

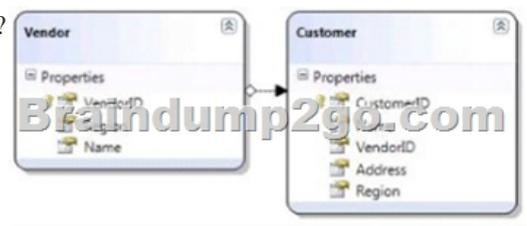
QUESTION 233 The database contains orphaned Color records that are no longer connected to Part records. You need to clean up the orphaned records. You have an existing ContosoEntities context object named context. Which code segment should you use? A. Dim unusedColors = context.Colors.Where(Function(c) Not C.Parts.Any()) context.DeleteObject(unusedColors) context.SaveChanges() B. Dim unusedColors = context.Colors.Where(Function(c) Not C.Parts.Any()).ToList() For Each unused As Color In unusedColors context.DeleteObject(unused) Next context.SaveChanges() C. context.Colors.ToList().RemoveAll(Function(c) Not C.Parts.Any()) context.SaveChanges() D. context.Colors.TakeWhile(Function(c) Not C.Parts.Any()) context.SaveChanges() Answer: B

QUESTION 234 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to create an application. The application connects to a Microsoft SQL Server 2008 database. You create classes by using LINQ to SQL based on the records shown in the exhibit. (Click the Exhibit button.) You need to create a LINQ query to retrieve a list of objects that contains the OrderID and CustomerID properties. You need to retrieve the total price amount of each Order record. What are two possible ways to achieve this goal? (Each correct answer presents a complete solution. Choose two.)



A. From details in dataContext.Order\_Details \_Group details By details.OrderID Into g \_Join order In dataContext.Orders On g.Key = order.OrderID \_ Select New With { \_ .OrderID = order.OrderID, \_ .CustomerID = order.CustomerID, \_ .TotalAmount = g.Sum(Function(od) od.UnitPrice \* od.Quantity) \_ } B. dataContext.Order\_Details.GroupJoin(dataContext.Orders, Function(d) D.OrderID, Function(o) o.OrderID, Function(dts,ord) New With { \_ .OrderID = dts.OrderID, \_ .CustomerID = dts.Order.CustomerID, \_ .TotalAmount = dts.UnitPrice \* dts.Quantity \_ }) C. From order in dataContext.Orders \_Group order By order.OrderID Into g \_Join details in dataContext.Order\_Details On g.Key = details.OrderID \_ Select New With { \_ .OrderID = details.OrderID, \_ .CustomerID = details.Order.CustomerID, \_ .TotalAmount = details.UnitPrice \* details.Quantity \_ } D. dataContext.Orders.GroupJoin(dataContext.Order\_Details, Function(o) o.OrderID, Function(d) D.OrderID, Function(ord, dts) New With { \_ .OrderID = ord.OrderID, \_ .CustomerID = ord.CustomerID, \_ .TotalAmount = dts.Sum(Function(od) od.UnitPrice \* od.Quantity) \_ }) Answer: A D Explanation: Alternative A. This is an Object Query. It looks at the Order Details EntitySet and creating a group g based on OrderID. - The group g is then joined with Orders EntitySet based on g.Key = OrderID - For each matching records a new dynamic object containing OrderID, CustomerID and TotalAmount is created. - The dynamic records are the results returned in an IEnumerable Object, for later processing Alternative D. This is an Object Query. The GroupJoin method is used to join Orders to OrderDetails. Parameters for GroupJoin: 1. An Order\_Details EntitySet 2. Order o (from the Orders in the Orders Entity Set, picking up Order\_id from both Entity Sets) 3. Order\_ID from the first Order\_Details record from the OD EntitySet 4.

Lambda Expression passing ord and dts (ord=o, dts=d) The body of the Lambda Expression is working out the total and Returning a Dynamic object as in A. QUESTION 235 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to develop an application that uses LINQ to SQL. The application contains the following model. Each region contains a single vendor. Customers order parts from the vendor that is located in their region. You need to ensure that each row in the Customer table references the appropriate row from the Vendor table. Which code segment should you use?



- A. Dim dc As SalesDataContext = New SalesDataContext( "...") Dim query = From c In dc.Customers Join v In dc.Vendors On c.Region Equals v.Region Select New With { .Customer = c, -Vendor = v } For Each u In query B. Vendor.VendorID = u.Customer.VendorID Next dc.SubmitChanges()
  - C. Dim dc As SalesDataContext = New SalesDataContext( "...") Dim query = From v In dc.Vendors Join c In dc.Customers On v.Region Equals c.Region Select New With { .Vendor = v, -Customer = c } For Each u In query D. Customer.VendorID = u.Vendor.VendorID Next dc.SubmitChanges()
  - E. Dim dc As SalesDataContext = New SalesDataContext( "...") Dim query = From c In dc.Customers Join v In dc.Vendors On c.VendorID Equals v.VendorID Select New With { .Customer = c, -Vendor = v } For Each u In query F. Vendor.Region = u.Customer.Region Next dc.SubmitChanges()
  - G. Dim dc As SalesDataContext = New SalesDataContext( "...") Dim query = From v In dc.Vendors Join c In dc.Customers On v.VendorID Equals c.VendorID Select New With { .Vendor = v, .Customer = c } For Each u In query H. Customer.Region = u.Vendor.Region Next dc.SubmitChanges()
- Answer: B QUESTION 236 Drag and Drop Question You have a Cable named Table1 that contains two columns named Column1 and Column2. Column1 contains string data. Column2 contains image files. You create the following code:

```
String sql =
    "SELECT Column2, Column1 FROM Table1";
SqlConnection connection =
    new SqlConnection(connectionString);
SqlCommand command =
    new SqlCommand(sql, connection);
connection.Open();
```

What code should you use? (To answer, drag the appropriate elements to the correct locations. Each element may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.)



Answer:



QUESTION 237 You are developing a new feature that displays an auto-complete list to users as they type color names. You have an existing ContosoEntities context object named context. To support the new feature you must develop code that will accept a string object named text containing a user's partial input and will query the Colors database table to retrieve all color names that begin with that input. You need to create an Entity SQL (ESQL) query to meet the requirement. The query must not be vulnerable to a SQL injection attack. Which code segment should you use? A. Dim parameter = New ObjectParameter("text", HttpUtility.HtmlEncode(text) & "%") Dim result = context.CreateQuery(Of String)("SELECT (c.Name) FROM Colors AS c WHERE C.Name LIKE '@text'", parameter) B. Dim parameter = New ObjectParameter("text", text & "%") Dim result = context.CreateQuery(Of String)("SELECT (c.Name) FROM Colors AS c WHERE C. Name LIKE @text", parameter) C. Dim parameter = New ObjectParameter("text", text & "%") Dim result = context.CreateQuery(Of String)("SELECT VALUE (c.Name) FROM Colors AS c WHERE C.Name LIKE @text", parameter) D. Dim parameter = New ObjectParameter("text", text & "%")

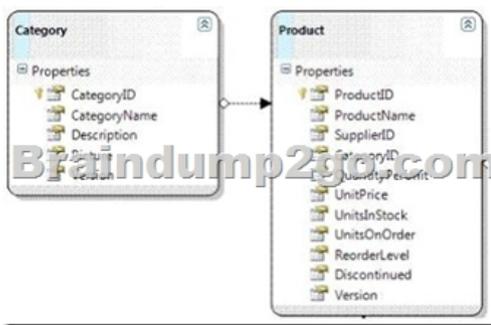
Dim result = context.CreateQuery(Of String)("SELECT VALUE (c.Name) FROM Colors AS c WHERE c.Name LIKE '@text'", parameter) Answer: C Explanation: Entity SQL supports two variants of the SELECT clause. The first variant, row select, is identified by the SELECT keyword, and can be used to specify one or more values that should be projected out. Because a row wrapper is implicitly added around the values returned, the result of the query expression is always a multiset of rows. Each query expression in a row select must specify an alias. If no alias is specified, Entity SQL attempts to generate an alias by using the alias generation rules. The other variant of the SELECT clause, value select, is identified by the SELECT VALUE keyword. It allows only one value to be specified, and does not add a row wrapper. A row select is always expressible in terms of VALUE SELECT, as illustrated in the following example. ESQ Select(<http://msdn.microsoft.com/en-us/library/bb399554.aspx>) QUESTION 238 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to create an application. You create a stored procedure to insert a new record in the Categories table according to following code segment. CREATE PROCEDURE dbo.InsertCategory @CategoryName nvarchar(15), @Identity int OUT AS INSERT INTO Categories (CategoryName) VALUES (@CategoryName) SET @Identity = SCOPE\_IDENTITY() RETURN @@ROWCOUNT You write the following code segment. (Line numbers are included for reference only).

```

01 private static void ReturnIdentity(string connectionString)
02 {
03     using (SqlConnection connection = new
SqlConnection(connectionString))
04 {
05     SqlDataAdapter adapter = new SqlDataAdapter("SELECT CategoryID, CategoryName
FROM dbo.Categories", connection);
06 adapter.InsertCommand = new SqlCommand("InsertCategory", connection);
07 adapter.InsertCommand.CommandType = CommandType.StoredProcedure;
08 SqlParameter rowCountParameter =
adapter.InsertCommand.Parameters.Add(10 "@RowCount", SqlDbType.Int);
12 adapter.InsertCommand.Parameters.Add(13 "@CategoryName", SqlDbType.NChar, 15, "CategoryName");
14 SqlParameter identityParameter =
adapter.InsertCommand.Parameters.Add(15 "@Identity", SqlDbType.Int, 0, "CategoryID");
17 DataTable categories = new
DataTable();
18 adapter.Fill(categories);
19 DataRow categoryRow = categories.NewRow();
20 categoryRow["CategoryName"] =
"New Beverages";
21 categories.Rows.Add(categoryRow);
22 adapter.Update(categories);
23 Int32 rowCount =
(Int32)adapter.InsertCommand.Parameters["@RowCount"].Value;
24 }
}

```

You need to retrieve the identity of the new record. You also need to retrieve the row count. What should you do? A. Insert the following code segment at line 11. rowCountParameter.Direction = ParameterDirection.ReturnValue; Insert the following code segment at line 16. identityParameter.Direction = ParameterDirection.ReturnValue; B. Insert the following code segment at line 11. rowCountParameter.Direction = ParameterDirection.Output; Insert the following code segment at line 16. identityParameter.Direction = ParameterDirection.Output; C. Insert the following code segment at line 11. rowCountParameter.Direction = ParameterDirection.ReturnValue; Insert the following code segment at line 16. identityParameter.Direction = ParameterDirection.Output; D. Insert the following code segment at line 11. rowCountParameter.Direction = ParameterDirection.Output; Insert the following code segment at line 16. identityParameter.Direction = ParameterDirection.ReturnValue; Answer: C Explanation: Input-The parameter is an input parameter. InputOutput-The parameter is capable of both input and output. Output-The parameter is an output parameter. ReturnValue-The parameter represents a return value from an operation such as a stored procedure, builtin function, or user-defined function. ParameterDirection Enumeration([http://msdn.microsoft.com/en-us/library/system.data.parameterdirection\(v=vs.71\).aspx](http://msdn.microsoft.com/en-us/library/system.data.parameterdirection(v=vs.71).aspx)) QUESTION 239 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to create an application. The application connects to a Microsoft SQL Server database. You use the following SQL statement to retrieve an instance of a DataSet object named ds. SELECT CustomerID, CompanyName, ContactName, Address, City FROM dbo.Customers You need to query the DataSet object to retrieve only the rows where the ContactName field is not NULL. Which code segment should you use? A. from row in ds.Tables(0).AsEnumerable() where DirectCast(row("ContactName"), String) IsNot Nothing select row B. from row in ds.Tables(0).AsEnumerable() where row.Field(Of String)("ContactName") IsNot Nothing select row C. from row in ds.Tables(0).AsEnumerable() where Not row.IsNull(DirectCast(row("ContactName"), String)) select row D. from row in ds.Tables(0).AsEnumerable() where Not Convert.IsDBNull(row.Field(Of String)("Region")) select row Answer: B Explanation: Field<T>(DataRow, String) Provides strongly-typed access to each of the column values in the specified row. The Field method also supports nullable types. QUESTION 240 You use Microsoft Visual Studio 2010 and Microsoft .NET Framework 4 to create an application. The application connects to a Microsoft SQL Server database. You use the ADO.NET LINQ to SQL model to retrieve data from the database. The application contains the Category and Product entities, as shown in the following exhibit. You need to ensure that LINQ to SQL executes only a single SQL statement against the database. You also need to ensure that the query retrieves the list of categories and the list of products. Which code segment should you use?



A. Using dc As New NorthwindDataContext(dc.ObjectTrackingEnabled = FalseDim categories As var = From c In dc.Categories \_ Select cFor Each category As var In categoriesConsole.WriteLine("{0} has {1} products", category.CategoryName, category.Products.Count)NextEnd UsingB. Using dc As New NorthwindDataContext(dc.DeferredLoadingEnabled = FalseDim dlOptions As New DataLoadOptions(dlOptions.LoadWith(Of Category)(Function(c As )C. Products) dc.LoadOptions = dlOptionsDim categories As var = From c In dc.Categories \_ Select cFor Each category As var In categoriesConsole.WriteLine("{0} has {1} products", category. CategoryName, category.Products.Count)NextEnd UsingC. Using dc As New NorthwindDataContext() dc.DeferredLoadingEnabled = FalseDim categories As var = From c In dc.Categories \_ Select cFor Each category As var In categoriesConsole.WriteLine("{0} has {1} products", category.CategoryName, category.Products.Count)NextEnd UsingD. Using dc As New NorthwindDataContext(dc.DeferredLoadingEnabled = FalseDim dlOptions As New DataLoadOptions()dlOptions.AssociateWith(Of Category)(Function(c As )C.Products) dc.LoadOptions = dlOptionsDim categories As var = From c In dc.Categories \_ Select cFor Each category As var In categoriesConsole.WriteLine("{0} has {1} products", category.CategoryName, category.Products.Count)NextEnd Using Answer: BExplanation:DataLoadOptions Class Provides for immediate loading and filtering of related data. DataLoadOptions.LoadWith(LambdaExpression) Retrieves specified data related to the main target by using a lambda expression.You can retrieve many objects in one query by using LoadWith. DataLoadOptions.AssociateWith(LambdaExpression) Filters the objects retrieved for a particular relationship.Use the AssociateWith method to specify sub-queries to limit the amount of retrieved data.DataLoadOptions Class(<http://msdn.microsoft.com/en-us/library/system.data.linq.dataloadoptions.aspx>)How to: Retrieve Many Objects At Once (LINQ to SQL) ([http://msdn.microsoft.com/en-us/library/Bb386917\(v=vs.90\).aspx](http://msdn.microsoft.com/en-us/library/Bb386917(v=vs.90).aspx))How to: Filter Related Data (LINQ to SQL)([http://msdn.microsoft.com/en-us/library/Bb882678\(v=vs.100\).aspx](http://msdn.microsoft.com/en-us/library/Bb882678(v=vs.100).aspx)) All Braindump2go 70-516 Exam Dumps are Promised One Year Free Updation -- We will inform you when your products have new questions and Answers updation! Download Microsoft 70-516 Practice Tests Questions Full Version Now - Pass 70-516 100% One Time!

**TS: Accessing Data with Microsoft .NET Framework**



Questions and Answers : 286 Q&As

Updated: Nov 22, 2015

~~\$429.99~~ **\$99.99**

[PDF DEMO](#)

[CHECK OUT](#)

**Product Description Exam Number/Code:**

**Exam Number/Code: 70-516**

"TS: Accessing Data with Microsoft .NET Framework 4", also Microsoft Certification. With the complete collection of questions has assembled to take you through 286 Q&As to your 70-516 exam resources, you will cover every field and category in it for your successful Microsoft Certification.

**Free Demo Download**

Braindump2go offers free demo for 70-516 exam (TS: Accessing Data with Microsoft .NET Framework 4). You can check out the interface, question questions before you decide to buy it.

**Printable PDF**  **Premium VCE + VCE Simulation**

FREE DOWNLOAD: NEW UPDATED 70-516 PDF Dumps & 70-516 VCE Dumps from Braindump2go:

<http://www.braindump2go.com/70-516.html> (286 Q&A)