**Lab 04 Use Join for Queries to multiple tables**

**Ex 1 (ex1.sql open)**

You no longer need the supplied mapping information between categoryid and categoryname because you now have the Production.Categories table with the needed mapping rows. Write a SELECT statement using an inner join to retrieve the productname column from the Production.Products table and the categoryname column from the Production.Categories table.

**Ex 2 (Ex2.sql open)**

Sales department would like a report of all customers who placed at least one order, with detailed information about each one. A developer prepared an initial SELECT statement that retrieves the custid and contactname columns from the Sales.Customers table and the orderid column from the Sales.Orders table. You should observe the supplied statement and add additional information from the Sales.OrderDetails table.

2.1 developer has written this query:

**SELECT custid, contactname, orderid**

**FROM Sales.Customers**

**INNER join Sales.Orders ON Customers.custid = Orders.custid;**

Execute the query exactly as written inside a query window and observe the result.

The error is shown. Change code and fix this problem:

2.1.1 Notice that there are full source table names written as table aliases.

2.1.2 Apply the needed changes to the SELECT statement so that it will run without an error. Test the changes by executing the T-SQL statement.

**2.2 Change Table Aliases**

2.2.1 Copy the T-SQL statement from task 2 and modify it to use the table aliases “c” for the Sales.Customers table and “o” for the Sales.Orders table.

2.2.2 Execute the written statement and compare the results with those in task 2.

2.2.3 Change the prefix of the columns in the SELECT statement with full source table names and execute the statement.

2.3 Add an Additional Table and Columns

1. Copy the T-SQL statement from task 3 and modify it to include three additional columns from the Sales.OrderDetails table: productid, qty, and unitprice.

**Ex3 3 (ex3.sql open)**

HR department would like a report showing employees and their managers. They want to see the lastname, firstname, and title columns from the HR.Employees table for each employee, and the same columns for the employee’s manager.

3.1 To better understand the needed tasks, you will first write a SELECT statement against the HR.Employees table showing the empid, lastname, firstname, title, and mgrid columns.

Write a Query That Uses a Self Join

3.2 Copy the SELECT statement from task 1 and modify it to include additional columns for the manager information (lastname, firstname) using a self join. Assign the aliases mgrlastname and mgrfirstname respectively, to distinguish the manager names from the employee names.

**Ex4 (ex4.sql open)**

The sales department was satisfied with the report you produced in exercise 2. Now sales staff would like to change the report to show all customers, even if they did not have any orders, and still include order information for the customers who did. You need to write a SELECT statement to retrieve all rows from Sales.Customers (columns custid and contactname) and the orderid column from the table Sales.Orders

4.1 Write a SELECT Statement Uses an Outer Join

Write a SELECT statement to retrieve the custid and contactname columns from the Sales.Customers table and the orderid column from the Sales.Orders table. The statement should retrieve all rows from the Sales.Customers table.

**Ex5 (ex55.sql open)**

The HR department would like to prepare a personalized calendar for each employee. The IT department supplied you with T-SQL code that will generate a table with all dates for the current year. Your job is to write a SELECT statement that would return all rows in this new calendar date table for each row in the HR.Employees table.

5.1 Write a SELECT Statement That Uses a Cross Join

Write a SELECT statement to retrieve the empid, firstname, and lastname columns from the HR.Employees table and the calendardate column from the HR.Calendar table.