**Olivia Denbu Wilhelmsson**

**Uppgift 1**

private SqlConnection conn = null;

public DataAccess()

{

var cs = "Data Source=localhost;Initial Catalog=Northwind;Integrated Security=SSPI;";

conn = new SqlConnection(cs);

}

public DataSet Find(SqlParameter[] inputCustomerID)

{

SqlCommand cmd = null;

DataSet dataset = new DataSet();

DataTable table = new DataTable();

cmd = conn.CreateCommand();

cmd.CommandType = CommandType.Text;

cmd.CommandText = "select companyName, phone from Customers";

SqlParameter param = new SqlParameter();

param.ParameterName = "@CustomerID";

param.Value = inputCustomerID;

if (param != null)

{

cmd.Parameters.AddRange(inputCustomerID);

}

conn.Open();

SqlDataAdapter da = null;

using (da = new SqlDataAdapter(cmd))

{

da.Fill(table);

}

dataset.Tables.Add(table);

cmd.Dispose();

cmd = null;

conn.Close();

return dataset;

}

**Uppgift 2**

public BindingList<OrderModel> GetOrders()

{

BindingList<OrderModel> searchedOrders;

using (var dataContext = new Model1())

{

var query = from order in dataContext.Orders

where order.Customers.CustomerID.Contains("ALFKI")

select new OrderModel

{

CustomerID = order.CustomerID,

OrderID = order.OrderID

};

searchedOrders = new BindingList<OrderModel>(query.Distinct().ToList());

}

return searchedOrders;

}

**Uppgift 3**

create view View (CompanyName, Country, ProductName, CategoryName)

as select cu.companyName, cu.country, p.ProductName, ca.CategoryName from Customers cu

left join Orders o on cu.CustomerID = o.CustomerID

left join [Order Details] orDe on o.OrderID = orDe.OrderID

left join Products p on orDe.ProductID = p.ProductID

left join Categories ca on p.CategoryID = ca.CategoryID

where cu.Country = 'USA'

create procedure StoredProcedure

@categoryID int,

@minimunStockValue

as

begin

select CompanyName from View

where (categoryID like '%' +

@categoryID + '%') and (products.UnitsInStock < ('%' + @minimunStockValue + '%'))

end

**Uppgift 4**

public BindingList<ProductModel> GetProducts()

{

BindingList<ProductModel> products;

using (var dataContext = new Model1())

{

var query = from product in dataContext.Products

join category in dataContext.Categories on product.CategoryID equals category.CategoryID

where (category.CategoryName.Contains("Beverages"))

select new ProductModel

{

UnitPrice = product.UnitPrice

};

products = new BindingList<ProductModel>(query.ToList());

}

return products;

}

**Uppgift 5**

select count(\*) as Antal from Customers c

where c.Fax IS NULL

**Uppgift 6**

private SqlConnection conn = null;

public DataAccess()

{

var cs = "Data Source=localhost;Initial Catalog=Northwind;Integrated Security=SSPI;";

conn = new SqlConnection(cs);

}

public bool UpdateProductPrice()

{

SqlCommand cmd = null;

cmd = conn.CreateCommand();

int result = 0;

cmd.CommandType = CommandType.Text;

cmd.CommandText = "update p set UnitPrice = UnitPrice \* 1.15 " +

"from Products p " +

"inner join Categories c on p.CategoryID = c.CategoryID " +

"where c.CategoryName = 'Seafood';";

conn.Open();

result = cmd.ExecuteNonQuery();

cmd.Dispose();

cmd = null;

conn.Close();

if (result > 0)

{

return true;

}

else

{

return false;

}

}

}

**Uppgift 7**

create table Toys (

ToyID int identity(1,1) PRIMARY KEY,

[Description] nvarchar(40),

MadeBySanta bit

);

**Uppgift 8**

create procedure AddToys

@Description nvarchar(40),

@MadeBySanta bit

as

begin

insert into Toys ([Description], MadeBySanta)

values (@Description, @MadeBySanta)

end