

Mikko Parkkinen

T074SN

SQL Server 2008 ja ASP.NET  
Tietokannan perustaminen ja sen web-  
pohjainen muokkaaminen

Opinnäytetyö  
Tietotekniikan koulutusohjelma


Tammikuu 2011




**MIKKELIN AMMATTIKORKEAKOULU**

Mikkeli University of Applied Sciences

## KUVAILULEHTI

 <b>MIKKELIN AMMATTIKORKEAKOULU</b> <small>Mikkeli University of Applied Sciences</small>	Opinnäytetyön päivämäärä				
<b>Tekijä(t)</b>  Mikko Parkkinen	<b>Koulutusohjelma ja suuntautuminen</b>  Tietotekniikan koulutusohjelma				
<b>Nimeke</b>  SQL Server 2008 ja ASP.NET – Tietokannan perustaminen ja sen web-pohjainen muokkaus					
<b>Tiivistelmä</b>  Työn tarkoituksena oli tehdä ohje SQL Server 2008 asentamiseen ja tietokannan luomiseen sekä ohje tietokannan muokkaamiseen ASP.NET -tekniikan avulla tehdyn web-sivun kautta. Työ oli jaettu kahteen osaan eli SQL- osioon ja ASP.NET-osioon. Työ aloitettiin SQL-ohjelmiston asentamisesta, josta siirryttiin askel kerrallaan eteenpäin kunnes tietokanta oli luotu serverille. Seuraava vaihe oli luoda aspx-sivu, jolla ottaa yhteys SQL Serveriin ja muokata tietokannan dataa.  Työn lopussa oli valmiina toimiva SQL Serveri sekä ASP-sivu, jolla muokata tietokannan dataa. Työn avulla lukijalle selviää tarvittavat tiedot nopeaan tietokantapalvelimen pystyttämiseen sekä kuinka luoda aspx-sivu tietokannan muokkaamiseen.					
<b>Asiasanat (avainsanat)</b>  SQL, ASP					
<b>Sivumäärä</b>  40	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;"><b>Kieli</b></td> <td style="width: 33%;"><b>UNR</b></td> </tr> <tr> <td>Suomi</td> <td>URN:NBN:fi:amk-201104184453</td> </tr> </table>	<b>Kieli</b>	<b>UNR</b>	Suomi	URN:NBN:fi:amk-201104184453
<b>Kieli</b>	<b>UNR</b>				
Suomi	URN:NBN:fi:amk-201104184453				
<b>Huomautus (huomautukset liitteistä)</b>					
<b>Ohjaavan opettajan nimi</b>  Timo Mynttinen	<b>Ohjaavan opettajan nimi</b>				

## DESCRIPTION

 <b>MIKKELIN AMMATTIKORKEAKOULU</b> Mikkeli University of Applied Sciences		<b>Date of the bachelor's thesis</b>	
<b>Author(s)</b> Mikko Parkkinen		<b>Degree programme and option</b> Information technology	
<b>Name of the bachelor's thesis</b>  SQL Server 2008 and ASP.NET – Creation of database and its web based editing			
<b>Abstract</b>  <p>Purpose of this thesis was to create instructions for installation of SQL Server 2008, also how to create a database and build web page using ASP.NET technique for browsing and editing the database.</p> <p>Thesis was divided in two sections, first was SQL Server 2008 part which started from installation of database program and ended with fully functioning database. Installation part included also tips for managing your computers user profiles to match them to needs of SQL Server 2008 installation program. Basic database security is also explained at first part. Second part was ASP.NET, in which was created an .aspx page for browsing and editing the database .At the end of the thesis, we had fully working SQL Server database and ASP page for browsing and editing it. With this thesis, reader gets information on how to create a database and how to make a web page for browsing and editing the database.</p>			
<b>Subject headings, (keywords)</b>  SQL, ASP			
<b>Pages</b>  40	<b>Language</b>  Finnish	<b>URN</b>  URN:NBN:fi:amk-201104184453	
<b>Remarks, notes on appendices</b>			
<b>Tutor</b>  Timo Mynttinen		<b>Bachelor's thesis assigned by</b>	

## SISÄLTÖ

1	JOHDANTO .....	1
2	SQL SERVER 2008 ESITTELY .....	1
2.1	Uusia ominaisuuksia .....	2
3	ASP.NET ESITTELY .....	2
3.1	ASP.NET historiaa .....	3
4	SQL SERVER 2008 ASENNUS .....	4
4.1	Asennus.....	4
4.2	Asennuksen jälkeen .....	8
4.3	SQL Server Configuration Manager.....	9
4.4	Yhteydet.....	11
5	SQL SERVER 2008 .....	10
5.1	Tietokannan luominen .....	11
5.2	Tietojen lisääminen ja muokkaus .....	14
5.3	Tietojen haku .....	16
5.4	Tietojen poisto .....	20
5.5	Taulun luominen ja tietojen lisääminen SQL-kielillä.....	22
5.6	Tietojen muokkaus SQL-kielillä.....	23
5.7	Tietojen haku SQL-kielillä.....	25
5.8	Tietojen poisto SQL-kielillä.....	26
6	ASP.NET.....	30
6.1	Tietokanta yhteyden luominen .....	31
6.1.2	Visual Studio ja SQL Server.....	31
6.2	ASP-lomake tietojen hakemiseen ja muokkaamiseen.....	32
6.2.1	Edit ja Delete-komentojen testaus.....	36
6.2.2	Tietojen lisääminen tietokantaan.....	41
6.2.3	Tietojen lisäämisen testaus.....	40
6.3	Ulkoasun muokkaus .....	41
7	JOHTOPÄÄTÖKSIÄ .....	44
	LÄHTEET .....	46

### LIITE/LIITTEET

1 Yksisivuinen liite

## 2 Monisivuinen liite

## **1 JOHDANTO**

Nykyaikana jokainen ihminen on tekemisissä tietokantojen kanssa tietoisesti tai tietämättään siitä. ASP-tekniikkaa käytetään internetsivujen tekoon monesti jos sivusto tarvitsee tietokantaa, esimerkiksi internetkauppa. ASP-sivut eivät aina ole pelkästään internetkauppojen sivuja vaan myös muut sivustot käyttävät ASP-tekniikkaa, hyvänä esimerkkinä ovat Microsoftin kotisivut.

Tietokanta puoli on näkymättömämpi, koska se jää ASP-sivujen taakse piiloon. Tietokanta on yksinkertaisuudessaan vaikkapa lukujärjestys, jota luetaan tietyt ehdot antamalla, esimerkiksi annetaan ehto ”maanantai kello 13.00” ja katsotaan mitä sillä kohdalla lukujärjestyksessä lukee. SQL-tietokantoja selataan samalla tavalla hakuehtoja antamalla ja luodaan antamalla samanlaisia selventäviä tietoja, jolloin tietojen hakeminen helpottuu.

Työn tarkoituksena on antaa tietoa SQL Server 2008:sta sekä ASP.NET:stä sekä luoda ohjeet Microsoft SQL Server 2008 asennukseen ja nopeaan käyttöönottoon. Työssä myös näytetään kuinka tietokantaa pääsee muokkaamaan internetsivulla, joka on tehty Microsoft Visual Studio 2008 ohjelmalla, ASP.NET tekniikalla.

## **2 SQL SERVER 2008 ESITTELY**

Microsoft SQL Server 2008 on yritystason relaatiotietokantapalvelin. Relatiotietokanta tarkoittaa tietokantaa, jossa voidaan tarkasti kuvata tietojen organisointi ja sekä tietokantaan tehtävät kyselyt, poistot ja päivitykset. Tietokannassa tiedot on sisällytetty tauluihin (eng. table). Taulut pitävät sisällään n-määrän rivejä (eng. row) ja sarakkeita (eng. column). Tietokannassa tietoja voidaan hakea useasta eri taulusta yhdellä kyselyllä (eng. query).

Microsoftin suurimmat kilpailijat tällä alalla ovat Oracle ja IBM:n DB2. Markkinoiden suurimmat ovat juuri Oracle sekä IBM ja Microsoft tulee kolmantena. Listalla neljäntenä tulee Sybase jonka kanssa Microsoft teki yhteistyötä aikoinaan ja teki Sybasen kanssa yhteisen tietokantapalvelin ohjelmiston (SQL Server) vuonna 1988. Microsoftin ja Sybasen yhteistyö jatkui vielä joitain vuosia 90-luvulla, kunnes vuonna

1994 yhtiöiden tiet erosivat ja Microsoft alkoi itsenäisesti kehittää SQL Server-ohjelmistoa.

## **2.1 Uusia ominaisuuksia**

SQL Server 2008:ssa on paljon uusia ominaisuuksia aikaisempiin versioihin verrattaessa. Tärkeimpiä uudistuksia ovat tuki monille uusille tiedostoformaateille sekä laajat tietojen import/export mahdollisuudet. SQL Server 2008:ssa on mahdollisuus ottaa yhteys ja linkittää tietoja eri valmistajien tietokantapalvelimiin (mm. Oracle, IBM DB2). Tämä ominaisuus helpottaa esimerkiksi yhteistyötä muiden yritysten kanssa, joilla on eri valmistajan tietokantapalvelimet tai jos osa omista palvelimista on eri ohjelmistolla.

Tietojen tuominen ja vieminen Microsoft Excel ja Access-ohjelmistoihin on tehty helppoksi. SQL Serveristä löytyy apuohjelma (SQL Server Import and Export Wizard), jonka avulla voi esimerkiksi tuoda valmiin taulukon Excelistä ja luoda siitä taulun tietokantapalvelimelle. Samalla apuohjelmalla voi luoda Excel-taulukon tietokannassa olevista tiedoista.

## **3 ASP.NET ESITTELY**

ASP.NET on Microsoftin verkkosovellusympäristö, jolla pystytään tekemään dynaamisia verkkosivuja, web-sovelluksia ja verkkopalveluja. ASP:n avulla tehdyillä internetsivuilla voidaan esimerkiksi hakea ja muokata tietokannoissa olevaa dataa. Parhaiten ASP soveltuu Microsoftin SQL ja Access-tietokantojen käsittelyyn.

ASP:n etuja on se että käyttäjä ei tarvitse erillistä ohjelmaa esim. tietokantojen käsittelyyn, vaan pelkkä internetselain riittää. Helppokäyttöisyyttä lisää se, että käyttäjän internetselain voi olla minkä tahansa valmistajan selain, lisäksi internetsivut näyttävät samalta oli selain MS-Explorer, Firefox tai mikä tahansa muu selain. Syy miksi internetsivut näyttävät samalta oli selain mikä tahansa on se, että asp sivut prosessoidaan

serverillä ja lähetetään vasta sitten html-muodossa käyttäjän selaimelle. Internetsivu on siis vain ns. ”tyhjä kuori”, johon serveripäässä luodaan sisusta.

### **3.1 ASP.NET historiaa**

ASP eli (Microsoft) Active Server Pages julkaistiin vuonna 1996 lokakuussa beta-versiona. Julkaisu oli osana päivitystä Internet Information Server (IIS) 2.0:llaa. Aluksi ASP kehittyi hitaasti versioon 2.0 ja siitä versioon 3.0. Kaikki kolme versiota käyttivät oletuksena VBScript-kieltä (Virtual Basic Script). Scriptin käyttäminen toi omat ongelmansa. Vuonna 1997 Microsoft alkoi suunnitella mahdollisuutta uudenlaisten verkkosovellusten tekoa ASP:n avulla, joka poistaisi joitain ASP:n käytön ongelmia. Vuonna 2000 Microsoft esitteli uuden .NET ympäristön (beta) ja samalla päivityksen ASP:n, joka oli nimeltään ASP.NET 1.0. Tämä avasi uusia mahdollisuuksia ja enää ei käyttö-kieli rajoittunut scriptaaviin kieliin vaan uudet C#, J# ja VB.NET kielet tulivat käyttöön. Oletuskielenä ASP.NET käyttää VB.NET-kieltä. Nykyään ASP.NET tukee miltei mitä tahansa kieltä, kunhan se on tuettu .NET ympäristössä.

ASP.NET on saanut uuden version miltei vuosittain. Vuodesta 2002 (virallinen julkaisu vuosi) lähtien on lähes joka vuosi tullut uusi versio ASP.NET:stä ja uusin versio ASP.NET 4.0 julkaistiin syksyllä 2010. Versio 4.0 tulee osana uutta Visual Studio 2010 RC:tä



## 4 SQL SERVER 2008 ASENNUS

### 4.1 Asennus

Asennuksen aluksi ohjelma kertoo vaativansa Microsoft .NET Framework 3.5 SP1-ohjelman toimiakseen. Jos käytettävässä tietokoneessa on vanhempi versio .NET Frameworkista, asennusohjelma pyytää lataamaan vaaditun version Microsoft Update-sivustolta.

Kun on hyväksytty käyttöehtosopimus, klikataan alareunasta Asenna jonka jälkeen siirrytään SQL Server Installation Center-valikkoon. Valitaan vasemmasta yläreunasta kohta Planning. Ennen itse ohjelman asennusta täytyy tarkastaa, onko tietokoneella SQL Serverin tarvitsemat ohjelmat ja päivitykset. Tämä onnistuu valitsemalla kohta - > System Configuration Checker.

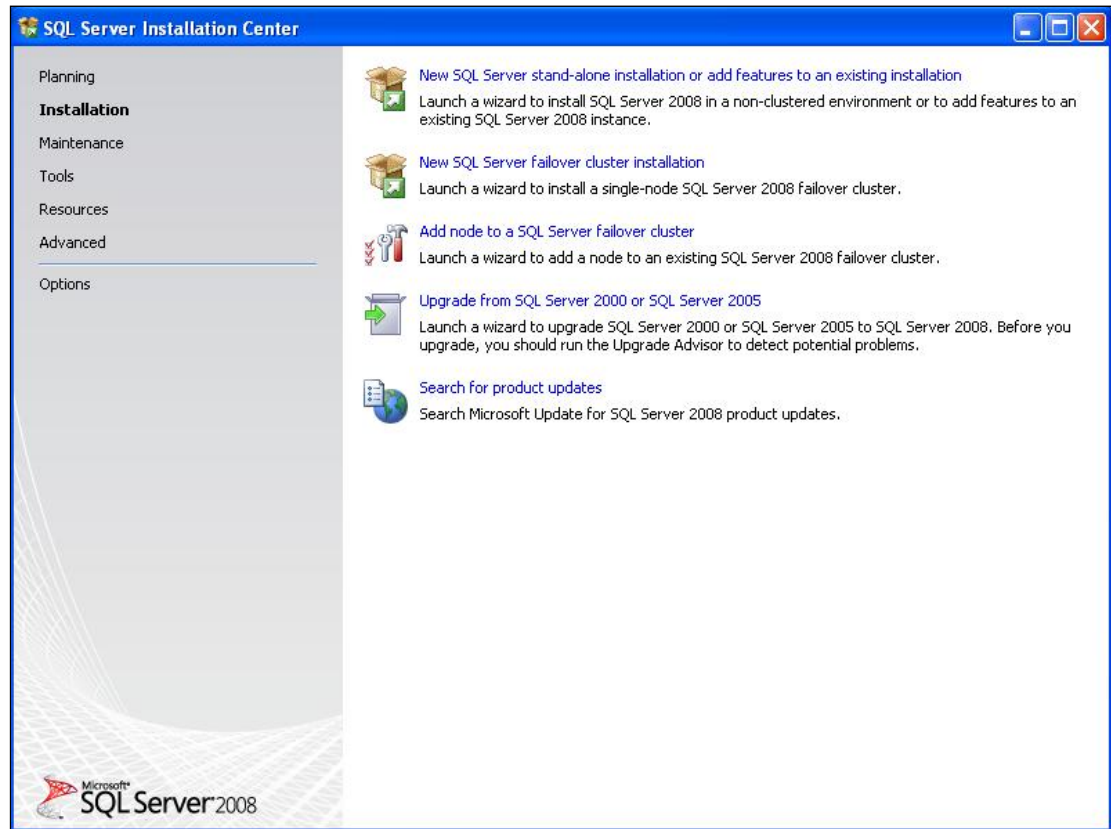


**Kuva 1. Planning**

System Configuration Checker listaa tarvitsemansa vaiheet ja ilmoittaa, jos jokin kohta ei täyttänyt ohjelman vaatimuksia. Asennus-ohjelma antaa tarvittaessa ilmoituksen

puutteesta/virheestä ja kehottaa korjaamaan puutteet/virheet. Jos Windows Power Shell tai .NET Application puuttuvat tai ovat liian vanhaa versiota ilmestyy listaan puutteen perään linkki joka vie suoraan Windows Update sivustolle josta voi ladata puuttuvan päivityksen. (Lista samanlainen kuin Setup-vaiheessa, katso Kuva 3.)

Seuraavaksi siirrytään Installation-kohtaan josta valitaan ylin vaihtoehto -> New SQL Server stand-alone installation or add features....



**Kuva 2.Installation**

Ohjelma tarkastaa uudestaan tietyt parametrit koneeltasi ja ilmoittaa niistä vaihtoehtoin: passed, failed, warning tai skipped. Varoitukset eivät vielä pysäytä asennusta, mutta jos jokin kohta on tilassa failed, joudut korjaamaan asian ohjelman antamien neuvojen avulla ennen kuin asennus jatkuu eteenpäin.

## **Tärkeää:**

SQL Server 2008 asennettaessa täytyy olla kirjautuneena Windowsiin pääkäyttäjänä, jos asennusta yrittää muulla kuin pääkäyttäjän profiililla Setup Support Rules kohdassa tulee Setup Administrator statukseksi Failed.

Joskus käyttäjä profiiliin on merkattu pääkäyttäjän/administrator vaikka todellisuudessa profiililla ei ole täysiä administrator oikeuksia. Tämä ongelma juontaa juurensa Windowsin ensiasennuksessa tehtyihin valintoihin (Windows XP).

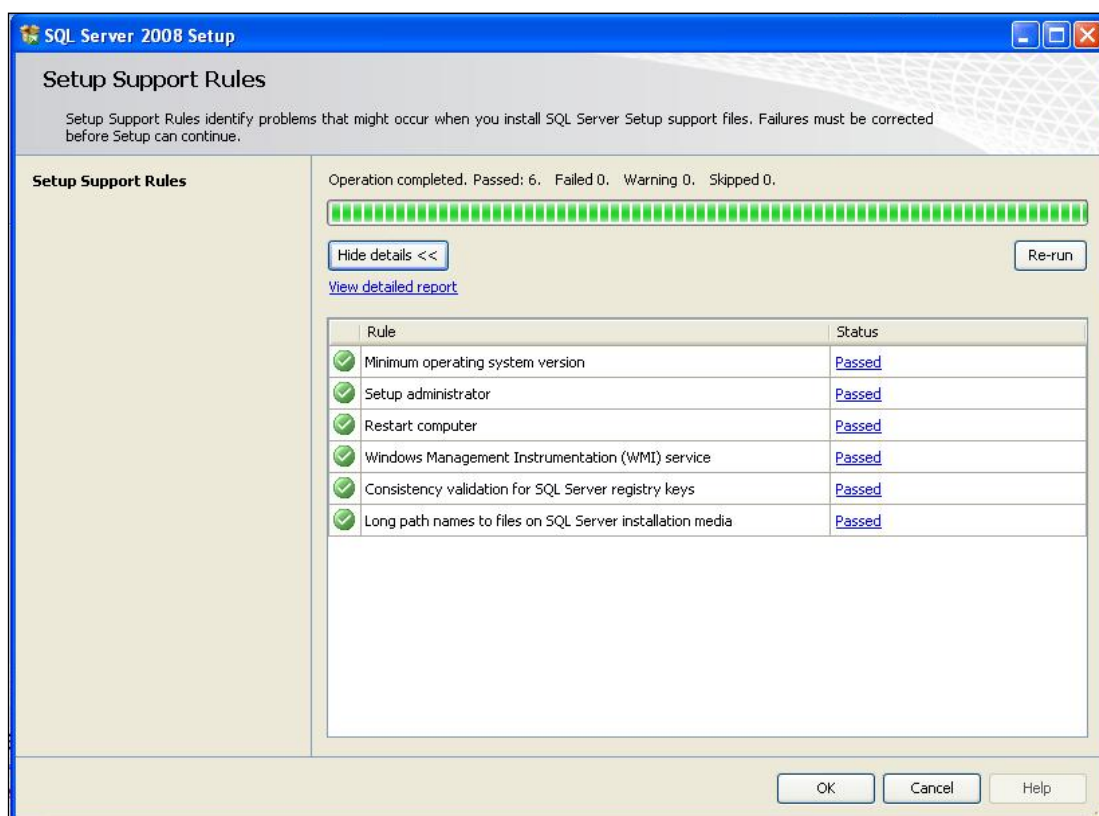
Pakettikoneet eli koneet joissa on esiasennettuna jokin käyttöjärjestelmä ovat ongelmallisia pääkäyttäjän suhteen koska näissä ”oikea” pääkäyttäjätunnus on piilotettuna ja sen on luonut tietokoneeseen käyttöjärjestelmän esiasentanut yritys. Tämän käyttöjärjestelmän imagen luonut henkilö on tehnyt esiasennuksen omalla tunnuksellaan ja tämä ensimmäinen käyttöjärjestelmän rekisteröimä tunnus on ainut oikea pääkäyttäjätunnus. Käyttöjärjestelmä ilmoittaa tietokoneen omistajalle joka on asentanut käyttöjärjestelmän asennus imagesta että omistajan luoma käyttäjätunnus olisi pääkäyttäjätunnus ja se onkin sitä mutta vasta sekundaarinen pääkäyttäjätunnus oikeasti. Tämänlaisia pääkäyttäjätunnuksia saa luotua Windowsin käyttäjätilit-valikossa ja käyttöoikeus erot huomaa vasta kun on käyttänyt käyttöjärjestelmää molemmilla pääkäyttäjätunnuksilla, oikeilla ja väärillä. Helpoiten selvyuden omiin käyttäjätunnuksiin saa asentamalla esimerkiksi Spybot-ohjelman ja yrittämällä ohjelman avulla poistaa väliaikaistiedostoja tai immunisoida järjestelmää, jos ohjelma suostuu poistamaan kaikki väliaikaistiedostot, niin sinulla on käytössä pääkäyttäjän oikeuksilla olevat tunnukset käyttöjärjestelmääsi. Immunisoidessa ohjelma jättää immunisoimatta MS Explorer selaimen mutta immunisoi muut selaimet koska ne on asennettu (oletetusti) eri käyttäjätunnuksella kuin alkuperäisen käyttöjärjestelmän imagen tekijän tunnuksella.

TweakUI-ohjelmalla pääkäyttäjää etsiessä tunnus löytyy melko helposti koska yleensä tietokonevalmistajat mainostavat tunnuksessakin itseään eli esimerkiksi Hewlett Packard suosii tunnusta nimeltä ”HP\_Administrator”. Tämänlaiset käyttäjätunnukset ovat helppo löytää listasta.

”Oikean” pääkäyttäjän pystyy määrittämään esimerkiksi TweakUI-ohjelmalla.  
(<http://www.microsoft.com/windowsxp/Downloads/powertoys/Xppowertoys.msp>)

Spybot-S&D-ohjelman löytää osoitteesta:

<http://www.safer-networking.org/>



**Kuva 3. Setup Support Rules**

Tämän jälkeen varsinainen asennus-vaihe pääsee alkamaan. Asennuksen aikana ohjelma kysyy tapaa, jolla haluat kirjautua SQL Serveriin. Vaihtoehtoina on:

- Windows Authentication
- SQL Server Authentication

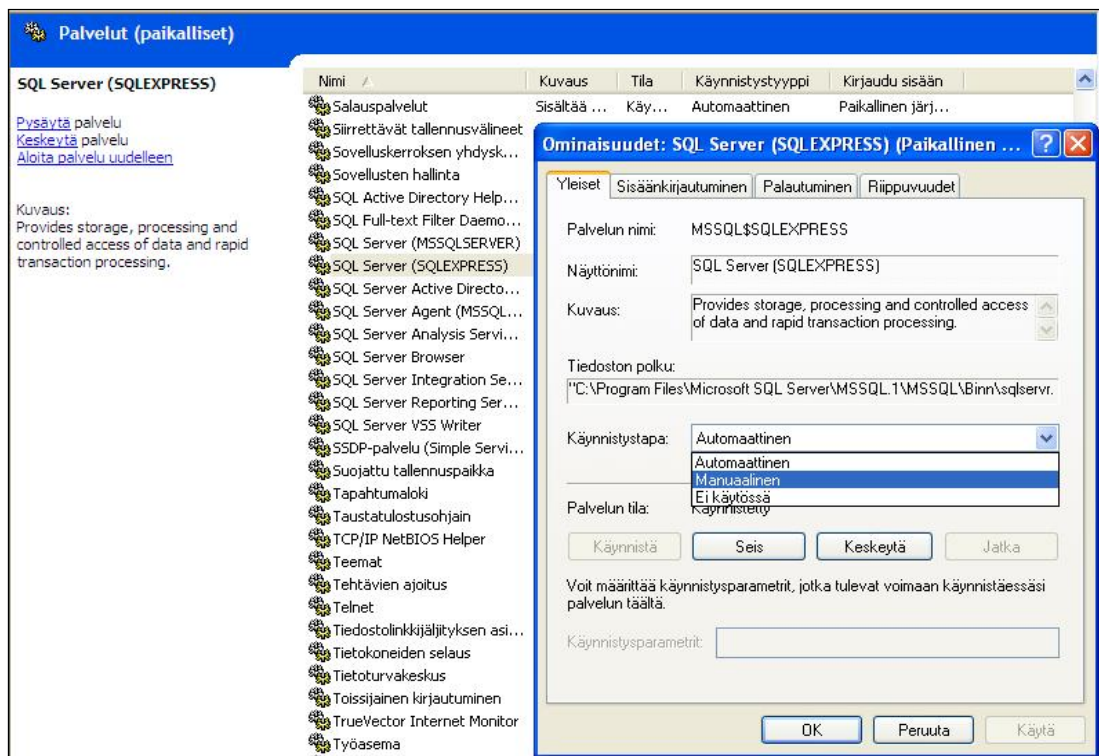
Valitaan vaihtoehto Windows Authentication, jos et ole vielä miettinyt sopivaa käyttäjätunnusta ja salasanaa. Kirjautumistapa ei ole sitova ja sen voi muuttaa asennuksen jälkeen. Asiasta tarkemmin kohdassa SQL Server Configuration Manager, sekä SQL Server 2008.

## 4.2 Asennuksen jälkeen

Tämä vaihe on vapaaehtoinen. Tällä työvaiheella pyritään nopeuttamaan koneen toimintaa ja vapauttamaan muistiresursseja silloin, kun SQL Serveriä ei tarvitse pitää päällä.

Jos käytät Windows xp:tä tee näin: **Käynnistä->Ohjauspaneeli->Valvontatyökalut->Palvelut**. Palvelut kuvaketta kaksoisklikkaamalla avautuu lista kaikista paikallisista palveluista. Etsitään listasta SQL Server (SQLEXPRESS). SQL Serverille on annettu oletusarvona automaattinen käynnistys, jos haluat esimerkiksi juuri muistin varauksen takia estää serveriä käynnistymästä automaattisesti, kaksoisklikkaa tekstin kohdalta ja valitse avautuneesta ikkunasta käynnistystavan kohdalta-> Manuaalinen (Kuva 4). Tämän valittuasi voit käynnistää ja sulkea SQL Serverin silloin kun itse haluat, joudut vain tulemaan palvelut-kansioon joka kerta sen tehdäksesi (pikakuvake työpöydällä olisi hyödyllinen).

Serverin käynnistäminen ja sulkeminen täällä on ainut keino "oikeasti" sulkea ja ns. nollata serveri. Kuten Windowsin, SQL Server tarvitsee joskus uudelleenkäynnistysten toimiakseen taas moitteettomasti.



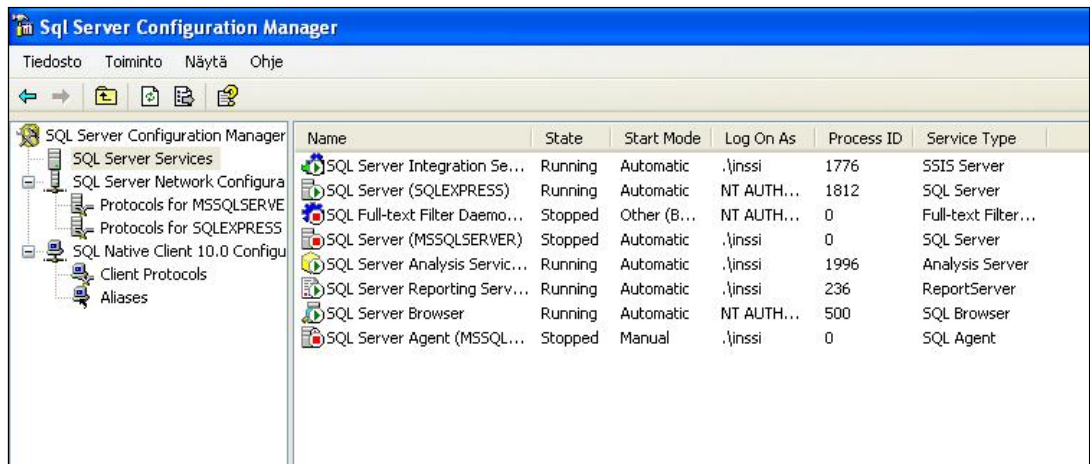
**Kuva 4. Käynnistystavan muuttaminen**

### 4.3 SQL Server Configuration Manager.

Tällä ohjelmalla voi tehdä osittain samoja asioita kuin Windowsin Palvelut-kansiossa. Tämä ohjelma on tehty serverin palveluiden muokkaamiseen, pysäyttämiseen, sulkemiseen, käynnistämiseen ja uudelleenkäynnistämiseen.

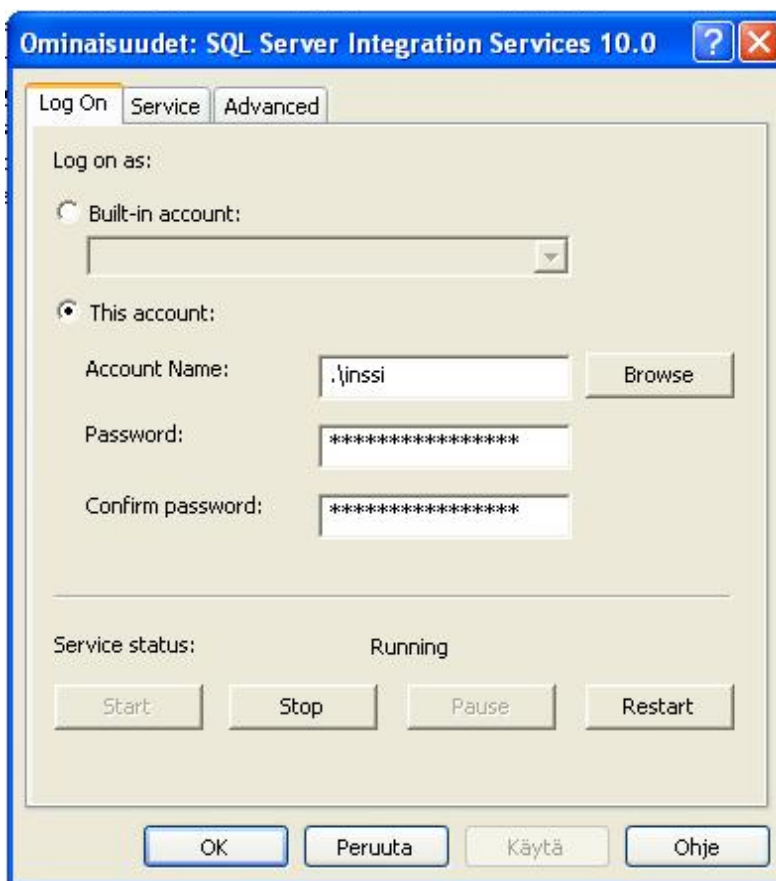
Kohdassa SQL Server Services näytetään:

- Palvelun nimi
- Tila
- Käynnistystapa
- Käyttäjä ("Log on as")
- Prosessin ID
- Palvelun tyyppi



**Kuva 5. SQL Server Configuration Manager**

Hiiren oikealla napilla klikataan halutun palvelun nimen päällä, jolloin saadaan uusi valikko näkyviin.

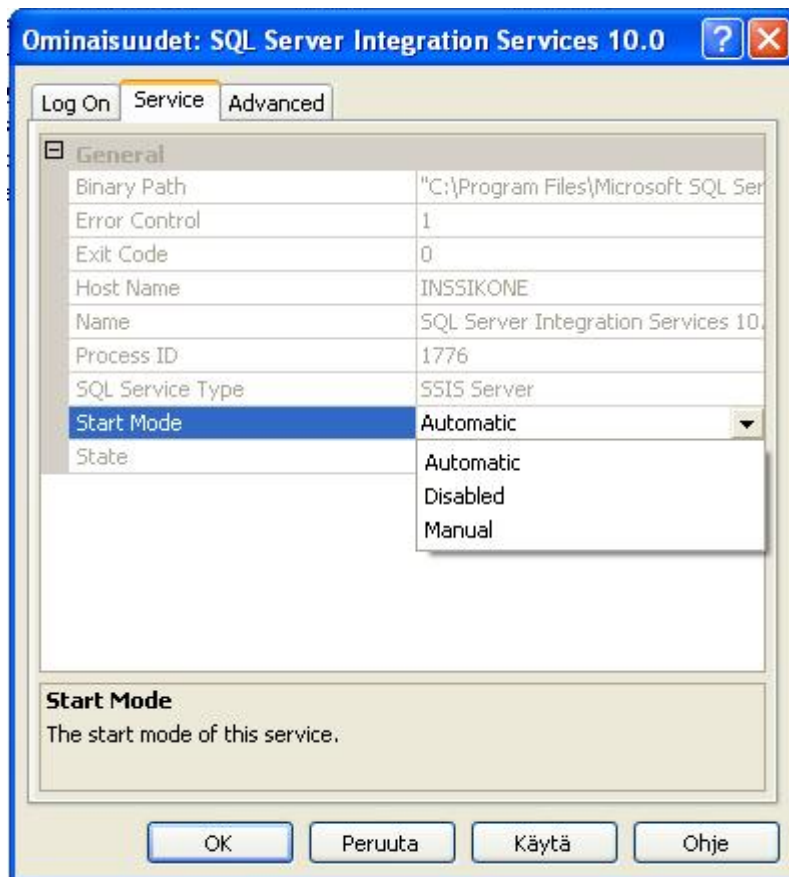


**Kuva 6. Log On**

Log on-välilehdellä voidaan valita, mitä tapaa käytetään kyseiseen palveluun kirjautumiseen. Built-in account-kohdassa on pudotusvalikossa vaihtoehtoina: Local System, Local Service ja Network service. Kohdassa This account voidaan valita tietty

käyttäjaprofiili tunnuksineen kirjautumisvaihtoehdoksi. Kuvassa 6, valittuna profiilina on profiili, jolla on kirjaututtu Windowsiin. Browse-napin kautta voidaan valita myös muu kuin tällä hetkellä kirjautuneena oleva profiili.

Välilehdellä Service (Kuva 7.), voidaan valita käynnistyykö kyseinen ohjelma automaattisesti vai manuaalisesti tai ohjelmaa voidaan estää käynnistymästä kokonaan.



**Kuva 7. Start Mode valikko**

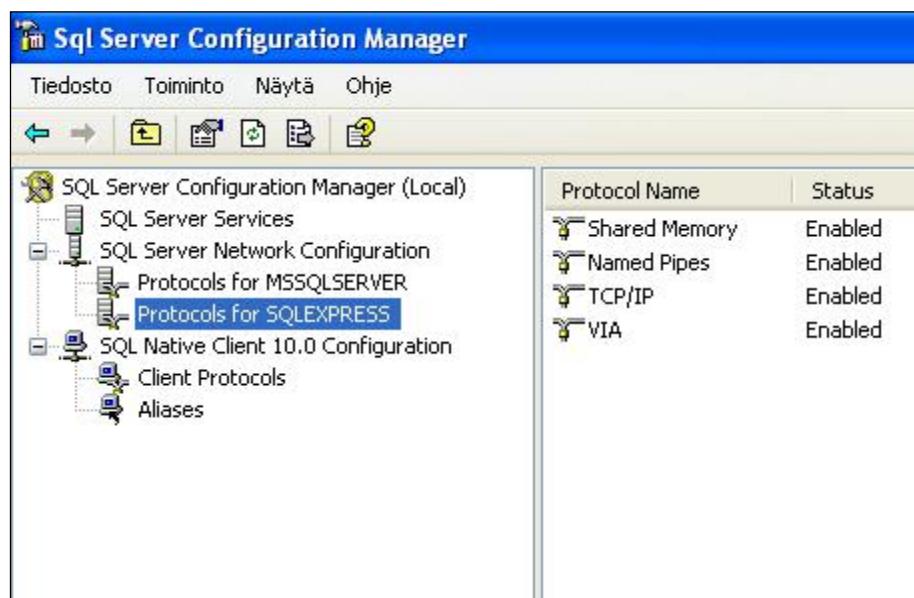
#### 4.4 Yhteydet

Voidakseen kommunikoida SQL Serveriä käyttävien ohjelmien kanssa, esimerkiksi ASP.NET tekniikalla tehdyt internet sivut, täytyy Configuration Managerissa enableoida siihen käytettävät protokollat (Kuva 8).



Näistä kolme tärkeintä protokollaa on:

- Shared Memory: Sallii yhteydet paikalliseen serveriin ilman verkkoyhteyttä. Serveriin voidaan ottaa ainoastaan yhteys paikallisesti. Tähän yhteysprotokollaan ei ole mahdollista tehdä konfiguraatiomuutoksia.
- Named Pipes: Suorittaa tiedonsiirron prosessien välissä ja on siten tehokas tiedonsiirtotapa lähiverkoissa (LAN).
- TCP/IP: Yleisin verkkoprotokolla, jonka avulla otetaan yhteys SQL Serveriin. TCP/IP yhteys on helpoin ja tuetuin yhteysmuoto erilaisissa verkkoympäristöissä.



**Kuva 8. Tiedonsiirto protokollat**

Kun SQL Serveriä käyttää paikallisena asennuksena, tarvitsee siis enableoida ainoastaan Shared Memory, jos otetaan internetin kautta yhteys serveriin täytyy TCP/IP enableoida.

TCP/IP:n kohdalta kaksi kertaa klikkaamalla saadaan auki valikko, jossa on kaksi sivua: Protocol ja IP Addressess. IP Addressess sivulta löytyy kaikki tarvittava tieto yhteyden luomiseen etätietokoneelta serveriin, sekä tietoja pystyy myös muokkaamaan haluamikseen.

## 5 SQL SERVER 2008

Kun ohjelma on käynnistetty, aukeaa ikkuna nimeltä ”Connect to server”.



**Kuva 10. Kirjautumisikkuna**

Tässä ikkunassa määritellään serverin tyyppi, nimi ja sisäänkirjautumistapa. Tarkistetaan, että kohdassa ”Server type” lukee ”Database Engine”, jos sen tilalla lukee jotain muuta, etsi oikealta pudotus-valikosta oikea teksti.

Seuraavaksi siirrytään pykälä alemmaksi ja muutetaan serverin nimi toiseksi. Huom. serveri toimii aivan yhtä hyvin oletus nimelläkin, mutta varmistetaan, että kaikki toimii mutkattomasti, kun päästään asp.net-osiossa ottamaan yhteyttä serveriin niin muutetaan serverin nimen polku toiseksi. Vaihda ”Server name” kohtaan teksti: ”localhost\SQLEXPRESS”.

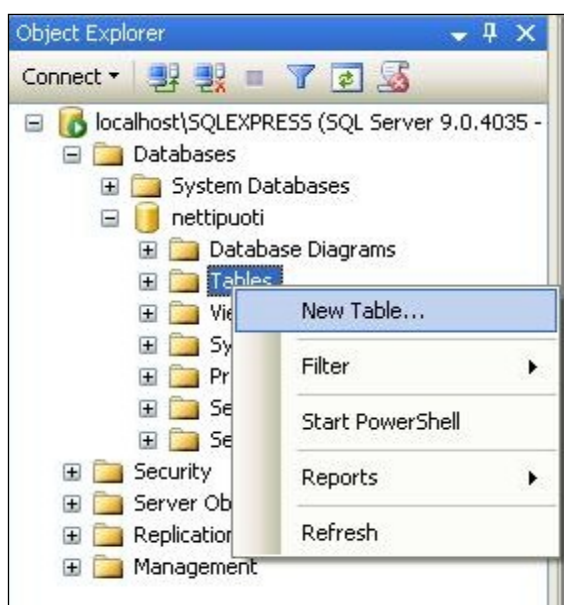
Seuraavaksi tulee ”Authentication” kohta. Tässä kohdassa valitaan kumpaa kirjautumistapaa halutaan käyttää, vaihtoehdot ovat ”Windows Authentication” tai ”SQL Server Authentication”. Tässä työssä valitsin Windows kirjautumisen, koska käytän ohjelmaa paikallisena, jolloin tietoturvariski on pieni, sekä valitsin ohjelmaa asentaessa Windows kirjautumisen enkä luonut SQL Server-tunnuksia. SQL Server-tunnukset kannattaa luoda, jos ottaa yhteyden serveriin verkon yli.

Valitaan siis ”Windows Authentication”, jolloin kirjaudut sisälle serveriin jo Windowsiin kirjautuessa antamalla tunnuksillasi. Sinun ei siis tarvitse kirjoittaa niitä uudestaan alla oleviin ”User name” ja ”Password” kohtiin.



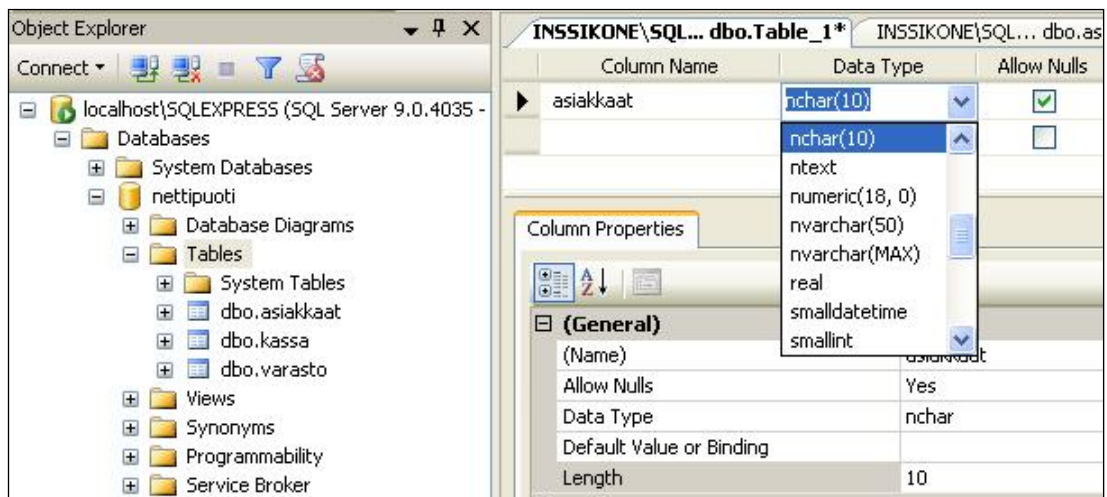
Ruudun vasemmassa reunasta Databases kansion sisältä pitäisi nyt löytyä juuri luotu ”nettipuoti”-tietokanta. Seuraavaksi lisäämme tietokantaan tietoja.

Klikataan nettipuoti-kansio auki, jolloin valikkoon aukeaa monia uusia kansioita. Valitaan ”Tables”-kohta ja klikataan hiiren oikealla sen päällä ja valitaan kohta ”New Table”.



**Kuva 12. New Table**

Ruudun keskelle aukeaa ikkuna, jossa kirjoitetaan taulun sisältämien eri sarakkeiden nimet kohtaan ”Column Name”, esim. etunimi, sukunimi, kaupunki. Seuraavassa kohdassa valitaan, minkä tyyppistä dataa nimi-kohta sisältää, vaihtoehtoja on valuatusta integeriin saakka. Ohjelma antaa oletuksena tyyppin nchar(10) ja kun klikkaat tekstiä char(10) niin saat esille pudotusvalikon josta voit valita mieleisesi ja tarpeisiisi sopivan datatyyppin (Kuva 13).



**Kuva 13. Taulun määrittely**

Tässä tapauksessa nchar(10) on liian lyhyt merkkimäärältään, joten sitä pitää muokata pidemmäksi. Tämä muokkaus onnistuu menemällä alla olevaan ”Column Properties” kohtaan ja etsiä sieltä kohta ”Length”. Length kohdassa on alkuarvona kymmenen, kuten nchar(10) jo sen kertoo, sen saa muutettua klikkaamalla numero kymmenen päällä ja painamalla Delete nappia, jonka jälkeen voi syöttää haluamansa luvun.

Kolmas kohta valikossa on ”Allow Nulls” eli sallitaan tyhjät sarakkeet ja/tai rivit. Klikataan valinta ruutua ”Allow Nulls”-kohdan alla.

Tietokannassa kannattaa olla myös sarake, joka pitää kirjaa tietokannassa olevista tiedoista ja jota voidaan käyttää haun vertaamiseen tietokannassa oleviin tietoihin. Tällainen sarake nimetään yleensä Id sarakkeeksi. Tällaisen sarakkeen ei tarvitse olla nimeltään Id, se voi olla mikä tahansa, kunhan sarakkeeseen on asetettu Primary Key arvo. Tämän lisääminen tehdään siten, että klikataan hiiren oikealla sarakkeen nimen vasemmalla puolella olevasta nuolesta (Kuten kuvassa Kuva 13) ja valitsemalla auneesta valikosta Set Primary Key. Nuolen seuraksi ilmestyy avain, joka ilmaisee sarakkeen olevan Primary Key. Tämän lisäksi tästä ”Id” sarakkeesta tehdään Identity Column, joka määritetään ruudun oikeassa reunassa olevassa Properties-valikossa. Valitaan haluttu sarake pudotusvalikosta, jos Primary Key on asetettu niin pudotusvalikossa ei pitäisi olla kuin Primary Key-sarake.

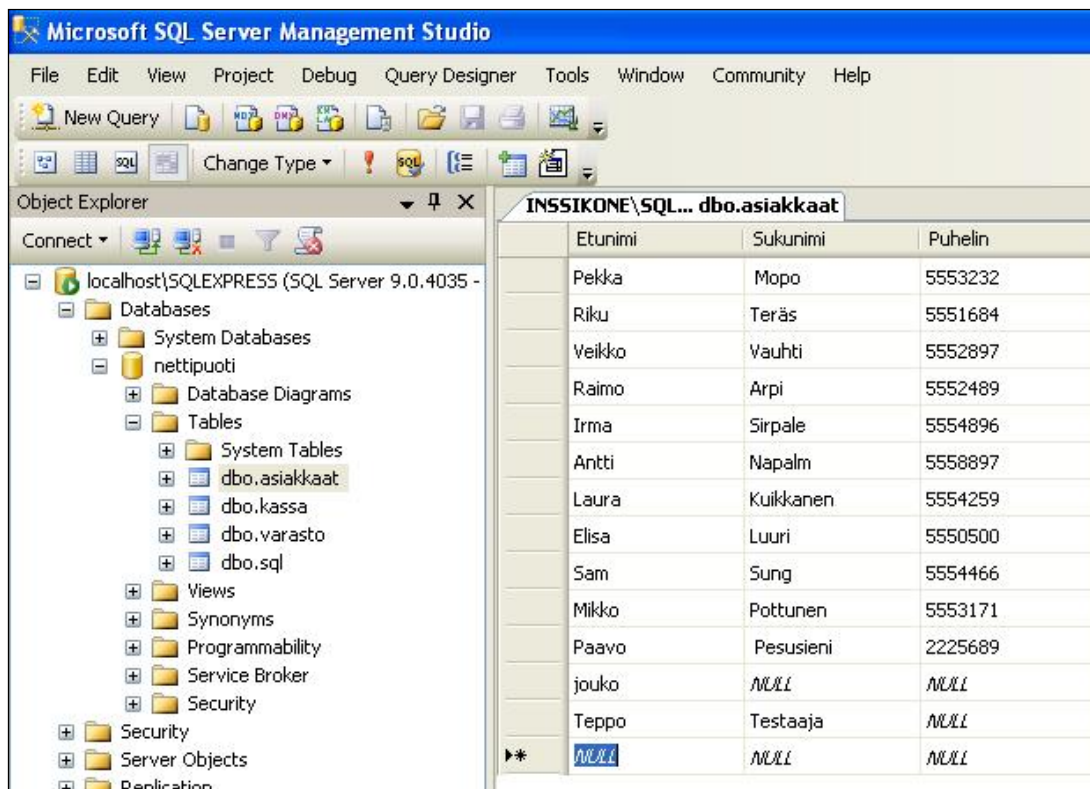
Tämän jälkeen tarvitsee enää tallentaa juuri tehty taulu. Ennen sitä annetaan taululle nimi ja sen pystyy tekemään ruudun oikeassa reunassa olevasta ”Properties”-valikosta kohdasta ”Identity”

Korvataan oletuksena oleva Table\_1, asiakkaat-nimellä. Tämän jälkeen klikataan hiiren oikealla napilla ruudun keskellä olevan ikkunan yläreunassa jossa lukee polku tähän tauluun. Tässä tapauksessa ”INSSIKONE\SQL...dbo.asiakkaat\*”. Auenneesta valikosta valitaan ensimmäinen vaihtoehto ”Save”, jolla tallennetaan työ.

Kun taulu on luotu, pitäisi seuraavaksi lisätä tietoja siihen, tämän voi tehdä SQL Server 2008:ssa monella eri tavalla. Seuraavaksi käyn läpi kaksi perinteisintä tapaa lisätä tietoja.

## **5.2 Tietojen lisääminen ja muokkaus**

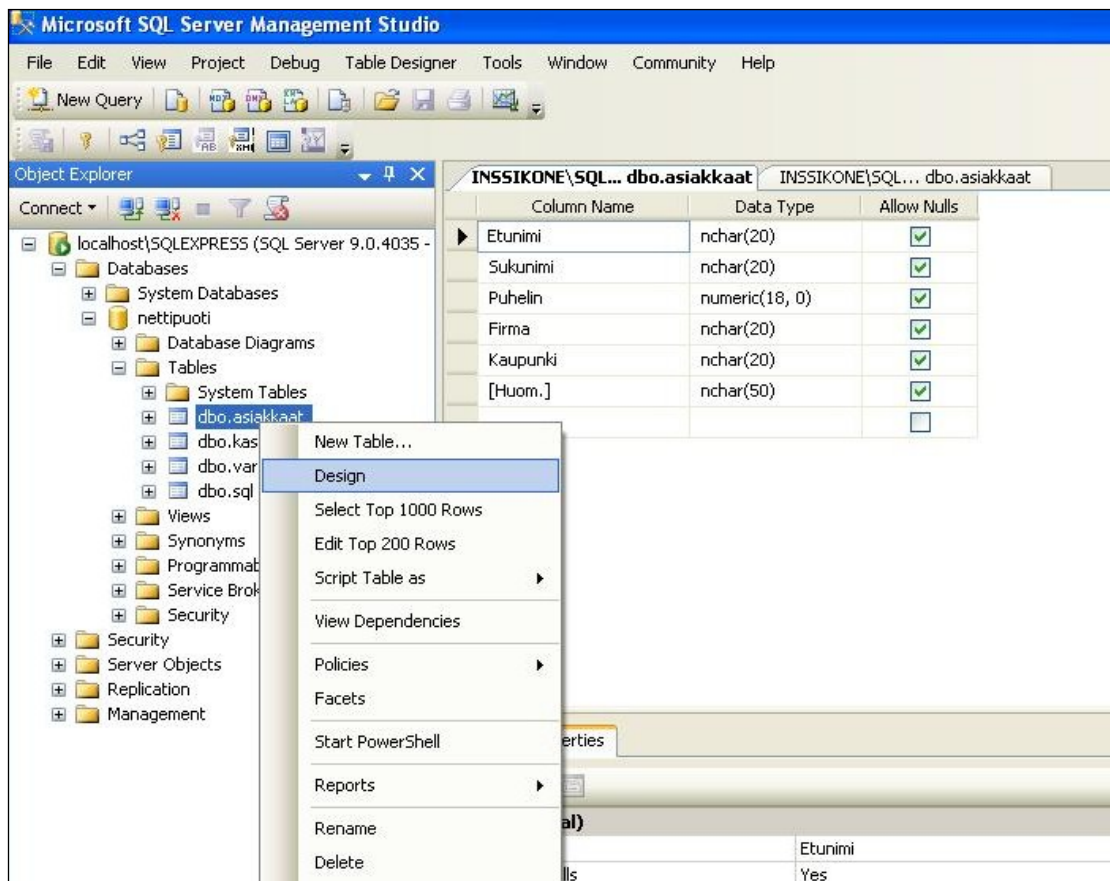
Toimitaan aluksi kuten edelläkin eli valitaan ”Edit Top 200 Rows”. Keskelle ruutua aukeaa ikkuna, jossa on yläreunassa jo taulun luonnin yhteydessä nimetyt sarakkeet. Jokaisen sarakkeen kohdalla on oletus arvo ”NULL”. Klikkaamalla *NULL*-kohtaa teksti saa sinisen taustan. Tämän jälkeen voidaan alkaa kirjoittamaan tietoja sarakkeisiin. Sarakkeille annetut datatyytit (char, varchar, money yms.) vaikuttavat nyt annettaviin tietoihin. Esimerkiksi sarake ”Puhelin” on tyybiltään numeric(18) ja ei siten hyväksy kirjaimia tai erikoismerkkejä, vaan ainoastaan numeroita. Kun tarvittavat tiedot on syötetty, työ tallennetaan joko hiiren oikean napin painalluksella saatavasta valikosta valitsemalla Execute SQL, tai vaihtoehtoisesti työkalupalkista löytyvästä punaisesta huutomerkistä.



**Kuva 14.**

Tietoja lisätessä voi tulla ongelma sen takia, että on antanut liian pienen arvon esimerkiksi Kaupunki sarakkeelle (nchar(10)) ja nimi ei mahdu ennalta määrättyyn merkkimäärään. Sarakkeen datatyyppiä voi vielä muuttaa.

Klikataan hiiren oikealla napilla dbo.asiakkaat kohdalla ja valitaan kohta ”Design” (Kuva 15.). Tällä valinnalla päästään muuttamaan kaikkia alussa annettuja arvoja, nimeä, datatyyppiä ja sallitaanko tyhjät arvot. Puhelin sarakkeenkin datatyyppin voi vaihtaa nchar(20) tyyppiä, jos haluaa erottaa suuntanumeron viivalla muusta numerosta, esim. 555-1234.



**Kuva 15. Design**

Tietojen muokkaus aloitetaan klikkaamalla hiiren oikealla napilla dbo.asiakkaat kohdalla ja valitaan valikosta kohta Edit Top 200 Rows. Klikataan muokkausta tarvitsevan solun päällä ja kirjoitetaan uudet tiedot tilalle. Kun tiedot on muutettu, työ tallennetaan. Tämän saman voi tehdä myös kirjoittamalla SQL-komennot, silloin valitaan hiiren oikealla napilla esiin saatavasta valikosta Pane->SQL ja kirjoitetaan haluttu SQL-lause. Lisää kohdassa Tietojen muokkaus SQL-kielellä.

### 5.3 Tietojen haku

Tietojen hakuun on muutamia eri tapoja SQL Server 2008:ssa, esittelen niistä kaksi. Tietojen haku ensimmäisellä tavalla aloitetaan klikkaamalla hiiren oikealla halutun tietokannan päällä ja valitsemalla listasta Select Top 1000 Rows. Tämän valitsemalla ruudun keskelle tulee ylimmäiseksi SQL-kielellä juuri valittu komento. SQL-



komennon alapuolelle tulee haku tulokset (Kuva 16.). Saadut hakutulokset sisältävät 1000 ensimmäistä riviä ilman mitään tarkentavaa ehtoa. Hakua voi rajata tämän jälkeen kirjoittamalla SQL-kielellä lisäehtoja valmiiseen SQL-komentoon, joka on sivun yläreunassa (Kuva 16.). Haun voi siis asettaa koskemaan koko taulua, eikä ainoastaan ensimmäistä tuhatta riviä, tällöin pitää poistaa kohta ”TOP 1000” SQL-lauseesta. Tietojen hakeminen tietokannasta onnistuu parhaiten käyttämällä SQL-komentoja.

The screenshot shows a SQL query window titled "SQLQuery7.sql - localhost\.....2))". The query is as follows:

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP 1000 [Etunimi]
, [Sukunimi]
, [Puhelin]
, [Firma]
, [Kaupunki]
, [Huom.]
, [ID]
FROM [nettipuoti].[dbo].[asiakkaat]
  
```

Below the query window, the "Results" pane displays the following data:

	Etunimi	Sukunimi	Puhelin	Firma	Kaupunki	Huom.	ID
1	0	Testaaja	5551234	Mokia	Mokia	NULL	1
2	Simo	Serveri	5554321	Oraakkeli	Turku	NULL	2
3	Riku	Teräs	5554664	Veitsi Oy	Kerimäki	NULL	3
4	Ilpo	Vati	5557948	Plasto	Oulu	NULL	4
5	Matti	Meikäläinen	NULL	?	?	NULL	5
6	Anssi	Kukkonen	5551111	P.I.	Helsinki	NULL	6
7	Erkki	Petäjä	5558237	Sahapojat	Inari	NULL	7

**Kuva 16. Haku**

Tietojen haku toisella tavalla, klikataan hiiren oikealla nettipuoti-kohdan päällä (Kuva 15.) ja valitaan kohta New Query. Ruudun keskelle tulee tyhjä ikkuna, jossa tietojen hakeminen täytyy tehdä sql-kielellä. Tietoja voi hakea kaikista tauluista, jotka sisältyvät nettipuoti-tietokantaan.

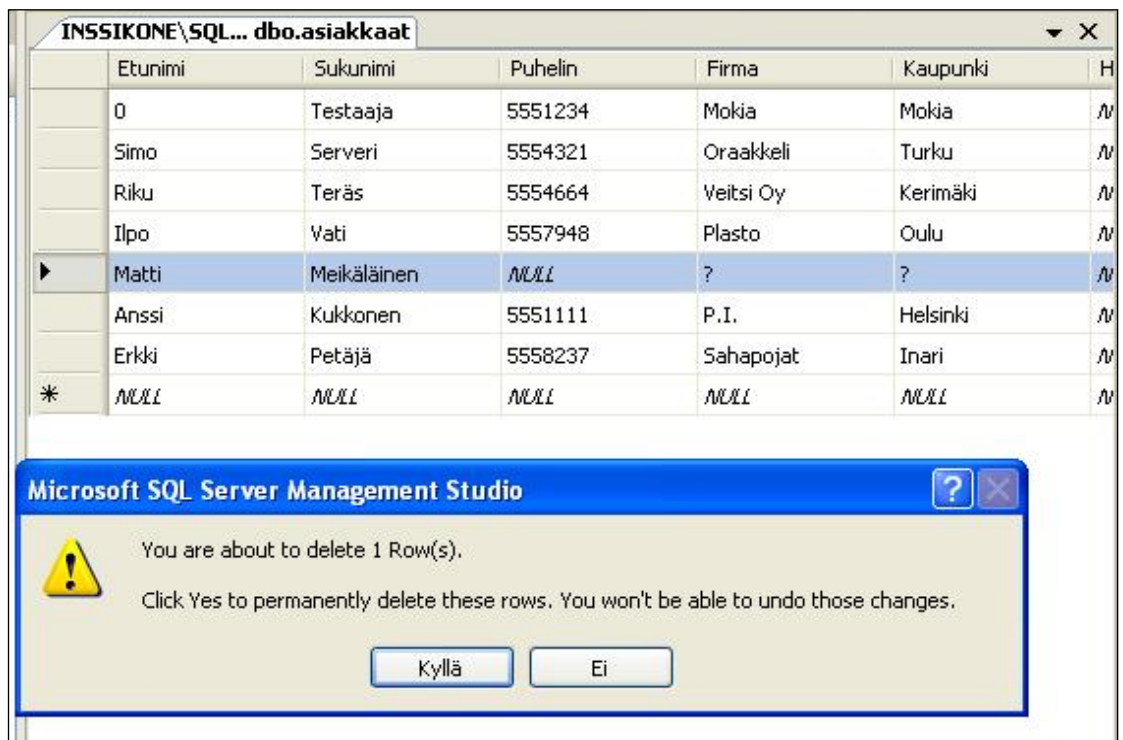
#### 5.4 Tietojen poisto

Tietojen poistaminen aloitetaan valitsemalla halutun tietokannan yllä ”Edit TOP 200 Rows”, jonka jälkeen ruudun keskelle aukeaa tiedot tietokannasta. Viedään hiiren kursori halutun rivin kohdalle, ja klikataan hiiren oikealla napilla vasemman reunimmaisesta nimen vieressä olevaa nuolta (Kuva 17.).



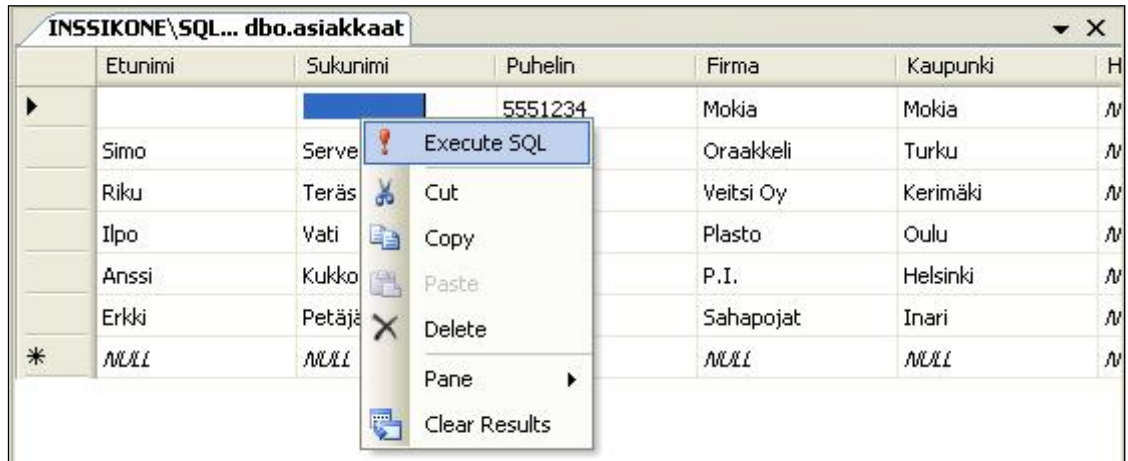
**Kuva 17. Delete**

Avautuneesta valikosta valitaan Delete, jolloin avautuu uusi ikkuna, jossa varmistetaan, haluatko poistaa kyseisen rivin. Valinta poistaa koko rivin, ei pelkästään yhden solun tietoa. Valinnalla ”Kyllä” poistetaan kaikki Matti Meikäläisen tiedot taulusta (Kuva 18.).



**Kuva 18. Varmistus**

Yksittäisten solujen tietojen poistaminen tapahtuu klikkaamalla haluttua solua, jonka jälkeen painamalla askelpalautinta poistetaan solun tiedot. Kun tiedot on hävinnyt solusta, täytyy poisto-operaatio hyväksyä painamalla Enteriä. Tämän jälkeen joko hiiren oikeata nappia painamalla avautuneesta valikosta Execute SQL valitsemalla, tai työkaluriviltä punaista huutomerkkiä painamalla toteutetaan tietojen poisto (Kuva 19).

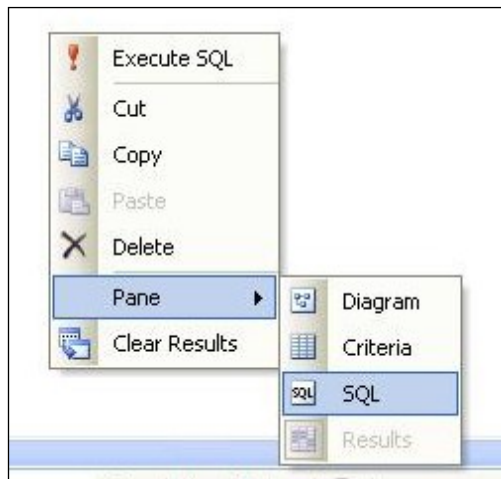


**Kuva 19. Execute SQL**

Jos tietoja poistaessa jättää pois Enterin painamisen ja siirtyy suoraan Execute SQL vaiheeseen, niin tiedot eivät poistu vaan generoituvat uudelleen. Tällä tavoin varmistetaan, että haluttu poisto operaatio on tehty tietoisesti eikä kyseessä ole tahaton askelpalauttimen painallus.

### 5.5 Taulun luominen ja tietojen lisääminen SQL-kielellä.

Klikataan hiiren oikealla napilla halutun taulun yllä ja valitaan valikosta ”Edit Top 200 Rows”. Ruudun keskelle aukeaa nyt taulu ”asiakkaat”, joka on täysin tyhjä. Klikataan hiiren oikealla napilla tyhjällä ruudulla, jolloin saadaan seuraavanlainen valikko näkyviin (Kuva 20.).



**Kuva 20. Pane**

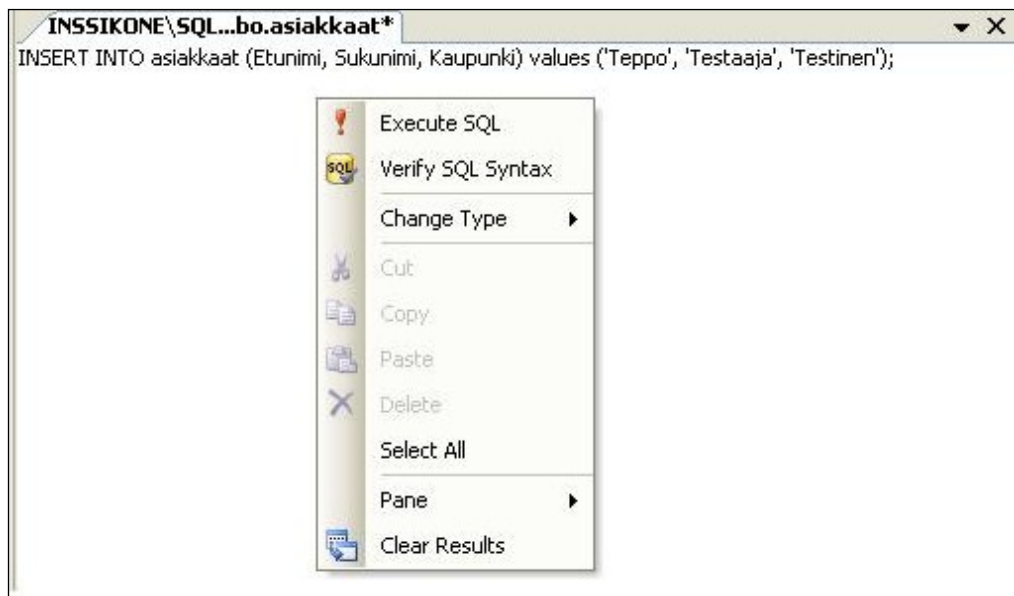
Valitaan valikosta kohta Pane, jolloin aukeaa uusi valikko, josta valitaan kohta SQL. Ruudun yläreunaan generoituu SQL-kielellä: `SELECT TOP (200) FROM asiakkaat`. Generoitunut sql-lauseke tekee saman toiminnon kuin äsken hiirellä valittu ”Edit Top 200 Rows”.

Poistetaan SELECT-lauseke (*lisää SELECT ehdosta sivulla 22*) ja kirjoitetaan tilalle tietojen syöttölauseke (INSERT). Oletetaan että edellisessä vaiheessa tauluun on luotu vaikkapa sarakkeet: etunimi, sukunimi ja kaupunki.

Kirjoitetaan seuraavanlainen SQL-lause:

**INSERT INTO asiakkaat (etunimi, sukunimi, kaupunki) values ('Teppo', 'Testaaja', 'Testinen');**

Tämän jälkeen hiiren oikealla napilla saadaan näkyviin valikko, jossa ylimpänä on ”Execute SQL”, jota klikkaamalla juuri tehty sql-lause suoritetaan ja tiedot lisätään tauluun (Kuvat 21 ja 22).



**Kuva 21. Lauseen toteuttaminen**



**Kuva 22. Tulos**

Insert-lauseen ei tarvitse sisältää vain yhden rivin verran tietoja vaan yhdellä insert-lauseella voi lisätä useita satoja rivejä kerrallaan sekä luoda taulun ensiksi ja sen perään lisätä tiedot sarakkeille ja riveille. Esimerkissä luodaan aluksi itse taulu asiakkaat, jonka jälkeen nimetään sarakkeet ja määritetään niiden tiedot.

Esim. 1

```
CREATE TABLE Asiakkaat (  
    id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,  
    Etunimi    char(20) NOT NULL,  
    Sukunimi   char(20),  
    Kaupunki  char(20),  
    PRIMARY KEY (id)  
) TYPE=InnoDB;  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Tep-  
po','Testaaja','Testilä');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Mat-  
ti','Meikäläinen','Mattila');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Töp-  
pö','Hönönen','Töppölä');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Ri-  
ku','Teräs','Sorsakoski');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Sep-  
po','Testaaja','Testilä');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Höp-  
pö','Tönönen','Töppölä');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Ri-  
ku','Rauta','Sorsakoski');  
insert into Asiakkaat (Etunimi, Sukunimi, Kaupunki) values ('Mai-  
ja','Meikä','Mattila');
```

## 5.6 Tietojen muokkaus SQL-kielillä

Tietoja lisätään INSERT-ehdolla, kun taas tietoja muokataan UPDATE-ehdolla. UPDATE-ehdolla ei pysty poistamaan tietoja vaan ainoastaan muokkaamaan aiemmin annettuja arvoja.

Esimerkiksi taulussa dbo.asiakkaat on viisi saraketta:

- Etunimi
- Sukunimi
- Puhelin
- Firma
- Kaupunki

Kahdella alimmalla rivillä olevista tiedoista ei selviä kuin toisesta etunimi ja toisesta etunimi, sukunimi ja kaupunki (Kuva 23).



	Etunimi	Sukunimi	Puhelin	Firma	Kaupunki
	Pekka	Mopo	5553232	Rengas oy	Mikkeli
	Riku	Teräs	5551684	Perintä Teräs ky	Pori
	Veikko	Vauhti	5552897	Vaunula	Lappeenranta
	Raimo	Arpi	5552489	Teräveikot	Keuruu
	Irma	Sirpale	5554896	Lasikoru oy	Mikkeli
	Antti	Napalm	5558897	Nafta Export	Kotka
	Laura	Kuikkanen	5554259	Koru ja posliini	Iisalmi
	Elisa	Luuri	5550500	Mobilesolutions	Helsinki
	Sam	Sung	5554466	Radio ja TV	Turku
	Mikko	Pottunen	5553171	Rymy rakentajat...	Ristiina
	Paavo	Pesusieni	2225689	Pesula	Mikkeli
	jouko	NULL	NULL	NULL	NULL
	Teppo	Testaaja	NULL	NULL	Testinen
▶*	NULL	NULL	NULL	NULL	NULL

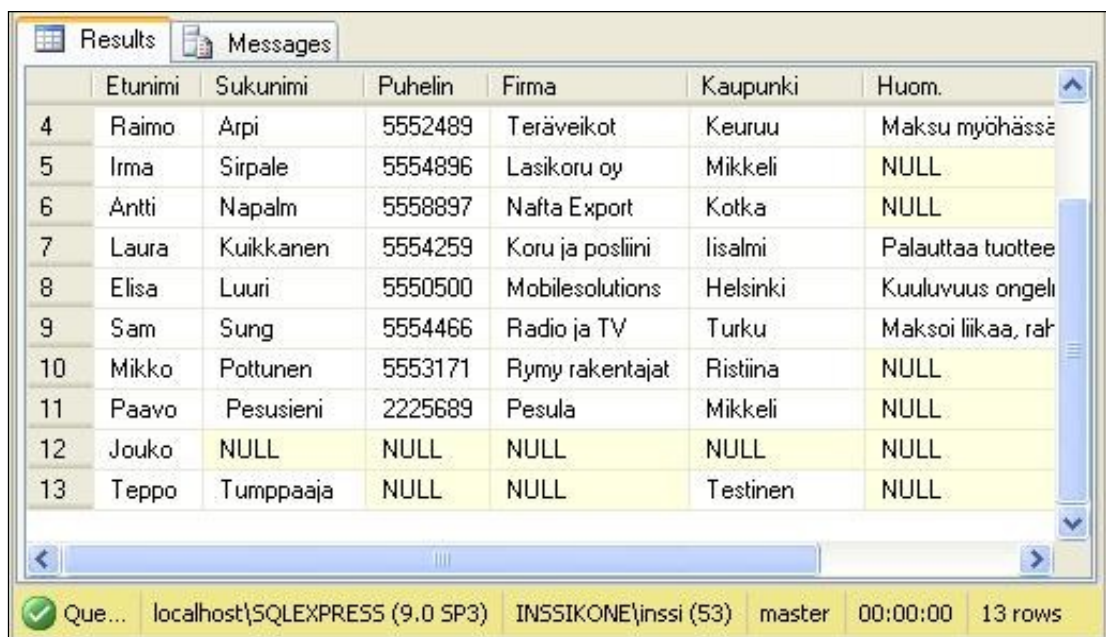
Kuva 23. Taulukko

Luodaan UPDATE-lause, jolla muutetaan toiseksi alimmalle riville etunimi isolla kirjaimella sekä alimmalle riville muutetaan sukunimi.

Esim. 2

**UPDATE asiakkaat SET Etunimi = 'Jouko' WHERE (Etunimi = 'jouko');**

**UPDATE asiakkaat SET Sukunimi = 'Tumppaaja' WHERE (Sukunimi = 'Testaaja');**



	Etunimi	Sukunimi	Puhelin	Firma	Kaupunki	Huom.
4	Raimo	Arpi	5552489	Teräveikot	Keuruu	Maksu myöhässä
5	Irma	Sirpale	5554896	Lasikoru oy	Mikkeli	NULL
6	Antti	Napalm	5558897	Nafta Export	Kotka	NULL
7	Laura	Kuikkanen	5554259	Koru ja posliini	Iisalmi	Palauttaa tuotteen
8	Elisa	Luuri	5550500	Mobilesolutions	Helsinki	Kuuluvuus ongelma
9	Sam	Sung	5554466	Radio ja TV	Turku	Maksoi liikaa, rah...
10	Mikko	Pottunen	5553171	Rymy rakentajat	Ristiina	NULL
11	Paavo	Pesusieni	2225689	Pesula	Mikkeli	NULL
12	Jouko	NULL	NULL	NULL	NULL	NULL
13	Teppo	Tumppaaja	NULL	NULL	Testinen	NULL

**Kuva 24. Tulokset SQL-lauseista**

Ehto WHERE täsmentää, mihin riviin muutos halutaan. Jos WHERE ehtoa ei käytetä, päivittää UPDATE esimerkiksi kaikki etunimet Joukoksi. Suuremmissa tietokannoissa tulee vastaan se, että on useita samannimisiä henkilöitä. Silloin kannattaa luoda jokaiselle riville oma id, jonka kautta kohdistaa UPDATE-ehto.

UPDATE-ehdolla voidaan muokata useampaa saraketta kerrallaan, esim:

**UPDATE asiakkaat SET Sukunimi = 'Koistinen', Kaupunki = 'Mikkeli'  
WHERE (Sukunimi = 'Tumppaaja');**

Riittää, että kerrot yhden sarakkeen tiedolla, mitä riviä haluat muokata, tässä tapauksessa sukunimellä Tumppaaja.



## 5.7 Tietojen haku SQL-kielellä

Tietojen hakemiseen käytetään SELECT-lausetta, siihen erilaisia kohdentavia ehtoja lisäten. SELECT-lauseella voidaan hakea tietoja yhdestä tai useammasta taulusta samalla kertaa.

Alla oleva lause hakee kaikki tiedot taulusta asiakkaat, koska siinä on käytetty \*-merkkiä.

```
SELECT * FROM asiakkaat;
```

Tässä lauseessa on kohdennettu, mitkä tiedot halutaan hakea taulusta asiakkaat.

```
SELECT Etunimi FROM asiakkaat;
```

Yleisimmin käytetyt ehto, määrittely ja koostefunktiot:

- FROM, Mistä taulusta tai tauluista tiedot haetaan
- WHERE, Mitkä rivit mukaan
- GROUP BY, Järjestele kaikki samannimiset arvot yhteen
- ORDER BY, Tulokset järjestetään esim. aakkosjärjestykseen
- HAVING, Rajataan ryhmiteltyä joukkoa

Funktiot GROUP BY ja ORDER BY tekevät miltei samat asiat kuin funktio WHERE, siksi niitä käytetäänkin usein yhdessä täsmentämään hakua.

Luodaan lause, jossa haetaan kuvitteellisesta taulusta tiedot asiakkaiden etunimistä ja sukunimistä, joiden sukunimi alkaa p-kirjaimella (LIKE 'P%') sekä järjestetään heidät järjestykseen etunimi, sukunimi.

```
SELECT Etunimi, Sukunimi FROM asiakkaat WHERE Sukunimi LIKE 'P%'  
GROUP BY Etunimi, Sukunimi;
```

## 5.8 Tietojen poisto SQL-kielillä

Tietoja poistetaan DELETE-ehdolla. DELETE-ehdolla voi poistaa yksittäisiä tietoja aina koko taulun tyhjentämiseen saakka. Taulun tuhoaminen tapahtuu DROP-ehdon avulla. DELETE / DROP ehdon kanssa täytyy olla varovainen, koska poistettuja tietoja ei saa takaisin kuin ne uudelleen luomalla.

Esimerkki X. Poistetaan yksittäisen henkilön tiedot.

**DELETE FROM asiakkaat WHERE Etunimi = 'Jouko';**

- Tällä lauseella on nyt poistettu Joukon tiedot taulukosta.

Esimerkki X. Koko taulun tyhjentäminen.

**DELETE \* FROM asiakkaat;**

- Tällä lauseella on nyt tyhjennetty koko taulu asiakkaat.

Esimerkki X. Taulun tuhoaminen

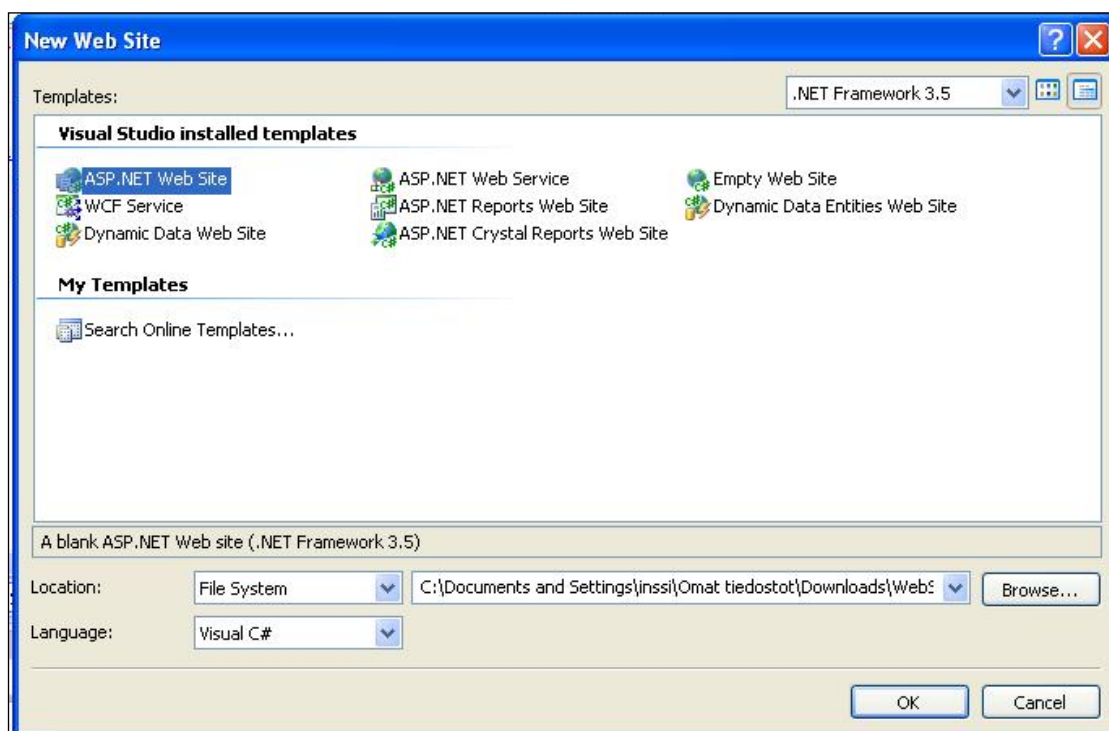
**DROP TABLE asiakkaat;**

- Taulu asiakkaat on poistettu kokonaan.

## 6 ASP.NET

ASP.NET-sivujen luomiseen käytetään Microsoft Visual Studio 2008-ohjelmaa. Visual Studio 2008 on hyvä työkalu niin aloitteleville koodaajille kuin myös ammattilaisille. Ohjelmassa on mahdollisuus suorittaa ohjelmointi täysin manuaalisesti kaiken koodin itse kirjoittamalla tai sitten käyttämällä apuna Toolbox-valikkoa, josta löytyy runsas valikoima valmiita ”työkaluja”. Toolbox-valikosta voi valita palikoita alasve-tovalikosta valmiiseen login-ohjelmaan saakka. Yleisin ohjelmointitapa on yhdistää manuaalinen ohjelmointi valmiisiin ohjelmapätkiin, joita Visual studiosta löytyy. Monesti pelkkä objektin valikosta valitseminen ja sen työhönsä asettaminen ei riitä, vaan objektille täytyy kertoa itse koodaamalla, kuinka sen täytyy toimia.

Kun ohjelma on käynnistynyt, aloitetaan luomaan uusi Web-sivu valitsemalla vasemmasta yläkulmasta File->New->Web Site, jolloin aukeaa seuraavanlainen ikkuna (Kuva 25.).

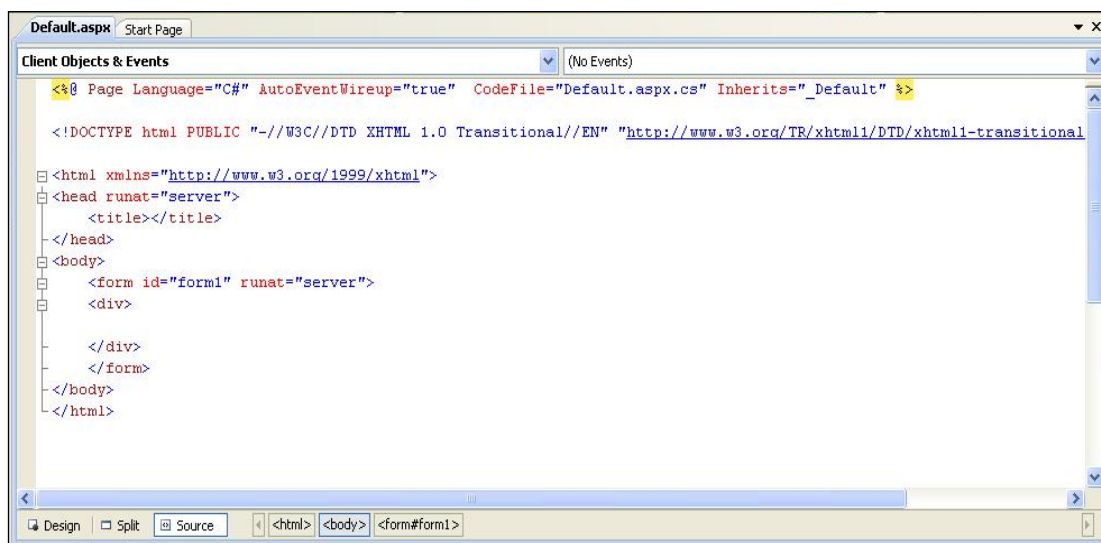


**Kuva 25. Valikko**

Valitaan ASP.NET Web Site vasemmalta yläreunasta. Valitaan vasemmasta alareunasta ohjelmointikieleksi Visual C# kohdasta Language. Location-kohtaan jätetään

oletuksena oleva File System, koska tallennamme internet-sivut C-asemalle tietokoneeseen, muut vaihtoehdot olisivat HTTP tai FTP-serveri. Klikataan OK.

Ohjelma luo pohjakoodin aspx-sivuille nimeltä Default.aspx (kuva 26.). Koodissa käy ilmi valittu ohjelmointikieli, josta voi vielä tarkistaa, että valitsi oikean kielen. Ohjelma loi valmiin html-pohjan sivua varten, johon tarvitsee vain lisätä haluamansa tiedot. Ohjelman valmiiksi luomaa tyhjää sivua voi esikatsella klikkaamalla koodin alapuolella olevasta palkista kohtaa Design.

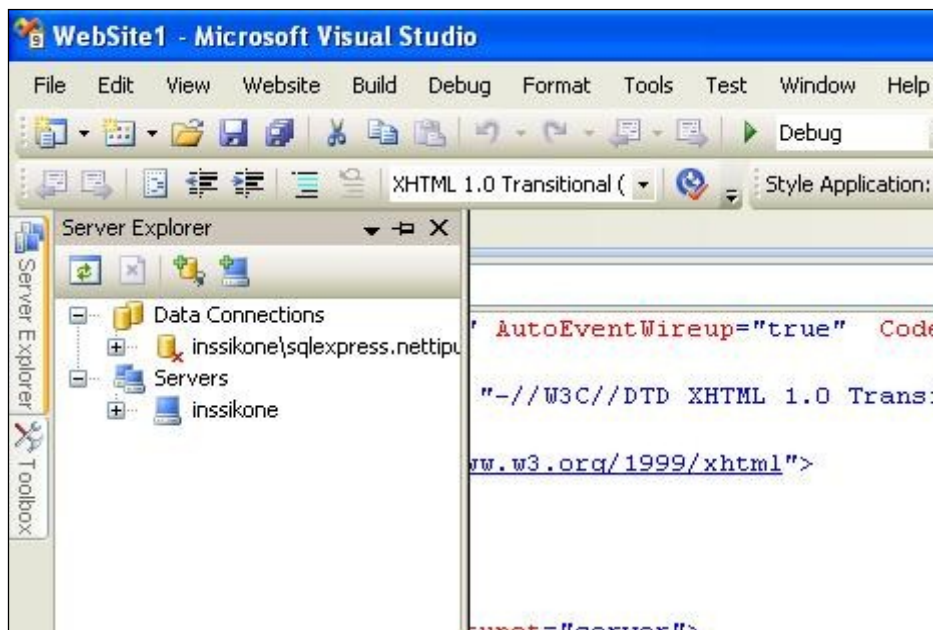


```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
</head>
<body>
<form id="form1" runat="server">
<div>
</div>
</form>
</body>
</html>
```

**Kuva 26. Generoitu koodi**

## 6.1 Tietokanta yhteyden luominen

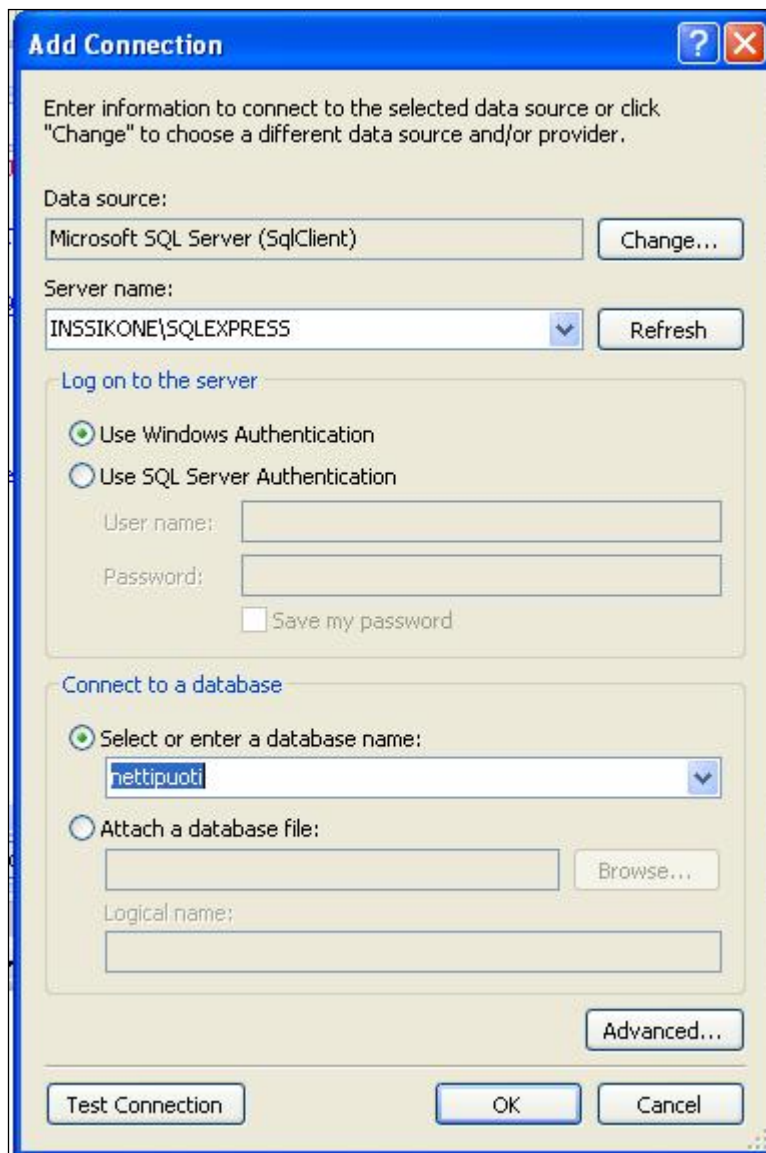
Viedään hiiren osoitin vasemmassa reunassa olevan Server Explorer valikon päälle, jolloin se liukuu näkyviin. Valikon saa pois näkyvistä esimerkiksi klikkaamalla html-koodia. Klikataan hiiren oikealla napilla Data Connections-kohdan päällä ja valitaan listasta Add Connection (Kuva 27.).



**Kuva 27. Yhteyden luominen**

Auennessa ikkunassa valitaan pudotusvalikosta Server Name oikea palvelin (Kuva 28.). Yleensä valikosta löytyy palvelin-vaihtoehdot heti, mutta joskus joutuu klikkaamaan Refresh-painiketta ennen kuin palvelimet ilmestyvät valikkoon.

Jos haluaa tietokantayhteyden olevan sidottuna tällä hetkellä käytettävään tietokoneeseen, silloin valitaan pudotusvalikosta palvelin, jonka nimessä on käytetyn tietokoneen nimi, tässä tapauksessa inssikone\sqlexpress. Tietokantayhteyden saa käytettävästä tietokoneesta riippumattomaksi kirjoittamalla Server Name kohtaan: localhost\sqlexpress.



**Kuva 28. Valikko**

Kun palvelin on valittu tai polku palvelimelle on kirjoitettu, siirrytään valitsemaan tietokanta, jonka kanssa aspx-sivut kommunikoivat. Valitaan pudotusvalikosta haluttu tietokanta, jonka jälkeen painetaan painiketta Test Connection, jolla varmistetaan tietokantayhteyden toimivan (Kuva 29.). Yhteyden testaamisen jälkeen valitaan OK. Näin on luotu yhteys Visual Studion ja SQL Server 2008:n välille.

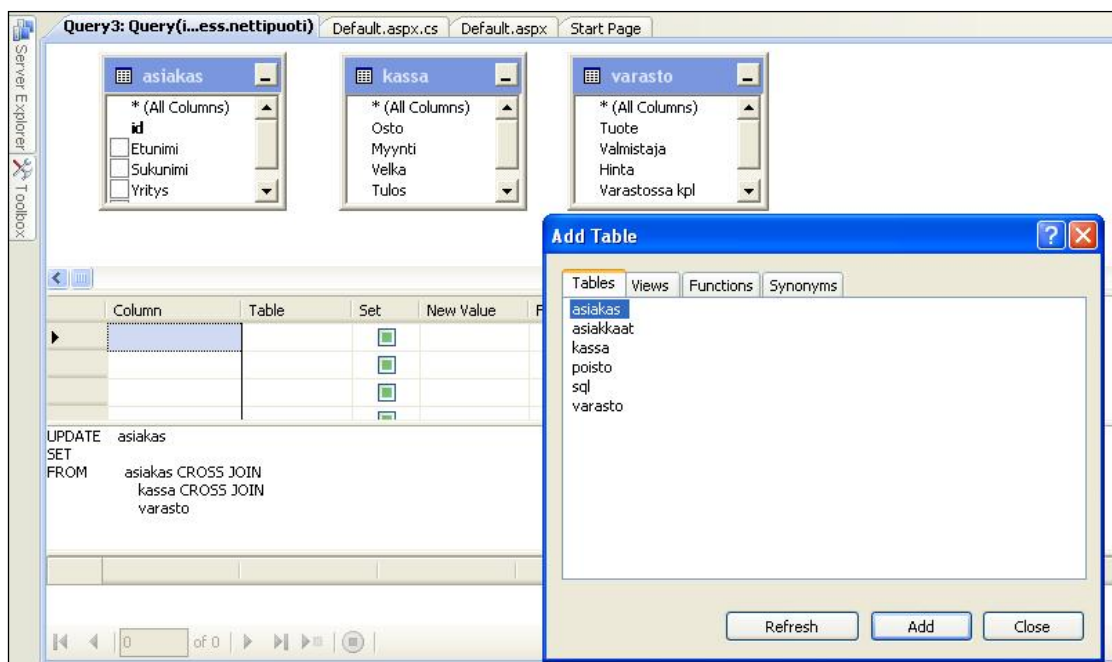


**Kuva 29. Testin tulos**

## 6.1.2 Visual Studio ja SQL Server

Kun tietokantayhteys on määritetty Visual Studion ja SQL Serverin välille, toimii Visual Studio työkaluna tietokannan hallintaan. Visual Studiossa voi tehdä miltei kaikkia samoja asioita kuin SQL Server Management Studiossa, paitsi yhtä tärkeää toimenpidettä Visual Studiossa ei voi tehdä ja se on tietokannan luominen. Tietokannan luomisen jälkeen voikin siis jättää Management Studion pois käytöstä ja tehdä kaiken ylläpidon yms. Visual Studion kautta. Graafinen käyttöliittymä on selkeä ja joissain asioissa parempi kuin SQL Server Management Studiossa.

Tietokantahaun tekeminen Visual Studion kautta on yksinkertaista, valitaan joko yksittäinen taulu tai sitten kansio Tables, joka sisältää kaikki tietokannan taulut ja klikataan hiiren oikealla napilla ja valitaan listasta New Query (Kuva 30.).



**Kuva 30. Haun näkymä**

Auennessa näkymässä valitaan halutut taulut mukaan hakuun, jonka jälkeen valitaan Close (Kuva X.). Hakuun valituista tauluista voi tämän jälkeen rasti ruutuunmenetelmällä rajata hakua mieleisekseen. Kuvassa SQL- lauseena on UPDATE käsky, sen saa vaihdettua haluamukseen klikkaamalla SQL-koodin yllä hiiren oikealla ja valitsemalla valikosta toisen vaihtoehdon. Kuvassa olevat taulut eivät liity toisiinsa, mutta jos liittyisivät, niiden välille olisi muodostunut viiva yhteyden merkiksi. Esimerkki hakuna voisi ajatella seuraavanlaisen haun, valitaan asiakas taulusta Yritys, kassa taulusta Osta ja varasto taulusta Tuote, jolloin haku tuloksena saadaan tietokannassa olevien yritysten ostamat tuotteet.

## **6.2 ASP-lomake tietojen hakemiseen ja muokkaamiseen**

Aloitetaan valmiin HTML-koodin muokkaaminen sillä, että kirjoitetaan <title> kohtaan esimerkiksi kuvaus web-sivusta. Tämä teksti näkyy selaimen kuvakkeessa Windowsin tehtäväpalkissa kuin myös web-sivun välilehdessä.

Siirrytään lähdekoodi näkymästä design näkymään valitsemalle se koodin alla olevasta palkista, jos käytössä on suuri näyttö niin valinta split näyttää puolet ruudusta koodia puolet graafista design-näkymää. Design näkymään tullessa yläreunassa vilkkuu tekstikursori, jonka yläpuolella lukee HTML-kielellä sen kohdan nimi, johon kirjoitettava teksti tulee koodissa. Klikataan pari kertaa kohdan alla, kunnes HTML-merkintä muuttuu muotoon div. Tämän jälkeen viedään hiiren kursori ruudun vasempaan reunaan Server Explorer- kohtaan ja näkyviin liukuvasta valikosta auki klikkailemalla valitaan asiakas-taulu. Raahataan asiakas-taulu tyhjälle HTML-sivulle ja pudotetaan se siihen. Ohjelma generoi siitä oletuksena Gridview-taulukon.

Taulukon alla oleva laatikko, nimeltään SqlDataSource1, pitää sisällään tiedon siitä, mihin yllä oleva taulukko viittaa. Tällä kertaa ohjelma generoi automaattisesti polun oikeaan tietokantaan, koska taulukko luotiin taulu suoraan tietokannasta työpöydälle raahaamalla. Muissa tilanteissa tietokantayhteys pitäisi itse määrittää. Toinen, mikä tapahtui automaattisesti, oli se, että GridView-taulukko viittaa automaattisesti oikeaan paikkaan eli SqlDataSource1 yhteyteen.



GridView –taulukko luo automaattisesti SELECT-lauseen koodisivulle, jolloin web-sivulle tulee tiedot automaattisesti ilman erillistä hakukäskyä. Koodin toimivuutta kokeillakseen mennään sivun yläpalkkiin ja sieltä valitaan Debug->Start Without Debugging tai vaihtoehtoisesti Ctrl+F5. Valinta Start Debugging on tarkoitettu virheellisen koodin tarkistamiseen ja virheen aiheuttajan etsimiseen. Oletuksena Start Debugging on poistettu käytöstä Visual Studio 2008:sta ja se pitää erikseen aktivoida, jos sitä haluaa käyttää.

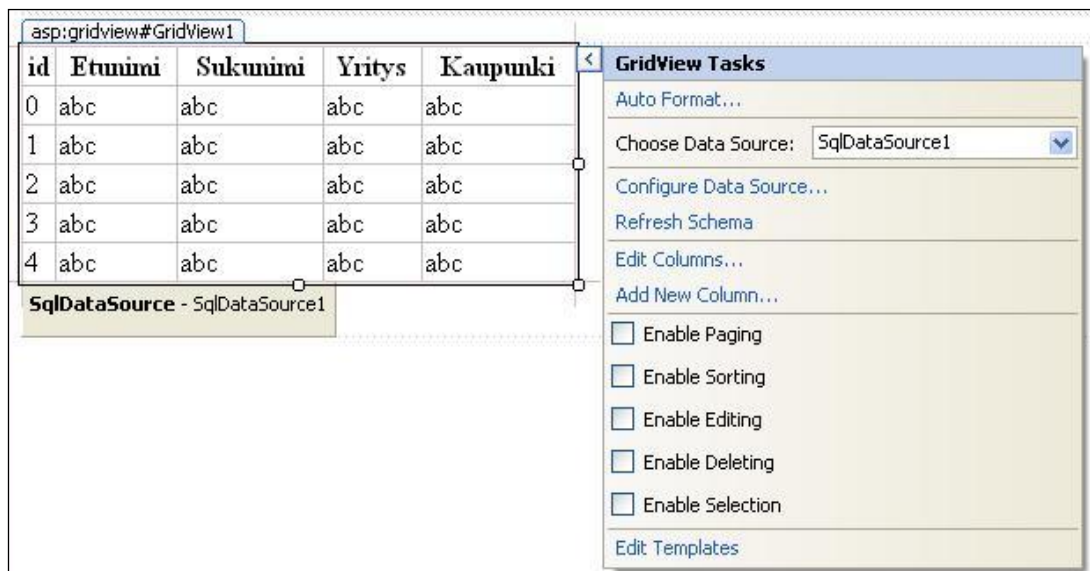
Ohjelman sivua kääntäessä ja käynnistäessä käynnistyy myös serveri, jolta sivu näkyy. Tämä ohjelma on nimeltään ASP.NET Development Server. Ohjelma valitsee jonkin vapaista porteista ja käyttää sitä toimiakseen. Tässä vaiheessa palomuuuri saattaa ilmoittaa toiminnasta portti numero n:ssä ja tämä toiminta kannattaa sallia, jotta juuri muokattua web-sivua päästään tarkastelemaan.

Web-sivu toimii ja HTML-koodin lisätty asp-koodi on hakenut SQL Serveriltä tiedot GridView-taulukkoon (Kuva 31.). Taulukon oletuskoko on liian pieni joten tiedot taulukossa menevät kahdelle eri riville.

id	Etunimi	Sukunimi	Yritys	Kaupunki
1	Matti	Meikäläinen	Metsurit Ky	Jurva
2	Simo	Sappi	Matkailu Oy	Lontoo
3	Rami	Russakka	Exterminator	Oulu
4	Petri	Parafini	Suksiveikot Ky	Nurmes
5	Olli	Kupponen		Mikkeli
6	Hilma	Hyppönen	Puhelinmyynti Ky	Espoo
7	Keijo	Karju	Maanviljelijä	Juva
8	Per	Stihl	Pienkone Oy	Loimaa
9	Leena	Lipasto	Compass Bank	Ontario
11	Elisa	Luuri	Känny Oy	Helsinki
12	Wihtori	Welho	Laihakaista Oy	Inari
13	Ida	Putkinen	R-kioski	Laukaa
14	Teppo	Testaaja	Testaus Oy	Testilä
16	Vallu	Vaahtopää	Saippukauppias	Imatra
18	Veijo	Varamies	Vuokrakuuni Oy	Vantaa

**Kuva 31. Gridview-taulukko**

Gridview-taulukon kokoa pystyy muokkaamaan helposti klikkaamalla taulukkoa, jolloin siihen generoituu reunat, joista taulukkoa voi venyttää haluamaansa kokoon. Kun taulu on tarpeeksi suuri, lisätään taulukkoon toimintoja. Avataan valikko GridViewn oikeasta yläkulmasta (Kuva 32.).



**Kuva 32. GridView-valikko**

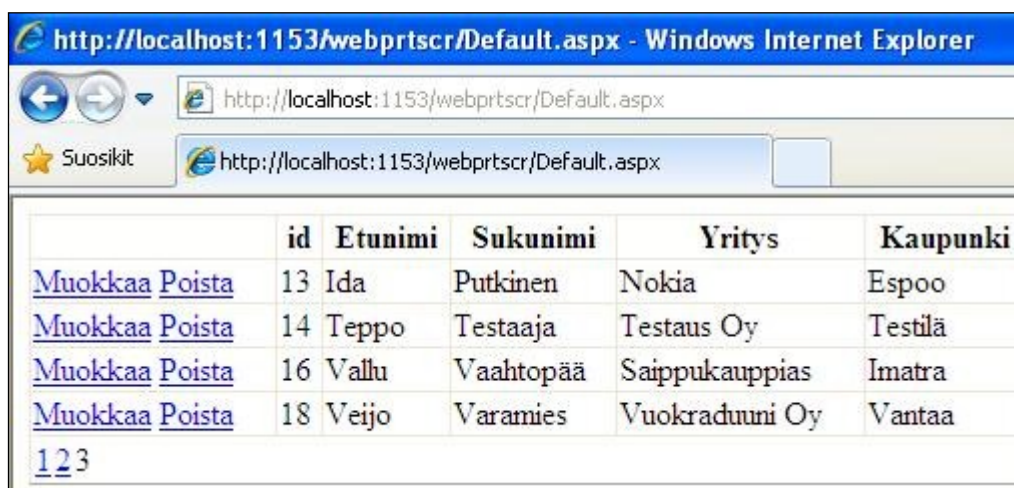
Valitaan valikosta kohdat Enable Paging, Editing ja Deleting, jolloin taulukon vasempaan reunaan generoituu Edit ja Delete-linkit sekä taulukon alareunaan numerot, jotka ilmaisevat monelleko sivulle tulokset on jaettu.

Enable Paging tarkoittaa sitä, että SELECT-lauseen tuloksia ei näytetä yhtenä pötkönä vaan tulokset voi jakaa useammalle sivulle. Paging toiminnon oletusarvona on 10 haku tulosta/sivu. Tätä arvoa voi muuttaa menemällä ruudun vasemmassa reunassa olevaan Properties-valikkoon. Valikosta Paging-kohdan alivalikosta löytyy kohta Page Size, jonka arvo on 10, muutetaan arvo viideksi.

Tämän jälkeen mennään Source näkymään, jossa nähdään, että ohjelma on generoinut valmiit SQL-lauseet sekä lisäksi tarvittavat ASP-koodit. Koska asiakas taulussa on id sarake asetettu Primary Key:ksi, ohjelma on osannut muodostaa SQL-lauseet viittaamaan oikeaan ”osoitteeseen”. Ilman id saraketta ohjelman luomista SQL-lauseista puuttuu kohdentava osa, jolloin ne eivät toimi ilman, että niitä joko muokkaa itse tai joissain tapauksissa ne eivät toimi ollenkaan ilma, että luo sarakkeen, jonka asettaa Primary Key:ksi.

## 6.2.1 Edit ja Delete-komentojen testaus

Käynnistetään web-sivu painamalla Ctrl+F5. Ensimmäinen asia, joka pistää silmään on se, että Visual Studiossa englannin kielellä olleet Edit ja Delete ovat nyt suomen kielellä web-sivulla (Kuva 33.).



	id	Etunimi	Sukunimi	Yritys	Kaupunki
<a href="#">Muokkaa</a> <a href="#">Poista</a>	13	Ida	Putkinen	Nokia	Espoo
<a href="#">Muokkaa</a> <a href="#">Poista</a>	14	Teppo	Testaaja	Testaus Oy	Testilä
<a href="#">Muokkaa</a> <a href="#">Poista</a>	16	Vallu	Vaahtopää	Saippukauppias	Imatra
<a href="#">Muokkaa</a> <a href="#">Poista</a>	18	Veijo	Varamies	Vuokradyuni Oy	Vantaa
<a href="#">1</a> <a href="#">2</a> <a href="#">3</a>					

**Kuva 33. Update/Muokkaa**

Testataan ensimmäiseksi toimiiko Paging, joten selataan kaikki kolme sivua läpi, sivujen pitäisi vaihtua moitteettomasti koska ohjelman generoima SQL-koodi oli virheetön. Seuraavaksi testataan toimiiko Poisto-linkki, valitaan toiselta sivulta poistettavaksi asiakas nimeltä Leena Lipasto. Klikataan Poista-linkkiä nimen vieressä ja kaikki tiedot kyseisestä henkilöstä on poistettu tietokannasta. Tämän voi vielä varmistaa käymällä läpi kaikki kolme sivua.

Seuraavaksi testataan toimiiko Muokkaa-toiminto. Valitaan kolmannelta sivulta henkilö Ida Putkinen ja klikataan kohtaa Muokkaa. Sivun muuttua ulkoasuun ja kaikkien neljän henkilötiedon ympärille on tullut laatikot sekä Muokkaa teksti on muuttunut Päivitä tekstiksi (Kuva 34.).

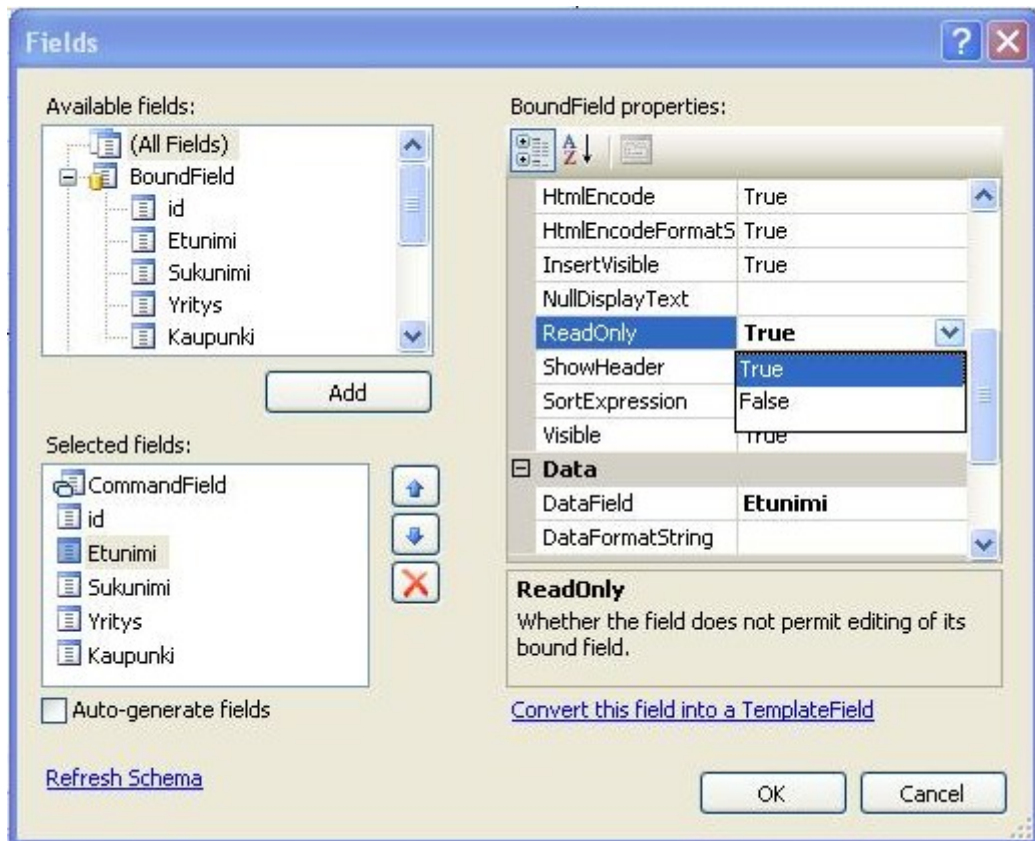
	id	Etunimi	Sukunimi	Yritys	Kaupunki
<a href="#">Päivitä</a> <a href="#">Peruuta</a>	13	Ida	Putkinen	R-kioski	Laukaa
<a href="#">Muokkaa</a> <a href="#">Poista</a>	14	Teppo	Testaaja	Testaus Oy	Testlä
<a href="#">Muokkaa</a> <a href="#">Poista</a>	16	Valtu	Vaahtopää	Saippukauppias	Imatra
<a href="#">Muokkaa</a> <a href="#">Poista</a>	18	Veijo	Varamies	Vuokrakuuni Oy	Vantaa

**Kuva 34. Muokkaus näkymä**

Muutetaan tiedot kohdissa Yritys ja Kaupunki. Tämän jälkeen klikataan kohtaa Päivitä. Päivitä käsky lähettää tiedot serverille, jossa tiedot käsitellään sekunninmurto-osassa, jonka jälkeen ne lähetetään takaisin web-sivulle. Tämä tietojen päivitys ei siis tapahtunut jossain selainohjelman sisällä vaan päivityskäsky lähetettiin suoraan serverille, jossa tieto käsiteltiin ja sen jälkeen päivitetty tieto lähetettiin takaisin web-sivulle.

Päivityskäskyä voi myös muokata siten, että se koskee ainoastaan haluttuja sarakkeita. Tämän muokkauksen pystyy tekemään valitsemalla Edit Columns, Gridview-valikossa (Kuva 33.). Auenneessa valikossa näkyy ylimmäisenä vasemmalla kaikki ne kentät, joita pystyy muokkaaman ja sen alla on kohdentava Selected Fields-valikko. Klikataan esimerkiksi Etunimi-kenttää Selected Fields-valikossa, jolloin ikkunan oikeaan reunaan ilmestyy kyseisen kentän ominaisuudet näyttävä valikko. Valinnalla Read Only = True estetään kentän Etunimi muokkaaminen, koska sille on nyt ainoastaan luku-oikeus.

Valikossa on myös kenttä NullDisplayText, johon voi kirjoittaa tekstin joka generoituu jokaiseen Null-arvon omaavaan kenttään. Tähän kenttään kirjoittamalla esimerkiksi Tyhjä, saadaan taulukko näyttämään visuaalisesti paremmalta, kun siellä ei ole tyhjiä aukkoja välissä (Saman voi tehdä myös Gridviewssä).



**Kuva 35. Kenttien muokkaus valikko**

## 6.2.2 Tietojen lisääminen tietokantaan

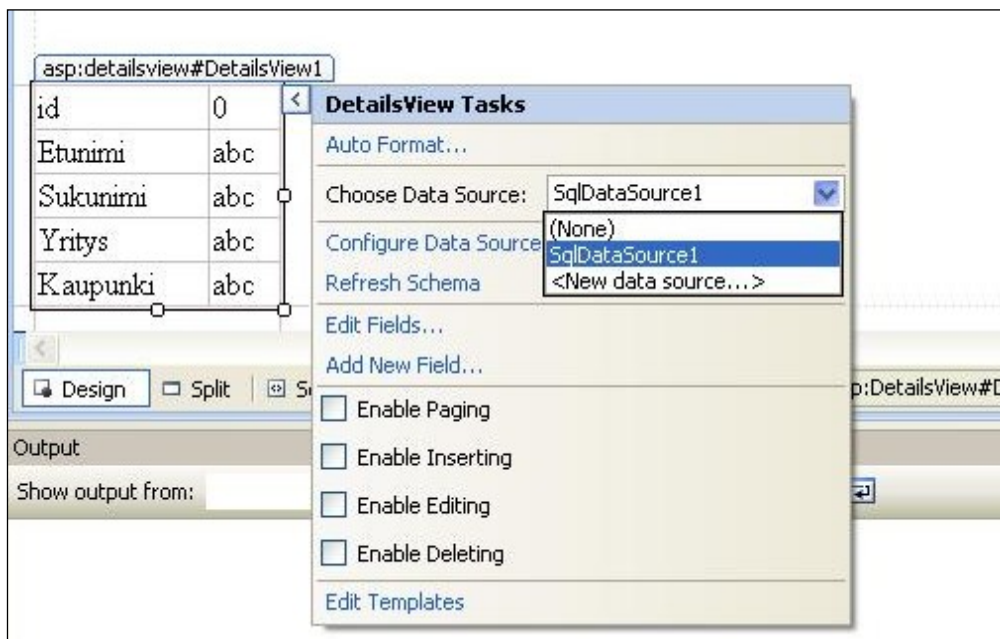
GridView on mainio työkalu tietojen hakemiseen, muokkaamiseen, poistamiseen jne. mutta jostain syystä Microsoft ei ole nähnyt tarpeelliseksi tehdä GridViewstä työkalua, jolla voisi myös lisätä tietoja. GridView ei siis tue insert-toimintoa, mutta internetistä etsimällä löytää helposti ohjeet, joilla muokata jokin GridViewn toimintoista toimimaan insert-toimintona. Tällä tavalla toimittaessa kannattaa miettiä tarkkaan, minkä toiminnon uhraa insertiksi, koska esimerkiksi muuttamalla delete-toiminnon insertiksi, ei delete-toimintoa enää ole, joten tietojen poistamisen voi siis unohtaa. Annetaan GridViewn olla alkuperäisessä muodossaan ja käytetään toista metodia tietojen lisäämiseksi.

Avataan Toolbox ruudun vasemmasta reunasta ja valitaan Data-valikosta DetailsView ja raahataan se GridViewn ja SqlDataSourceen alapuolelle. Detailsviewtä käytetään yleisesti juuri GridViewn apuna ja DetailsViewssä näytetään yhden rivin kaikki tiedot. Ruudulla näkyy ainoastaan kolme Column-kenttää ja niiden data-kentät.

Column0	abc
Column1	abc
Column2	abc

**Kuva 36. DetailsView**

Viedään hiiren osoitin taulukon päälle ja klikataan vasemmassa yläreunassa olevasta nuolesta valikko auki. Kohdassa Choose Data Source lukee (None), jolloin taulukko ei ole yhteydessä mihinkään tietokantaan. Valitaan pudotusvalikosta SqlDataSource1, joka on sama, jota GridView käyttää. Heti tämän valittua muuttuu DetailsView taulukkoon Column sanojen tilalle asiakas-tietokannan sarakkeiden nimet (Kuva 36.).



**Kuva 37. DetailsView valikko**

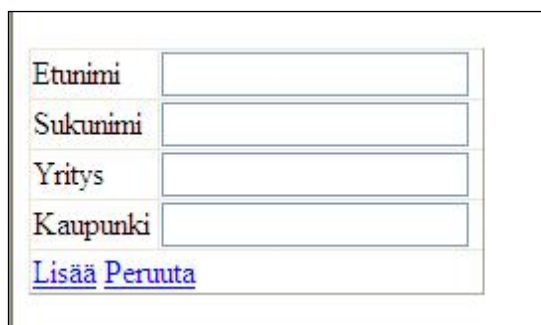
DetailsViewn valikossa on valittavissa miltei samat toiminnot kuin GridViewssäkin, ainoat erot ovat GridViewssä olevat Sorting ja Selection-valinnat, jotka puuttuvat DetailsViewstä sekä DetailsViewssä oleva Inserting-valinta, joka puuttuu GridViewstä. Riippuen web-sivun tarpeista saattaa pärjätä pelkällä GridViewllä tai DetailsViewllä, mutta nämä kaksi työkalua täydentävät loistavasti toisiaan.

Valitaan Enable Inserting, jolloin taulukon alareunaan generoituu linkki New. Seuraavaksi mennään tuttuun Edit Fields-valikkoon. Tarkoituksena on poistaa id-kenttä taulukosta, koska siihen kenttään SQL Server automaattisesti generoi luvun. Valitaan id-kenttä selected Fields-valikosta ja sen jälkeen poistetaan se klikkaamalla punaista rastia. Tämän jälkeen valitaan Ok, jolloin id-kenttä häviää taulukosta.

### 6.2.3 Tietojen lisäämisen testaus

Vedetään taulukkoa leveämmäksi, noin 230 pikseliä leveäksi, jolloin saadaan taulukko näyttämään symmetrisemmältä ja tekstit mahtuu kenttiin paremmin. Tekstien menemisen useammalle riville voi estää muuttamalla Edit Fields-valikossa kohdan Wrap=False, tällöin teksti tulee yhdelle riville ainoastaan. Tämän jälkeen käynnistetään sivu painamalla Ctrl+F5.

DetailsView-tilaus näyttää tietokannassa ensimmäisenä olevat tiedot (id1). Lisätään tietoja klikkaamalla linkkiä Uusi, joka on muuttunut suomeksi Visual Studion sanasta New. Sivun vaihtuu uudelleen ainoastaan DetailsViewn kohdalla, johon on generoitu tyhjät kentät tietojen lisäämiseen (Kuva 38.).



Etunimi	<input type="text"/>
Sukunimi	<input type="text"/>
Yritys	<input type="text"/>
Kaupunki	<input type="text"/>
<a href="#">Lisää</a> <a href="#">Peruuta</a>	

**Kuva 38. Update näkymä**

Taulukon alareunaan on tullut valinnat Lisää ja Peruuta. Testataan ensin toiminnon peruuttamista ja klikataan valintaa Peruuta. Sivun palaa takaisin aikaisempaan näkymään. Valitaan uudestaan Lisää ja aloitetaan lisäämään tietoja tyhjiin kenttiin. Lisätään tiedot kenttiin Etunimi, Sukunimi ja Kaupunki, jätetään Yritys-kenttä tyhjäksi jotta näimme sallisiko SQL Serveri tyhjän kentän. Serveriin on annettu Allow Nulls-ehto, joten sen pitäisi sallia tyhjät kentät. Klikataan Lisää linkkiä.



Sivu muuttuu taas alkuperäisen näköiseksi, DetailsView näyttää Matti Meikäläistä, mutta GridViewn kolmannelle sivulle on ilmestynyt juuri lisätyt tiedot (Kuva 39).

	id	Etunimi	Sukunimi	Yritys	Kaupunki
<a href="#">Muokkaa</a> <a href="#">Poista</a>	13	Ida	Putkinen	Nokia	Espoo
<a href="#">Muokkaa</a> <a href="#">Poista</a>	14	Teppo	Testaaja	Testaus Oy	Testilä
<a href="#">Muokkaa</a> <a href="#">Poista</a>	16	Vallu	Vaahtopää	Saippukauppias	Imatra
<a href="#">Muokkaa</a> <a href="#">Poista</a>	18	Veijo	Varamies	Vuokrakuuni Oy	Vantaa
<a href="#">Muokkaa</a> <a href="#">Poista</a>	19	Sami	Sorsa	Tyhjä	Rovaniemi
<a href="#">1</a> <a href="#">2</a> <a href="#">3</a>					

Etunimi	Matti
Sukunimi	Meikäläinen
Yritys	Metsurit Ky
Kaupunki	Jurva
<a href="#">Uusi</a>	

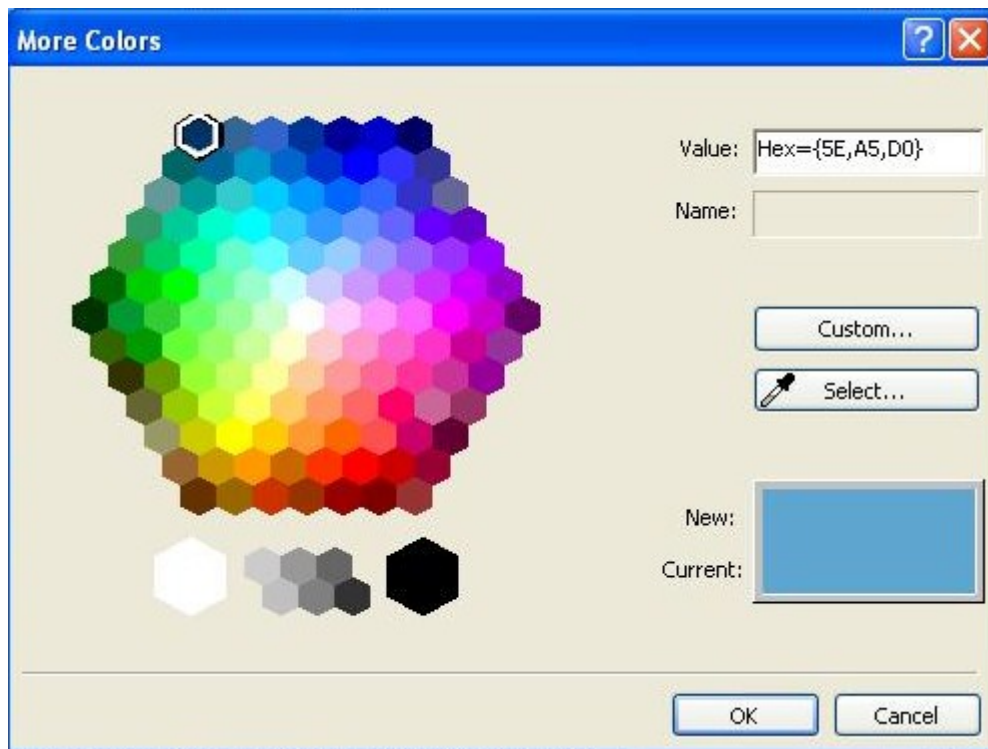
**Kuva 39. Web näkymä**

Tyhjäksi jätetty Yritys-kenttä ei siis kaatanut tiedon lisäystä ja lisäksi GridViewn lisäty NullDisplayText=Tyhjä toimii.

### 6.3 Ulkoasun muokkaus

Pelkkä koodin toimivuus ei aina riitä vaan visuaalinen puoli pitää myös muistaa. Seuraavaksi tehdään pieniä muutoksia ulkoasuun kuin myös taulukoiden toimintaan.

Aloitetaan ulkoasun muokkaus vaihtamalla tausta tutun siniseksi. Avataan Properties-valikko ruudun oikeasta reunasta ja sen yläreunasta alavetovalikosta DOCUMENT. Tämän valikon puolella välissä on kohta BgColor ja sen oikealla puolella olevaa napia klikkaamalla aukeaa ikkuna, jossa voi määrittää haluamansa värin monella eri tavalla (Kuva DFDG). Kirjoitetaan Value kohtaan Hex-arvot 5EA5D0 ja valitaan ok. Näin tausta on saanut uuden värin.



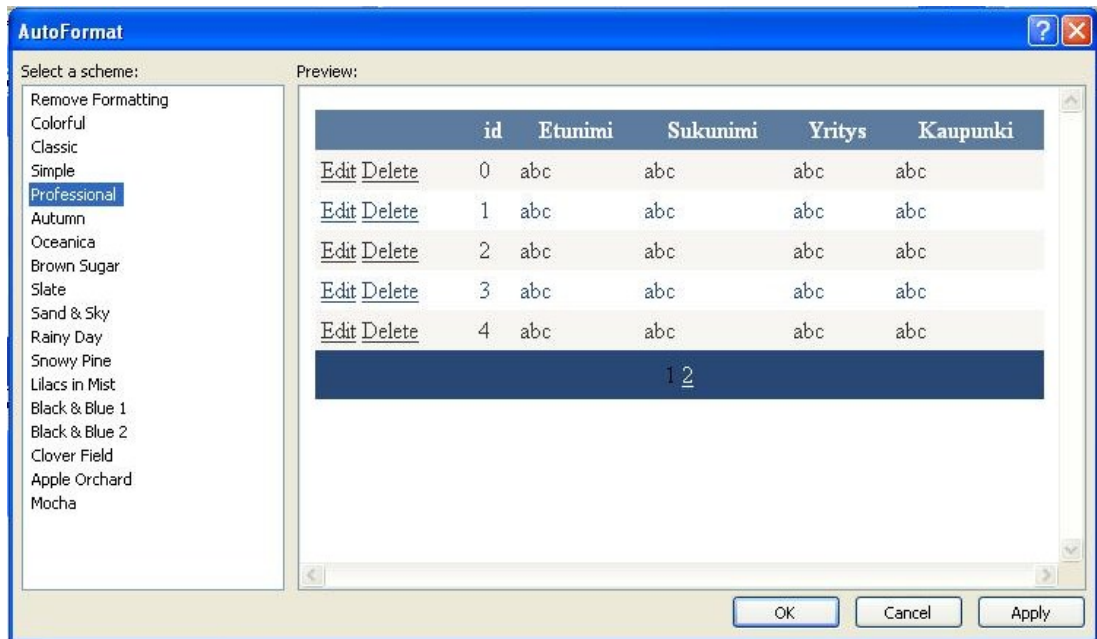
**Kuva 40. Värin valinta**

Tämän saman voi tehdä myös Source-näkymässä suoraan HTML-koodian kirjoittamalla. Taustavärin lisääminen tehdään HTML-koodin body-osiossa lisäämällä seuraavanlainen koodi:

```
<BODY bgcolor=""#808000">
```

Tässä koodissa heksadesimaali arvo 808000 on oliivinvihreä.

Nyt kun tausta on saatu oikean väriseksi, vaihdetaan taulukot sattuvamman väriseksi. Tämän voi tehdä valitsemalla jonkin useasta valmiista värityylistä. Aloitetaan Grid-Vieswstä ja klikataan taulukon reunassa olevasta nuolesta jolloin päästään taulukon valikkoon. Valikosta valitaan ylimpänä listassa oleva Auto Format, josta aukeaa uusi valinta ikkuna (Kuva 41).



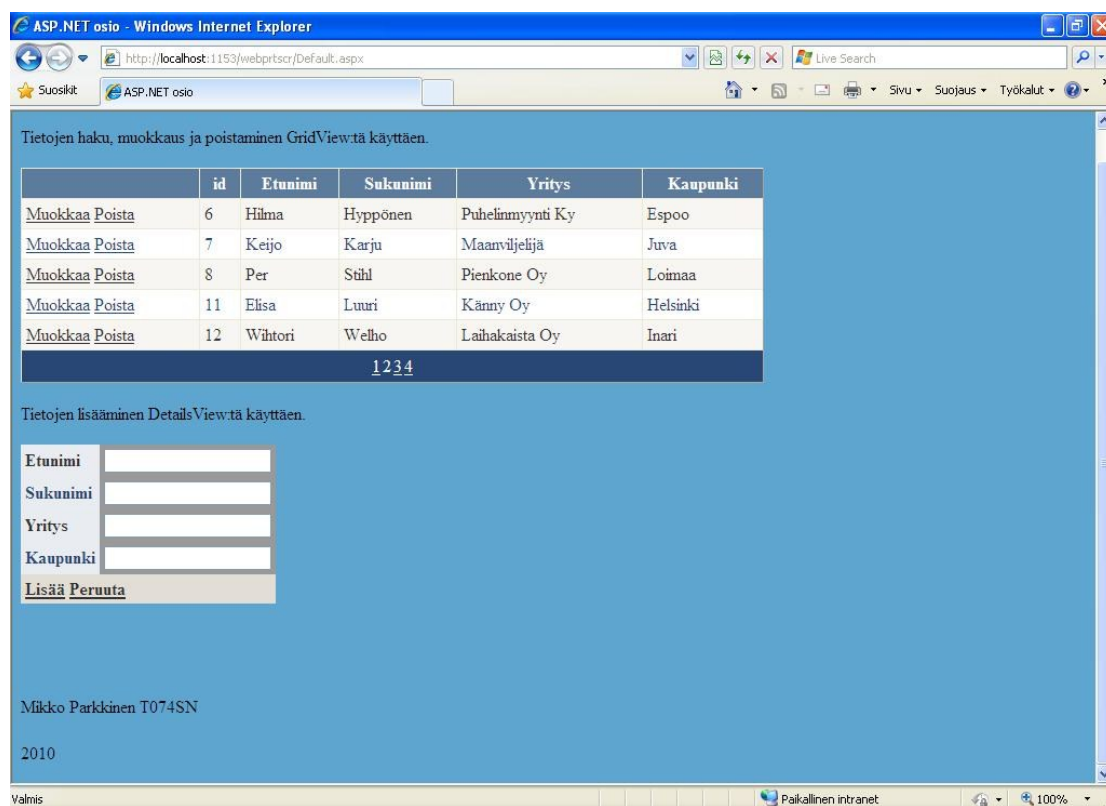
**Kuva 41. Värimaailma**

Ikkunan vasemmasta reunasta löytyviä väri teemoja klikkaamalla saa esikatsella oikean värityylin löytämiseksi. Eri värityyleissä on myös monesti erilaiset fontit ja fonttikoot, jos mieleinen väri löytyy mutta fontti ei miellytä, niin fonttia voi muuttaa (kuten muitakin ominaisuuksia) Properties-valikosta. Värityylin saa myös poistettua valitsemalla listan ylimpänä olevan Remove Formatting. Valitaan värityyliksi Professional ja tehdään sama muutos myös DetailsView-taulukolle.

Seuraavaksi muutetaan DetailsViewn ulkoasua siten, että se ei näytä tietokannassa ensimmäisenä olevaa tietoa vaan se on pelkästään tietojen lisäämistä varten ja näyttää aina samalta kuten kuvassa 38.

Avataan Properties-valikko ruudun oikeasta reunasta ja valitaan se näyttämään DetailsViewn tietoja. Etsitään kohta Behavior ja sieltä DefaultMode. DefaultMode on oletuksena ReadOnly-tilassa, jolloin DetailView toimii kuten aikaisemmin on todettu. Eri tila vaihtoehtoina on ReadOnly lisäksi Edit ja Insert. Valitaan Insert, jolloin DetailsView-taulukosta tulee pelkästään tietojen lisäämiseen dedikoitunut taulukko. Tämän valinnan jälkeen taulukko ei näytä tietokannassa olevia tietoja ollenkaan, eikä siinä ole muita linkkejä kuin Lisää ja Peruuta. Tietojen kirjoittamisen jälkeen, kun painaa Lisää-linkkiä, tiedot lisätään tietokantaan ja DetailsView-taulukko tyhjentää kentiin kirjoitetut tekstit.

Tämän jälkeen lisätään taulukoitten ylle niiden toimintoja selostavat tekstit ja ajetaan sivun kääntäjästä läpi viimeisen kerran painamalla Ctrl+F5. Valmis ASP-sivu SQL-tietokannan käyttöä varten (Kuva 42).



**Kuva 42. Valmis sivu**

## 7 JOHTOPÄÄTÖKSIÄ

Tarkoitukseni oli tässä työssä näyttää, kuinka luoda nopeasti tietokantapalvelin ja sen sisällön muokkaamista varten ASP-sivu. Työni on kirjoitettu ohjeenomaiseksi, jotta työ olisi helppo toistaa henkilöiden, joilla on jo hieman tietämystä tietokoneista. Tämän työn toivoisi antavan kipinän aloittaa SQL- ja ASP-ohjelmointi kenelle tahansa.

Ennen tietokantojen luominen ja muokkaaminen vaati paljon aikaa ja suurta kooditietämystä, nykytyökaluilla jo pienelläkin tietämyksellä voi aloittaa tietokantojen rakentamisen. SQL Server 2008 on tehty siten, että ohjelmaa käyttäessä oppii paljon ja jos on aikaisemmin tietokantoja koodilla rakentanut, niin Server 2008:ssa tulee vastaan paljon tuttuja asioita myös eri tavalla tehtynä.

SQL-osiota tehdessä tuli vastaan myös ongelma, jonka osasi ratkaista teoriassa, mutta itse ohjelmasta ei tuntunut löytyvän oikeaa valikkoa ongelman ratkaisuun. Ongelma vaikutti myös ASP.NET osioon, jonka tekemistä ongelma jarrutti. Ongelman ratkettua työ saatiin nopeasti valmiiksi.

ASP-osiossa tarkoitukseni oli painottaa käsin koodaamisen vähäisyyttä. Käsin koodaaminen on yksi suurimmista syistä, miksi ihmisiä ei koodaaminen kiinnosta ja miksi monet arastelevat aloittaa koodaamisen. Tietenkään ilman käsin koodaamista ei nykyään pärjää eikä käsin koodaamista voida tulla koskaan pois jättämään. Moni asia, jonka hetkessä kirjoittaisi käsin lähdekoodiin, vie pitkän aikaa etsiä Visual Studion eri valikoista, mutta henkilö, joka ei ole tottunut käsin koodia kirjoittamaan etsii hetkessä haluamansa valikoista.

Itselläni mielenkiinto SQL- ja ASP-kieleen kasvoi tätä työtä tehdessä ja huomasin, kuinka suuri ero on aikaisemmin käyttämieni ilmaisten ja nyt käyttämieni maksullisten ohjelmointityökalujen välillä.

## LÄHTEET

Chapple, Mike 2008: Microsoft® SQL Server® 2008 For Dummies®. Wiley Publishing, Indianapolis, Indiana USA

Dewson, Robin 2008: Beginning SQL Server 2008 for Developers. From Novice to Professional. Apress, USA

What's New. Microsoft Corporation. WWW-Dokumentti.  
<http://www.microsoft.com/sqlserver/en/us/product-info/whats-new.aspx>  
Päivitetty 9.3.2011 Luettu 1.12.2010

SQL aloittelijoille. Mureakuha. WWW-Dokumentti.  
[http://wiki.mureakuha.com/wiki/SQL\\_aloittelijoille](http://wiki.mureakuha.com/wiki/SQL_aloittelijoille)  
Päivitetty 11.4.2007. Luettu 1.12.2010

What is ASP.NET. 2005. O'Reilly Media, Inc. WWW-Dokumentti.  
<http://ondotnet.com/pub/a/dotnet/2005/09/19/what-is-asp-net.html>  
Päivitetty 23.3.2011, Luettu 1.12.2010



**Liite 1.**  
**Yksisivuinen liite**