

Microsoft 70-561

**TS: Microsoft .NET Framework 3.5, ADO.NET
Application Development
Version: 34.0**

Topic 1, C#**QUESTION NO: 1**

You create an application by using the Microsoft .NET Framework 3.5 and Microsoft ADO.NET. The application connects to a Microsoft SQL Server 2005 database.

You write the following code fragment to generate a storage schema definition for a stored procedure from the database.

```
<Function Name="GetSuppliersForCountry" Aggregate="false"
BuiltIn="false" NiladicFunction="false"
IsComposable="false"
ParameterTypeSentinel="AllowImplicitConversion"
Schema="dfoo">
<Parameter Name="country" Type="char" Mode="In" />
</Function>
```

The application uses two namespaces named ContosoModel.Store and ContosoModel1.

The ContosoModel.Store namespace contains the storage schema. The ContosoModel1 namespace contains the conceptual schema that has an entity named Supplier.

You need to create a function named GetSuppliersInCountry that returns a list of Supplier entity instances.

What should you do?

A. Create the following code fragment in the conceptual schema.

```
<FunctionImport EntitySet="Supplier" Name="GetSuppliersInCountry"
ReturnType="ContosoModel.Supplier">
</FunctionImportMapping>
```

B. Create the following code fragment in the conceptual schema.

```
<FunctionImport EntitySet="Supplier" Name="GetSuppliersInCountry"
ReturnType="ContosoModel.Supplier">
<Parameter Name="country" Mode="In" Type="String" />
</FunctionImport>
```

Create the following code fragment in the mapping schema.

```
<FunctionImportMapping FunctionImportName="GetSuppliers"
FunctionName="ContosoModel.Store.GetSuppliers">
<ResultMapping>
<EntityTypeMapping TypeName ="MultiSet" />
</ResultMapping>
</FunctionImportMapping>
```

C. Create the following code fragment in the conceptual schema.

```
<FunctionImport EntitySet="Supplier" Name="GetSuppliersIncountry"
ReturnType="Collection(ContosoModel.Supplier)">
<Parameter Name="country" Mode="In" Type="String" />
</FunctionImport>
```

Create the following code fragment in the mapping schema.

```
<FunctionImportMapping FunctionImportName="GetSuppliers"
FunctionName="ContosoModel.Store.GetSuppliers">
<ResultMapping>
<EntityTypeMapping TypeName ="MultiSet" />
</ResultMapping>
</FunctionImportMapping>
```

D. Create the following code fragment in the conceptual schema.

```
<FunctionImport EntitySet="Supplier" Name="GetSuppliersIncountry"
ReturnType="Collection(ContosoModel.Supplier)">
<Parameter Name="country" Mode="In" Type="String" />
</FunctionImport>
```

Create the following code fragment in the mapping schema.

```
<FunctionImportMapping FunctionImportName="GetSuppliers"
FunctionName="ContosoModel.Store.GetSuppliers">
<ResultMapping>
<EntityTypeMapping TypeName ="ContoaoModel.Supplier" />
</ResultMapping>
</FunctionImportMapping>
```

Answer: D

Explanation:

QUESTION NO: 2

You create an application by using the Microsoft .NET Framework 3.5 and Microsoft ADO.NET.

The application uses anObjectContext named ContosoEntities that contains an entity named Customer.

You write the following code segment. (Line numbers are included for reference only.)

```
01 ContosoEntities entities=new ContosoEntities();  
02 string sql; 03  
04 foreach (string vol in new ObjectQuery<stnng> (sql, entities))  
05 {  
06 //Retrieve Name of Customer  
07 }
```

You need to retrieve the names of all the Customer entity instances as a collection of strings.

Which line of code should you insert at line 03?

- A. sql = "SELECT c.Name FROM ContosoEntities.Customer as c";
- B. sql = "SELECT ROW(c.Name) FROM ContosoEntities.Customer as c";
- C. sql = "SELECT MULTISSET(c.Name) FROM ContosoEntities.Customer as c";
- D. sql = "SELECT VALUE c.Name FROM ContosoEntities.Customer as c";

Answer: A

Explanation: This simple statement returns a collection of strings. It is the default behavior for SELECT statements.

Note:For example, if customers is a collection of type Customer that has a property Name that is of type string, selecting Name from customers will yield a collection of strings, as illustrated in the following example.

```
SELECT customers.Name FROM customers AS c
```

Reference:.NET Framework 4,SELECT

QUESTION NO: 3

You create an application by using the Microsoft .NET Framework 3.5 and Microsoft ADO.NET.

The application contains two entities named Customer and Order. The Customer entity has a navigable property named Orders. The Orders property returns a collection of Order entity instances.

You write the following code segment.

```
ContosoEntities context = new ContosoEntities();  
  
ObjectQuery<Customer> query;
```

You need to ensure that each time a Customer entity instance is queried the related Order entity instances are retrieved.

Which code segment should you add?

- A. `query = context.CreateQuery<Customer>("Orders");`
- B. `query = context.Customer; query.Select("Orders") ;`
- C. `query = context.Customer;`
`query.Parameters.Add(new ObjectParameter("Orders", ""));`
- D. `query = context.Customer.Include("Orders");`

Answer: D

Explanation:

QUESTION NO: 4

You create an application by using the Microsoft .NET Framework version 3.5. You use the Microsoft ADO.NET Entity Framework for persistence. The application connects to a Microsoft SQL Server 2005 database.

You write the following code segment. (Line numbers are included for reference only.)

```
01 EntityConnection con = new EntityConnection();  
02 EntityCommand cmd=con.CreateCommand();  
03 //Set the Entity SQL command  
04 string statement; 05  
06 //Log statement
```

You need to log the T-SQL statements that will be executed by the cmd object against the SQL Server 2005 database.

Which line of code should you insert at line 05?

- A. `statement = cmd.ToTraceString();`
- B. `statement = cmd.CommandText;`
- C. `statement = cmd.UpdatedRowSource.ToString();`
- D. `statement = cmd.ToString();`

Answer: A

Explanation:

QUESTION NO: 5

You create an application by using Microsoft .NET Framework 3.5. The application uses the The application has an entity named Customer.

You write the following code segment.

```
ContosoEntities context = new ContosoEntities();
```

```
Customer cust = new Customer();
```

```
//Set properties for cust
```

You need to save the cust object by using the object context variable.

Which code segment should you add?

- A. `context.AcceptAllChanges();`
`context.Detach(cust);`
`context.SaveChanges(true);`
- B. `context.Attach(cust);`
`context.SaveChanges();`
- C. `context.AcceptAllChanges();`
`context.AttachTo("Customer", cust);`
`context.SaveChanges(true);`
- D. `context.AddToCustomer(cust);`
`context.SaveChanges();`

Answer: B

Explanation: Call **Attach** on the ObjectContext to attach the object to the object context. Do this when the object already exists in the data source but is currently not attached to the context.

TheObjectContext.SaveChangesmethodpersists all updates to the data source and resets change tracking in the object context.

Incorrect answer:

A, C: there should be no input parameter true:

context.SaveChanges(true);

D: No such methodAddToCustomer.

Reference: **.NET Framework 4**,ObjectContext.Attach Method

Reference: **.NET Framework 4**,ObjectContext.SaveChanges Method

QUESTION NO: 6

You create an application by using the Microsoft .NET Framework 3,5 and Microsoft ADO.NET.

The application contains an ObjectContext named ContosoEntities. This ObjectContext contains an entity named Supplier. The entity has a property named Id that is set as the key.

You write the following code segment. (Line numbers are included for reference only.)

```
01 EntityCommand cmd = new EntityCommand();
02 string sql;
03 //Setup cmd object 04
05 cmd.CommandText = sql;
06 IDataReader rd =
cmd.ExecuteReader(CommandBehavior.SequentialAccess) ;
07 while (rd.Read())
08 {
09 //Retrieve Supplier informationIO
10 }
```

You need to retrieve all the properties of the Supplier entity for which the Id value is 1.

Which line of code should you insert at line 04?

- A. sql = "DEREF(CREATEREF(ContosoEntities.Supplier, KEY(I)))";
- B. sql = "DEREF(CREATEREF(ContosoEntities.Supplier, ROW(1)))";
- C. sql = "CREATEREF(ContosoEntities.Supplier, KEY(1))";
- D. sql = "CREATEREF (ContosoEntities. Supplier, ROW(1))";

Answer: B

Explanation: We use ROW(1) to get the row with the ID Value of 1. We then create a reference to the object with CREATEREF. Finally we use DEREF to get the specific row.

Note:

CREATEREF fabricates references to an entity in an entityset.

Syntax:

CreateRef(entityset_identifier, row_typed_expression)

Arguments:

entityset_identifier

The entityset identifier, not a string literal.

row_typed_expression

A row-typed expression that corresponds to the key properties of the entity type.

Note: KEY: Extracts the key of a reference or of an entity expression.

KEY(createref_expression)

Reference: .NET Framework 4, CREATEREF

QUESTION NO: 7

You create an application by using the Microsoft .NET Framework 3.5 and Microsoft ADO.NET. The application uses Microsoft SQL Server 2005.

You write the following code segment. (Line numbers are included for reference only.)

```
01 Seeing myConnString = "User
02 ID=<username>;password=<strong password>;Initial
03 Catalog=pubs;Data Source=myServer";
```



```
04 SqlConnection myConnection = new
05 SqlConnection(myConnString) ;
06 SqlCommand myCommand = new SqlCommand();
07 SqlDataReader myReader;
08 myCommand.CommandType =
09 CommandType.Text;
10 myCommand.Connection = myConnection;
11 myCommand.CommandText = "Select * from Table1;
Select * from Table2;";
12 int RecordCount = 0;
13 try
14 {
15 myConnection.Open(); 16
17 }
18 catch (Exception ex)
19 {
20 }
21 finally
22 {
23 myConnection.Close();
24 }
```

You need to compute the total number of records processed by the Select queries in the RecordCount variable.

Which code segment should you insert at line 16?

A. Add the following code segment to line 16:
myReader = myCommand.ExecuteReader();
RecordCount = myReader.RecordsAffected;

B. Add the following code segment to line 16:

```
while (myReader.Read())  
{  
//Write logic to process data for the first result.  
}
```

RecordCount = myReader.RecordsAffected;

C. Add the following code segment to line 16:

```
while (myReader.HasRows)  
{  
while (myReader.Read())  
{  
//Write logic to process data for the second result.  
RecordCount = RecordCount + 1;  
}  
myReader.NextResult();  
}
```

D. Add the following code segment to line 16:

```
while (myReader.HasRows)  
{  
while (myReader.Read())  
{  
//Write logic to process data for the second result.  
RecordCount = RecordCount + 1; myReader.NextResult();  
}  
}
```

Answer: C

Explanation:

QUESTION NO: 8

You create an application by using the Microsoft .NET Framework 3.5 and Microsoft ADO.NET. The application connects to a Microsoft SQL Server 2005 database.

The application contains the following objects:

- A SqlDataReader object named reader
- A SqlCommand object named cmdRecords
- A SqlCommand object named cmdUpdate
- A SqlConnection object named conn