Onderzoek naar LINQ to SQL

**Wat is LINQ to SQL?**

Met LINQ to SQL kun je tabellen uit een database behandelen als objecten zonder hierbij de mogelijkheid te verliezen om querries uit te voeren op deze tabellen.

Voordat de volgende onderwerpen uitgevoerd kunnen worden moet er eerst een object model worden gemaakt van de database in visual studio, dit kan met de “Object Relational Designer” voor een kleine tot middelgrote database of met “SQLMetal code-generation tool” voor een grote database.

**Querries in LINQ to SQL**

Makkelijk querries opstellen vanuit c#

Voorbeeld:

var q =

from c in db.Customers

where c.City == "London"

select c;

Vervolgens kun je dan makkelijk door de teruggegeven data gaan op de volgende manier:

foreach (Customer c in q)

Console.WriteLine(c.CompanyName);

Dit hoeft natuurlijk niet alleen op deze manier, je kan gewoon met de teruggegeven data omgaan zoals je ook met objecten omgaat.

**Objecten**

Toevoegen

Dit kan op dezelfde manier als je een object aanmaakt.

Voorbeeld:

Customer cust = new Customer();

cust.CompanyName = "SomeCompany";

cust.City = "London";

cust.CustomerID = "98128";

cust.PostalCode = "55555";

cust.Phone = "555-555-5555";

nw.Customers.InsertOnSubmit(cust);

(nw is de database, en Costumers is de tabel waar het object wordt ingevoerd)

Updaten

var cityNameQuery =

from cust in nw.Customers

where cust.City.Contains("London")

select cust;

foreach (var customer in cityNameQuery)

{

if (customer.City == "London")

{

customer.City = "London - Metro";

}

}

nw.SubmitChanges();

Deleten

var deleteIndivCust =

from cust in nw.Customers

where cust.CustomerID == "98128"

select cust;

if (deleteIndivCust.Count() > 0)

{

nw.Customers.DeleteOnSubmit(deleteIndivCust.First());

nw.SubmitChanges();

}

Zoals je ziet is het ontzettend makkelijk om objecten op deze manier aan te passen in de database.

**Nadelen van LINQ to SQL**

* LINQ to SQL heeft geen cascade delete functie(dit is wel mogelijk als je het zelf programmeert).
* Een object kan maar gekoppeld zijn aan één tabel.
* Omdat je niet altijd goed ziet wat er onder water gebeurt op het SQL gebied, kan het zijn dat er performance loss optreedt.