7

Programming in VB.NET

Chapter 7 - Programming in VB.NET

DB Visual ARCHITECT (DB-VA) can generate C#.NET source code so you can implement your application by C# programming language directly but you also can choose another language (VB.NET or C++) based on your taste in the .NET Framework. DB-VA generates DLL files and persistent libraries that can be referenced by .NET projects of language other than C#.

In this chapter:

- Introduction
- Generating DLL file
- Creating VB.NET Project
- Adding Referenced Project
- Working with the Generated Code and Persistent Library
- Running the Application

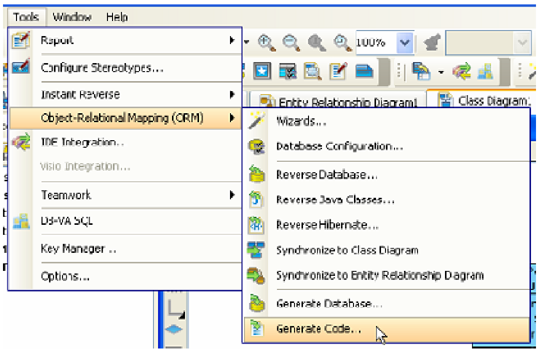
Introduction

Visual Basic .NET (VB.NET) is an Object-Oriented computer language that can be viewed as an evolution of Microsoft’s Visual Basic (VB) implemented on the Microsoft .NET framework. The .NET Framework contains a virtual machine called Common Intermediate Language (CIL). Simply put, programs are compiled to produce CIL and the CIL is distributed to user to run on a virtual machine. VB.NET, C++, C# compilers are available from Microsoft for creating CIL. In DB-VA you can generate C# persistent source code and DLL files, so you can reference the DLL file and Persistent Library in Visual Studio .NET 2003 and develop the VB.NET application.

In the following section, you will develop a VB.NET application. The application is exactly the same as the one in Chapter 4 – Developing Standalone .NET Application sample but this time you use VB.NET instead of C# for development. You need to download Chapter 4 sample application because it contains the DLL file and persistent libraries for your VB.NET project.

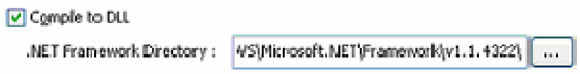
Generating DLL File

1. From the menu bar, select **Tools** > **Object-Relational Mapping (ORM)** > **Generate Database…** to open the Database Code Generation dialog box.



2. Check the **Compile to DLL** option to generate the DLL file.
   - **Compile to DLL**

     By checking this option, DB-VA will generate DLL files which can be referenced by .NET projects of language other than C#. DB-VA generates a batch file for this DLL file at the same time. You can modify the configure file (hibernate.cfg.xml) manually and use the batch file to recompile and build an up-to-date DLL file for referenced project.

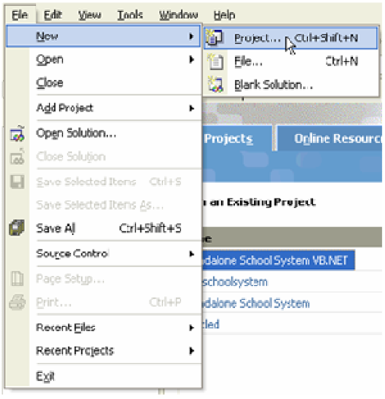


☑ Compile to DLL

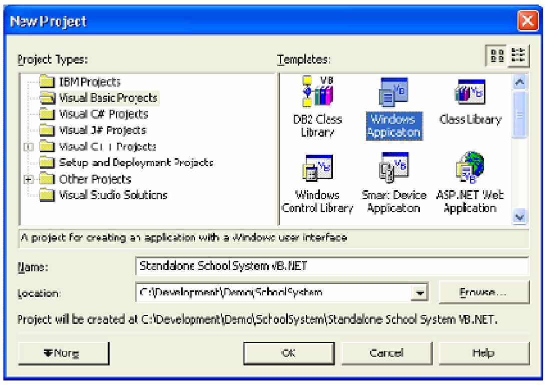
.NET Framework Directory : VS\Microsoft.NET\Framework\v1.1.4322\ ...

Creating VB.NET Project

1. Open Microsoft Visual Studio .NET 2003.
2. Select **File** > **New** > **Project …** from the menu.



3. Select the Project Types as Visual Basic Projects and Templates as Windows Application and enter the Name and Location for the new application.



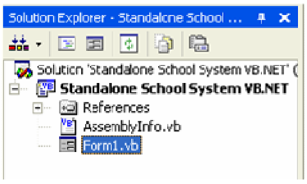
A project for creating an application with a Windows user interface

Name: Standalone School System VB.NET

Location: C:\Development\Demo\SchoolSystem

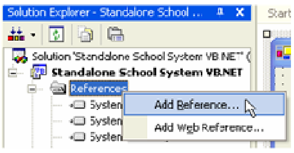
Project will be created at C:\Development\Demo\SchoolSystem\Standalone School System VB.NET.

4. The School System Project is created.

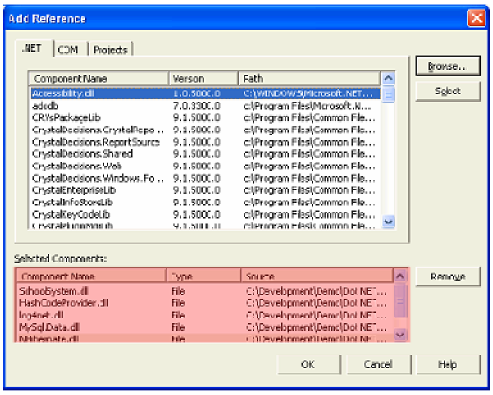

Solution Explorer - Standalone School ...
Solution 'Standalone School System VB.NET' (
Standalone School System VB.NET
References
AssemblyInfo.vb
Form1.vb

Adding Referenced Project

1. Right click References under the Standalone School System VB.NET project and select **Add Reference…** to add the C# example DDL file and persistent libraries.

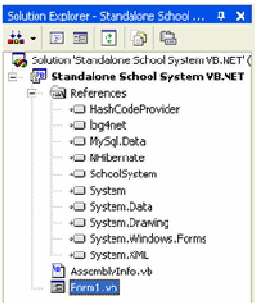

Solution Explorer - Standalone School ...
Start
Solution 'Standalone School System VB.NET' (
Standalone School System VB.NET
References
System
System
System
Add Reference...
Add Web Reference...

2. Click **Browse…** on the Add Reference dialog box to select the folder of the downloaded C# standalone application sample. Select **C# sample folder/bin/SchoolSystem.dll** and **all libraries in C# sample folder/lib**.



Component Name	Version	Path
Accessibility.dll	1.0.5000.0	C:\WINDOWS\Microsoft.NET...
adodb	7.0.3300.0	c:\Program Files\Microsoft.N...
CRVsPackageLib	9.1.5000.0	c:\Program Files\Common File...
CrystalDecisions.CrystalRepo...	9.1.5000.0	c:\Program Files\Common File...
CrystalDecisions.ReportSource	9.1.5000.0	c:\Program Files\Common File...
CrystalDecisions.Shared	9.1.5000.0	c:\Program Files\Common File...
CrystalDecisions.Web	9.1.5000.0	c:\Program Files\Common File...
CrystalDecisions.Windows.Fo...	9.1.5000.0	c:\Program Files\Common File...
CrystalEnterpriseLib	9.1.5000.0	c:\Program Files\Common File...
CrystalInfoStoreLib	9.1.5000.0	c:\Program Files\Common File...
CrystalKeyCodeLib	9.1.5000.0	c:\Program Files\Common File...

Component Name	Type	Source
SchoolSystem.dll	File	C:\Development\Demo\Dot NE...
HashCodeProvider.dll	File	C:\Development\Demo\Dot NE...
log4net.dll	File	C:\Development\Demo\Dot NE...
MySql.Data.dll	File	C:\Development\Demo\Dot NE...
NHibernate.dll	File	C:\Development\Demo\Dot NE...



Solution 'Standalone School System VB.NET' (
- Standalone School System VB.NET
  - References
    - HashCodeProvider
    - log4net
    - MySql.Data
    - NHibernate
    - SchoolSystem
    - System
    - System.Data
    - System.Drawing
    - System.Windows.Forms
    - System.XML
  - AssemblyInfo.vb
  - Form1.vb

Working with the Generated Code and Persistent Library

Both the C# and VB.NET languages are built on the .NET framework. Both languages accomplish the same thing using different language syntax. In this section, we will point out some examples of using the generated code and persistent library.

- Importing DLL and Persistent Libraries namespace for the Class.

VB.NET	C#

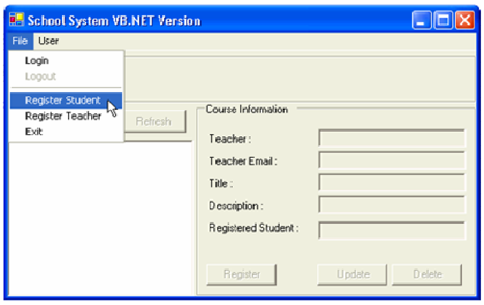
`Imports school`
`Imports Orm`
	`using school;`
	`using Orm;`

- Creating PersistentTransaction when manipulating the database.

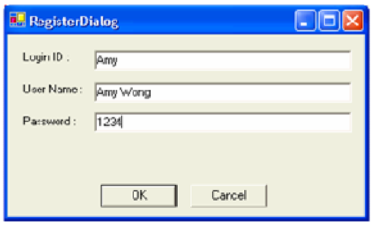
VB.NET	C#
`Dim t As PersistentTransaction = SchoolSystemPersistentManager.Instance.GetSession.BeginTransaction`	`PersistentTransaction t = SchoolSystemPersistentManager.Instance().GetSession().BeginTransaction();`

- Creating Object and Save to Database

1. From menu bar, select **File** > **Register Student** to open the Add Student dialog box.



2. Fill in the student information. Click OK to create the new Student record in School System.



Login ID : Amy
User Name : Amy Wong
Password : 1234

3. After click OK, the system create the new Student persistent object.

```
Private Sub okButton_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles okButton.Click
    If (loginIDTextBox.Text.Length = 0 Or
passwordTextBox.Text.Length = 0) Then
        MessageBox.Show("Login id or password missing")
        Return
    End If
    Dim t As PersistentTransaction =
SchoolSystemPersistentManager.Instance.GetSession.BeginTransaction
    Try
        If (_userType = CREATE_TEACHER) Then
            User = TeacherFactory.CreateTeacher()
        ElseIf (_userType = CREATE_STUDENT) Then
            User = StudentFactory.CreateStudent()
        End If
```

Source File : Standalone School System VB.NET\RegisterDialog.vb

4. Set the student information from the text fields value to the Student object.

```
User.Password = passwordTextBox.Text
User.LoginID = loginIDTextBox.Text
If (TypeOf User Is Teacher) Then
    CType(User, Teacher).Email = emailTextBox.Text
End If
```

Source File : Standalone School System VB.NET\RegisterDialog.vb

5. Call Save() method of Student Persistent Object and Commit() method of PersistentTransaction., then the new Student object will be saved to the database. If any error occurred during the transaction, you can call the Rollback() method to cancel the proposed changes in a pending database transaction

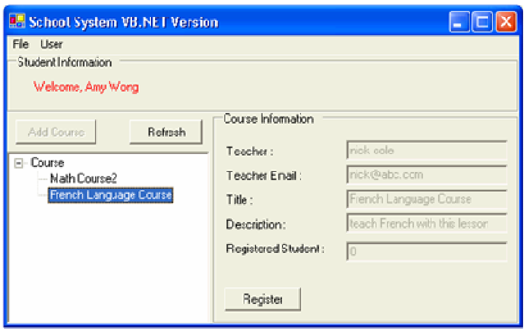
```
    User.Save()
    t.Commit()
    DialogResult = DialogResult.OK
    Close()
Catch ex As Exception
    Console.WriteLine(ex.InnerException.Message)
    DialogResult = DialogResult.Cancel
    t.RollBack()
End Try
```

Source File : Standalone School System VB.NET\RegisterDialog.vb

- Query Object from Database

After the user login the School System, the system query different Course objects from the database according to user role. If the user is a student, the system shows all the courses. The student can select and register the course. If the user is a teacher, the system shows courses are created by that teacher. The teacher can update or delete the course in the system.

Student Login:



Welcome, Amy Wong

Teacher : nick cole

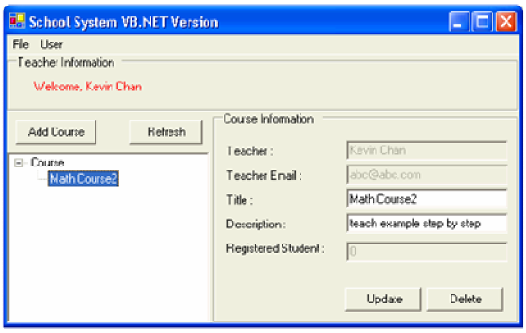
Teacher Email : nick@abc.com

Title : French Language Course

Description: teach French with this lessor

Registered Student : 0

Teacher Login:



Welcome, Kevin Chan

Teacher : Kevin Chan

Teacher Email : abc@abc.com

Title : Math Course2

Description: teach example step by step

Registered Student : 0

1. Query the course objects when user login. When Student login, the system will call **ListCourseByQuery()** method in **CourseFactory** to get all available courses. When Teacher login, the system will call **courses** collection variable in Teacher object.

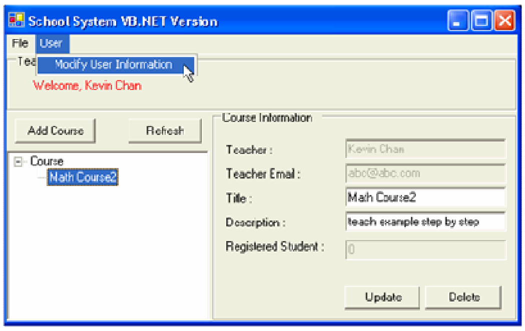
```
Private Sub UpdateTreeView()
    Dim lCourses() As Course
    If (TypeOf CurrentUser Is Student) Then
        lCourses = CourseFactory.ListCourseByQuery(Nothing, Nothing)
    ElseIf (TypeOf CurrentUser Is Teacher) Then
        lCourses = CType(CurrentUser, Teacher).courses.ToArray()
    End If
    ...
End Sub
```

Source File : Standalone School System VB.NET \SchoolSystemForm.vb

- Update Object and Save to Database

You can modify the user information and update the record in database. After the user login, the User object is stored in the application, so you can set new information to the user object and update the database record.

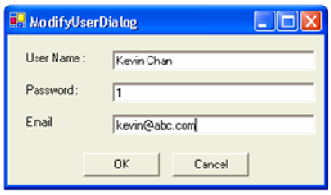
1. From the menu bar, select **User** > **Modify User Information** to open the Modify User Information dialog box.



Welcome, Kevin Chan

Teacher : Kevin Chan
Teacher Email : abc@abc.com
Title : Math Course2
Description : teach example step by step
Registered Student : 0

2. Enter new user information and click **OK** to update the User record.



User Name : Kevin Chan
Password : 1
Email : kevin@abc.com

3. Update the information for the User object by setting password, name and email address.

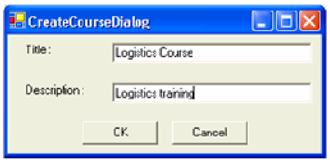
```
Private Sub okButton_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles okButton.Click
    If (nameTextBox.Text.Length = 0 Or passwordTextBox.Text.Length =
0) Then
        MessageBox.Show("Missing login id and password")
    Else
        Dim t As PersistentTransaction =
SchoolSystemPersistentManager.Instance.GetSession.BeginTransaction
        Try
            _user.Name = nameTextBox.Text
            _user.Password = passwordTextBox.Text
            If (TypeOf _user Is Teacher) Then
                CType(_user, Teacher).Email = emailTextBox.Text
            End If
            _user.Save()
            DialogResult = DialogResult.OK
            t.Commit()
        Catch ex As Exception
            DialogResult = DialogResult.Cancel
            t.RollBack()
        End Try
    End If
    Close()
End Sub
```

Source File : Standalone School System VB.NET\ModifyUserDialog.cs

- Delete Object in Database

Teacher can create course for students to register, and they can cancel courses in the system. They only need to click delete button of the course then the course information will be deleted in the database and all its relationships with registered students will be removed.

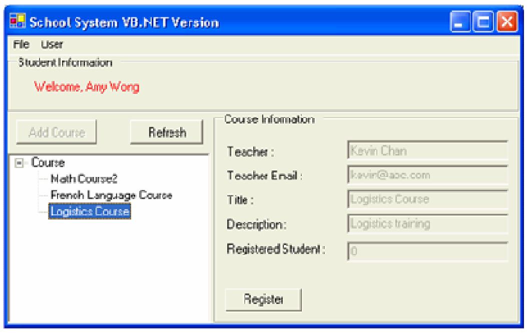
1. Teacher can create the course by clicking the **Add Course** button, fill in Course name and Description.



Title: Logistics Course

Description: Logistics training

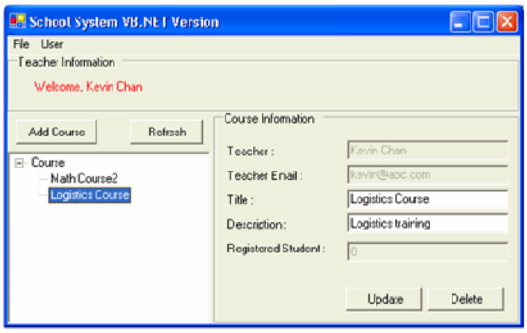
2. Student can register the course by clicking the **Register** button.



Welcome, Amy Wong

Teacher : Kevin Chan
Teacher Email : kevin@abc.com
Title : Logistics Course
Description: Logistics training
Registered Student : 0

3. The teacher can view how many students registered his course, and he can delete courses in the system.



Welcome, Kevin Chan

Teacher : Kevin Chan
Teacher Email : kevin@abc.com
Title : Logistics Course
Description: Logistics training
Registered Student : 0

4. Click Delete of a course then it will trigger the **delButton_Click()** method .

```
Private Sub deleteButton_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles deleteButton.Click
    If (MessageBox.Show("Delete", "Delete",
MessageBoxButtons.OKCancel).Equals(DialogResult.OK)) Then
```

Source File : Standalone School System VB.NET\CoursePanel.vb

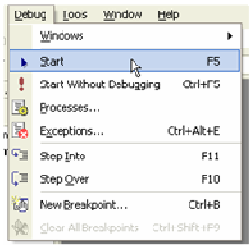
5. Call **deleteAndDissociate()** method to delete the course object and remove the relationships of students and teachers with it.

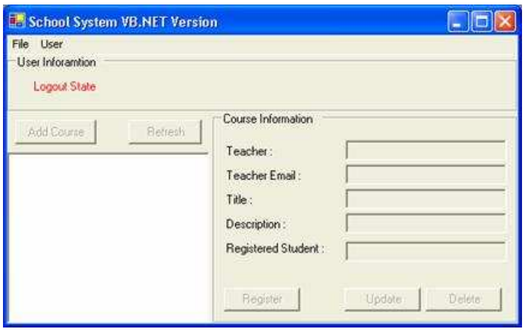
```
        Dim t As PersistentTransaction =
SchoolSystemPersistentManager.Instance().GetSession().BeginTransactio
n()
        Try
            _CourseNode.Course().DeleteAndDissociate()
            _CourseNode.Remove()
            t.Commit()
        Catch ex As Exception
            Console.WriteLine(ex.InnerException.Message)
            t.RollBack()
        End Try
    End If
End Sub
```

Source File : Standalone School System VB.NET\CoursePanel.vb

Running Application

The VB.NET project is implemented in Visual Studio .NET 2003. To execute the VB.NET Application, select **Debug** > **Start (F5)** from the menu bar.


Debug
Tools
Window
Help
Windows
Start F5
Start Without Debugging Ctrl+F5
Processes...
Exceptions... Ctrl+Alt+E
Step Into F11
Step Over F10
New Breakpoint... Ctrl+B
Clear All Breakpoints Ctrl+Shift+F9


School System VB.NET Version
File
User
User Inforamtion
Logout State
Add Course
Refresh
Course Information
Teacher :
Teacher Email :
Title :
Description :
Registered Student :
Register
Update
Delete