**CSC 312 – Database Systems**

**Mini-Project 2**

In this set exercises, we will have hands-on practice with the AdventureWorks database again to:

* First, use table joins, including INNER and OUTER joins to retrieve information;
* Second, practice with subqueries and views;

**Group 1 – INNER and OUTER Joins**

Joins that we learned so far are usually based on equality conditions between PK and FK. Therefore, products that have never been purchased will not show when we ***inner***-join the order detail table with the product table. In certain cases, such as a manager reviewing sales for all products, it’s crucial to include all products, whether have been ordered or not. Queries like this will need to use ***outer*** joins, which lead to result sets with NULL values added to fill up the data that are absent.

**Left** (**Outer**) **Join**

Q1. Using the Adventure Works sample database, list FirstName, LastName, PersonType (from Person.Person table), and JobTitle (from HumanResources.Employee table) using inner join.

Q2. Now let’s modify on Q1, replacing the (inner) join with ***left*** (***outer***) join. Make sure name of the Person.Person table is listed **before** the join operator. Explain the result and why it’s different from the result before the change.

Q3. Modify the query again to display BusinessEntityID (from Person. BusinessEntity table) in front the other 4 columns, using inner join here.

Q4. Modify Q3, replacing the (inner) join with ***left*** (***outer***) join, with BusinessEntity listed **before** the join operator.

**Right** (**Outer**) **Join**

Q5. Still using the Adventure Works sample database, list SalesOrderId, OrderQty, and product Name (using inner join).

Q6. Modify on Q5 by replacing the (inner) join with ***right*** (***outer***) join. Make sure name of the table product names is listed **after** the join operator. Explain the result and why it’s different from the result before the change.

Q7. Assuming your manager wants to know quantity sold for each product, make changes to Q6 to show product name followed with the quantity sold. [Hint: order number is not of concern now.]

Q8. Make change to Q7 to display 0 where NULL presence. [Hint: use the IsNull(expr, repl\_val) function.]

**Full** (**Outer**) **Join**

Q9. Now using the Payables sample database, list Vendor names and State names, using inner join on vendor zip equal to first zip in the state. Describe and explain the result.

Qa. Modify on Q9 by replacing the (inner) join with ***full*** (***outer***) join. Explain the result and why it’s different from the result before the change. Is there a row that has non-null value for both names? You may list vendor zip code and first zip code in state in the result to explain.

Qb. To generate some matched pairs, set the last digits in vendor zip code to ‘000’. Describe and explain results. [You may get the first two digits and concatenate three 0 to it.]

Qc. Does it make a difference when you swap the two table names with regards to the join operator? And why? What would happen when you swap table order in the previous two sections?

**Group 2 – Subquery and Views**

Subqueries can be used in various clauses in a SELECT statement (that serves as the main query). Let’s use the Payables database for the queries in this section. [Handout will be given out in class to give hints for syntax details.]

Qd. Subquery in the WHERE clause: Retrieve invoices with an InvoiceTotal greater than the average total of all invoices in the table. For each invoice in the result set, list its number, date, and total amount.

Qe. Subquery in the WHERE clause, comparing the ALL/ANY/SOME keywords: List number, date, and total amount for invoices with a total amount greater than all/some invoice amounts for vendor 34.

* First try to write the query without any of the three keywords;
* Then try out each of them and explain the differences as shown in the result sets.

Qf. Subquery in the WHERE clause, using NOT EXISTS: Retrieve vendors without any invoices.

Qg. Subquery in the FROM clause:

* Write a query to generate a temporary table with top 10 vendors by average invoice total.
* Write the main query using the query above in its FORM clause to find out the latest invoice date for each of the top 10 vendors.

Qh. Subquery in the SELECT clause:

* Write a query to retrieve vendor name and the latest invoice date for that vendor. This needs to left outer join Vendors and Invoices tables such that all vendor names show in the result, with or without any invoice.
* Write the same query using a (correlated) subquery: with a subquery used in the SELECT clause to retrieve the maximum (or latest) invoice date, alongside with a column to retrieve vendor name. Use the handout to see how it works.

Qi. Using a view: Retrieve all contents available through the HumanResources.vEmployeeDepartment-History view. Click the Design option on the short menu to check what is the underlying SELECT statement that defines the view.

Qj. Using the view Production.vProductAndDescription:

* Write a query retrieve distinct Culture types.
* Write the main query to list products that has descriptions in all these (6) Cultures.

All the queries are due by Friday, April 15.

use AdventureWorks2012

select Person.BusinessEntity.BusinessEntityID, Person.Person.PersonType, Person.Person.LastName, humanresources.employee.JobTitle

from Person.BusinessEntity inner join

Person.Person

on (Person.Person.BusinessEntityID = Person.BusinessEntity.BusinessEntityID)

left join humanresources.employee

on (Person.Person.BusinessEntityID = humanresources.employee.BusinessEntityID)

BusinessEntityID PersonType LastName JobTitle

…

284 SP Mensa-Annan Sales Representative

285 SP Abbas Pacific Sales Manager

286 SP Tsoflias Sales Representative

287 SP Alberts European Sales Manager

288 SP Valdez Sales Representative

289 SP Pak Sales Representative

290 SP Varkey Chudukatil Sales Representative

291 SC Achong NULL

293 SC Abel NULL

295 SC Abercrombie NULL

…

19972 rows

use AdventureWorks2012

select salesLT.SalesOrderDetail.SalesOrderID, salesLT.SalesOrderDetail.OrderQty, salesLT.Product.Name

from salesLT.SalesOrderDetail right join salesLT.Product

on (salesLT.Product.ProductID = salesLT.SalesOrderDetail.ProductID)

**Name Count()**

Bikes 0

Components 0

Clothing 0

Accessories 0

Mountain Bikes 32

Road Bikes 43

Touring Bikes 22

Handlebars 8

Bottom Brackets 3

Brakes 2

…

41 rows

use AdventureWorks2012

select salesLT.SalesOrderDetail.SalesOrderID, salesLT.SalesOrderDetail.OrderQty, salesLT.Product.Name

from salesLT.SalesOrderDetail right join salesLT.Product

on (salesLT.Product.ProductID = salesLT.SalesOrderDetail.ProductID)

**ID Qty Name**

NULL NULL All-Purpose Bike Stand

71782 10 AWC Logo Cap

71783 11 AWC Logo Cap

71784 10 AWC Logo Cap

71797 6 AWC Logo Cap

71816 4 AWC Logo Cap

71938 1 AWC Logo Cap

71858 3 AWC Logo Cap

71897 4 AWC Logo Cap

71902 3 AWC Logo Cap

71782 10 Bike Wash - Dissolver

71783 8 Bike Wash - Dissolver

71784 8 Bike Wash - Dissolver

71797 17 Bike Wash - Dissolver

71858 1 Bike Wash - Dissolver

71902 5 Bike Wash - Dissolver

71938 6 Bike Wash - Dissolver

NULL NULL Cable Lock

71845 3 Chain

71858 1 Chain

71898 1 Chain

71936 3 Chain

NULL NULL Classic Vest, L

71782 4 Classic Vest, M

…

695 rows

select salesLT.Product.Name, isnull(sum(salesLT.SalesOrderDetail.OrderQty), 0)

from salesLT.SalesOrderDetail right

join salesLT.Product

on (salesLT.Product.ProductID = salesLT.SalesOrderDetail.ProductID)

group by salesLT.Product.Name

**Name (No column name)**

All-Purpose Bike Stand 0

AWC Logo Cap 52

Bike Wash - Dissolver 55

Cable Lock 0

Chain 8

Classic Vest, L 0

Classic Vest, M 34

Classic Vest, S 87

Fender Set - Mountain 0

Front Brakes 12

Front Derailleur 13

Full-Finger Gloves, L 0

use Payables

select name, States.StateName

from Vendors full join States

on (left(ZipCode,2)+'000' = FirstZipCode)

…

Gostanian General Building NULL

Kent H Landsberg Co NULL

Malloy Lithographing Inc Michigan

Net Asset, Llc NULL

Office Depot California

Pollstar NULL

Postmaster NULL

Roadway Package System, Inc NULL

State of California NULL

Suburban Propane NULL

Unocal NULL

Yesmed, Inc NULL

Dataforms/West NULL

Zylka Design NULL

United Parcel Service NULL

Federal Express Corporation NULL

NULL Rhode Island

NULL Oregon

NULL New Hampshire

NULL New York

…

158 rows

select top 1 salesLT.Product.Color as ProductColor, count(salesLT.Product.Name)

from salesLT.Product

group by salesLT.Product.Color

order by 2 desc

ProductColor (No column name)

Black 89

ii.

use Payables

select Vendors.VendorID, Vendors.Name, Vendors.State

from Vendors

where VendorID not in

(select distinct VendorID

from Invoices)

VendorID Name State

2 National Information Data Ctr DC

3 Register of Copyrights DC

4 Jobtrak CA

5 Newbrige Book Clubs NJ

6 California Chamber Of Commerce CA

7 Towne Advertiser's Mailing Svcs CA

8 BFI Industries CA

9 Pacific Gas & Electric CA

10 Robbins Mobile Lock And Key CA

11 Bill Marvin Electric Inc CA

12 City Of Fresno CA

13 Golden Eagle Insurance Co CA

14 Expedata Inc CA

16 Internal Revenue Service CA

17 Blanchard & Johnson Associates CA

18 Fresno Photoengraving Company CA

19 Crown Printing CA

20 Diversified Printing & Pub CA

21 The Library Ltd MO

use AdventureWorks2012

select productid from

(

select productid, count(distinct culture) as C\_count

from SalesLT.vProductAndDescription

group by productid) as a

where C\_count = 6

productid

864

865

866

862

863

…

294 rows