2011



USER GUIDE

Appointment Manager

Suppose that you need to create an appointment manager for your business. You have a receptionist in the front office and salesmen ready to service customers. Whenever a customer comes in requesting a consultation, the receptionist will make an appointment in the *Appointment Manager*. The next automatically assigned salesman will be rotated by the application. If a customer requests a specific salesman, then that appointment will not affect the rotation process.

When a new appointment is created, the salesman that performed the smallest number of appointments will be assigned. The rotation process will ensure fair distribution of appointments.

New Appointments						
Please fill this form and dick OK button to create a new appointments record. Click Cancel to return to the previous screen.						
New Appointments						
Complete the form. Make sure to enter all required fields.						
Client Name * Date * 10/26/2011 10:10 AM						
Salesman Full Name (select) 📾 Assigned Salesman Full Kumar Rajkumar Name						
* - indicates a required field OK Cancel						

If you enter a client name and press *Ok* to save the appointment, the assigned salesman will be inserted into the salesman field, and the salesman's number of appointments will increase by one. When another appointment is made, the salesman with the next lowest number of appointments will be assigned. This salesman will be different for each appointment.

New Appointments	New Appointments X						
Please fill this form and dick OK button to create a new appointments record. Click Cancel to return to the previous screen.							
New Appointments	New Appointments						
Complete the form. Make s	sure to enter all required fields.						
Client Name *	Harry						
Date *	10/26/2011	10:11 AM 👻					
Salesman Full Name	(select)						
Assigned Salesman Full George Fischer Name							
* - indicates a required field	3			ОК	Cancel		

If the customer specifically requests a salesman, a salesman will be selected by the receptionist using the lookup for the field *Salesman Full Name*. When the appointment is saved, the number of appointments will not be changed for either salesman.

New Appointments X							
Please fill this form and dick OK button to create a new appointments record. Click Cancel to return to the previous screen.							
New Appointments	New Appointments						
Complete the form. Make s	ure to enter all required fields.						
Client Name * Date * Salesman Full Name Assigned Salesman Full	Harry 10/26/2011 10:11 AM Kumar Rajkumar						
Name * - indicates a required field	OK Cancel						

We will show you how to set up the database using *Microsoft SQL Server 2008 Management Studio*, generate the application using *Code OnTime Generator*, and use the *Designer* in combination with *Visual Web Developer 2008 Express Edition* to implement this algorithm.

To start, we created a database called *ServiceDesk*. This database has two tables, *Appointments* and *Salesmen*. *Salesmen* is a table of all the salesmen in the business, and have the columns *SalesmanID* (an automatically assigned number), *FullName*, and *AssignedAppointments* (a count of how many

appointments each salesperson has been assigned). Appointments table is a list of the appointments, and it has AppointmentID (an automatically assigned number), ClientName (name of the customer), Date (the date and time the appointment was made), AssignedSalesmanID (indicates the assigned salesperson according to rotation), and SalesmanID (the ID of the salesman who actually performed the consultation). Value of SalesmanID is only different from AssignedSalesmanID if a client specifically requests a salesperson. Both AssignedSalesmanID and SalesmanID are foreign key references to the Salesmen table.

You can see the database schema in the picture to the right.

	Column Name	Data Type	Allow Nulls
8	SalesmanID	int	
	FullName	nvarchar(50)	
	AssignedAppointments	int	
pt	oointments		8
pt	oointments Column Name	Data Type	Allow Nulls
		Data Type int	Allow Nulls
8	Column Name		Allow Nulls
8	Column Name AppointmentID	int	Allow Nulls
8	Column Name AppointmentID ClientName	int nvarchar(50)	Allow Nulls Allow Control
8	Column Name AppointmentID ClientName Date	int nvarchar(50) datetime	Allow Nulls Allow Control

SQL Script Creating the Sample Database

```
USE [ServiceDesk]
GO
/***** Object: Table [dbo].[Salesmen] *****/
SET ANSI NULLS ON
GO
SET QUOTED IDENTIFIER ON
GO
CREATE TABLE [dbo].[Salesmen](
       [SalesmanID] [int] IDENTITY(1,1) NOT NULL,
       [FullName] [nvarchar] (50) NOT NULL,
       [AssignedAppointments] [int] NOT NULL,
CONSTRAINT [PK Salesmen] PRIMARY KEY CLUSTERED
(
       [SalesmanID] ASC
)WITH (PAD INDEX = OFF, STATISTICS NORECOMPUTE = OFF, IGNORE DUP KEY = OFF, ALLOW ROW LOCKS =
ON, ALLOW PAGE LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/****** Object: Table [dbo].[Appointments] *****/
SET ANSI NULLS ON
GO
SET QUOTED IDENTIFIER ON
GO
CREATE TABLE [dbo].[Appointments](
       [AppointmentID] [int] IDENTITY(1,1) NOT NULL,
       [ClientName] [nvarchar] (50) NOT NULL,
       [Date] [datetime] NOT NULL,
       [AssignedSalesmanID] [int] NOT NULL,
       [SalesmanID] [int] NULL,
CONSTRAINT [PK Appointments] PRIMARY KEY CLUSTERED
(
       [AppointmentID] ASC
)WITH (PAD INDEX = OFF, STATISTICS NORECOMPUTE = OFF, IGNORE DUP KEY = OFF, ALLOW ROW LOCKS =
ON, ALLOW PAGE LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Default [DF Salesmen Appointments] *****/
ALTER TABLE [dbo].[Salesmen] ADD CONSTRAINT [DF Salesmen Appointments] DEFAULT ((0)) FOR
[AssignedAppointments]
GO
/***** Object: Default [DF Appointments Date] *****/
ALTER TABLE [dbo]. [Appointments] ADD CONSTRAINT [DF Appointments Date] DEFAULT (getdate()) FOR
[Date]
GO
/****** Object: ForeignKey [FK_Appointments_Salesmen] ******/
ALTER TABLE [dbo]. [Appointments] WITH CHECK ADD CONSTRAINT [FK Appointments Salesmen] FOREIGN
KEY([AssignedSalesmanID])
REFERENCES [dbo].[Salesmen] ([SalesmanID])
GO
ALTER TABLE [dbo].[Appointments] CHECK CONSTRAINT [FK Appointments Salesmen]
GO
/***** Object: ForeignKey [FK_Appointments_Salesmen1] ******/
ALTER TABLE [dbo]. [Appointments] WITH CHECK ADD CONSTRAINT [FK Appointments Salesmen1] FOREIGN
KEY([SalesmanID])
REFERENCES [dbo].[Salesmen] ([SalesmanID])
GO
ALTER TABLE [dbo]. [Appointments] CHECK CONSTRAINT [FK Appointments Salesmen1]
GO
```

Run *Code OnTime Generator* by clicking on the shortcut on your desktop. Click on the *Create new project* link. Create a new *Web Site Factory* project by choosing it from the list. Enter *Appointments* as the project name.

🕲 Create Project - Code OnTime	- O X
NITIME SCODE	
Please enter the project name and click the <i>Create</i> button.	
New Project	
Vendor: Code OnTime LLC http://www.codeontime.com	
Web Site Factory	
Create an ASP.NET/AJAX web site with navigation system, role-based security, membership manager, built-in data export and reporting, adaptive filtering, advar options, and much more. The generated project is managed via the file system folder structure.	nced search
Review a complete feature list of this project. Try a live demo now.	
Name:	
Appointments	
Language:	
C# •	
Create	Cancel
© 2011 Code OnTime LLC. All rights reserved. Version 5.0.0.72.	

Press the *Create* button in the bottom right corner to create your project. Select .NET Framework 4.0 and press *Next*. On this page, click on the "..." button to the left of the *Connection String* field.

Database Connection - Code OnTime	
	😴 💸 🌆 🕒 📼 ^
Please specify a database connection string for this project.	
Database Connection	
A valid connection string compatible with the selected data provider is required to gener and are easy to maintain with the provided XML schemas. Data Provider:	rate the project. Data controller descriptors are automatically built from your database
.Net Framework Provider for SqlServer (System.Data.SqlClient)	•
Connection String: Connection String: Connection string. Connection string. Select the database tables and views included in the project. All database tables and views are included in this project. Change	 Configure database connection string and install database features.
	Cancel Back Next
10/26/2011 10:18:00 AM if test="not(file/@name='DataAquarium.Project.xml')" 10/26/2011 10:18:00 AM test passed 10/26/2011 10:18:00 AM copy input="[Documents]\Code On Time\Library\Data Aquarium]	\DataAquarium.Project xml" output="DataAquarium.Project xml"

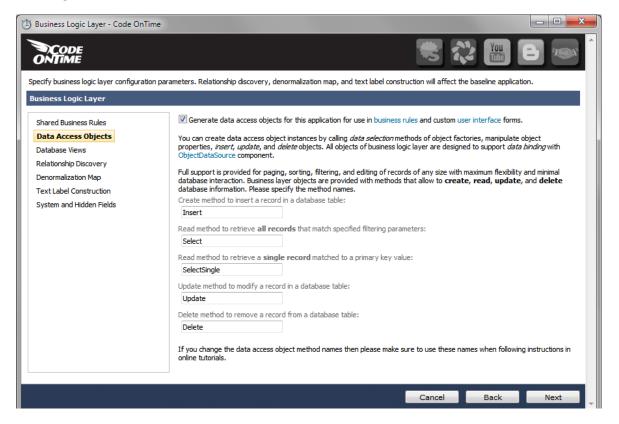
It will take you to a page where you will type in the database server name, and select the database. Press *Test* to confirm.

街 SQL Server Connection - Code OnTime		3
	💌 🔁 🖏 🖏 💌	*
Please specify SQL Server name and database.		
SQL Server Connection		
Server:		
Windows authentication		
SQL Server authentication		
User name:		
Password:		
Database:		
ServiceDesk Create		
Sample Tables: You can popluate the application database with sample tables.		
(select)		
Membership;		
Your application can be configured to implement user and role manager based on Microsoft ASP.NET Membership	ip.	
If you own Unlimited edition then you may consider implementing custom membership and role providers instead.	J.	
Status Add Remove		
	Test OK Cancel	-
10/26/2011 10:18:00 AM if test="not file/@name="DataAquarium.Project.xml")"		
10/26/2011 10:18:00 AM test passed 10/26/2011 10:18:00 AM copy input="[Documents]\Code On Time\Library\Data Aquarium\DataAquarium.Project.xml	m" output="Data Aquarium Project xml"	
10/26/2011 10:18:00 AM end if		-

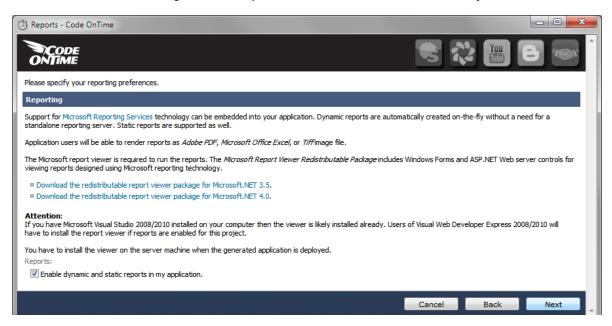
If the connection succeeds, press the "Add" button under *Membership* to add *ASP.NET Membership* to your application.

SQL Server Connection - Code OnTime	
	s 🔁 🔛 🖘
Please specify SQL Server name and database.	
SQL Server Connection	
Server:	
Windows authentication	
SQL Server authentication	
User name:	
Password:	Message from webpage
Database: ServiceDesk Sample Tables: You can populate the application database with sample tables.	Click OK to create tables and stored procedures required to support ASP.NET Membership and Role Manager in the database.
(select) #embership: Your application can be configured to implement user and role manager b f you own Unlimited edition then you may consider implementing custom	
Status Add Remove	
	Test OK Cancel
//26/2011 10:18:00 AM if test="notifile/@name=DataAquarium.Project.xm //26/2011 10:18:00 AM test passed //26/2011 10:18:00 AM copy input="[Documents]\Code On Time\Library\[//26/2011 10:18:00 AM codi	i)")ata Aquarium \DataAquarium .Project xml" output="DataAquarium .Project xml"

Press OK, and go back to the Database Connection page. Press Next to continue. In the list of items, switch to Data Access Objects. Enable generation of data access objects by checking the box. These will help you write data manipulation logic for the application, as explained later in this article. The options do not need to be changed beyond that on this page, so proceed by pressing the *Next* button in the lower right corner.



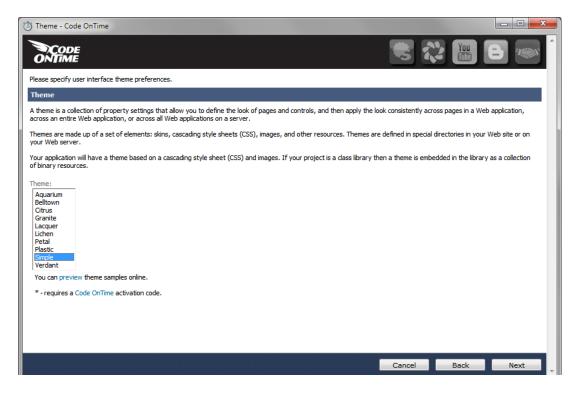
You can enable reports by checking the box on the next page. This requires *Microsoft Report Viewer* 2008 Redistributable Package to view reports. The viewer comes with *Microsoft Visual Studio 2008*.



Proceed to the *Authentication and Membership* page by pressing *Next*. If you installed ASP.NET Membership previously, then these settings should be configured correctly.

🕲 Membership - Code OnTime		• ×
Please select authentication and membersh	ip options for your web app. Options depend on product edition, may require custom coding, or are not compatible with eac	th other.
Authentication and Membership		
Authentication	$\boxed{\mathbb{V}}$ Enable support for ASP.NET Membership with membership bar and user manager.	
Login Window Membership Bar	Enable Windows Authentication. Recommended for Intranet applications only.	
Logout	Enable custom authentication and membership implementation. Requires additional coding.	
	$\overline{\mathbb{C}}$ Membership will use a standalone database that already exists.	
	Specify a valid connection string compatible with the selected data provider for a standalone membership database. Data Provider:	
	.Net Framework Provider for SqlServer (System.Data.SqlClient)	
	Connection String:	
	Data Source=.;Initial Catalog=AppointmentManager;Integrated Security=True;	
	Click here if you need assistance to build the connection string.	
	If the connection string is left blank, membership is enabled, and System.Data.SqlClient is selected as the data prov ASP.NET will expect that Microsoft SQL Express is installed on this computer.	ider then
	Cancel Back N	Vext -

Keep on pressing *Next* until you reach the *Theme* page. This page allows you to choose the theme of the project. Themes are made up of cascading style sheets, images, and other resources. Choose *Simple* theme from the list box.



Keep on pressing the *Next* button until you reach the *Data Controllers* page. It will display a list of the data controllers on the left, and some buttons on the right side of the page.

🕲 Data Controllers - Code OnTime	
	😸 🗞 🔚 E 📟 É
The analysis of your database has been completed. Code genera	ation will start immediately when you dick the <i>Next</i> button.
Data Controllers	
The following data controllers will be created.	
Appointments Salesmen	You can customize the properties of this code generation project with Code OnTime Designer. Customization is preserved between code generation sessions and is automatically re-applied when your database schema has changed. Start Designer You can customize the project metadata in Microsoft Excel spreadsheet. Project customization via spreadsheet is applied to the raw project metadata prior to any changes done in Designer. Start Excel Refresh Sync Merge
If your database has changed then not all of the tables may be one meta data.	displayed in the list. Please return to the <i>Database Connection</i> section and request to update the database
All existing files in the current project folder are automatically ba	cked up.
	Cancel Back Generate
10/26/2011 10:38:05 AM load path="[Documents]\Code On Time\F	DnTime\Library\Web Site Factory\MapBuilder.4.xstt" output="Map4.xml" Projects\Web Site Factory\Appointments\Controllers.xml" useLog="true" DnTime\Library\Web Site Factory\Application.xstt" output="Application.xml"

Press the *Start Designer* button on the Data Controllers page. This will bring up the *Designer*, and display the *All Controllers* page.

😕 Data Controllers - Code OnTime 📃 💷 🔫							
ONTIME					1	8 💸 🔠	8
Review and modify properties of the project items and select <i>Close</i> to return to code generator.							
Project Designer	r						exit
🕒 Home							
All Controllers	All Commands	All Fields All V	/iews All Data Fields	All Pages All Use	Controls		
'his is a list of data	controllers. View	tutorial that show	s how to work with the da	ta controllers.			
Quick Find	P New 🔹	Preview				View	Controllers -
lame		Generate	Conflict Dete	ection	Handle	r Annotations	
Appointments		Yes	Overwrite Ch	nanges	n/a	n/a	
Salesmen		Yes	Overwrite Cł	nanges	n/a	n/a	
						Showing 1-2 c	of 2 items Refresh
All customized pressure of the second sec	oject settings are	stored in *.Log.x	cml files located in the roo	ot of your project fol	der.		
🗈 2011 Code OnTime	e LLC. All rights re	served. Version 5	.0.0.72.				

Now, click on the *Salesmen* controller in the *All Controllers* list (selections are made by clicking on the link in the first column). You can see that the tab set has changed, so click on the *Fields* tab. Select *AssignedAppointments* field. Press *Edit*, and change *Code Default* field to "0". This way, the default value of the field is presented as 0 in the user interface. Press *Ok* to save changes.

Project Designer							
Ontroller: Salesmen > Field: AssignedAppointments (Int32)							
Field Items Validators Data Fields Field Outputs							
Please review the field information below. Click Edit to change this record, click Delete to delete the record, or click Cancel/Close to return back.							
Record -	View: Field -						
* - indicates a required field OK Delete Cancel							
General	Name *						
Specify field name, type, and data properties of the field.	AssignedAppointments						
Server Defaultis a SQL expression used as a field value when no value is provided for the field in INSERT and UPDATE statement.							
Indicate that the field is <i>computed</i> if the field is not physically present in the dataset produced by controller's command. Computed field requires a mandatory <i>formula</i> that must be defined as a valid SQL expression. This expression is automatically inserted in SELECT statements when needed.	Type* Int32						
	Allow null values.						
Calculated field values can be produced by business rule methods with attribute ControllerAction. You must list the context fields that will cause the calculation. Optional code formula is is embedded into an automatically created business rule and is calculated	The value of this field is computed at run-time by SQL expression.						
whenever any context field is changed.	\blacksquare The value of the field is calculated by a business rule expression.						
Code Default is an expression written in the programming language of your project. The expression is evaluated in an automatically created business rule to produce a default value for the field of the new row before it is presented in the user interface.	Server Default ((0))						
<i>Code Value</i> is an expression written in the programming language of your project. This expression is evaluted every time a record is saved to the database.	Code Default						

Go back to the *All Controllers* page by pressing the back icon on the left-hand side of page path. Select the *Appointments* controller. Press *Edit*, and change the *Handler* field to "AppointmentBusinessRules". This will create a business rule class, which allows altering the controller's behavior.

Controller Commands Fields Views Categories Data Field	ls Action Groups Actions
Please review data controller information below. Click Edit to change this re	ecord, click Delete to delete the record, or click Cancel/Close to return back.
Record -	View: Controller •
General Name of data controller.	Controller Name * Appointments Include in code generation
Miscellaneous Specify conflict detection strategy and optional connection string name. Specify a connection string name only if the controller is working with the database other than the one selected for this project.	Conflict Detection * Overwrite Changes Compare All Values Connection String Name
Business Rules Specify the name of the business rules class that can be extended to respond to the controller actions.	Handler AppointmentBusinessRules

Press *Save*. Click on the *Appointments* controller. Navigate to the *Fields* tab using the tabs at the top, and select the field *Date*. In the *Data Format String* field, type in "g", so that date and time will be displayed for the appointment date when presented in the user interface. Press *Ok* to save.

Project Designer	exit
O Home > Controller: Appointments > Field: Date (DateTime)	
Field Items Validators Data Fields Field Outputs	
Please review the field information below. Click Edit to change this record, click Delete to delet	te the record, or click Cancel/Close to return back.
Record -	View: Field -
* - indicates a required field	OK Delete Cancel
General	Name *
Specify field name, type, and data properties of the field.	Date
Server Default is a SQL expression used as a field value when no value is provided for the field in INSERT and UPDATE statement.	Controller Appointments
Indicate that the field is <i>computed</i> if the field is not physically present in the dataset produced by controller's command. Computed field requires a mandatory <i>formula</i> that must be defined as a valid SQL expression. This expression is automatically inserted in SELECT	Type * DateTime ▼
statements when needed. <i>Calculated</i> field values can be produced by business rule methods with attribute	Allow null values.
ControllerAction. You must list the context fields that will cause the calculation. Optional code formula is is embedded into an automatically created business rule and is calculated whenever any context field is changed.	The value of this field is computed at run-time by SQL expression. The value of the field is calculated by a business rule expression.
Code Default is an expression written in the programming language of your project. The	
expression is evaluated in an automatically created business rule to produce a default value for the field of the new row before it is presented in the user interface.	Server Default (getdate())
Code Value is an expression written in the programming language of your project. This expression is evaluted every time a record is saved to the database.	Code Default
The field must be marked as <i>on-demand</i> if the field is a large binary object (BLOB) or text in order to speed up record retrieval.	Code Value
	Value is retrieved on demand
	The field is included in all data views but remains hidden from users.
Presentation	Label *
Presentation interface properties of the field.	Date
A data format string is applied to the field value on the dient via JavaScript <i>String.format</i> function. You can overide the data format string on the data field level in views.	Values of this field cannot be edited.
A data format string is applied to the field value on the dient via JavaScript String.format function. You can enable a server-side formatting via .NET System.String.Format function if Format On Client is set to No.	Show In Summary
	☑ HTML encoding
Editor is a custom user control responsible for rendering of the field value in "edit" mode. Standard editor with name <i>RichEditor</i> is available in each project. We recommend using the	Data Format String
TextMode property with the corresponding data fields to enable "rich" editing of the field value.	g Use data format strings compatible with <i>String, format</i> functions.

Press the *back* icon in the upper left corner. Navigate to the *Views* tab, and select *grid1*. Press the *Edit* button, and change the *Sort Expression* field to "Date desc", so that the latest appointment will always appear at the top of the list.

Sort and Filter	Sort Expression	
Sort expression is a list of data field names of this view, each followed by optional <i>asc</i> or <i>desc</i> suffix.	Date desc	
Filter expression must be compatible with the back-end database server syntax. Data field names used in filter are automatically expanded into appropriate SQL expressions as defined in command.	Filter Expression	*

Press *Ok* to save. Now, navigate to *All Controllers* tab using the *Back* button in the upper left corner. Then, navigate to the *All Pages* tab. Select *Appointments*, and change the *Path* field from "Salesmen | Appointments" to just "Appointments", so that it appears on the main tabs, not underneath Salesmen.

Project Design	er						exit
G Home							
All Controllers	All Commands All Fields All Views	All Data Fields	All Pages	All User Controls			
This is a list of ap	plication pages.		-				
Quick Find	P Record -						View: Pages -
Name	Index Title		Path		Style	Roles	
Home	1000 ^HomeTitle^Start^HomeTitle^		^HomeF	ath^Home^HomePa	Home	n/a	
Salesmen	1010 Salesmen		Salesme	n	n/a	n/a	
Appointments	1020 Appointments		Appoint	Appointments			•
Save	Cancel						
Membership	1030 MembershipTitle^Membership Manager^MembershipTitle^		^Membe	ershipPath ^Members	hip^MembershipPath^	Users	Administrators
Showing 1-4 of 4							

Press *Ok* to save. Press the *Exit* link in the upper right corner to go back to the Generator, and then press *Generate* to start code generation. When the code generator finishes, a webpage will pop up with your generated application.

You can see that a sophisticated web application has been generated. You will need to log in using the membership bar at the top of the page. Use standard account *admin/admin123%* to sign in.

	Login to this website	Help	
💕 MyCompany		User Name: admin	
Home		Password: ••••••	15 🔻
Home > Home Start	Remember me next time		
Start		Forgot your password?	
Please select a page link in the table of contents below.	Sign in to access the protected site content.	Sign up now	
Site Map	Instructions	Login	
Home	Two standard user accounts are automaticall membership option has been selected for this		lif
	The administrative account admin is authori: membership manager. The standard user ac web site with the exception of membership m	count is allowed to access all areas of	
	Move the mouse pointer over the link <i>Login to</i> the top of the page and sign in with one of th		it
	Administrative account: admin / admin123%		
	Standard user account: user / user123%		

This will take you to the Home screen. Click on the Appointments tab. Press *New Appointment*, and you will be led to the *New Appointments* page. Here, you can see that a few optimizations still need to be

made. The date is blank, and will only be inserted when the appointment is saved. There is no automatically assigned *Salesman* or *Assigned Salesman* in the respective fields.

		Welcome admin, Today is Wednesday, October 26, 2011 My Accour	t Logout Help
f MyCor	npany		
Home Sales	men Appointments	Membership	ite Actions 🔹
	Home > App	pointments	
About	New Appointments	x	
This page allow appointments management. See Also • Home • Salesmen • Membership	New Appointments	k OK button to create a new appointments record. Click Cancel to return to the previous screen.	ments •

Press *Ok* to save the appointment. This will take you back to list view. Notice that *Salesman* field is not automatically assigned if not selected in the *New Appointment* page.

Return to code generator and click on the *Develop* link next to the name of your project. The project will open in Visual Studio. Using the *Solution Explorer*, navigate to the location *App_Code\Rules\AppointmentBusinessRules.cs(vb)* as shown in the picture.

Solution Explorer 🗸 🗸	>
🖶 🖉 🍋 🖸 🍅	
C:\\Appointments\	
🖕 🗁 App_Code	
👳 🚈 Data	
🚛 🔤 Handlers	
🚊 🗁 Rules	
🐒 AppointmentBusinessRules.cs	
🔤 🚰 Salesmen.Generated.cs	
🗊 🗁 📴 Security	
🖬 🔤 Services	
😥 🖓 🔁 Web	
🗄 🔤 🗹 App_Themes	
🗄 📷 Bin	ľ
🖕 🗁 Controllers	
🔮 Appointments.xml	
😰 aspnet_Membership.xml	
😰 aspnet_Roles.xml	
🛃 DataAquarium.xsd	
🔤 Salesmen.xml	
🗄 🗁 Controls	

This is how you will implement the required functionality:

Create method called *AssignDefaultSalesman* adorned with the *RowBuilder* attribute to handle new rows in *Appointments* data controller. When a user starts entering a new row, then several things will occur. The field *Date* will be updated with the current time, using *UpdateFieldValue*. The *Salesmen* rotation will be requested from the database in ascending order of *AssignedAppointments*. The first *SalesmanID* found will be inserted into the field *AssignedSalesmanID* of the new row. Also, the salesman's *FullName* will be copied into *AssignedSalesmanFullName* of the row.

Create another method named *UpdateAppointmentSalesman* with the *ControllerAction* attribute. The attribute will ensure that the code is executed on the server for new appointments just before the record is inserted in the database. If *SalesmanID* is null, then *AssignedSalesmanID* will be copied into *SalesmanID*.

One more method needs to be created, with the name *UpdateAssignedSalesman*. After an appointment is inserted, *SalesmanID* will be compared to *AssignedSalesmanID*. If the fields have the same value, then that specific *Salesman* record will have its *AssignedAppointments* field increased by one.

The actual code is presented next in C# and VB.NET on the next pages. Copy and paste the code into your own application.

AppointmentBusinessRules.cs

```
using System;
using System.Data;
using System.Collections.Generic;
using System.Ling;
using MyCompany.Data.Objects;
using MyCompany.Data;
namespace MyCompany.Rules
{
    public partial class AppointmentBusinessRules : MyCompany.Data.BusinessRules
    {
        [RowBuilder("Appointments", RowKind.New)]
        public void AssignDefaultSalesman()
        {
            UpdateFieldValue("Date", DateTime.Now);
            using (SqlText findSalesman = new SqlText(
                   "select * from Salesmen order by AssignedAppointments"))
            {
                if (findSalesman.Read())
                {
                    UpdateFieldValue("AssignedSalesmanID", findSalesman["SalesmanID"]);
                    UpdateFieldValue("AssignedSalesmanFullName",
                                      findSalesman["FullName"]);
                }
            }
        }
        [ControllerAction("Appointments", "Insert", ActionPhase.Before)]
        public void UpdateAppointmentSalesman(int assignedSalesmanID,
            FieldValue salesmanID)
        {
            if (salesmanID.Value == null)
            {
                salesmanID.NewValue = assignedSalesmanID;
                salesmanID.Modified = true;
            }
        }
        [ControllerAction("Appointments", "Insert", ActionPhase.After)]
        public void UpdateAssignedSalesman(int assignedSalesmanID, int salesmanID)
        {
            if (assignedSalesmanID == salesmanID)
            {
                Salesmen s = Salesmen.SelectSingle(salesmanID);
                s.AssignedAppointments = s.AssignedAppointments + 1;
                s.Update();
            }
        }
    }
}
```

AppointmentBusinessRules.vb

```
Imports MyCompany.Data
Imports System
Imports System.Collections.Generic
Imports System.Data
Imports System.Linq
Imports MyCompany.Data.Objects
Namespace MyCompany.Rules
    Partial Public Class AppointmentBusinessRules
       Inherits MyCompany.Data.BusinessRules
        <RowBuilder("Appointments", RowKind.New)>
        Public Sub AssignDefaultSalesman()
           UpdateFieldValue("Date", DateTime.Now)
           Using findSalesman As SqlText = New SqlText(
                "select * from Salesmen order by AssignedAppointments")
                If (findSalesman.Read()) Then
                    UpdateFieldValue("AssignedSalesmanID", findSalesman("SalesmanID"))
                    UpdateFieldValue("AssignedSalesmanFullName", findSalesman("FullName"))
               End If
           End Using
        End Sub
        <ControllerAction("Appointments", "Insert", ActionPhase.Before)>
        Public Sub UpdateAppointmentsSalesman(ByVal assignedSalesmanID As Integer,
                                              ByVal salesmanID As FieldValue)
            If salesmanID.Value Is Nothing Then
               salesmanID.NewValue = assignedSalesmanID
                salesmanID.Modified = True
           End If
        End Sub
        <ControllerAction("Appointments", "Insert", ActionPhase.After)>
        Public Sub UpdateAssignedSalesman(ByVal assignedSalesmanID As Integer,
                                          ByVal salesmanID As Integer)
            If (assignedSalesmanID = salesmanID) Then
                Dim s As Salesmen = Salesmen.SelectSingle(salesmanID)
                s.AssignedAppointments = s.AssignedAppointments + 1
                s.Update()
           End If
        End Sub
    End Class
```

End Namespace

Let's see if this code works. If you go back to the *Code OnTime Generator*, and press *Browse* next to the project name, the website will run. Navigate to the *Appointments* tab. If you press *New Appointments* on the action bar, you will be taken to the *New Appointments* page. You can see that date is automatically assigned into the field, and an *Assigned Salesman Full Name* will be automatically assigned, based on the *Assigned Appointments* of each salesman.

		Welcome admin, Today is	Wednesday, October 26, 2011 My Account	Logout Help
MyCom	ipany			
Home Salesm	en Appointments	1embership	Site	e Actions 🔻
	Home > App Appoin	aintments ntments		
About	New Appointments		x	
This page allows appointments	Please fill this form and d	ck OK button to create a new appointments record. Click Cancel to return to the previous screen.	. itn	nents 🔹
management.	New Appointments			
See Also	Complete the form. Make	sure to enter all required fields.		
= Home	Client Name *			
 Salesmen Membership 	Date *	10/26/2011 10:50 AM •		
 Membership 	Assigned Salesman Full Name *	George Fischer 🧷 🚔		
	Salesman Full Name	(select)	ms	
	* - indicates a required fie	ld	OK Cancel	

If you save this appointment, and then create another appointment, the next salesman in line will be automatically inserted. In the picture below, we created a new appointment, and then pressed *New Appointments* again. Notice how the *Assigned Salesman Full Name* has changed to cycle to the next salesman.

		Welcome admin, Today is Wednesday, October 26, 2011 My Account L	.ogout Help
í MyCom	pany		
Home Salesme	Appointments Membership	Site	Actions -
	Home > Appointments Appointments		
About	New Appointments	x	
This page allows appointments	Please fill this form and click OK button to create a new appointments record. Click Cancel	el to return to the previous screen.	ients 🔻
management.	New Appointments		
See Also	Complete the form. Make sure to enter all required fields.		
= Home	Client Name *		
 Salesmen Membership 	Date* 10/26/2011 10:50 AM -		
• Hendership	Assigned Salesman Full Kumar Rajkumar 🥢 🚔		
	Salesman Full Name (select)	ms	
	* - indicates a required field	OK Cancel	

Making Assigned Salesman Full Name a Read-only Field

Field *AssignedSalesmanID* is calculated on the server. It travels to the client, and when the record is sent to the server for insertion into the database, the client will send the field value to the server, along with all the other editable field values, and primary key field values. It is possible that you will want to have field *AssignedSalesmanID* presented as read-only text, so that the rotation of salespeople cannot be affected by users.

If you simply mark the field as read-only in *Designer*, then the client browser will not send the field value to the server, and the calculation of *AssignedSalesmanID* will be lost.

You can overcome this by creating a computed field that uses *AssignedSalesmanFullName* as an alias. Here's how you can do that.

Switch back to the Generator. Click on the project name, and then press *Design*. On the *All Controllers* tab, click on the *Appointments* controller, and switch to the *Fields* tab.

Project Designer												exit
G Home > Controller: Appo	ointmer	nts										
Controller Commands	Fields	Views C	ategories	Data Fields	Action Groups	Action	s					
This is a list of fields.												
Quick Find 🔎 📘	New 🔻	Preview									View: Fi	ields 🔻
Name		New Field Create a nev		s Is Primary Key	Read Only	Hidden	QBE	Sort	LEV	Label	Items Style	Format
AppointmentID		1 Int32	No	Yes	Yes	No	Yes	Yes	No	Appointment#	n/a	n/a
ClientName		2 String	No	No	No	No	Yes	Yes	No	Client Name	n/a	n/a
Date		3 DateTime	No	No	No	No	Yes	Yes	No	Date	n/a	g
AssignedSalesmanID		4 Int32	No	No	No	No	Yes	Yes	No	Assigned Salesman#	Lookup	n/a
AssignedSalesmanFullName		5 String	Yes	No	Yes	No	Yes	Yes	No	Assigned Salesman Full Name	n/a	n/a
SalesmanID		6 Int32	Yes	No	No	No	Yes	Yes	No	Salesman#	Lookup	n/a
SalesmanFullName		7 String	Yes	No	Yes	No	Yes	Yes	No	Salesman Full Name	n/a	n/a
										Showing 1	-7 of 7 items	Refres

On the action bar, press New, then New Field.

Enter the field name as "AssignedSalesmanDisplayFullName", choose "String" as field type. Allow null values.

Check the box next to "The value of this field is computed at run-time". The field *SQL Formula* will appear. Enter text "NULL" as the field value.

In the *Label* field, type "Assigned Salesman". Check the checkbox next to "Values of this field cannot be edited".

Project Designer	exit
G Home > Controller: Appointments	
Controller Commands Fields Views Categories Data Fields Action Groups Please fill this form and dick OK button to create a new field record. Click Cancel to return to the	
* - indicates a required field	OK Cancel
 Specify field name, type, and data properties of the field. Server Default is a SQL expression used as a field value when no value is provided for the field in INSERT and UPDATE statement. Indicate that the field is computed if the field is not physically present in the dataset produced by controller's command. Computed field requires a mandatory formula that must be defined as a valid SQL expression. This expression is automatically inserted in SELECT statements when needed. Calculated field values can be produced by business rule methods with attribute Controller Action. You must list the context fields that will cause the calculation. Optional code formula is is embedded into an automatically created business rule and is calculated whenever any context field is changed. Code Default is an expression written in the programming language of your project. The expression is evaluated in an automatically created business rule to produce a default value for the field of the new row before it is presented in the user interface. Code Value is an expression written in the programming language of your project. This expression is evaluated every time a record is saved to the database. The field must be marked as on-demand if the field is a large binary object (BLOB) or text in order to speed up record retrieval. 	Name * AssignedSalesmanDisplayFullName Type * String Allow null values. Allow null values. The value of this field is computed at run-time by SQL expression. SQL Formula NULL The value of the field is calculated by a business rule expression. Server Default Code Default Code Value Value is retrieved on demand The field is included in all data views but remains hidden from users.
Presentation Presentation interface properties of the field. A data format string is applied to the field value on the dient via JavaScript <i>String, format</i> function. You can overide the data format string on the data field level in views.	Label * Assigned Salesman Values of this field cannot be edited.

Press *Ok* to save. You will be taken back to the grid view of fields.

Click on *AssignedSalesmanDisplayFullName*, the record you just created. Navigate to the *Data Fields* tab (which should be empty), and on the action bar, press *New*, then *New Field*. For *View* lookup, choose "grid1". To use the lookup, click on (*select*), and choose your option from the list.

1	tion				Colur	nns						
This is a list of data controller views.										x		
	Quick Find		P						١	/iew: 📘	Views	-
	Id	Туре	Command	Label	Header Text	Group	Selector	Report Label	Orientation	Font	Templa	ite
	grid 1	Grid	command1	Appointments	<pre>\$DefaultGridViewDescription</pre>	n/a	Yes	n/a	(Auto)	(Auto)) No	
	editForm1	Form	command 1	Review Appointments	<pre>\$DefaultEditViewDescription</pre>	n/a	Yes	n/a	(Auto)	(Auto)) No	
	createForm1	Form	command 1	New Appointments	<pre>\$DefaultCreateViewDescription</pre>	n/a	Yes	n/a	(Auto)	(Auto)	No	
								Sho	wing 1-3 of	3 items	Refre	sh
												_

The *Alias* field should be *AssignedSalesmanFullName*. Use the lookup window to select the correct option.

Project Designer						
Ontroller: Appointments > Field: AssignedSalesmanDisplayFullName (String)						
Field Items Validators Data Fields Field Outputs Please fill this form and click OK button to create a new data field record. Click Cancel to return Field Compared to create a new data field record. Click Cancel to return	to the previous screen.					
	View: New Data Field 🔻					
* - indicates a required field	OK Cancel					
New Data Field Complete the form. Make sure to enter all required fields.	View * grid1 2 Category (select)					
	Alias AssignedSalesmanFullName					

Press *Ok* to save the data field. Again, press *New* on the action bar, then *New Data Field* on the dropdown menu. This time, *View* field will be "editForm1", *Category* field will be "Appointments", and the *Alias* field will be "AssignedSalesmanFullName".

Project Designer exit						
G Home > Controller: Appointments > Field: AssignedSalesmanDisplayFullName (String)						
Field Items Validators Data Fields Field Outputs						
Please fill this form and click OK button to create a new data field record. Click Cancel to return	n to the previous screen.					
	View: New Data Field 🔻					
* - indicates a required field	OK Cancel					
New Data Field	View *					
Complete the form. Make sure to enter all required fields.	editForm1 🖉					
	Category Appointments					
	Alias AssignedSalesmanFullName					

Press the *Ok* button to save the new data field. Create one last data field by pressing *New* on the action bar, then *New Data Field*. This will have *View* field as "createForm1", *Category* field as "New Appointments", and *Alias* field as "AssignedSalesmanFullName".

Project Designer	exit
O Home > Controller: Appointments > Field: AssignedSalesmanDisplayFullName (Strin	g)
Field Items Validators Data Fields Field Outputs	
Please fill this form and click OK button to create a new data field record. Click Cancel to return	to the previous screen.
	View: New Data Field 🝷
* - indicates a required field	OK Cancel
New Data Field	View *
Complete the form. Make sure to enter all required fields.	createForm1 🖉
	Category New Appointments
	Alias AssignedSalesmanFullName

Press Ok to save.

Go back to the fields list by pressing the back icon in the upper left corner. Click on the field *AssignedSalesmanID*. Go to the *Data Fields* tab. Mouse over the *New Appointments* link in the first column, activate the dropdown, and press *Edit*. Change the *Hidden* field to "Yes". Do this for all three data fields, so that the original field *AssignedSalesmanID* will not be displayed in any of the pages.

Project Designer											
Ontroller: Appointments > Field: AssignedSalesmanID (Int32)											
Field Items	Validators	Data Field	s	Field C	outputs						
This is a list of o	data fields.										
Quick Find	P F	Record 🝷									
Category	View	Read Only	Cols	Rows	Data Type	Format	Alias	Hidden	Text Mode	Search	
New Appointments	createForm1	Default	n/a	n/a	Int32	n/a	AssignedSalesmanFullName	Yes	Auto	Default	
Appointments	editForm1	Default	n/a	n/a	Int32	n/a	AssignedSalesmanFullName	Yes	Auto	Default	
n/a	grid 1	Default 🔻			Int32		AssignedSalesmanFullName 🥥		Auto	 Default 	
Save	Car	icel									

Press the *Exit* link in the upper right corner to go back to the Generator. Press *Generate* to initiate generation. When the Generator finishes, a webpage will open with the freshly generated application.

Go to the *Appointments* page. Press *New Appointments* on the action bar. You can see that *Assigned Salesman Full Name* field is no longer editable. If you press *Ok*, the assigned salesman will be transferred into the *Salesman Full Name* field, unless you modified that field.

		Welcome admin, Today is Wednesday, October 26, 2011 My Account Logout Help
MyCon	npany	
Home Saler		te Actions •
	New Appointments	x
	Please fill this form and clid	KOK button to create a new appointments record. Click Cancel to return to the previous screen.
	New Appointments	
About	Complete the form. Make s	ure to enter all required fields.
This page allow appointments	Client Name *	nents 🔻
management.	Date *	10/26/2011 11:16 AM •
See Also	Salesman Full Name	(select)
= Home	Assigned Salesman Full	George Fischer
 Salesmen 	Name	
 Membership 		
	* - indicates a required field	OK Cancel s Refresh
	© 2011 MvC	omoanv. All rights reserved.