Name:\_\_Bryan Bibb\_\_\_\_\_\_\_ Date:\_09/17/2023\_\_\_\_

# Exercise Instructions

1. Type your name and the date into the space provided.
2. Use the SQL Server Management Studio and the IST272EagleCorp database to complete this lab.
3. Write T-SQL statements to query the tables contained in the IST272EagleCorp database and 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 statement that uses an inner join and returns all columns from the Machine table and all columns from the the MachineProcess table.

Paste below the **code** you wrote and **type the number of rows returned** for exercise 1:

SELECT \*

FROM Machine

JOIN MachineProcess

ON Machine.MachineID = MachineProcess.MachineID

ORDER BY Machine.MachineID;

Rows returned: 957

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

2. Write a SELECT statement that uses an inner join and returns four columns:

MachineID From the Machine table

ManufacturerName From the Machine table

TimeStandard From the MachineProcess table

PartNumber From the MachineProcess table

Only return rows where the PartNumber has one of the following values ('PS-001','PS-002','PS-003', 'PS-004') and the MachineID has one of the following values('LABEL','SEAL’)

Paste below the **code** you wrote **and the run results** you obtained for exercise 2:

SELECT Machine.MachineID, ManufacturerName, TimeStandard, PartNumber

FROM Machine

JOIN MachineProcess

ON Machine.MachineID = MachineProcess.MachineID

WHERE MachineProcess.PartNumber LIKE 'PS-00%'

AND (Machine.MachineID LIKE 'LABEL' OR Machine.MachineID LIKE 'SEAL')

ORDER BY MachineProcess.PartNumber;

MachineID ManufacturerName TimeStandard PartNumber

LABEL SERGE 20 PS-001

SEAL SERGE 50 PS-001

SEAL SERGE 50 PS-002

LABEL SERGE 20 PS-002

LABEL SERGE 20 PS-003

SEAL SERGE 50 PS-003

SEAL SERGE 50 PS-004

LABEL SERGE 20 PS-004

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

1. Generate the same result set described in exercise 2, but use the implicit join syntax.

Hint: If the number of rows that comes back is different than what you got for exercise 2, you probably forgot to include the join condition in the WHERE clause (see page 139 of the book).

Paste below the **code** you wrote **and the run results** you obtained for exercise 3:

SELECT Machine.MachineID, ManufacturerName, TimeStandard, PartNumber

FROM Machine, MachineProcess

WHERE Machine.MachineID = MachineProcess.MachineID

AND MachineProcess.PartNumber LIKE 'PS-00%'

AND (Machine.MachineID LIKE 'LABEL' OR Machine.MachineID LIKE 'SEAL')

ORDER BY MachineProcess.PartNumber;

MachineID ManufacturerName TimeStandard PartNumber

LABEL SERGE 20 PS-001

SEAL SERGE 50 PS-001

SEAL SERGE 50 PS-002

LABEL SERGE 20 PS-002

LABEL SERGE 20 PS-003

SEAL SERGE 50 PS-003

SEAL SERGE 50 PS-004

LABEL SERGE 20 PS-004

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

1. Write a SELECT statement that uses an inner join and returns four columns:

ShipmentID from the PackingSlip table

LastName from the Employee table

JobTitle from the Employee table

ShippedDate from the PackingSlip table

Only return rows where the ShippedDate >= '2017-02-27'

and sort the result set by LastName

Paste below the **code** you wrote and **type the number of rows returned** for exercise 4:

SELECT ShipmentID, LastName, JobTitle, ShippedDate

FROM PackingSlip, Employee

WHERE Employee.EmployeeID = PackingSlip.EmployeeID

AND PackingSlip.ShippedDate >= '2017-02-27'

ORDER BY Employee.LastName;

Rows Returned: 561

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

1. Write a SELECT statement that returns four columns from three tables, all using column aliases:

Title CustTitle Column

Company CompanyName Column

LastName CustLastName Column

Date OrderDate Column

Part PartNumber Column

Assign the following correlation names to the tables:

C Customer Table

CO CustOrder Table

COL CustOrderLine Table

Only include rows in the result set that have a partnumber that begins with PR and a CompanyName that is not NULL.

Paste below the **code** you wrote **and the run results** you obtained for exercise 5:

SELECT CustTitle AS Title, CompanyName AS Company, CustLastName AS LastName, OrderDate AS Date, PartNumber AS Part

FROM Customer AS C

JOIN CustOrder CO

ON C.CustomerID = CO.CustomerID

JOIN CustOrderLine COL

ON CO.OrderID = COL.OrderID

WHERE PartNumber LIKE 'PR%'

AND CompanyName IS NOT NULL

ORDER BY Title;

Title Company LastName Date Part

Dr. Recreation Supply Burns 2016-08-06 00:00:00 PRT-004

Dr. Investment Specialties Valle 2016-12-06 00:00:00 PRT-001

Dr. Investment Specialties Valle 2017-01-18 00:00:00 PRT-004

Dr. Main St. Bar and Grill Kluth 2017-03-23 00:00:00 PRT-001

Mr. City Bus Transport Osmanova 2017-03-30 00:00:00 PRT-006

Mr. BMA Advertising Design Purcell 2017-02-17 00:00:00 PRT-006

Mr. Bankruptcy Help Lichty 2017-02-28 00:00:00 PRT-003

Mr. First National Bank Barrick 2017-01-27 00:00:00 PRT-002

Mr. South Street Rehabilitation Meers 2016-12-16 00:00:00 PRT-004

Mr. North Street Church Elston 2017-01-11 00:00:00 PRT-001

Mr. Cleaning Supply Schofield 2016-10-06 00:00:00 PRT-003

Mr. Ceramic Supply Talauage 2016-12-07 00:00:00 PRT-006

Mr. Phone Corporation Katpally 2016-12-10 00:00:00 PRT-006

Mr. Security Installation Gray 2016-12-10 00:00:00 PRT-001

Mr. Regency Hospital Gasper 2016-09-17 00:00:00 PRT-003

Mr. Powerful Employment Sokeland 2016-09-28 00:00:00 PRT-006

Mr. Bankruptcy Help Lichty 2016-10-14 00:00:00 PRT-001

Mr. Karate Made Easy Scott 2016-10-22 00:00:00 PRT-002

Mr. Realty Specialties Tarter 2016-10-27 00:00:00 PRT-006

Mr. Pools For You Ezra 2016-10-31 00:00:00 PRT-004

Mr. Gards Auto Repair Sammons 2016-11-10 00:00:00 PRT-003

Mr. Karate Made Easy Scott 2016-11-11 00:00:00 PRT-001

Mr. Wadake Critters Gillespie 2016-11-17 00:00:00 PRT-001

Mr. Security Installation Gray 2016-12-01 00:00:00 PRT-004

Mr. Pools For You Ezra 2016-08-25 00:00:00 PRT-003

Mr. BMA Advertising Design Purcell 2016-08-27 00:00:00 PRT-006

Mr. Regency Hospital Gasper 2016-09-01 00:00:00 PRT-002

Mr. Vacation Car Rentals Strehle 2016-09-10 00:00:00 PRT-006

Mr. South Street Rehabilitation Meers 2016-07-01 00:00:00 PRT-006

Mr. Greenpart Steet Metal Doremski 2016-07-07 00:00:00 PRT-003

Mr. Regency Hospital Gasper 2016-07-13 00:00:00 PRT-004

Mr. Realty Specialties Tarter 2016-07-28 00:00:00 PRT-004

Mr. Laser Graphics Associates Kaakeh 2016-08-02 00:00:00 PRT-006

Mr. Copy Center Robles 2016-08-05 00:00:00 PRT-006

Mrs. Sharons Shamrock Rouls 2016-08-30 00:00:00 PRT-004

Mrs. Camping Supplies Deets 2016-08-09 00:00:00 PRT-004

Mrs. Family Medical Center Strong 2016-08-24 00:00:00 PRT-006

Mrs. Sampson Home Mortgages Kyger 2016-12-06 00:00:00 PRT-003

Mrs. Rydell High School Franks 2016-10-05 00:00:00 PRT-004

Mrs. Family Medical Center Strong 2016-10-12 00:00:00 PRT-001

Mrs. Bryant Accounting Xiao 2016-12-16 00:00:00 PRT-004

Mrs. Sharons Shamrock Rouls 2016-12-16 00:00:00 PRT-003

Mrs. Family Medical Center Strong 2016-07-07 00:00:00 PRT-004

Ms. Conner National Jacobs 2017-02-16 00:00:00 PRT-004

Ms. Cottingham Plastics Cottingham 2016-09-20 00:00:00 PRT-001

Ms. Automart Nakamura 2016-12-15 00:00:00 PRT-002

Ms. Automart Nakamura 2016-12-15 00:00:00 PRT-004

Ms. Conner National Jacobs 2016-09-16 00:00:00 PRT-003

Ms. Vets Inc. Cassens 2016-09-08 00:00:00 PRT-004

Ms. Tippe Inn Scheetz 2016-07-02 00:00:00 PRT-003

Ms. Needle Center Plyman 2016-07-05 00:00:00 PRT-006

Ms. Computer Consultants Jones 2016-07-06 00:00:00 PRT-004

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

6. Write a SELECT statement that returns three columns

LastName From the Employee table

FirstName From the Employee table

SalaryWage From the Employee table

The result set should have one row for each employee whose SalaryWage is the same as another employee’s SalaryWage. Sort the final result set by SalaryWage.

Hint: Use a self-join.

Paste below the **code** you wrote **and the run results** you obtained for exercise 6:

SELECT DISTINCT e1.LastName, e1.FirstName, e1.SalaryWage

FROM Employee e1

JOIN Employee e2

ON e1.SalaryWage = e2.SalaryWage

AND e1.EmployeeID <> e2.EmployeeID

ORDER BY SalaryWage;

LastName FirstName SalaryWage

Albregts Nicholas 13.50

Hettinger Gregory 13.50

Ortman Austin 16.50

Underwood Patricha 16.50

Boden John 16.75

Nairn Michelle 16.75

Cochran Steve 145000.00

Voltare John 145000.00

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

7. Write a SELECT statement that returns three columns:

EmployeeID From the Employee table

LastName From the Employee table

ShippedDate From the PackingSlip table

The result set should have at least one row for each employee regardless of whether or not the employee has ever packed an order.

Hint: Use an outer join to the PackingSlip table

Paste below the **code** you wrote and **type the number of rows returned** for exercise 7:

SELECT e.EmployeeID, LastName, ShippedDate

FROM Employee e

LEFT JOIN PackingSlip p

ON e.EmployeeID = p.EmployeeID

ORDER BY e.EmployeeID;

Rows returned: 3533

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

8. Modify the solution to exercise 7 to filter for Employees

who never packed an order (only include rows in the result set for employees that have never packed an order).

Paste below the **code** you wrote and **type the number of rows returned** for exercise 8:

SELECT e.EmployeeID, LastName, ShippedDate

FROM Employee e

LEFT JOIN PackingSlip p

ON e.EmployeeID = p.EmployeeID

WHERE p.ShippedDate IS NULL

ORDER BY e.EmployeeID;

Rows returned: 37

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

9. Generate a result set that contains two columns:

CompanyName and Phone. The result set should have one row for each

Customer with a CompanyName that is not null and one row for each Shipper with a ShipperName that is not null with following exception:

Do not include

Hint: Use UNION or UNION ALL

Paste below the **code** you wrote and **type the number of rows returned** for exercise 9:

SELECT CompanyName, Phone

FROM Customer

WHERE CompanyName IS NOT NULL

UNION

SELECT ShipperName, Phone

FROM Shipper

WHERE ShipperName IS NOT NULL

ORDER BY Customer.CompanyName;

Rows returned: 82

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

10. Write a SELECT statement that returns for each order line item with a status of PENDING and an Order Date > ‘2017-03-30’, the Order ID and Order Date along with the First Name, Last Name and Phone of the customer who placed the order. List the order information only once.

Hint: You will most likely have to join three tables

Paste below the **code** you wrote **and the run results** you obtained for exercise 10:

SELECT DISTINCT CO.OrderDate, COL.Status, COL.OrderID AS OID, c.CustFirstName AS First, c.CustLastName AS Last, c.Phone

FROM CustOrder CO

JOIN CustOrderLine COL

ON CO.OrderID = COL.OrderID

JOIN Customer c

ON c.CustomerID = CO.CustomerID

WHERE COL.Status = 'PENDING'

AND CO.OrderDate > '2017-03-30';

OrderDate Status OID First Name Last Name Phone

2017-03-31 00:00:00 PENDING 2000000805 Marjorie Vandermay 308-489-1137

2017-03-31 00:00:00 PENDING 2000000806 Daniel Rodkey 719-748-3205

2017-03-31 00:00:00 PENDING 2000000807 Cecil Scheetz 207-679-9822

2017-03-31 00:00:00 PENDING 2000000808 Andy Huegel 302-620-1366

2019-04-30 00:00:00 PENDING 2000000809 Jim Blough 610-261-4677

2020-01-30 00:00:00 PENDING 2000000810 Jim Baggins 419-376-9228

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