**BSYS 3105 | SSMS Stored Procedure Assignment**

# Instructions

Create and document the following Stored Procedures for the **Northwind** database. Under each question below, copy/paste the SQL code used to create the Stored Procedure.

For example:

**Q: Display all records from the Quarterly Orders view**

CREATE PROCEDURE ShowQuarterlyOrders

AS

SELECT \* FROM [Quarterly Orders]

*(Hint: You could use the query editor to help you write SQL statement:* ***Query Menu 🡪 Design Query in Editor*** *and/or use the shortcut* ***CTRL|SHIFT|Q*** *while editing in the SQL window. Check out the separate hint notes posted as well)*

1. **List Products for user defined Product Category name**

Alter procedure ProductsbyID

(@categoryid int)

as

begin

SELECT \* FROM Products

where (CategoryID = @categoryid)

end;

1. **List customer Orders from user defined Country and City**

Create procedure Locationsearch

@city char(15),

@country char(15)

as

begin

Select \*

from Orders

where (@city = ShipCity) and (@country = ShipCountry)

end;

1. **List Orders for user defined Employee last name**

create procedure LastNameOrderSearch

@LastName nchar(15)

as

begin

SELECT Orders.\*, Employees.LastName

FROM Employees INNER JOIN

Orders ON Employees.EmployeeID = Orders.EmployeeID

Where (@LastName = lastname)

end;

1. **Display Total sales for user defined Product Category name**

Create procedure ListTotalByCategoryName

@CategoryName nvarchar(MAX)

as

begin

SELECT Sum([Order Details].[Quantity]\*[Order Details].[UnitPrice]) as [total cost],

Products.ProductID,

Products.CategoryID,

Categories.CategoryName

FROM [Order Details] INNER JOIN

Products ON [Order Details].ProductID = Products.ProductID INNER JOIN

Categories ON Products.CategoryID = Categories.CategoryID

where CategoryName = @CategoryName

Group by

CategoryName,

Products.ProductID,

Products.CategoryID;

end

1. **Display the Number of sales for user defined Supplier and Year (Use YEAR function)**

create proc CompanyOrdersByYear

@Companyname nvarchar(max),

@Year nchar(4)

as

begin

SELECT Suppliers.CompanyName,

Count([Orders].[OrderID]) as OrderCount,

YEAR([Orders].orderdate) as OrderYear

FROM Suppliers INNER JOIN

Products ON Suppliers.SupplierID = Products.SupplierID INNER JOIN

[Order Details] ON Products.ProductID = [Order Details].ProductID INNER JOIN

Orders ON [Order Details].OrderID = Orders.OrderID

where CompanyName = @CompanyName and Orders.OrderDate = @Year

Group by

Suppliers.CompanyName,

YEAR(Orders.OrderDate)

Order by

Suppliers.CompanyName,

OrderYear

end;

1. **As above, but assume we know only part of Supplier name (Keyword Search – Use LIKE operator)**

alter proc CompanyOrdersByYear

@Companyname nvarchar(max),

@Year nchar(4)

as

begin

SELECT Suppliers.CompanyName,

Count([Orders].[OrderID]) as OrderCount,

YEAR([Orders].orderdate) as OrderYear

FROM Suppliers INNER JOIN

Products ON Suppliers.SupplierID = Products.SupplierID INNER JOIN

[Order Details] ON Products.ProductID = [OrderDetails].ProductID INNER JOIN

Orders ON [Order Details].OrderID = Orders.OrderID

where Suppliers.CompanyName Like '%' + CompanyName + '%'

and Orders.OrderDate = @Year

Group by

Suppliers.CompanyName,

YEAR(Orders.OrderDate)

Order by

Suppliers.CompanyName,

OrderYear

end;

1. **Create a customer Order with one Product (INSERT statement for Orders and Order Details tables) One SP that includes two statements:**
2. Add a new order in the **Orders** table with parameters for CustomerID and EmployeeID and the current date. (Remember: NO need for OrderID in the INSERT as it is an Identity field. Use the GETDATE() function for the current date.)
3. Add a record in the **Order Details** table for the Order you just created above. Instead of OrderID enter **@@IDENTITY** (SQL Server stores the last created identity value in this variable). Check out the Order Details table design to see the **required** fields.

create Proc NewCustomerOrderOneProduct

@customerID nchar(5),

@employeeID int,

@ProductID int,

@UnitPrice money,

@Quantity smallint,

@discount real

as

begin

insert into [Orders](CustomerID,EmployeeID,OrderDate)

Values(@customerID,@employeeID,GETDATE())

insert into [Order Details](OrderID,ProductID,UnitPrice,Quantity,Discount)

Values(@@IDENTITY,@ProductID,@UnitPrice,@Quantity,@discount)

end