielle 7

Tesba, George 2

Underwood, Patricha 7

Vigus, Todd 7

Voltare, John 4

Wendling, Jason 7

Xolo, Kathleen 7

Yates, Tina 4

Zobitz, Beth 5

Zollman, Calie 7

-----------------------------------------------------------------------

4. Write a select that list each Assembly employee’s name, job title, and age (in years).

Paste below the **code** you wrote **and the run results** you obtained for this exercise:

SELECT (LastName + ', ' + FirstName) AS Name, JobTitle,

DATEDIFF(year, BirthDate, GETDATE()) AS Age

FROM Employee

ORDER BY Name ASC;

Name JobTitle Age

Abbott, Michael Engineer 46

Albregts, NicholasAssembly 41

Alvarez, Melissa Assembly 32

Blume, Meghan Engineer 51

Boden, John Assembly 31

Brose, Jack Assembly 48

Bush, Rita VP Operations 43

Butler, Ronald Assembly 50

Cheswick, Sherman President 55

Cochran, Steve Assembly Engineer 43

Deagen, Kathryn Assembly 50

Deppe, David Assembly 39

Eckelman, Paul Accountant 39

Edaton, Michael Design 27

German, Gary Chief Sales Officer 49

Gustavel, Kristen Operations Officer 47

Hess, Steve Assembly 49

Hettinger, GregoryAssembly 49

Hickman, Steven Programmer Analyst50

Jones, Charles DBA 48

Keck, David Assembly 52

Krasner, Jason Sales 50

Lilley, Edna VP Information 35

Manaugh, Jim Chief Financial Officer 45

Moore, Kristey Assembly 41

Nairn, Michelle Assembly 36

Ortman, Austin Assembly 42

Osman, Jamie Programmer Analyst 53

Platt, Joseph Assembly 42

Ray, Sunny Spokes Person 24

Reece, Phil Assembly 45

Rodgers, Laura Sales 28

Roland, Allison Assembly 45

Rosner, Joanne Assembly 48

Shlick, James Marketing Manager 25

Stahley, Ryan Engineer 37

Stevenson, Gabrielle Chief Information Officer 47

Tesba, George Design 25

Underwood, Patricha Assembly 43

Vigus, Todd Accountant 43

Voltare, John New Products 29

Wendling, Jason Operations Officer 45

Xolo, Kathleen VP Finance 54

Yates, Tina Senior Engineer 51

Zobitz, Beth Engineer 52

Zollman, Calie Sales 46

------------------------------------------

5. Write a select to list names of eagle’s customers (use the Customer Table). If the company name is not null list it, if the company name is null list the customer name (first name and last name). Your results should come back looking similar to the following (the database is subject to insertions and deletions so these partial results are shown only to convey the nature of the results not necessarily specific records).  
  
Customer\_Name  
----------------------------------------  
Baker and Company  
Cole Sales and Associates  
Tippe Inn  
Franklin Trinkets  
.  
.  
.  
Linda Li  
  
(231 row(s) affected)  
  
Hint: use ISNULL or COALESCE function in your SELECT list to control what is displayed.

Paste below the **code** you wrote and **type the number of rows returned** for this exercise:

SELECT (COALESCE(CompanyName, (CustFirstName + ' ' + CustLastName))) AS Customer\_Name

FROM Customer;

257 rows

------------------------------------------

6. Write a select that produces a list that shows the company name of each company that ordered something from the eagle company during a month of October (year does not matter). Do not include rows for companies with a CompanyName that begins with the letter ‘F’. Also, do not include rows for companies that have a null value for Company Name. Sort the list by CompanyName in descending sequence.

Paste below the **code** you wrote **and the run results** you obtained for this exercise:

SELECT c.CompanyName

FROM Customer c

JOIN CustOrder co

on c.CustomerID = co.CustomerID

WHERE CompanyName IS NOT NULL AND

LEFT(c.CompanyName, 1) NOT LIKE 'F' AND

MONTH(co.OrderDate) = 10

ORDER BY CompanyName DESC;

CompanyName

Water Analysts

Vets Inc.

Trailor Rentals

The Employment Agency

TAS

Store It Here

South Street Rehabilitation

Signs Signs Signs

Rydell High School

Regency Hospital

Realty Specialties

R and R Air

Powerful Employment

Pools For You

Photography Niche

Phone Corporation

Needle Center

Kids Recreation Inc.

Karate Made Easy

Gards Auto Repair

Cottingham Plastics

Collectibles Inc.

Cole Sales and Associates

Cleaning Supply

Bankruptcy Help

Average Burgers

------------------------------------------

7. Write a select that produces a list that shows the Customer FirstName, LastName, and Phone of each of eagles non-corporate customers that has ordered something during a month of October (year does not matter).Hint: A non-corporate customer has a null value for company name.

Paste below the **code** you wrote and **type the number of rows returned** for this exercise:

SELECT c.CustFirstName, c.CustLastName, c.Phone

FROM Customer c

JOIN CustOrder co

ON c.CustomerID = co.CustomerID

WHERE CompanyName IS NULL AND

MONTH(OrderDate) = 10

ORDER BY c.CustLastName ASC;

39 rows

------------------------------------------

8. Write a select that produces a list that shows the company name of each company that ordered something from the eagle company during a spring. Year does not matter (it does not matter what spring). Do not include rows for companies with a CompanyName that begins with the letter ‘F’. Also, do not include rows for companies with a null value

for CompanyName. Sort the list by CompanyName. Consider a date to be in the spring if its dayofyear value is between 79 and 170 (inclusive between)

\*/

Paste below the **code** you wrote and **type the number of rows returned** for this exercise:

SELECT c.CompanyName

FROM Customer c

JOIN CustOrder co

ON c.CustomerID = co.CustomerID

WHERE CompanyName IS NOT NULL AND

LEFT(c.CompanyName, 1) NOT LIKE 'F' AND

DATENAME(dayofyear, co.OrderDate) BETWEEN 79 AND 170

ORDER BY CompanyName ASC;

16 rows

------------------------------------------------------------