
Lab 2: ASP.NET 2.0 Data Access

Estimated time to
complete this lab:
60 minutes

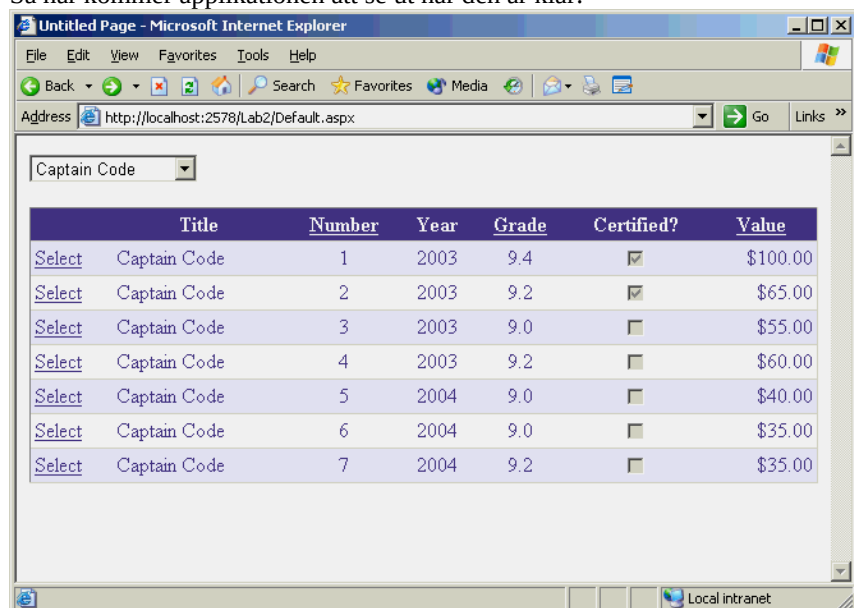
Access till bakomliggande databaser och andra data objekt är en viktig del av databaserade webbapplikationer. Med ASP.NET 2.0 blir det ännu enklare än tidigare med hjälp av nya datakontroller så som "GridView" och "DetailsView" för att rendera data i HTML, och datakällkontroller så som "SqlDataSource" och "ObjectDataSource" för att deklarativt binda datakontroller till datakällor. Tack vare dessa kontroller har dataaccess som tidigare krävde hundratals rader kod i ASP.NET 1.x blivit möjligt att utföra med ett minimum av kod eller helt utan kod i ASP.NET 2.0.

Innan du börjar med labb 2 är det lämpligt att ha läst kapitlen 1 till 6 och 11 i boken.

I den här labben skall du skapa en webb site med namnet MyComics som är en virtuell katalog för serieböcker. Även i denna labb används en demodatabas från Microsoft kallad MyComics. Du skall bygga en applikation där man skall kunna bläddra ibland serieböcker i en "GridView". Du skall även lägga till en sida detaljer för en seriebok i en "DetailsView". Slutligen skall du lägga till en administrations sida där man kan lägga till, uppdatera och radera böcker. Du kommer att lära dig mer om "SqlDataSource" och "ObjectDataSource" kontroller och mer om editerings möjligheterna med "GridView" och "DetailsView".

När dina sidor är färdiga skall du lägga till caching för att optimera prestanda genom att minimera databas I/O. Slutligen skall du använda SQL cache beroenden för att för att säkerställa att data alltid är korrekt även om underliggande data förändras.

Så här kommer applikationen att se ut när den är klar:



Övning 1

Verifiera att databasen MyComics finns

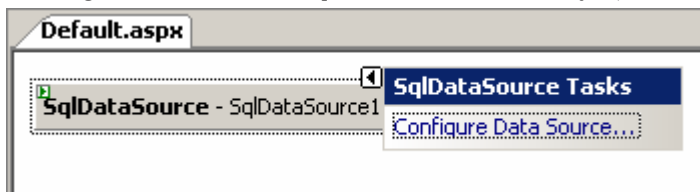
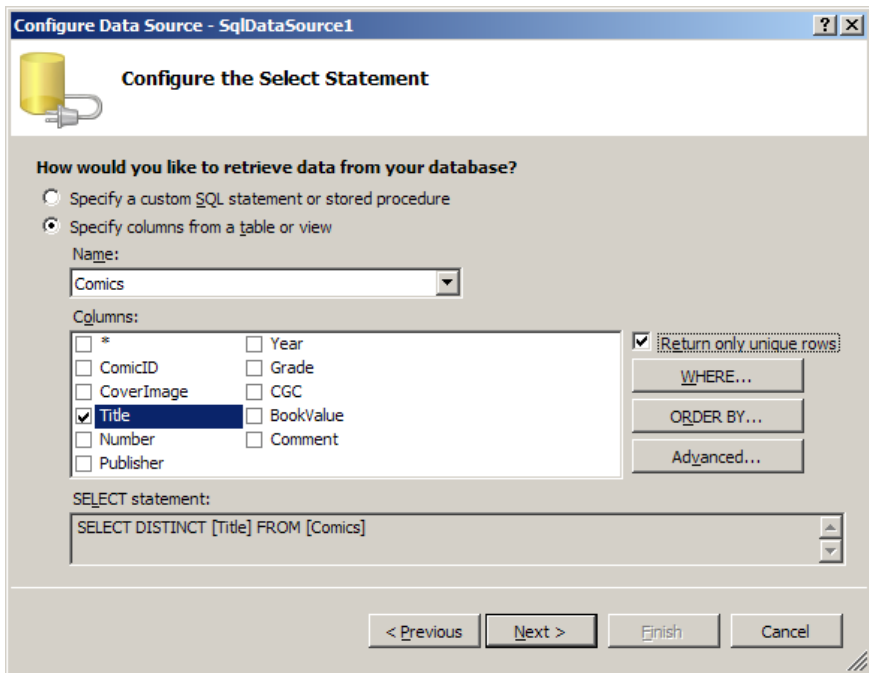
Du skall i denna övning verifiera att SQL Server databasen med namn MyComics finns och kan och kan tjäna som applikationens primära datakälla.

uppgift	Detaljerad beskrivning
1. Installera databasen	a. Hämta databasfilen MyComics.sql från http://apachepersonal.miun.se/~parfor/DTAB89 och spara de lokalt. b. Installera den på din dator med hjälp av SQL Server 2005 Management Studio Express på samma sätt som i föregående laboration.
2. Anslut till databasen	c. Starta Visual Studio 2005. d. Högerklicka på "Data Connections" i Server Explorer fönstret och välj "Add Connection" (OBS! du skall inte skapa någon ny site ännu). <i>Om Server Explorer fönstret inte är synligt välj View Server Explorer.</i> <i>Om dialogen "Change Data Source" visas, välj Microsoft SQL Server och klicka OK, då bör "Add Connection" dialogen visas.</i> e. I "Add Connection" dialogen, kontrollera så att "Data source" är Microsoft SQL Server (SqlClient). Om en annan datakälla är vald klicka på Change knappen för att byta datakälla. f. I dialogen "Add Connection" ange anslutningsegenskaperna enligt nedan: Server Name: localhost eller DATORNAMN\SQLEXPRESS Use Windows Authentication Välj eller skriv databasnamnet: MyComics g. Klicka på Test Connection h. När du har verifierat att du kan ansluta till databasen, klicka på OK för att stänga dialogen. i. Klicka på plustecknet framför "MyComics" i Server Explorer (i Visual Studio). j. Expandera noden "Tables" och verifiera att det finns en tabell "Comics".

Övning 2

Använd SqlDataSource för att hämta data till en DropDownList

I denna övning skall du skapa en ny ASP.NET webbplats med Visual Studio. Sedan skall du lägga till en "DropDownList" kontroll till sedan och använda "SqlDataSource" för att fylla den med data som ett resultat av en databasfråga.

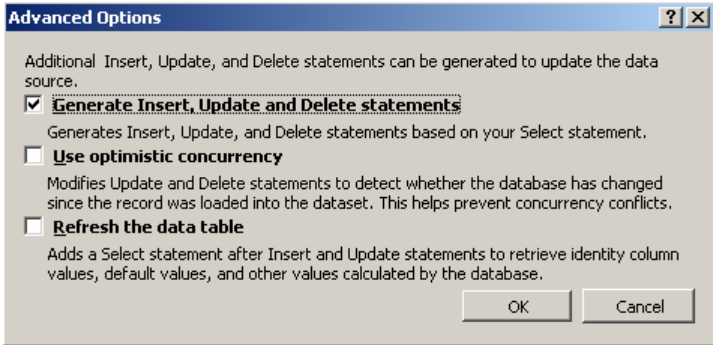
Uppgift	Detaljerad beskrivning
1. Skapa en ny webbplats	a. Starta Microsoft Visual Studio om du inte redan har den igång. b. Välj "New Web Site" i Visual Studio-s File meny. c. I dialogen "New Web Site" välj "Visual C#" som språk och "ASP.NET Web Site" som template. Navigera till eller skriv in sökvägen till den plats där du tänker placera webbplatsen ex. "C:\nnn\...\Lab2" i "Location" fältet och klicka OK för att skapa webbplatsen.
2. Lägg till en SqlDataSource	a. Klicka på Design. b. Dra en "SqlDataSource" kontroll från verktygsfältet till sidan. c. Klicka på "Configure Data Source" i "SqlDataSource Tasks" menyn (se nedan).  d. När dialogen "Configure Data Source" dyker upp, klicka på "New Connection" knappen. e. I dialogen "Add Connection" försäkra dig om att datakällan (Data source) är "Microsoft SQL Server (SqlClient)", skriv därefter in eller välj antingen "localhost" eller "DATORNAMN\SQLSERVEREXPRES" i rutan "Server name" och välj "Use Windows Authentication.". Välj sedan "MyComics" under "Select or enter a database name" och klicka OK. f. Kontrollera så att din nya anslutning är vald i fältet "What data connection should your application use to connect to the database?", klicka Next. g. När du får frågan om anslutningssträngen skall sparas i applikationens konfigurationsfil, svara "yes" och kontrollera att "MyComicsConnectionString" är anslutningssträngens namn. Klicka Next för att fortsätta. h. I följande dialog, kryssa i "Title" i kolumn valet. Kryssa även i "Return only unique rows," så som visas nedan. Klicka Next. 

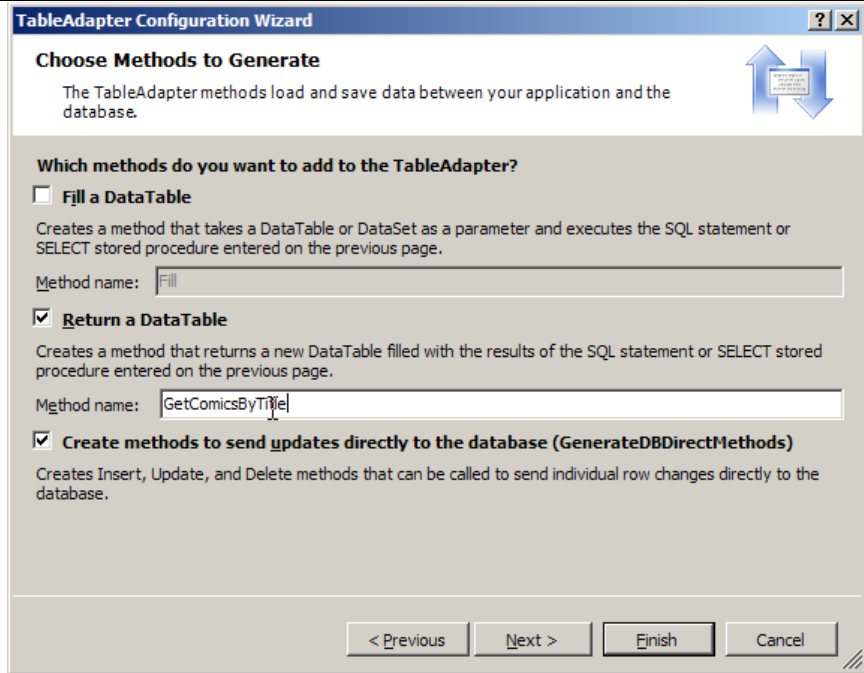
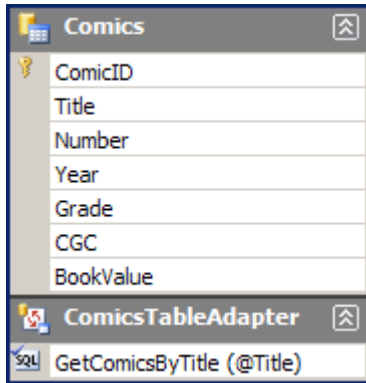
	i. Klicka Finish för att slutföra konfigurationen av "SqlDataSource". j. Titta i Solution Explorer fönstret i Visual Studio så skall det nu finnas en Web.config fil har lagts till. Dubbelklicka på Web.config för att öppna den. Titta vad som finns under <connectionStrings> sektionen? k. Stäng Web.config. l. Byt till Source för att se koden för Default.aspx. Leta efter <asp:SqlDataSource> taggen och titta på dess "ConnectionString" attribut. Vad står det på höger sida om likhetstecknet?
3. Lägg till en DropDownList	a. Byt till Design vyn igen och dra en DropDownList kontroll till sidan. b. Klicka på "Choose Data Source" i "DropDownList Tasks" menyn. c. Välj "SqlDataSource1" i listan över datakällor och klicka OK. d. Kryssa i rutan "Enable AutoPostBack" i "DropDownList Tasks" menyn. e. Välj "Start Without Debugging" i Visual Studio-s Debug meny (eller tryck Ctrl+F5) för att starta Default.aspx i webbläsaren. Kontrollera så att dropdown listen har ett innehåll enligt nedan.  f. Stäng webbläsaren och återvänd till Visual Studio.

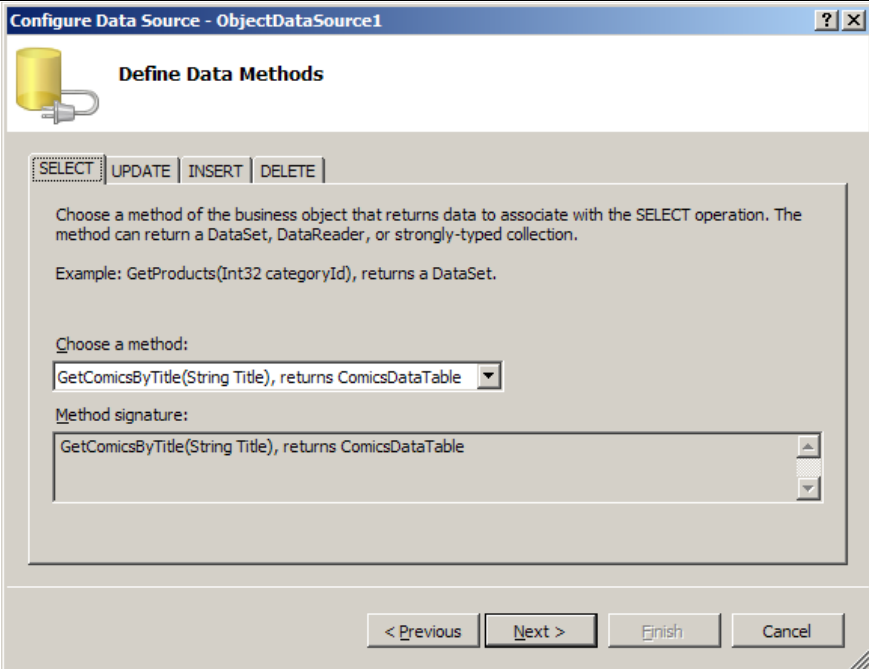
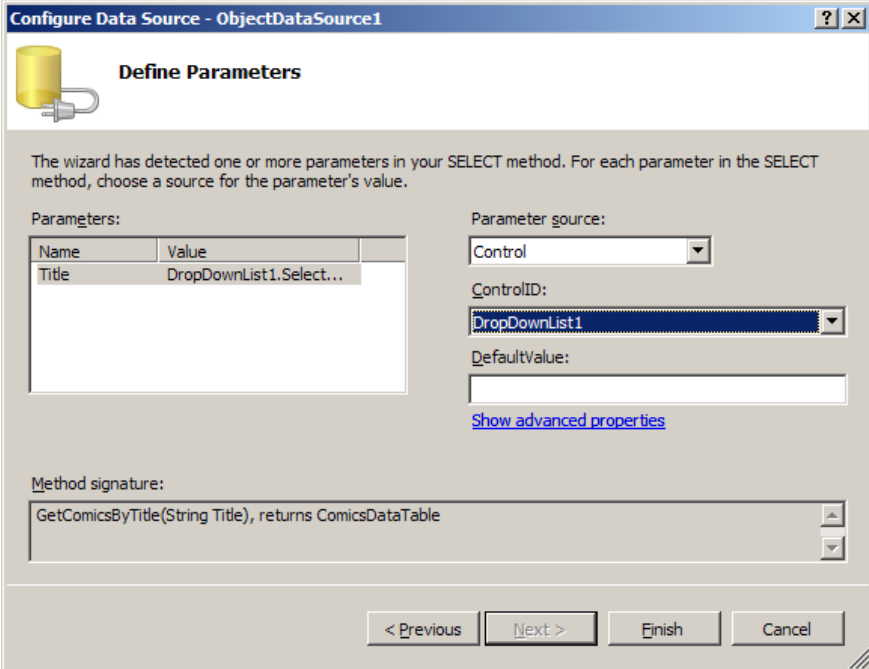
Övning 3

Använd ObjectDataSource för att fylla en GridView

I den här övningen skall du lägga till en "GridView" kontroll på sidan. Den här gången skall du inte använda en "SqlDataSource" för att fylla "GridView"-n med data, istället skall du skapa en datakomponent som interagerar med databasen för att binda data till "GridView"-n med en "ObjectDataSource" komponent. I tillägg skall du skapa parametrar för "ObjectDataSource" så att den skickar den title som valts i en "DropDownList" till datakomponenten för användning i en WHERE satts.

Tasks	Detailed Steps
1. Lägg till en datakomponent	a. Lägg till en mapp med namnet "App_Code" genom att högerklicka på C:\..\Lab2 i Solution Explorer och välj "Add ASP.NET Folder ->App_Code." b. Högerklicka på mappen "App_Code" och välj "Add New Item." c. I efterföljande dialog välj "DataSet" som "template type" och skriv "MyComics.xsd" som filnamn. Klicka på "Add". Efter en kort paus öppnas dialogen "TableAdapter Configuration Wizard". d. Om "MyComicsConnectionString (Web.config)" dyker upp i listan, välj den. Annars, klicka på knappen "New Connection" och skapa en ny "MyComics" anslutning och välj sedan den under "Select or enter a database name." Klicka Next. e. På sidan "Choose a Command Type", välj "Use SQL statements." Klicka Next. f. I dialogen "Generate the SQL Statements", skriv "SELECT ComicID, Title, Number, Year, Grade, CGC, BookValue FROM Comics WHERE Title=@Title" i fältet med namn "What data should be loaded into the table?" Klicka inte på Next ännu. g. Klicka på knappen "Advanced Options". Kryssa i "Generate Insert, Update, and Delete statements" i den efterföljande dialogen. Kontrollera så att de andra två rutorna inte är ikryssade enligt nedan. Klicka OK följt av Next.  h. I sedan "Choose Methods to Generate" se till så att inte "Fill a DataTable" är ikryssad. Under sektionen "DataTable" ändra metodnamn till "GetComicsByTitle" enligt nedan. Guiden kommer nu inkludera en metod för SELECT i datakomponenten (som använder frågan du skapade tidigare för att fylla en "DataTable") och även metoder för INSERTs, UPDATEs, och DELETE.

	 i. Klicka Next, följt av Finish. "TableAdapter Configuration Wizard" skriver inställningarna du just valt till "MyComics.xsd". Filen "MyComics.xsd" innehåller ett XML schema som beskriver en datakomponent. Under körning autokompilerar ASP.NET starkt typad data från XSD filer. Du ser inte komponenterna eftersom de inte är en del av projektet men du ser en bild av innehållet I XSD-filen I Visual Studio designer, enligt nedan.  j. Välj "Save All" i Visual Studio-s File meny för att spara ändringar.
2. Lägg till en ObjectDataSource kontroll	a. Återvänd till Default.aspx i design och dra en "ObjectDataSource" kontroll till sidan. b. Klicka "Configure Data Source" i "ObjectDataSource Tasks" menyn. c. Välj "MyComicsTableAdapters.ComicsTableAdapter" (namnet på den component som autokompilerats från XSD filen från förra uppgiften) från listen i "Choose your business object". Klicka Next. d. Kontrollera så att "GetComicsByTitle" är valt i listen "Choose a method" enligt nedan. Klicka på UPDATE, INSERT, och DELETE flikarna en efter en och välj "None" i deras "Choose a method" listor. Detta förhindrar att Visual Studio konfigurerar "ObjectDataSource" till att stödja INSERT, UPDATE och DELETE. När du är klar, klicka Next.

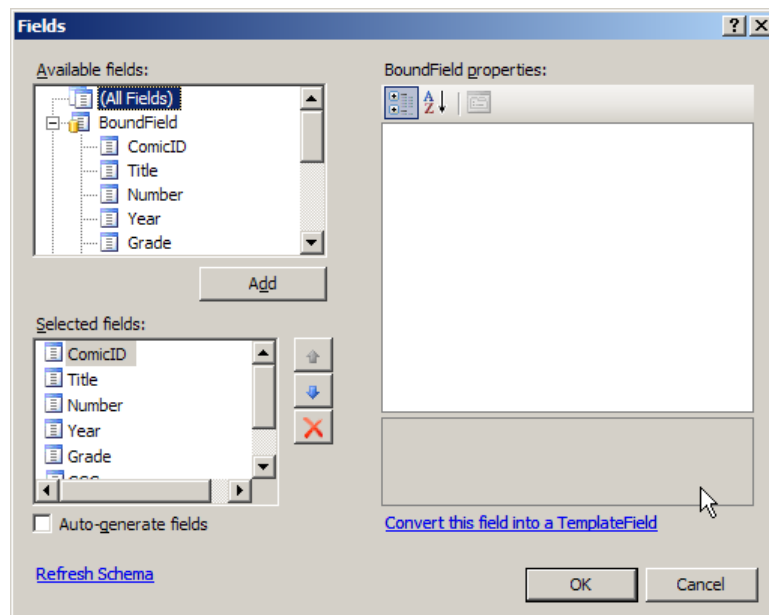
	
	e. Här näst tillfrågas du om att ange en källa för metoden "GetComicsByTitle"-s Title parameter. Välj "Control" i drop-down listen "Parameter source" och "DropDownList1" i "ControlID" drop-down listen enligt nedan. Klicka Finish. 
3. Lägg till en GridView	f. Växla till Source och undersök <asp:ObjectDataSource> elementet. Vad ser du som skapar kopplingen mellan "ObjectDataSource" och DropDownList? a. Växla tillbaka till Design och dra en "GridView" kontroll till sidan, Infoga ett par radbryt för att skapa lite utrymme mellan "GridView"-n och DropDownListen. b. Klicka på pilen uppe till höger hörn på "GridView"-n för att visa menyn "GridView Tasks" och välj "ObjectDataSource1" som "GridView"-s datakälla. c. Använd "Auto Format" valet i "Common GridView Tasks" menyn och välj temat "Slate". d. Markera GridView kontrollen i design. Sätt Width egenskapen för GridView-n till 100% i Properties fönstret. e. Tryck Ctrl+F5 för att starta Default.aspx. Resultatet skall se ut ungefär som nedan, kontrollera så att listan av serier i GridWiew-n ändras när man väljer en annan titel i DropDownListen.

ComicID	Title	Number	Year	Grade	CGC	BookValue
1004	.NET Man	1	2003	9.0	☐	175.0000
1005	.NET Man	2	2003	9.0	☐	100.0000
1006	.NET Man	3	2003	9.0	☐	65.0000
1007	.NET Man	4	2003	9.0	☐	50.0000
1008	.NET Man	5	2003	9.0	☒	50.0000

f. Stäng webbläsaren och återvänd till Visual Studio.

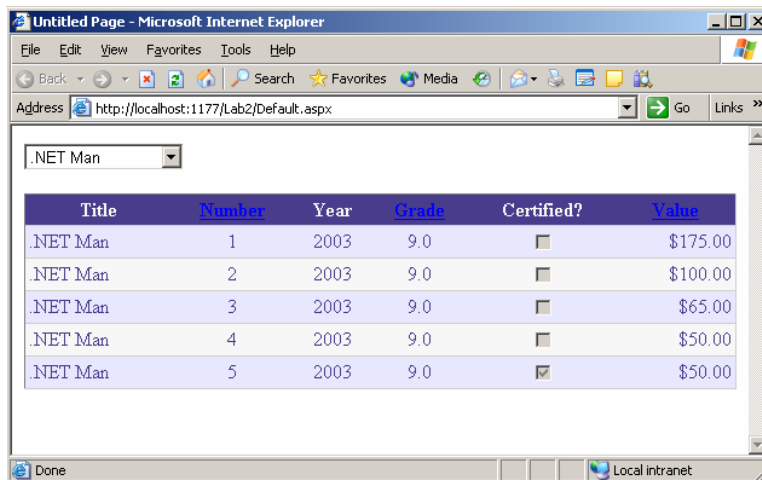
4. snygga till GridView-ns utseende

- Öppna Default.aspx i Design view.
- Klicka på pilen i det övre högra hörnet av "GridView"-n för att visa Tasks menyn, och välj "Edit Columns" för att visa dialogen "Fields" enligt nedan.



- Välj "ComicID" i "Selected Fields". I properties fältet till höger, ange ComicID "HeaderText" egenskap till "Comic ID" och dess Visible egenskap till false.
- Välj "Title" i "Selected Fields". Ange dess "SortExpression" egenskap till en tom sträng.
- Välj "Number" fältet. Ange dess "ItemStyle-HorizontalAlign" egenskap till center.
- Välj "Year" fältet. Ange dess "ItemStyle-HorizontalAlign" egenskap till "Center" och dess "SortExpression" egenskap till en tom string.
- Välj "Grade" fältet. Ange dess "ItemStyle-HorizontalAlign" egenskap till "Center".
- Välj "CGC" fältet. Ange dess "HeaderText" egenskap till "Certified?", dess "ReadOnly" egenskap till true, dess "ItemStyle-HorizontalAlign" egenskap till "Center", och dess "SortExpression" egenskap till en tom sträng.
- Välj "BookValue" fältet. Ange dess "HeaderText" egenskap till "Value", dess "DataFormatString" egenskap till "{0:c}", dess "HtmlEncode" egenskap till "False", och dess "ItemStyle-HorizontalAlign" egenskap till "Right".
- Klicka OK bekräfta ändringarna och stänga "Fields" dialogen.
- I design, visa "GridView"-ns "Tasks" menyn igen om den inte redan visas. Kryssa i menyns "Enable Sorting".

- l. Tryck Ctrl+F5 för att stata Default.aspx i webbläsaren. Sidan skall nu se ut ungefär så här. Kontrollera även att sortering efter number, grade, och value fungerar, samt att vid upprepad klickning på samma rubrik, sorteringen ändras mellan stigande och fallande.

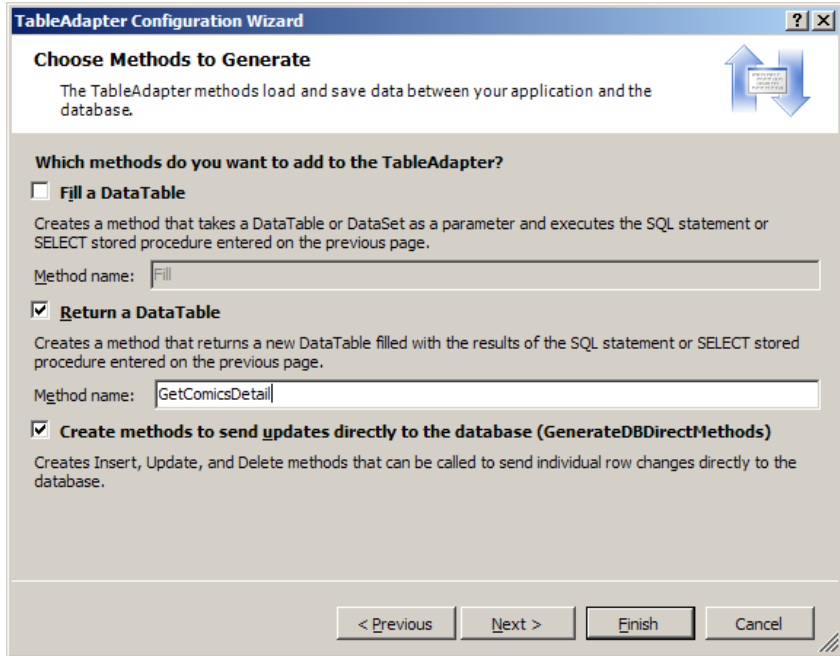


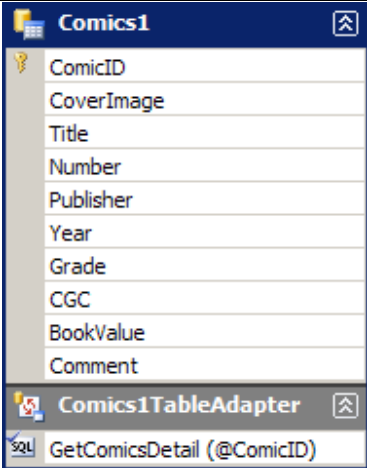
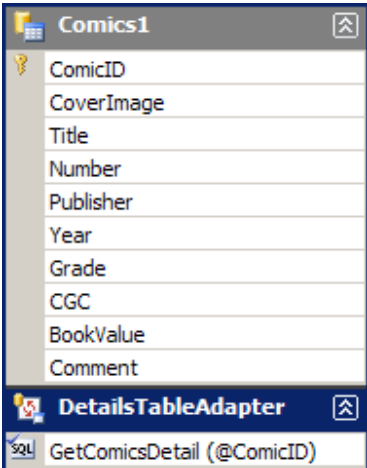
- m. Stäng webbläsaren och återvänd till Visual Studio.

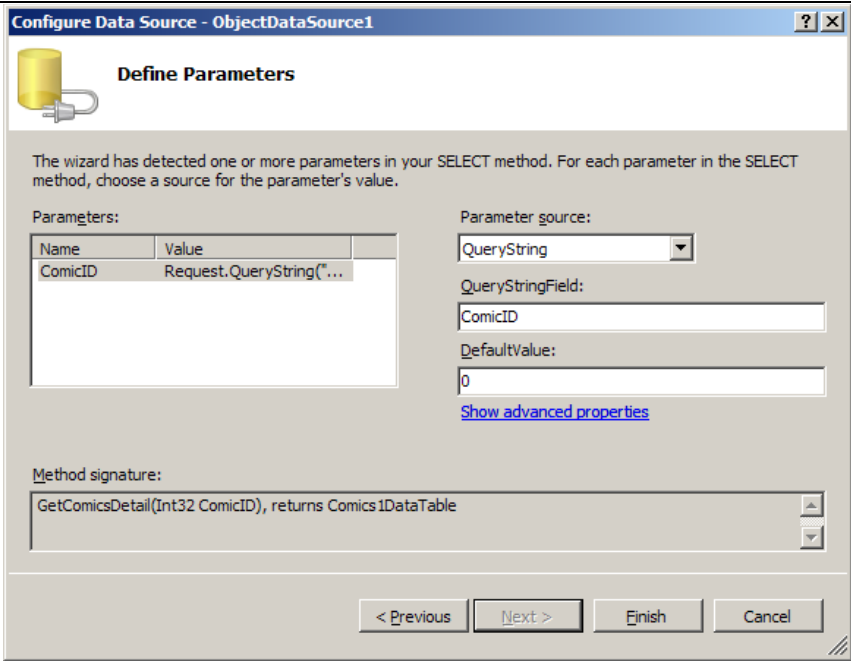
Övning 4

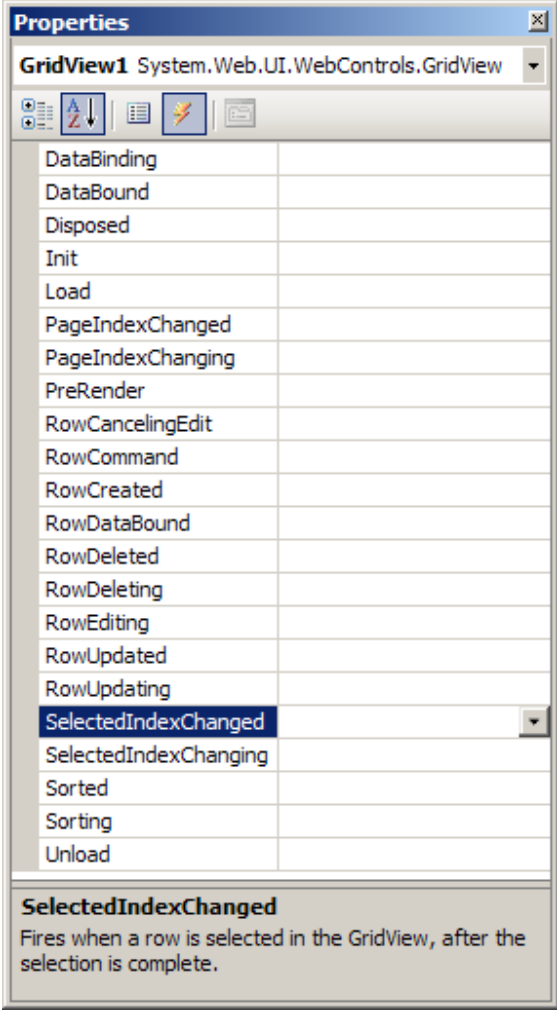
Skapa en detalj info sida

I denna övning skall du lägga till en sida som visar detaljer för en vald bok, Details.aspx. Du skall använda en DetailsView kontroll. Sedan skall du koppla ihop de två sidorna så att när du klickar på ett val i Default.aspx så visas detaljer om denna i Details.aspx.

Uppgift	Detaljerad beskrivning
1. Lägg till en sida till webbplatsen	- Högerklicka på C:\..\Lab2 i "Solution Explorer" och välj "Add New Item." - Välj "Web Form" och namnge den "Details.aspx". Se till så att kryssrutan "Place code in separate file" och att språket är satt till C# som vanligt innan du klickar på Add.
2. Lägg till en till data komponent	- Dubbeltklicka på MyComics.xsd i "Solution Explorer" fönstret för att den för edditering. - Högerklicka i design ytan och välj "Add->TableAdapter" för att visa "TableAdapter Configuration Wizard". - Välj "MyComicsConnectionString (Web.config)" i "data connections" listan och klicka Next. - På sidan "Choose a Command Type", välj "Use SQL statements." Klicka Next. - På sidan "Generate the SQL statements", skriv "SELECT * FROM Comics WHERE ComicID=@ComicID". Klicka på "Advanced Options" knappen se till så att "Use optimistic concurrency" och "Refresh the data table" inte är ikryssade. Klicka OK, följt av Next. - Fyll i "Choose Methods to Generate" sidan enligt nedan. Notera särskilt att namnet i "Method Name" är GetComicDetails, inte GetComicsByTitle.  - Klicka Next, följt av Finish. Kontrollera så att den nya datakomponenten ser ut så här:

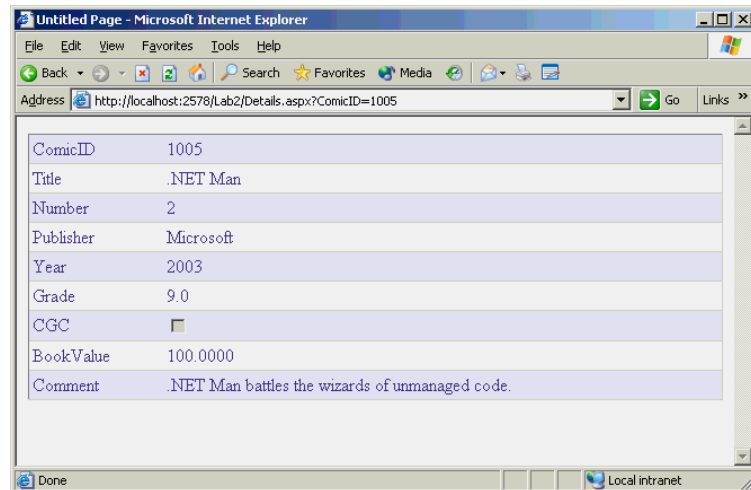
	 h. Högerklicka på "Comics1TableAdapter" och använd Rename valet för att ändra namnet till "DetailsTableAdapter":  i. Välj "Save All" från file menyn för att spara ändringar.
3. Lägg till en ObjectDataSource kontroll	a. Återvänd till utvecklingsmiljön och öppna "Details.aspx" i Design. b. Dra en "ObjectDataSource" kontroll från verktygsfältet till sedan. c. Klicka "Configure Data Source" i "ObjectDataSource Tasks" menyn. d. Under "Enter the name of your business object," välj "MyComicsTableAdapters.DetailsTableAdapter". Klicka Next. e. Kontrollera så att ComicDetails är valt i "Choose a method" listan i SELECT page. Välj "None" i "Choose a Method" listan på UPDATE, INSERT, och DELETE sidorna. klicka Next. f. Du blir nu ombedd att specificera en källa för "GetComicDetails"-s "ComicID" parameter. Välj "QueryString" i "Parameter source" drop-downlisten, skriv "ComicID" i fältet "QueryStringField", och skriv "0" i fältet "DefaultValue" enligt nedan. Klicka Finish.

	 g. Växla till Source och titta på <asp:ObjectDataSource> elementet. Vad gör <SelectParameters> elementet?
4. Lägg till Select knappar till GridView-n	a. Öppna Default.aspx i Design läge. b. välj "GridView" kontrollen och ange dess "AutoGenerateSelectButton" egenskap till true. c. Tryck Ctrl+F5 för att stata Default.aspx i wewbbläsaren. Vad har ändrats i GridView kontrollen? d. Stäng webbläsaren och återvänd till Visual Studio. e. Välj GridView kontrollen. f. Klicka på blyxtikonen i Poroperties fönstret för att visa en lista över GridView events.

	 g. Dubbelklicka på "SelectedIndexChanged" för att lägga till en "SelectedIndexChanged" event hanterare till Default.aspx.cs. h. Lägg till följande kod till eventhanteraren: <pre>C# Response.Redirect ("Details.aspx?ComicID=" + GridView1.SelectedValue);</pre> i. Återvänd till Default.aspx. i properties fönstret, klicka på knappen till vänster om blixten för att visa egenskaper i stället för event. j. Välj GridView kontrollen och kontrollera så att dess "DataKeyNames" egenskap är angiven till "ComicID" så att "GridView1.SelectedValue" kommer att returnera värdet på "ComicID" fältet i den rad som just nu är vald. k. Tryck Ctrl+F5 för att stata Default.aspx i webbläsaren. l. Klicka på en av GridView kontrollens Select knappar. Vad händer? Vad ser du i webbläsarens adress fält? m. Stäng webbläsaren och återvänd till Visual Studio.
5. Lägg till en DetailsView kontroll	a. Öppna Details.aspx i Design läge. b. Dra en "DetailsView" kontroll från verktygsfältet och släpp den på sidan. c. Använd "DetailsView Tasks" meny för att välja "ObjectDataSource1" som DetailsView-ns data källa. d. Använd "Auto Format" valet i "DetailsView Tasks" menyn för att lägga till

utseendet men namn "Slate."

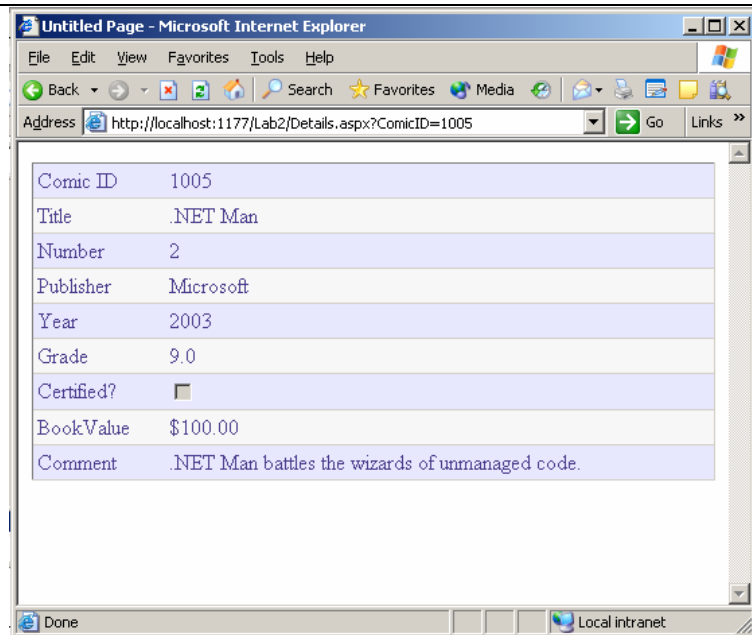
- e. Välj "DetailsView" kontrollen i design. Gå till Properties fönstret och ange DetailsView kontrollens Width egenskap till 100%.
- f. Välj "Default.aspx" i Solution Explorer fönstret. tryck Ctrl+F5 för att starta den.
- g. Klicka på en av GridView kontrollens Select knappar och kontrollera att Details.aspx visas med detaljer om vald seriebok enligt nedan.



- h. Stäng webbläsaren och återvänd till Visual Studio.

6. Snygga till DetailsView-ns utseende

- a. Öppna Details.aspx i Design läge.
- b. Klicka på pilen längst upp till höger på DetailsView kontrollen för att visa "DetailsView Tasks" menyn, klicka "Edit Fields" för att visa "Fields" dialogen.
- c. Markera "ComicID" i fältet "Selected Fields". Ange dess "HeaderText" egenskap till "Comic ID".
- d. Markera "CGC" i fältet "Selected Fields". Ange dess "HeaderText" egenskap till "Certified?" och dess ReadOnly egenskap till true.
- e. Markera "BookValue" i fältet "Selected Fields". Ange dess "HeaderText" egenskap till "Value", dess DataFormatString egenskap till "{0:c}", och dess HtmlEncode egenskap till "False".
- f. Klicka OK för att spara ändringarna och stänga dialogen.
- g. Välj Default.aspx i Solution Explorer fönstret och kör applikationen.
- h. Klicka på en av GridView kontrollens Select knappar och kontrollera så att resultatet set ut ungefär som nedan.

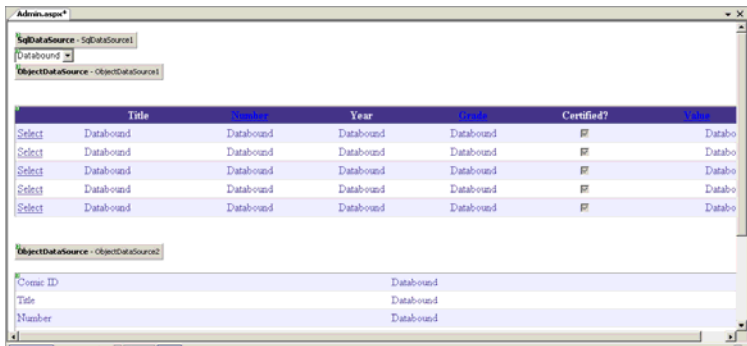


- i. Stäng webbläsaren och återvänd till Visual Studio.

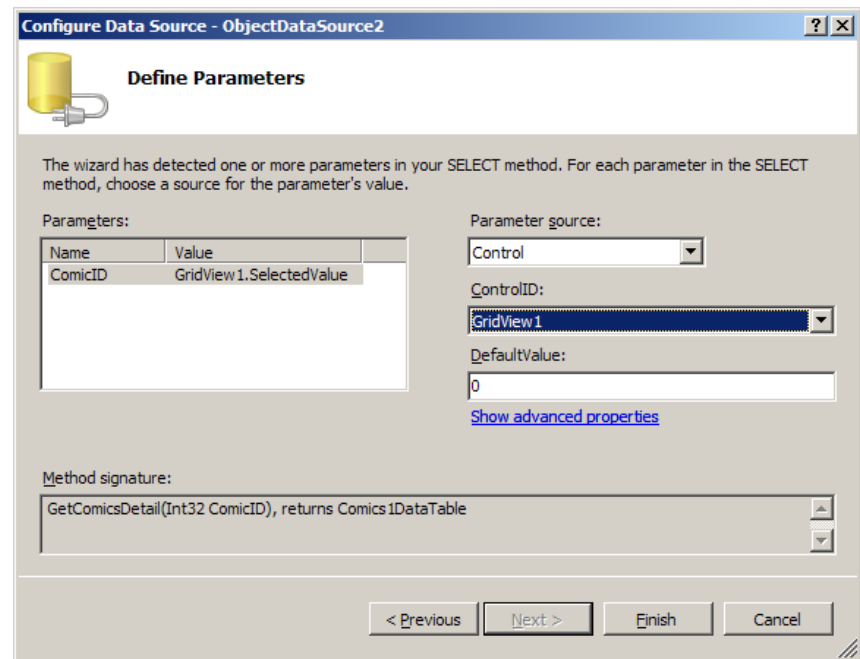
Övning 5

skapa en admin sida

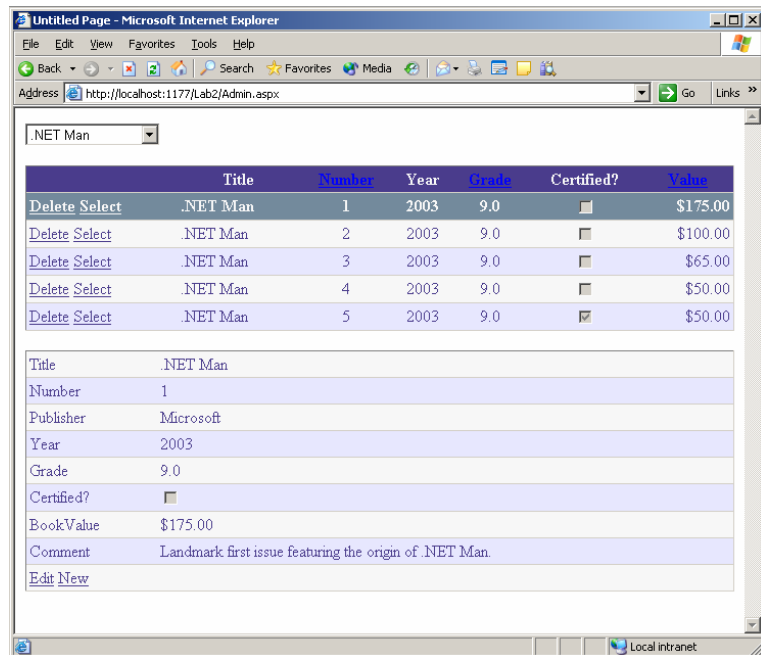
Du skall i denna övning lägga till en sida med namn Admin.aspx till din applikation. På denna sida skall du lägga en edditerbar huvud detalj vy för läsning, redigering, tillägg och radering av förekomster. Du skall använda datakomponenter som du byggt i tidigare övningar för att koppla till databasen, du skall använda edditerings funktionen hos "GridView" kontrollen och "DetailsView" kontrollen och två vägs databindningsstödet i "ObjectDataSource" för arbetet med uppdatering av databasen.

Tasks	Detailed Steps
1. Lägg till ytterligare en sida till ditt projekt	j. Lägg till en ny sida på samma sätt som tidigare. Sidan skall vara av typen "WebForm" och ha namnet Admin.aspx och samma inställningar som tidigare gäller.
2. Kopiera kontroller till sidan	a. Öppna Default.aspx i Source läge och kopiera allt mellan taggarna <form id="form1" runat="server"> och </form> till urklipp. b. Öppna Admin.aspx sidan som du just skapat i Source läge och klistra in det du kopierat mellan taggarna <form id="form1" runat="server"> och </form>. c. Radera den del som lyder OnSelectedIndexChanged="GridView1_SelectedIndexChanged" som finns i taggen <asp:GridView>. d. Öppna Details.aspx i Source läge och kopiera allt mellan taggarna <form id="form1" runat="server"> och </form>. e. Gå tillbaka till Admin.aspx i Source läge och klistra in det du kopierat under det du klistrade in i steg b. f. Gå till Design läge och infoga en blank rad mellan "GridView" och "DetailsView" för att göra sedan lite luftigare. Sidan bör nu se ut ungefär som nedan i design läge:  g. Använd "DetailsView" kontrollens "Tasks" meny för att ange dess "data source" till "ObjectDataSource2". Om du får frågan om du vill skapa referenser till fält och nycklar i "DetailsView", svara No. h. Välj "GridView" kontrollen. Ange dess "AutoGenerateDeleteButton" egenskap till true. i. Välj "DetailsView" kontrollen. Ange dess "AutoGenerateEditButton" och "AutoGenerateInsertButton" egenskaper till true. j. Kontrollera så att "DetailsView" kontrollens "DataKeyNames" egenskap är satt till "ComicID". k. Gå till "DetailsView" kontrollens "Tasks" meny och välj "Edit Fields." I "Fields" dialogen, ange "ComicID" fältets "Visible" egenskap till False och "Certified?" fältets ReadOnly egenskap till False. → OK. l. Välj "ObjectDataSource2" kotrollens "Tasks" meny. Välj "Configure Data Source" och klicka Next tills du kommer fram till "Define parameters" sidan. "ObjectDataSource2" är nu konfigurerad till att hämta ComicID parametern som den använder från databasfrågan. Fyll i dialogen enligt nedan för att ändra

konfigurationen för "ObjectDataSource2" att hämta parametern från "GridView".
→Finish. Om Visual Studio erbjuder dig att uppdatera "DetailsView" kontrollen, svara No.



- m. Starta applikationen (Admin.aspx). Kontrollera så att en klickning på en av "GridView" kontrollens "Select" knappar visar detaljer om den valda serieboken i "DetailsView" kontrollen, enligt nedan.

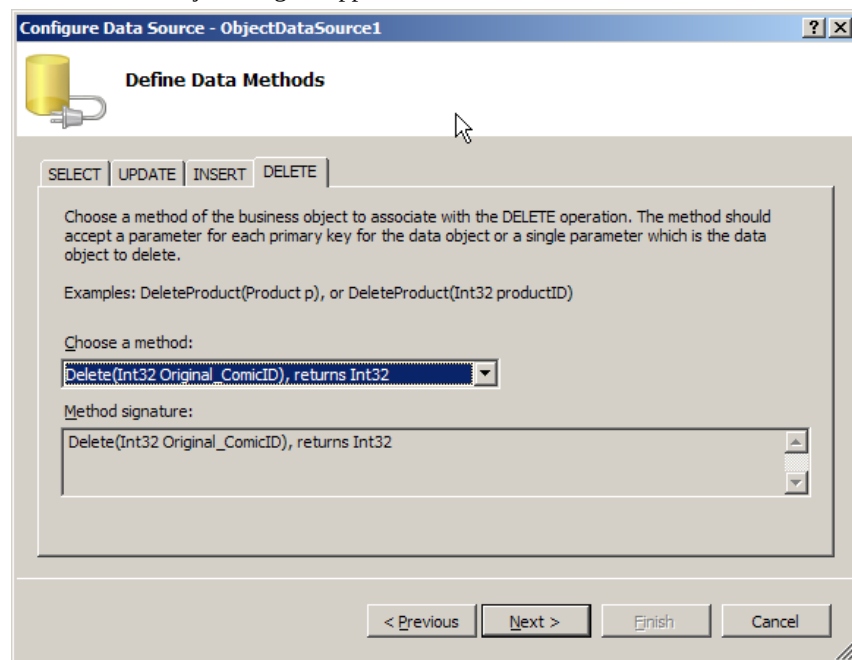


- n. Stäng applikationen och återvänd till Visual Studio.

3. Ställ in "ObjectDataSource" kontrollen så att den stöder 2-vägs data bindning

- a. Visa Admin.aspx i design läge. Välj "Configure Data Source" kommandot från "ObjectDataSource1" kontrollens "Tasks" meny. Klicka Next tills du kommer till "Define data methods" sidan.
- b. Ange "ObjectDataSource1" kontrollens SELECT och DELETE metoder till: "GetComicsByTitle" och "Delete". (DELETE inställningarna visas nedan.) Ange dess UPDATE och INSERT metoder till "None" eftersom denna "ObjectDataSource" inte

kommer att användas för att infoga eller uppdatera poster. →Next, följt av Finish. Om Visual Studio erbjuder dig att uppdatera "DetailsView" kontrollen, svara No.



- c. Välj "Configure Data Source" kommandot från "ObjectDataSource2" objektets "Common ObjectDataSource Tasks" meny. Klicka Next tills du kommer till "Define data methods" sidan.
- d. Ange "ObjectDataSource2"-s SELECT, UPDATE, och INSERT metoder till "GetComicDetails". Ange dess DELETE metod till "None" eftersom denna "ObjectDataSource" inte kommer att användas för att radera poster. →Next, följt av Finish. Om Visual Studio erbjuder dig att uppdatera "DetailsView" kontrollen, svara No.
- e. Markera "ObjectDataSource1" i design läge och klicka på blix symbolen i Properties fönstret för att visa en lista över event för ObjectDataSource.
- f. Dubbelklicka på "Deleted" i Properties fönstret för att lägga till en eventhanterare för "ObjectDataSource1.Deleted" eventet.
- g. Lägg till följande kod till eventhanteraren så att du kan vara säker på att drop-down listen är uppdaterad om den sista posten i en kategori är raderad:

C#

```
DropDownList1.DataBind();
```

- h. Återvänd till Admin.aspx och markera "ObjectDataSource2" i design läge.
- i. Gå till Properties fönstret (som fortfarande visar en lista av event för ObjectDataSource) och dubbelklicka på "Inserted" för att lägga till en eventhanterare för "ObjectDataSource2.Inserted".
- j. Lägg till följande kod till eventhanteraren så att du kan vara säker på att GridView kontrollen och ddrop-down listen är uppdaterad om en seriebok läggs till till databasen:

C#

```
DropDownList1.DataBind();  
GridView1.DataBind();
```

- k. Återvänd till Admin.aspx och markera ObjectDataSource2 i design läge.
- l. Gå till Properties fönstret (som fortfarande visar en lista av event för ObjectDataSource), Dubbelklicka på "Updated" för att lägga till en eventhanterare för "ObjectDataSource2.Updated".
- m. Lägg till följande kod till eventhanteraren för att försäkra dig om att "GridView" kontrollen är uppdaterad när en post i databasen uppdateras:

	C# <pre>GridView1.DataBind ();</pre>
--	--

4. Testkör Admin sidan	a. Starta Admin.aspx i webbläsaren. b. Prova möjligheten att lägga till poster genom att välja en av "Captain Code" böckerna i "GridView" kontrollen och klicka på New knappen i "DetailsView" kontrollen. Fyll fälten enligt nedan:	
	Name	Value
	Title	Captain Code
	Number	8
	Publisher	Microsoft
	Year	2004
	Grade	9.0
	Certified?	Unchecked
	Value	30
	Comment	Introducing CodeGirl and her sidekick, CodeKid
	c. Klicka "Insert" längst ner på DetailsView för att infoga boken i databasen. Kontrollera så att "Captain Code 8" syns i GridView kontrollen. d. Testa sidans uppdaterings möjlighet genom att välja "Captain Code 8" i "GridView" kontrollen och klicka på "Edit" i "DetailsView". Ändra "grade" från 9.0 till 7.0. Klicka "Update." Kontrollera att grade ändras till 7.0 i "GridView". e. Testa sidans raderings funktion genom att klicka på "Delete" knappen bredvid "Captain Code 8" i "GridView". Kontrollera så att den försvinner. f. Stäng webbläsaren och återvänd till Visual Studio.	