

www.bytescout.com

How to generate barcode in crystal reports application c# 2015 in Crystal Reports using ByteScout Barcode SDK

How to generate barcode in crystal reports application c# 2015 in Crystal Reports

On this page you will learn from code samples for programming in Crystal Reports. Writing of the code to generate barcode in crystal reports application c# 2015 in Crystal Reports can be done by developers of any level using ByteScout Barcode SDK. ByteScout Barcode SDK is the robust SDK that generates high quality barcode images and pdf. Can generate all popular types of barcodes from QR Code, Code 39, Code 128, UPC, GS1, GS-128, PDF417, Datamatrix to more exotic barcode types. Fully customizable fonts, colors, print sizes. Includes special functions to ensure output quality, and tools for adding barcodes to new or existing pdf files and images. It can generate barcode in crystal reports application c# 2015 in Crystal Reports.

Crystal Reports code samples for Crystal Reports developers help to speed up coding of your application when using ByteScout Barcode SDK. Follow the instructions from the scratch to work and copy the Crystal Reports code. This basic programming language sample code for Crystal Reports will do the whole work for you to generate barcode in crystal reports application c# 2015.

ByteScout free trial version is available for download from our website. It includes all these programming tutorials along with source code samples.

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout Barcode SDK](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout Barcode SDK](#)

[Get Free API key for Web API](#)

[visit www.ByteScout.com](#)

Source Code Files:

BarcodeInCrystalReports.sln

```
Microsoft Visual Studio Solution File, Format Version 12.00
# Visual Studio 14
VisualStudioVersion = 14.0.25420.1
MinimumVisualStudioVersion = 10.0.40219.1
Project("{FAE04EC0-301F-11D3-BF4B-00C04F79EFBC}") = "BarcodeInCrystalReports", "Barcode
EndProject
Global
    GlobalSection(SolutionConfigurationPlatforms) = preSolution
        Debug|Any CPU = Debug|Any CPU
        Debug|x86 = Debug|x86
        Release|Any CPU = Release|Any CPU
        Release|x86 = Release|x86
    EndGlobalSection
    GlobalSection(ProjectConfigurationPlatforms) = postSolution
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Debug|Any CPU.ActiveCfg = Debug
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Debug|Any CPU.Build.0 = Debug|Any
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Debug|x86.ActiveCfg = Debug|x86
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Debug|x86.Build.0 = Debug|x86
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Release|Any CPU.ActiveCfg = Release
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Release|Any CPU.Build.0 = Release
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Release|x86.ActiveCfg = Release
        {75AA680D-9434-4C36-AB16-29EB20693F0B}.Release|x86.Build.0 = Release|x86
    EndGlobalSection
    GlobalSection(SolutionProperties) = preSolution
        HideSolutionNode = FALSE
    EndGlobalSection
EndGlobal
```

CrystalReport1.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//     This code was generated by a tool.
//     Runtime Version:4.0.30319.34014
//
//     Changes to this file may cause incorrect behavior and will be lost if
//     the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

namespace BarcodeInCrystalReports {
    using System;
```

```
using System.ComponentModel;
using CrystalDecisions.Shared;
using CrystalDecisions.ReportSource;
using CrystalDecisions.CrystalReports.Engine;

public class CrystalReport1 : ReportClass {

    public CrystalReport1() {
    }

    public override string ResourceManagerName {
        get {
            return "CrystalReport1.rpt";
        }
        set {
            // Do nothing
        }
    }

    public override bool NewGenerator {
        get {
            return true;
        }
        set {
            // Do nothing
        }
    }

    public override string FullResourceManagerName {
        get {
            return "BarcodeInCrystalReports.CrystalReport1.rpt";
        }
        set {
            // Do nothing
        }
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali
    public CrystalDecisions.CrystalReports.Engine.Section Section1 {
        get {
            return this.ReportDefinition.Sections[0];
        }
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali
    public CrystalDecisions.CrystalReports.Engine.Section Section2 {
        get {
            return this.ReportDefinition.Sections[1];
        }
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali
    public CrystalDecisions.CrystalReports.Engine.Section Section3 {
        get {
            return this.ReportDefinition.Sections[2];
        }
    }
}
```

```
}

[Browsable(false)]
[DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali...
public CrystalDecisions.CrystalReports.Engine.Section Section4 {
    get {
        return this.ReportDefinition.Sections[3];
    }
}

[Browsable(false)]
[DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali...
public CrystalDecisions.CrystalReports.Engine.Section Section5 {
    get {
        return this.ReportDefinition.Sections[4];
    }
}
}

[System.Drawing.ToolboxBitmapAttribute(typeof(CrystalDecisions.Shared.ExportOptions...
public class CachedCrystalReport1 : Component, ICachedReport {

    public CachedCrystalReport1() {
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali...
    public virtual bool IsCacheable {
        get {
            return true;
        }
        set {
            //
        }
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali...
    public virtual bool ShareDBLogonInfo {
        get {
            return false;
        }
        set {
            //
        }
    }

    [Browsable(false)]
    [DesignerSerializationVisibilityAttribute(System.ComponentModel.DesignerSeriali...
    public virtual System.TimeSpan CacheTimeOut {
        get {
            return CachedReportConstants.DEFAULT_TIMEOUT;
        }
        set {
            //
        }
    }

    public virtual CrystalDecisions.CrystalReports.Engine.ReportDocument CreateRepo...
        CrystalReport1 rpt = new CrystalReport1();
```

```

        rpt.Site = this.Site;
        return rpt;
    }

    public virtual string GetCustomizedCacheKey(RequestContext request) {
        String key = null;
        // // The following is the code used to generate the default
        // // cache key for caching report jobs in the ASP.NET Cache.
        // // Feel free to modify this code to suit your needs.
        // // Returning key == null causes the default cache key to
        // // be generated.
        //
        // key = RequestContext.BuildCompleteCacheKey(
        //     request,
        //     null,          // sReportFilename
        //     this.GetType(),
        //     this.ShareDBLogonInfo );
        return key;
    }
}
}

```

Form1.Designer.cs

```

namespace BarcodeInCrystalReports
{
    partial class Form1
    {
        private System.ComponentModel.IContainer components = null;

        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        private void InitializeComponent()
        {
            this.crystalReportViewer1 = new CrystalDecisions.Windows.Forms.
            this.CrystalReport11 = new BarcodeInCrystalReports.CrystalReport11();
            this.SuspendLayout();
            //
            // crystalReportViewer1
            //
            this.crystalReportViewer1.ActiveViewIndex = 0;
        }
    }
}

```

```

        this.crystalReportViewer1.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
        this.crystalReportViewer1.Cursor = System.Windows.Forms.Cursors.Default;
        this.crystalReportViewer1.Dock = System.Windows.Forms.DockStyle.Fill;
        this.crystalReportViewer1.Location = new System.Drawing.Point(0, 0);
        this.crystalReportViewer1.Name = "crystalReportViewer1";
        this.crystalReportViewer1.ReportSource = this.CrystalReport11;
        this.crystalReportViewer1.Size = new System.Drawing.Size(799, 566);
        this.crystalReportViewer1.TabIndex = 0;
        this.crystalReportViewer1.Load += new System.EventHandler(this.crystalReportViewer1_Load);
        //
        // Form1
        //
        this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(799, 566);
        this.Controls.Add(this.crystalReportViewer1);
        this.Name = "Form1";
        this.Text = "Form1";
        this.ResumeLayout(false);

    }

}

#endregion

private CrystalDecisions.Windows.Forms.CrystalReportViewer crystalReportViewer1;
private CrystalReport1 CrystalReport11;
}
}

```

Form1.cs

```

using System;
using System.Data;
using System.Data.OleDb;
using System.Windows.Forms;
using Bytescout.BarCode;

namespace BarcodeInCrystalReports
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void crystalReportViewer1_Load(object sender, EventArgs e)
        {
            try

```

```
        OleDbConnection connection = new OleDbConnection("Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\Users\Public\Documents\Crystal Reports\Barcode.mdb");
        OleDbDataAdapter dataAdapter = new OleDbDataAdapter("SELECT * FROM [Table1]", connection);
        DataSet dataSet = new DataSet();
        dataAdapter.Fill(dataSet);
        connection.Close();

        // add virtual column into the result table
        dataSet.Tables[0].Columns.Add(new DataColumn("BarcodeImage"));

        // create barcode object
        Barcode barcode = new Barcode(SymbologyType.Code128);
        barcode.DrawCaption = false;

        // Fill BarcodeImage column with generated barcode image
        foreach (DataRow row in dataSet.Tables[0].Rows)
        {
            // set barcode value
            barcode.Value = Convert.ToString(row["ID"]);

            // retrieve generated image bytes
            byte[] barcodeBytes = barcode.GetImageBytesWMF();
            row["BarcodeImage"] = barcodeBytes;
        }

        // set filled DataSet as report's data source
        CrystalReport11.SetDataSource(dataSet.Tables[0]);
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: " + ex.Message);
    }
}
```

ProductsDataSet.xsc

```
<TableUISettings />
</DataSetUISetting>
```

ProductsDataSet.xsd

```
<?xml version="1.0" encoding="utf-8"?>
<xss: schema id="ProductsDataSet" targetNamespace="http://tempuri.org/ProductsDataSet.xsd" xmlns:xss="http://www.w3.org/2001/XMLSchema">
  <xss:annotation>
    <xss:appinfo source="urn:schemas-microsoft-com:xml-msdatasource">
      <DataSource DefaultConnectionIndex="0" FunctionsComponentName="QueriesTableAdapter" />
      <Connections>
        <Connection AppSettingsObjectName="Settings" AppSettingsPropertyName="productsConnectionString" />
      </Connections>
      <Tables>
        <TableAdapter BaseClass="System.ComponentModel.Component" DataAccessorModified="false" />
        <MainSource>
          <DbSource ConnectionRef="productsConnectionString (Settings)" DbObjectName="Products" />
          <DeleteCommand>
            <DbCommand CommandType="Text" ModifiedByUser="false">
              <CommandText>DELETE FROM `Products` WHERE ((`ID` = ?) AND ((? = 1 AND `Discontinued` = 0) OR (? = 0 AND `Discontinued` = 1)))</CommandText>
              <Parameters>
                <Parameter AllowDBNull="false" AutogeneratedName="" DataSourceName="Products" />
                <Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Products" />
              </Parameters>
            </DbCommand>
          </DeleteCommand>
          <InsertCommand>
            <DbCommand CommandType="Text" ModifiedByUser="false">
              <CommandText>INSERT INTO `Products` (`ProductName`, `ProductDescription`, `UnitPrice`, `UnitsInStock`, `Discontinued`) VALUES (?, ?, ?, ?, ?)</CommandText>
              <Parameters>
                <Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Products" />
                <Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Products" />
              </Parameters>
            </DbCommand>
          </InsertCommand>
          <SelectCommand>
            <DbCommand CommandType="Text" ModifiedByUser="false">
              <CommandText>SELECT ID, ProductName, ProductDescription, ProductPrice, UnitsInStock, UnitsPerOrder, Discontinued FROM `Products`</CommandText>
              <Parameters />
            </DbCommand>
          </SelectCommand>
          <UpdateCommand>
            <DbCommand CommandType="Text" ModifiedByUser="false">
              <CommandText>UPDATE `Products` SET `ProductName` = ?, `ProductDescription` = ?, `UnitPrice` = ?, `UnitsInStock` = ?, `UnitsPerOrder` = ?, `Discontinued` = ? WHERE ((`ID` = ?) AND ((? = 1 AND `Discontinued` = 0) OR (? = 0 AND `Discontinued` = 1)))</CommandText>
              <Parameters>
                <Parameter AllowDBNull="false" AutogeneratedName="" DataSourceName="Products" />
                <Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Products" />
              </Parameters>
            </DbCommand>
          </UpdateCommand>
        </MainSource>
      </Tables>
    </xss:appinfo>
  </xss:annotation>
</xss: schema>
```

```
<Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Northwind" />
<Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Northwind" />
<Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Northwind" />
<Parameter AllowDBNull="false" AutogeneratedName="" DataSourceName="Northwind" />
<Parameter AllowDBNull="true" AutogeneratedName="" DataSourceName="Northwind" />
<Parameters>
</Parameters>
</DbCommand>
</UpdateCommand>
</DbSource>
</MainSource>
<Mappings>
<Mapping SourceColumn="ID" DataSetColumn="ID" />
<Mapping SourceColumn="ProductName" DataSetColumn="ProductName" />
<Mapping SourceColumn="ProductDescription" DataSetColumn="ProductDescription" />
<Mapping SourceColumn="ProductPrice" DataSetColumn="ProductPrice" />
</Mappings>
<Sources />
</TableAdapter>
</Tables>
<Sources />
</DataSource>
</xs:appinfo>
</xs:annotation>
<xs:element name="ProductsDataSet" msdata:IsDataSet="true" msdata:UseCurrentLocale="true" type="msdata:DataSet">
<xs:complexType>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element name="Products" msprop:Generator_TableClassName="ProductsDataTable" msprop:Generator_RowClassName="ProductsRow" msprop:Generator_ColumnVarNameInTable="co">
<xs:complexType>
<xs:sequence>
<xs:element name="ID" msdata:AutoIncrement="true" msdata:AutoIncrementSeed="1" msdata:AutoIncrementStep="1" msprop:Generator_ColumnVarNameInTable="co" type="xs:int" />
<xs:element name="ProductName" msprop:Generator_ColumnVarNameInTable="co" type="xs:string" />
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255" />
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ProductDescription" msprop:Generator_ColumnVarNameInTable="co" type="xs:string" />
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255" />
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ProductPrice" msprop:Generator_ColumnVarNameInTable="co" type="xs:decimal" />
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:maxLength value="255" />
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="BarcodeImage" msprop:Generator_ColumnVarNameInTable="co" type="xs:base64Binary" />
</xs:sequence>
</xs:complexType>
</xs:element>

```

```
</xs:choice>
</xs:complexType>
<xs:unique name="Constraint1" msdata:PrimaryKey="true">
  <xs:selector xpath=".//mstns:Products" />
  <xs:field xpath="mstns:ID" />
</xs:unique>
</xs:element>
</xs:schema>
```

ProductsDataSet.xss

```
<?xml version="1.0" encoding="utf-8"?>
<!!--<autogenerated>
  This code was generated by a tool to store the dataset designer's layout information.
  Changes to this file may cause incorrect behavior and will be lost if
  the code is regenerated.
</autogenerated>-->
<DiagramLayout xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Shapes>
    <Shape ID="DesignTable:Products" ZOrder="1" X="70" Y="70" Height="172" Width="196" />
  </Shapes>
  <Connectors />
</DiagramLayout>
```

ProductsDataSet1.Designer.cs

```
//------------------------------------------------------------------------------
// <auto-generated>
//   This code was generated by a tool.
//   Runtime Version:4.0.30319.34014
//
//   Changes to this file may cause incorrect behavior and will be lost if
//   the code is regenerated.
// </auto-generated>
//------------------------------------------------------------------------------

#pragma warning disable 1591

namespace BarcodeInCrystalReports {

  /// <summary>
```

```
///Represents a strongly typed in-memory cache of data.
///</summary>
[global::System.Serializable()]
[global::System.ComponentModel.DesignerCategoryAttribute("code")]
[global::System.ComponentModel.ToolboxItem(true)]
[global::System.Xml.Serialization.XmlSchemaProviderAttribute("GetTypedDataSetSchema")]
[global::System.Xml.Serialization.XmlRootAttribute("ProductsDataSet")]
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.DataSet")]
public partial class ProductsDataSet : global::System.Data.DataSet {

    private ProductsDataTable tableProducts;

    private global::System.Data.SchemaSerializationMode _schemaSerializationMode =
        [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
        [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.TypedDataSetSchemaGenerator", "1.0.0.0")]
    public ProductsDataSet() {
        this.BeginInit();
        this.InitClass();
        global::System.ComponentModel.CollectionChangeEventHandler schemaChangedHandler =
            delegate { base.Tables.CollectionChanged += schemaChangedHandler; };
        base.Tables.CollectionChanged += schemaChangedHandler;
        base.Relations.CollectionChanged += schemaChangedHandler;
        this.EndInit();
    }

    [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.TypedDataSetSchemaGenerator", "1.0.0.0")]
    protected ProductsDataSet(global::System.Runtime.Serialization.SerializationInfo info, global::System.IFormatterContext context, bool fFromBinary) {
        if ((this.IsBinarySerialized(info, context) == true)) {
            this.InitVars(false);
            global::System.ComponentModel.CollectionChangeEventHandler schemaChangedHandler1 =
                delegate { this.Tables.CollectionChanged += schemaChangedHandler1; };
            this.Relations.CollectionChanged += schemaChangedHandler1;
            return;
        }
        string strSchema = ((string)(info.GetValue("XmlSchema", typeof(string))));
        if ((this.DetermineSchemaSerializationMode(info, context) == global::System.Data.SchemaSerializationMode.IncludeSchema)) {
            global::System.Data.DataSet ds = new global::System.Data.DataSet();
            ds.ReadXmlSchema(new global::System.Xml.XmlTextReader(new global::System.IO.StringReader(strSchema)));
            if ((ds.Tables["Products"] != null)) {
                base.Tables.Add(new ProductsDataTable(ds.Tables["Products"]));
            }
            this.DataSetName = ds.DataSetName;
            this.Prefix = ds.Prefix;
            this.Namespace = ds.Namespace;
            this.Locale = ds.Locale;
            this.CaseSensitive = ds.CaseSensitive;
            this.EnforceConstraints = ds.EnforceConstraints;
            this.Merge(ds, false, global::System.Data.MissingSchemaAction.Add);
            this.InitVars();
        }
        else {
            this.ReadXmlSchema(new global::System.Xml.XmlTextReader(new global::System.IO.StringReader(strSchema)));
        }
        this.GetSerializationData(info, context);
        global::System.ComponentModel.CollectionChangeEventHandler schemaChangedHandler =
            delegate { base.Tables.CollectionChanged += schemaChangedHandler; };
        this.Relations.CollectionChanged += schemaChangedHandler;
    }

}
```

```
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
[global::System.ComponentModel.Browsable(false)]
[global::System.ComponentModel.DesignerSerializationVisibility(global::System.C
public ProductsDataTable Products {
    get {
        return this.tableProducts;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
[global::System.ComponentModel.BrowsableAttribute(true)]
[global::System.ComponentModel.DesignerSerializationVisibilityAttribute(global
public override global::System.Data.SchemaSerializationMode SchemaSerialization
    get {
        return this._schemaSerializationMode;
    }
    set {
        this._schemaSerializationMode = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
[global::System.ComponentModel.DesignerSerializationVisibilityAttribute(global
public new global::System.Data.DataTableCollection Tables {
    get {
        return base.Tables;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
[global::System.ComponentModel.DesignerSerializationVisibilityAttribute(global
public new global::System.Data.DataRelationCollection Relations {
    get {
        return base.Relations;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
protected override void InitializeDerivedDataSet() {
    this.BeginInit();
    this.InitClass();
    this.EndInit();
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
public override global::System.Data.DataSet Clone() {
    ProductsDataSet cln = ((ProductsDataSet)(base.Clone()));
    cln.InitVars();
    cln.SchemaSerializationMode = this.SchemaSerializationMode;
    return cln;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
```

```
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
protected override bool ShouldSerializeTables() {
    return false;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
protected override bool ShouldSerializeRelations() {
    return false;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
protected override void ReadXmlSerializable(global::System.Xml.XmlReader reader) {
    if ((this.DetermineSchemaSerializationMode(reader) == global::System.Data.SchemaSerializationMode.IncludeSchema))
        this.Reset();
    global::System.Data.DataSet ds = new global::System.Data.DataSet();
    ds.ReadXml(reader);
    if ((ds.Tables["Products"] != null)) {
        base.Tables.Add(new ProductsDataTable(ds.Tables["Products"]));
    }
    this.DataSetName = ds.DataSetName;
    this.Prefix = ds.Prefix;
    this.Namespace = ds.Namespace;
    this.Locale = ds.Locale;
    this.CaseSensitive = ds.CaseSensitive;
    this.EnforceConstraints = ds.EnforceConstraints;
    this.Merge(ds, false, global::System.Data.MissingSchemaAction.Add);
    this.InitVars();
}
else {
    this.ReadXml(reader);
    this.InitVars();
}
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
protected override global::System.Xml.Schema.XmlSchema GetSchemaSerializable()
{
    global::System.IO.MemoryStream stream = new global::System.IO.MemoryStream();
    this.WriteXmlSchema(new global::System.Xml.XmlTextWriter(stream, null));
    stream.Position = 0;
    return global::System.Xml.Schema.XmlSchema.Read(new global::System.Xml.XmlTextReader(stream));
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
internal void InitVars()
{
    this.InitVars(true);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
internal void InitVars(bool initTable)
{
    this.tableProducts = ((ProductsDataTable)(base.Tables["Products"]));
    if ((initTable == true)) {
        if ((this.tableProducts != null)) {
            this.tableProducts.InitVars();
        }
    }
}
```

```
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
private void InitClass() {
    this.DataSetName = "ProductsDataSet";
    this.Prefix = "";
    this.Namespace = "http://tempuri.org/ProductsDataSet.xsd";
    this.EnforceConstraints = true;
    this.SchemaSerializationMode = global::System.Data.SchemaSerializationMode
    this.tableProducts = new ProductsDataTable();
    base.Tables.Add(this.tableProducts);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
private bool ShouldSerializeProducts() {
    return false;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
private void SchemaChanged(object sender, global::System.ComponentModel.Collection
    if ((e.Action == global::System.ComponentModel.CollectionChangeAction.Remov
        this.InitVars();
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
public static global::System.Xml.Schema.XmlSchemaComplexType GetTypedDataSetSche
    ProductsDataSet ds = new ProductsDataSet();
    global::System.Xml.Schema.XmlSchemaComplexType type = new global::System.Xml.Sch
    global::System.Xml.Schema.XmlSchemaSequence sequence = new global::System.Xml.Sch
    global::System.Xml.Schema.XmlSchemaAny any = new global::System.Xml.Schema.AnyT
    any.Namespace = ds.Namespace;
    sequence.Items.Add(any);
    type.Particle = sequence;
    global::System.Xml.Schema.XmlSchema dsSchema = ds.GetSchemaSerializable();
    if (xs.Contains(dsSchema.TargetNamespace)) {
        global::System.IO.MemoryStream s1 = new global::System.IO.MemoryStream();
        global::System.IO.MemoryStream s2 = new global::System.IO.MemoryStream();
        try {
            global::System.Xml.Schema.XmlSchema schema = null;
            dsSchema.Write(s1);
            for (global::System.Collections.IEnumerator schemas = xs.Schemas(dsS
                schema = ((global::System.Xml.Schema.XmlSchema)(schemas.Current));
                s2.SetLength(0);
                schema.Write(s2);
                if ((s1.Length == s2.Length)) {
                    s1.Position = 0;
                    s2.Position = 0;
                    for (; ((s1.Position != s1.Length)
                            && (s1.ReadByte() == s2.ReadByte())); ) {
                        ;
                    }
                    if ((s1.Position == s1.Length)) {
                        return type;
                    }
                }
            }
        }
    }
}
```

```

        }
    }
    finally {
        if ((s1 != null)) {
            s1.Close();
        }
        if ((s2 != null)) {
            s2.Close();
        }
    }
}
xs.Add(dsSchema);
return type;
}

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
public delegate void ProductsRowChangeEventHandler(object sender, ProductsRowCh
/// <summary>
///Represents the strongly named DataTable class.
///</summary>
[global::System.Serializable()]
[global::System.Xml.Serialization.XmlSchemaProviderAttribute("GetTypedTableSche
public partial class ProductsDataTable : global::System.Data.DataTable, global

    private global::System.Data.DataColumn columnID;

    private global::System.Data.DataColumn columnNameProduct;

    private global::System.Data.DataColumn columnProductDescription;

    private global::System.Data.DataColumn columnProductPrice;

    private global::System.Data.DataColumn columnBarcodeImage;

    [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
public ProductsDataTable() {
    this.TableName = "Products";
    this.BeginInit();
    this.InitClass();
    this.EndInit();
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
internal ProductsDataTable(global::System.Data.DataTable table) {
    this.TableName = table.TableName;
    if ((table.CaseSensitive != table.DataSet.CaseSensitive)) {
        this.CaseSensitive = table.CaseSensitive;
    }
    if ((table.Locale.ToString() != table.DataSet.Locale.ToString())) {
        this.Locale = table.Locale;
    }
    if ((table.Namespace != table.DataSet.Namespace)) {
        this.Namespace = table.Namespace;
    }
    this.Prefix = table.Prefix;
    this.MinimumCapacity = table.MinimumCapacity;
}

```

```
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected ProductsDataTable(global::System.Runtime.Serialization.SerializationInfo
    base(info, context) {
    this.InitVars();
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public global::System.Data.DataColumn IDColumn {
    get {
        return this.columnID;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public global::System.Data.DataColumn ProductNameColumn {
    get {
        return this.columnProductName;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public global::System.Data.DataColumn ProductDescriptionColumn {
    get {
        return this.columnProductDescription;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public global::System.Data.DataColumn ProductPriceColumn {
    get {
        return this.columnProductPrice;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public global::System.Data.DataColumn BarcodeImageColumn {
    get {
        return this.columnBarcodeImage;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
[global::System.ComponentModel.Browsable(false)]
public int Count {
    get {
        return this.Rows.Count;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public ProductsRow this[int index] {
```

```
        get {
            return ((ProductsRow)(this.Rows[index]));
        }
    }

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public event ProductsRowChangeEventHandler ProductsRowChanging;

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public event ProductsRowChangeEventHandler ProductsRowChanged;

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public event ProductsRowChangeEventHandler ProductsRowDeleting;

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public event ProductsRowChangeEventHandler ProductsRowDeleted;

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public void AddProductsRow(ProductsRow row) {
    this.Rows.Add(row);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public ProductsRow AddProductsRow(string ProductName, string ProductDescription) {
    ProductsRow rowProductsRow = ((ProductsRow)(this.NewRow()));
    object[] columnValuesArray = new object[] {
        null,
        ProductName,
        ProductDescription,
        ProductPrice,
        BarcodeImage};
    rowProductsRow.ItemArray = columnValuesArray;
    this.Rows.Add(rowProductsRow);
    return rowProductsRow;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public ProductsRow FindByID(int ID) {
    return ((ProductsRow)(this.Rows.Find(new object[] {
        ID})));
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public virtual global::System.Collections.IEnumerator GetEnumerator() {
    return this.Rows.GetEnumerator();
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public override global::System.Data.DataTable Clone() {
    ProductsDataTable cln = ((ProductsDataTable)(base.Clone()));
    cln.InitVars();
    return cln;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
```

```
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override global::System.Data.DataTable CreateInstance() {
    return new ProductsDataTable();
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
internal void InitVars() {
    this.columnID = base.Columns["ID"];
    this.columnProductName = base.Columns["ProductName"];
    this.columnProductDescription = base.Columns["ProductDescription"];
    this.columnProductPrice = base.Columns["ProductPrice"];
    this.columnBarcodeImage = base.Columns["BarcodeImage"];
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
private void InitClass() {
    this.columnID = new global::System.Data.DataColumn("ID", typeof(int), null);
    base.Columns.Add(this.columnID);
    this.columnProductName = new global::System.Data.DataColumn("ProductName");
    base.Columns.Add(this.columnProductName);
    this.columnProductDescription = new global::System.Data.DataColumn("ProductDescription");
    base.Columns.Add(this.columnProductDescription);
    this.columnProductPrice = new global::System.Data.DataColumn("ProductPrice");
    base.Columns.Add(this.columnProductPrice);
    this.columnBarcodeImage = new global::System.Data.DataColumn("BarcodeImage");
    base.Columns.Add(this.columnBarcodeImage);
    this.Constraints.Add(new global::System.Data.UniqueConstraint("Constraints1", new DataColumn[] { this.columnID }, true));
    this.columnID.AutoIncrement = true;
    this.columnID.AutoIncrementSeed = -1;
    this.columnID.AutoIncrementStep = -1;
    this.columnID.AllowDBNull = false;
    this.columnID.Unique = true;
    this.columnProductName.MaxLength = 255;
    this.columnProductDescription.MaxLength = 255;
    this.columnProductPrice.MaxLength = 255;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public ProductsRow NewProductsRow() {
    return ((ProductsRow)(this.NewRow()));
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override global::System.Data.DataRow NewRowFromBuilder(global::System.Data.DataRowBuilder builder) {
    return new ProductsRow(builder);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override global::System.Type GetRowType() {
    return typeof(ProductsRow);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
```

```

protected override void OnRowChanged(global::System.Data.DataRowChangeEvent e)
{
    base.OnRowChanged(e);
    if ((this.ProductsRowChanged != null)) {
        this.ProductsRowChanged(this, new ProductsRowChangeEvent(((ProductsRow) e.NewRow), ((ProductsRow) e.OldRow)));
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override void OnRowChanging(global::System.Data.DataRowChangeEvent e)
{
    base.OnRowChanging(e);
    if ((this.ProductsRowChanging != null)) {
        this.ProductsRowChanging(this, new ProductsRowChangeEvent(((ProductsRow) e.NewRow), ((ProductsRow) e.OldRow)));
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override void OnRowDeleted(global::System.Data.DataRowChangeEvent e)
{
    base.OnRowDeleted(e);
    if ((this.ProductsRowDeleted != null)) {
        this.ProductsRowDeleted(this, new ProductsRowChangeEvent(((ProductsRow) e.NewRow)));
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
protected override void OnRowDeleting(global::System.Data.DataRowChangeEvent e)
{
    base.OnRowDeleting(e);
    if ((this.ProductsRowDeleting != null)) {
        this.ProductsRowDeleting(this, new ProductsRowChangeEvent(((ProductsRow) e.NewRow)));
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public void RemoveProductsRow(ProductsRow row) {
    this.Rows.Remove(row);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public static global::System.Xml.Schema.XmlSchemaComplexType GetTypedTable(global::System.Xml.Schema.XmlSchemaComplexType type)
{
    global::System.Xml.Schema.XmlSchemaComplexType type1 = new global::System.Xml.Schema.XmlSchemaComplexType();
    global::System.Xml.Schema.XmlSchemaSequence sequence = new global::System.Xml.Schema.XmlSchemaSequence();
    ProductsDataSet ds = new ProductsDataSet();
    global::System.Xml.Schema.XmlSchemaAny any1 = new global::System.Xml.Schema.XmlSchemaAny();
    any1.Namespace = "http://www.w3.org/2001/XMLSchema";
    any1.MinOccurs = new decimal(0);
    any1.MaxOccurs = decimal.MaxValue;
    any1.ProcessContents = global::System.Xml.Schema.XmlSchemaContentProcessMode.Strict;
    sequence.Items.Add(any1);
    global::System.Xml.Schema.XmlSchemaAny any2 = new global::System.Xml.Schema.XmlSchemaAny();
    any2.Namespace = "urn:schemas-microsoft-com:xml-diffgram-v1";
    any2.MinOccurs = new decimal(1);
    any2.ProcessContents = global::System.Xml.Schema.XmlSchemaContentProcessMode.Preserve;
    sequence.Items.Add(any2);
    global::System.Xml.Schema.XmlSchemaAttribute attribute1 = new global::System.Xml.Schema.XmlSchemaAttribute();
    attribute1.Name = "namespace";
    attribute1.FixedValue = ds.Namespace;
}

```

```

        type.Attributes.Add(attribute1);
        global::System.Xml.Schema.XmlSchemaAttribute attribute2 = new global::System.Xml.Schema.XmlSchemaAttribute();
        attribute2.Name = "tableName";
        attribute2.FixedValue = "ProductsDataTable";
        type.Attributes.Add(attribute2);
        type.Particle = sequence;
        global::System.Xml.Schema.XmlSchema dsSchema = ds.GetSchemaSerializable();
        if (xs.Contains(dsSchema.TargetNamespace)) {
            global::System.IO.MemoryStream s1 = new global::System.IO.MemoryStream();
            global::System.IO.MemoryStream s2 = new global::System.IO.MemoryStream();
            try {
                global::System.Xml.Schema.XmlSchema schema = null;
                dsSchema.Write(s1);
                for (global::System.Collections.IEnumerator schemas = xs.Schemas.GetEnumerator(); schemas.MoveNext();)
                    schema = ((global::System.Xml.Schema.XmlSchema)(schemas.Current));
                s2.SetLength(0);
                schema.Write(s2);
                if ((s1.Length == s2.Length)) {
                    s1.Position = 0;
                    s2.Position = 0;
                    for (; ((s1.Position != s1.Length) && (s1.ReadByte() == s2.ReadByte())); )
                        ;
                }
                if ((s1.Position == s1.Length)) {
                    return type;
                }
            }
        }
    }
    finally {
        if ((s1 != null))
            s1.Close();
        if ((s2 != null))
            s2.Close();
    }
}
xs.Add(dsSchema);
return type;
}
}

/// <summary>
/// Represents strongly named DataRow class.
/// </summary>
public partial class ProductsRow : global::System.Data.DataRow {

    private ProductsDataTable tableProducts;

    [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design", "4.0.0.0")]
    internal ProductsRow(global::System.Data.DataRowBuilder rb) :
        base(rb) {
        this.tableProducts = ((ProductsDataTable)(this.Table));
    }

    [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design", "4.0.0.0")]
}

```

```
public int ID {
    get {
        return ((int)(this[this.tableProducts.IDColumn]));
    }
    set {
        this[this.tableProducts.IDColumn] = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public string ProductName {
    get {
        try {
            return ((string)(this[this.tableProducts.ProductNameColumn]));
        }
        catch (global::System.InvalidCastException e) {
            throw new global::System.Data.StrongTypingException("The value
        }
    }
    set {
        this[this.tableProducts.ProductNameColumn] = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public string ProductDescription {
    get {
        try {
            return ((string)(this[this.tableProducts.ProductDescriptionColu
        }
        catch (global::System.InvalidCastException e) {
            throw new global::System.Data.StrongTypingException("The value
        }
    }
    set {
        this[this.tableProducts.ProductDescriptionColumn] = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public string ProductPrice {
    get {
        try {
            return ((string)(this[this.tableProducts.ProductPriceColumn]));
        }
        catch (global::System.InvalidCastException e) {
            throw new global::System.Data.StrongTypingException("The value
        }
    }
    set {
        this[this.tableProducts.ProductPriceColumn] = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design")]
public byte[] BarcodeImage {
```

```
get {
    try {
        return ((byte[])(this[this.tableProducts.BarcodeImageColumn]));
    }
    catch (global::System.InvalidCastException e) {
        throw new global::System.Data.StrongTypingException("The value
    }
}
set {
    this[this.tableProducts.BarcodeImageColumn] = value;
}
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public bool IsProductNameNull() {
    return this.IsNull(this.tableProducts.ProductNameColumn);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public void SetProductNameNull() {
    this[this.tableProducts.ProductNameColumn] = global::System.Convert.DBM
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public bool IsProductDescriptionNull() {
    return this.IsNull(this.tableProducts.ProductDescriptionColumn);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public void SetProductDescriptionNull() {
    this[this.tableProducts.ProductDescriptionColumn] = global::System.Conv
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public bool IsProductPriceNull() {
    return this.IsNull(this.tableProducts.ProductPriceColumn);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public void SetProductPriceNull() {
    this[this.tableProducts.ProductPriceColumn] = global::System.Convert.DB
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public bool IsBarcodeImageNull() {
    return this.IsNull(this.tableProducts.BarcodeImageColumn);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design
public void SetBarcodeImageNull() {
    this[this.tableProducts.BarcodeImageColumn] = global::System.Convert.DB
}
```

```

}

/// <summary>
/// Row event argument class
///</summary>
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type
public class ProductsRowChangeEvent : global::System.EventArgs {

    private ProductsRow eventRow;

    private global::System.Data.DataRowAction eventAction;

    [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.
public ProductsRowChangeEvent(ProductsRow row, global::System.Data.DataRowA
    this.eventRow = row;
    this.eventAction = action;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.
public ProductsRow Row {
    get {
        return this.eventRow;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.
public global::System.Data.DataRowAction Action {
    get {
        return this.eventAction;
    }
}
}

namespace BarcodeInCrystalReports.ProductsDataSetTableAdapters {

    /// <summary>
    /// Represents the connection and commands used to retrieve and save data.
    ///</summary>
    [global::System.ComponentModel.DesignerCategoryAttribute("code")]
    [global::System.ComponentModel.ToolboxItem(true)]
    [global::System.ComponentModel.DataObjectAttribute(true)]
    [global::System.ComponentModel.DesignerAttribute("Microsoft.VSDesigner.DataSource.
        ", Version=10.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")]
    [global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapter")]
    public partial class ProductsTableAdapter : global::System.ComponentModel.Component
        {
            private global::System.Data.OleDb.OleDbDataAdapter _adapter;

            private global::System.Data.OleDb.OleDbConnection _connection;

            private global::System.Data.OleDb.OleDbCommand[] _commandCollection;

            private bool _clearBeforeFill;

            [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]

```

```
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
public ProductsTableAdapter() {
    this.ClearBeforeFill = true;
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
private global::System.Data.OleDb.OleDbDataAdapter Adapter {
    get {
        if ((this._adapter == null)) {
            this.InitAdapter();
        }
        return this._adapter;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
internal global::System.Data.OleDb.OleDbConnection Connection {
    get {
        if ((this._connection == null)) {
            this.InitConnection();
        }
        return this._connection;
    }
    set {
        this._connection = value;
        if ((this.Adapter.InsertCommand != null)) {
            this.Adapter.InsertCommand.Connection = value;
        }
        if ((this.Adapter.DeleteCommand != null)) {
            this.Adapter.DeleteCommand.Connection = value;
        }
        if ((this.Adapter.UpdateCommand != null)) {
            this.Adapter.UpdateCommand.Connection = value;
        }
        for (int i = 0; (i < this.CommandCollection.Length); i = (i + 1)) {
            if ((this.CommandCollection[i] != null)) {
                ((global::System.Data.OleDb.OleDbCommand)(this.CommandCollection[i])).Connection = value;
            }
        }
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
protected global::System.Data.OleDb.OleDbCommand[] CommandCollection {
    get {
        if ((this._commandCollection == null)) {
            this.InitCommandCollection();
        }
        return this._commandCollection;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
public bool ClearBeforeFill {
    get {
        return this._clearBeforeFill;
```

```
        }
    set {
        this._clearBeforeFill = value;
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Typed")]
private void InitAdapter() {
    this._adapter = new global::System.Data.OleDb.OleDbDataAdapter();
    global::System.Data.Common.DataTableMapping tableMapping = new global::System.Data.Common.DataTableMapping();
    tableMapping.SourceTable = "Table";
    tableMapping.DataSetName = "Products";
    tableMapping.ColumnMappings.Add("ID", "ID");
    tableMapping.ColumnMappings.Add("ProductName", "ProductName");
    tableMapping.ColumnMappings.Add("ProductDescription", "ProductDescription");
    tableMapping.ColumnMappings.Add("ProductPrice", "ProductPrice");
    this._adapter.TableMappings.Add(tableMapping);
    this._adapter.DeleteCommand = new global::System.Data.OleDb.OleDbCommand();
    this._adapter.DeleteCommand.Connection = this.Connection;
    this._adapter.DeleteCommand.CommandText = "DELETE FROM `Products` WHERE ((`ProductID` = ?) AND ((? = 1 AND `ProductDescription` IS NULL) OR (? = 1 AND `ProductPrice` IS NULL) OR (? = 1 AND `ProductName` = ?)))";
    this._adapter.DeleteCommand.CommandType = global::System.Data.CommandType.Text;
    this._adapter.DeleteCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@ProductID", System.Data.OleDbType.Int32));
    this._adapter.DeleteCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Count", System.Data.OleDbType.Int32));
    this._adapter.DeleteCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Description", System.Data.OleDbType.VarChar));
    this._adapter.DeleteCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Price", System.Data.OleDbType.Decimal));
    this._adapter.DeleteCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Name", System.Data.OleDbType.VarChar));
    this._adapter.InsertCommand = new global::System.Data.OleDb.OleDbCommand();
    this._adapter.InsertCommand.Connection = this.Connection;
    this._adapter.InsertCommand.CommandText = "INSERT INTO `Products` (`ProductID`, `ProductName`, `ProductDescription`, `ProductPrice`) VALUES (?, ?, ?, ?)";
    this._adapter.InsertCommand.CommandType = global::System.Data.CommandType.Text;
    this._adapter.InsertCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@ProductID", System.Data.OleDbType.Int32));
    this._adapter.InsertCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Name", System.Data.OleDbType.VarChar));
    this._adapter.InsertCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Description", System.Data.OleDbType.VarChar));
    this._adapter.InsertCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Price", System.Data.OleDbType.Decimal));
    this._adapter.UpdateCommand = new global::System.Data.OleDb.OleDbCommand();
    this._adapter.UpdateCommand.Connection = this.Connection;
    this._adapter.UpdateCommand.CommandText = @"UPDATE `Products` SET `ProductName` = ?, `ProductDescription` = ?, `ProductPrice` = ? WHERE `ProductID` = ?";
    this._adapter.UpdateCommand.CommandType = global::System.Data.CommandType.Text;
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Name", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Description", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Price", System.Data.OleDbType.Decimal));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@ProductID", System.Data.OleDbType.Int32));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@Count", System.Data.OleDbType.Int32));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@OldDescription", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@OldPrice", System.Data.OleDbType.Decimal));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@OldName", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@NewDescription", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@NewPrice", System.Data.OleDbType.Decimal));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@NewName", System.Data.OleDbType.VarChar));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@OldCount", System.Data.OleDbType.Int32));
    this._adapter.UpdateCommand.Parameters.Add(new global::System.Data.OleDb.OleDbParameter("@NewCount", System.Data.OleDbType.Int32));
}
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Typed")]
private void InitConnection() {
    this._connection = new global::System.Data.OleDb.OleDbConnection();
}
```

```
        this._connection.ConnectionString = global::BarcodeInCrystalReports.Properties.  
    }  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
private void InitCommandCollection() {  
    this._commandCollection = new global::System.Data.OleDb.OleDbCommand[1];  
    this._commandCollection[0] = new global::System.Data.OleDb.OleDbCommand();  
    this._commandCollection[0].Connection = this.Connection;  
    this._commandCollection[0].CommandText = "SELECT ID, ProductName, ProductDe  
    this._commandCollection[0].CommandType = global::System.Data.CommandType.T  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapte  
[global::System.ComponentModel.DataObjectMethodAttribute(global::System.Compon  
public virtual int Fill(ProductsDataSet.ProductsDataTable dataTable) {  
    this.Adapter.SelectCommand = this.CommandCollection[0];  
    if ((this.ClearBeforeFill == true)) {  
        dataTable.Clear();  
    }  
    int returnValue = this.Adapter.Fill(dataTable);  
    return returnValue;  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapte  
[global::System.ComponentModel.DataObjectMethodAttribute(global::System.Compon  
public virtual ProductsDataSet.ProductsDataTable GetData() {  
    this.Adapter.SelectCommand = this.CommandCollection[0];  
    ProductsDataSet.ProductsDataTable dataTable = new ProductsDataSet.ProductsD  
    this.Adapter.Fill(dataTable);  
    return dataTable;  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapte  
public virtual int Update(ProductsDataSet.ProductsDataTable dataTable) {  
    return this.Adapter.Update(dataTable);  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapte  
public virtual int Update(ProductsDataSet dataSet) {  
    return this.Adapter.Update(dataSet, "Products");  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]  
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type  
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapte  
public virtual int Update(global::System.Data.DataRow dataRow) {  
    return this.Adapter.Update(new global::System.Data.DataRow[] {  
        dataRow});  
}  
  
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
```

```

[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapter")]
public virtual int Update(global::System.Data.DataRow[] dataRows) {
    return this.Adapter.Update(dataRows);
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapter")]
[global::System.ComponentModel.DataObjectMethodAttribute(global::System.ComponentModel.DataObjectMethodType.Update)]
public virtual int Delete(int Original_ID, string Original_ProductName, string Original_Description, decimal Original_Price) {
    this.Adapter.DeleteCommand.Parameters[0].Value = ((int)(Original_ID));
    if ((Original_ProductName == null)) {
        this.Adapter.DeleteCommand.Parameters[1].Value = ((object)(1));
        this.Adapter.DeleteCommand.Parameters[2].Value = global::System.DBNull;
    }
    else {
        this.Adapter.DeleteCommand.Parameters[1].Value = ((object)(0));
        this.Adapter.DeleteCommand.Parameters[2].Value = ((string)(Original_ProductName));
    }
    if ((Original_ProductDescription == null)) {
        this.Adapter.DeleteCommand.Parameters[3].Value = ((object)(1));
        this.Adapter.DeleteCommand.Parameters[4].Value = global::System.DBNull;
    }
    else {
        this.Adapter.DeleteCommand.Parameters[3].Value = ((object)(0));
        this.Adapter.DeleteCommand.Parameters[4].Value = ((string)(Original_ProductDescription));
    }
    if ((Original_ProductPrice == null)) {
        this.Adapter.DeleteCommand.Parameters[5].Value = ((object)(1));
        this.Adapter.DeleteCommand.Parameters[6].Value = global::System.DBNull;
    }
    else {
        this.Adapter.DeleteCommand.Parameters[5].Value = ((object)(0));
        this.Adapter.DeleteCommand.Parameters[6].Value = ((string)(Original_ProductPrice));
    }
    global::System.Data.ConnectionState previousConnectionState = this.Adapter.Connection.State;
    if (((this.Adapter.DeleteCommand.Connection.State & global::System.Data.ConnectionState.Open) != global::System.Data.ConnectionState.Open)) {
        this.Adapter.DeleteCommand.Connection.Open();
    }
    try {
        int returnValue = this.Adapter.DeleteCommand.ExecuteNonQuery();
        return returnValue;
    }
    finally {
        if ((previousConnectionState == global::System.Data.ConnectionState.Closed))
            this.Adapter.DeleteCommand.Connection.Close();
    }
}
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.Type")]
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapter")]
[global::System.ComponentModel.DataObjectMethodAttribute(global::System.ComponentModel.DataObjectMethodType.Insert)]
public virtual int Insert(string ProductName, string ProductDescription, string ProductPrice) {
    if ((ProductName == null)) {
        this.Adapter.InsertCommand.Parameters[0].Value = global::System.DBNull;
    }
}

```

```

        else {
            this.Adapter.InsertCommand.Parameters[0].Value = ((string)(ProductName));
        }
        if ((ProductDescription == null)) {
            this.Adapter.InsertCommand.Parameters[1].Value = global::System.DBNull;
        }
        else {
            this.Adapter.InsertCommand.Parameters[1].Value = ((string)(ProductDescription));
        }
        if ((ProductPrice == null)) {
            this.Adapter.InsertCommand.Parameters[2].Value = global::System.DBNull;
        }
        else {
            this.Adapter.InsertCommand.Parameters[2].Value = ((string)(ProductPrice));
        }
        global::System.Data.ConnectionState previousConnectionState = this.Adapter.Connection.State;
        if (((this.Adapter.InsertCommand.Connection.State & global::System.Data.ConnectionState.Open) != global::System.Data.ConnectionState.Open)) {
            this.Adapter.InsertCommand.Connection.Open();
        }
        try {
            int returnValue = this.Adapter.InsertCommand.ExecuteNonQuery();
            return returnValue;
        }
        finally {
            if ((previousConnectionState == global::System.Data.ConnectionState.Closed))
                this.Adapter.InsertCommand.Connection.Close();
        }
    }
}

[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Data.Design.TypedTableAdapter", "")]
[global::System.ComponentModel.Design.HelpKeywordAttribute("vs.data.TableAdapter")]
[global::System.ComponentModel.DataObjectMethodAttribute(global::System.ComponentModel.DataObjectMethodType.Update)]
public virtual int Update(string ProductName, string ProductDescription, string ProductPrice, int Original_ID) {
    if ((ProductName == null)) {
        this.Adapter.UpdateCommand.Parameters[0].Value = global::System.DBNull;
    }
    else {
        this.Adapter.UpdateCommand.Parameters[0].Value = ((string)(ProductName));
    }
    if ((ProductDescription == null)) {
        this.Adapter.UpdateCommand.Parameters[1].Value = global::System.DBNull;
    }
    else {
        this.Adapter.UpdateCommand.Parameters[1].Value = ((string)(ProductDescription));
    }
    if ((ProductPrice == null)) {
        this.Adapter.UpdateCommand.Parameters[2].Value = global::System.DBNull;
    }
    else {
        this.Adapter.UpdateCommand.Parameters[2].Value = ((string)(ProductPrice));
    }
    this.Adapter.UpdateCommand.Parameters[3].Value = ((int)(Original_ID));
    if ((Original_ProductName == null)) {
        this.Adapter.UpdateCommand.Parameters[4].Value = ((object)(1));
        this.Adapter.UpdateCommand.Parameters[5].Value = global::System.DBNull;
    }
    else {

```

```

        this.Adapter.UpdateCommand.Parameters[4].Value = ((object)(0));
        this.Adapter.UpdateCommand.Parameters[5].Value = ((string)(Original_Pro
    }
    if ((Original_ProductDescription == null)) {
        this.Adapter.UpdateCommand.Parameters[6].Value = ((object)(1));
        this.Adapter.UpdateCommand.Parameters[7].Value = global::System.DBNull
    }
    else {
        this.Adapter.UpdateCommand.Parameters[6].Value = ((object)(0));
        this.Adapter.UpdateCommand.Parameters[7].Value = ((string)(Original_Pro
    }
    if ((Original_ProductPrice == null)) {
        this.Adapter.UpdateCommand.Parameters[8].Value = ((object)(1));
        this.Adapter.UpdateCommand.Parameters[9].Value = global::System.DBNull
    }
    else {
        this.Adapter.UpdateCommand.Parameters[8].Value = ((object)(0));
        this.Adapter.UpdateCommand.Parameters[9].Value = ((string)(Original_Pro
    }
    global::System.Data.ConnectionState previousConnectionState = this.Adapter
    if (((this.Adapter.UpdateCommand.Connection.State & global::System.Data.Con
        != global::System.Data.ConnectionState.Open)) {
        this.Adapter.UpdateCommand.Connection.Open();
    }
    try {
        int returnValue = this.Adapter.UpdateCommand.ExecuteNonQuery();
        return returnValue;
    }
    finally {
        if ((previousConnectionState == global::System.Data.ConnectionState.Cl
            this.Adapter.UpdateCommand.Connection.Close();
        }
    }
}
}

#pragma warning restore 1591

```

Program.cs

```

using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace BarcodeInCrystalReports
{
    static class Program
    {
        [STAThread]

```

```
        static void Main()
    {
        Application.EnableVisualStyles();
        Application.SetCompatibleTextRenderingDefault(false);
        Application.Run(new Form1());
    }
}
```

app.config

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
    <configSections>
    </configSections>
    <connectionStrings>
        <add name="CrystalReportsApplication1.Properties.Settings.productsConnectionString"
            connectionString="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\products.mdb"
            providerName="System.Data.OleDb" />
    </connectionStrings>
    <startup useLegacyV2RuntimeActivationPolicy="true">
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/>
    </startup>
</configuration>
```

VIDEO

<https://www.youtube.com/watch?v=REnj3A-oSPI>

ON-PREMISE OFFLINE SDK

[60 Day Free Trial](#) or [Visit ByteScout Barcode SDK Home Page](#)
[Explore ByteScout Barcode SDK Documentation](#)
[Explore Samples](#)
[Sign Up for ByteScout Barcode SDK Online Training](#)

[Get Your API Key](#)

[Explore Web API Docs](#)

[Explore Web API Samples](#)

[visit www.ByteScout.com](#)

[visit www.PDF.co](#)

[www.bytescout.com](#)