ExamCode: 70-483
ExamName: Microsoft Programming in C#
Vendor Name: Microsoft

Edition = DEMO

Question: 1

You are developing an application that includes a class named Order. The application will store a collection of Order objects.
The collection must meet the following requirements:

- Use strongly typed members.
- Process Order objects in first-in-first-out order.
- Store values for each Order object.
- Use zero-based indices.

You need to use a collection type that meets the requirements.
Which collection type should you use?

A. Queue<T>
B. SortedList
C. LinkedList<T>
D. HashTable
E. Array<T>

Answer: A

Explanation:
Queues are useful for storing messages in the order they were received for sequential processing. Objects stored in a Queue<T> are inserted at one end and removed from the other.
http://msdn.microsoft.com/en-us/library/7977ey2c.aspx

Question: 2

You are developing an application. The application calls a method that returns an array of integers named employeeIds. You define an integer variable named employeeIdToRemove and assign a value to it. You declare an array named filteredEmployeeIds.
You have the following requirements:

- Remove duplicate integers from the employeeIds array.
- Sort the array in order from the highest value to the lowest value.
- Remove the integer value stored in the employeeIdToRemove variable from the employeeIds array.

You need to create a LINQ query to meet the requirements.
Which code segment should you use?

A.

```
int[] filteredEmployeeIds = employeeIds.Where(value => value !=
  employeeIdToRemove).OrderBy(x => x).ToArray();
```

B.

```
int[] filteredEmployeeIds = employeeIds.Where(value => value !=
  employeeIdToRemove).OrderByDescending(x => x).ToArray();
```

C.

```
int[] filteredEmployeeIds = employeeIds.Distinct().Where(value => value !=
  employeeIdToRemove).OrderByDescending(x => x).ToArray();
```

D.

```
int[] filteredEmployeeIds = employeeIds.Distinct().OrderByDescending(x => x).ToArray();
```

Answer: C

Question: 3

You are developing an application that includes the following code segment. (Line numbers are included for reference only.)

```
01 class Animal
02 {
03   public string Color { get; set; }
04   public string Name { get; set; }
05 }
06 private static IEnumerable<Animal> GetAnimals(string sqlConnectionString)
07 {
08   var animals = new List<Animal>();
09   SqlConnection sqlConnection = new SqlConnection(sqlConnectionString);
10   using (sqlConnection)
11   {
12     SqlCommand sqlCommand = new SqlCommand("SELECT Name, ColorName FROM Animals", sqlConnection);
13
14     using (SqlDataReader sqlDataReader = sqlCommand.ExecuteReader())
15     {
16
17       {
18         var animal = new Animal();
19         animal.Name = (string)sqlDataReader["Name"];
20         animal.Color = (string)sqlDataReader["ColorName"];
21         animals.Add(animal);
22       }
23     }
24   }
25   return customers;
26 }
```

The GetAnimals() method must meet the following requirements:
Connect to a Microsoft SQL Server database.
Create Animal objects and populate them with data from the database.
Return a sequence of populated Animal objects.
You need to meet the requirements.
Which two actions should you perform? (Each correct answer presents part of the solution. Choose two.)

A. Insert the following code segment at line 16:

while(sqlDataReader.NextResult())
B. Insert the following code segment at line 13:
sqlConnection.Open();
C. Insert the following code segment at line 13:
sqlConnection.BeginTransaction();
D. Insert the following code segment at line 16:
while(sqlDataReader.Read())
E. Insert the following code segment at line 16:
while(sqlDataReader.GetValues())

Answer: B, D

Explanation:
SqlConnection.Open - Opens a database connection with the property settings specified by the ConnectionString.
http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqlconnection.open.aspx
SqlDataReader.Read - Advances the SqlDataReader to the next record. http://msdn.microsoft.com/en-us/library/system.data.sqlclient.sqldatareader.read.aspx

Question: 4

You are developing an application that uses the Microsoft ADO.NET Entity Framework to retrieve order information from a Microsoft SQL Server database. The application includes the following code. (Line numbers are included for reference only.)

```
01 public DateTime? OrderDate;
02 IQueryable<Order> LookupOrdersForYear(int year)
03 {
04   using (var context = new NorthwindEntities())
05   {
06     var orders =
07      from order in context.Orders
08
09       select order;
10     return orders.ToList().AsQueryable();
11   }
12 }
```

The application must meet the following requirements:
Return only orders that have an OrderDate value other than null.
Return only orders that were placed in the year specified in the OrderDate property or in a later year.
You need to ensure that the application meets the requirements.
Which code segment should you insert at line 08?

A. Where order.OrderDate.Value != null && order.OrderDate.Value.Year > = year
B. Where order.OrderDate.Value = = null && order.OrderDate.Value.Year = = year
C. Where order.OrderDate.HasValue && order.OrderDate.Value.Year = = year
D. Where order.OrderDate.Value.Year = = year

Answer: A

Explanation:
*For the requirement to use an OrderDate value other than null use:
OrderDate.Value != null
*For the requirement to use an OrderDate value for this year or a later year use:
OrderDate.Value>= year

Question: 5

You are developing an application. The application includes a method named ReadFile that reads data from a file.
The ReadFile() method must meet the following requirements:
It must not make changes to the data file.
It must allow other processes to access the data file.
It must not throw an exception if the application attempts to open a data file that does not exist.
You need to implement the ReadFile() method.
Which code segment should you use?

A. var fs = File.Open(Filename, FileMode.OpenOrCreate, FileAccess.Read, FileShare.ReadWrite);
B. var fs = File.Open(Filename, FileMode.Open, FileAccess.Read, FileShare.ReadWrite);
C. var fs = File.Open(Filename, FileMode.OpenOrCreate, FileAccess.Read, FileShare.Write);
D. var fs = File.ReadAllLines(Filename);
E. var fs = File.ReadAllBytes(Filename);

Answer: A

Explanation:
FileMode.OpenOrCreate - Specifies that the operating system should open a file if it exists; otherwise, a new file should be created. If the file is opened with FileAccess.Read, FileIOPermissionAccess.Read permission is required. If the file access is FileAccess.Write, FileIOPermissionAccess.Write permission is required. If the file is opened with FileAccess.ReadWrite, both FileIOPermissionAccess.Read and FileIOPermissionAccess.Write permissions are required.
http://msdn.microsoft.com/en-us/library/system.io.filemode.aspx
FileShare.ReadWrite - Allows subsequent opening of the file for reading or writing.If this flag is not specified, any request to open the file for reading or writing (by this process or another process) will fail until the file is closed.However, even if this flag is specified, additional permissions might still be needed to access the file.
http://msdn.microsoft.com/pl-pl/library/system.io.fileshare.aspx

Question: 6

An application receives JSON data in the following format:

```
{ "FirstName" : "David",
  "LastName" : "Jones",
  "Values" : [0, 1, 2] }
```

The application includes the following code segment. (Line numbers are included for reference only.)

```
01 public class Name
02 {
03   public int[] Values { get; set; }
04   public string FirstName { get; set; }
05   public string LastName { get; set; }
06 }
07 public static Name ConvertToName(string json)
08 {
09   var ser = new JavaScriptSerializer();
10
11 }
```

You need to ensure that the ConvertToName() method returns the JSON input string as a Name object.
Which code segment should you insert at line 10?

A. Return ser.ConvertToType<Name>(json);
B. Return ser.DeserializeObject(json);
C. Return ser.Deserialize<Name>(json);
D. Return (Name)ser.Serialize(json);

Answer: C

Explanation:
JavaScriptSerializer.Deserialize<T> - Converts the specified JSON string to an object of type T.
http://msdn.microsoft.com/en-us/library/bb355316.aspx

Question: 7

You are developing an application. The application converts a Location object to a string by using a method named WriteObject. The WriteObject() method accepts two parameters, a Location object and an XmlObjectSerializer object.

The application includes the following code. (Line numbers are included for reference only.)

```
01 public enum Compass
02 {
03   North,
04   South,
05   East,
06   West
07 }
08 [DataContract]
09 public class Location
10 {
11   [DataMember]
12   public string Label { get; set; }
13   [DataMember]
14   public Compass Direction { get; set; }
15 }
16 void DoWork()
17 {
18   var location = new Location { Label = "Test", Direction = Compass.West };
19   Console.WriteLine(WriteObject(location,
20
21   ));
22 }
```

You need to serialize the Location object as a JSON object.
Which code segment should you insert at line 20?

A. New DataContractSerializer(typeof(Location))
B. New XmlSerializer(typeof(Location))
C. New NetDataContractSenalizer()
D. New DataContractJsonSerializer(typeof(Location))

Answer: D

Explanation:
The DataContractJsonSerializer class serializes objects to the JavaScript Object Notation (JSON) and deserializes JSON data to objects.
Use the DataContractJsonSerializer class to serialize instances of a type into a JSON document and to deserialize a JSON document into an instance of a type.

Question: 8

An application includes a class named Person. The Person class includes a method named GetData.
You need to ensure that the GetData() from the Person class.
Which access modifier should you use for the GetData() method?

A. Internal
B. Protected
C. Private
D. Protected internal
E. Public

Answer: B

Explanation:
Protected - The type or member can be accessed only by code in the same class or structure, or in a class that is derived from that class.
http://msdn.microsoft.com/en-us/library/ms173121.aspx
The protected keyword is a member access modifier. A protected member is accessible within its class and by derived class instances.

Question: 9

You are developing an application by using C#. The application includes the following code segment. (Line numbers are included for reference only.)

```
01 public interface IDataContainer
02 {
03   string Data { get; set; }
04 }
05 void DoWork(object obj)
06 {
07
08   if (dataContainer != null)
09   {
10     Console.WriteLine(dataContainer.Data);
11   }
12 }
```

The DoWork() method must not throw any exceptions when converting the obj object to the IDataContainer interface or when accessing the Data property.
You need to meet the requirements. Which code segment should you insert at line 07?

A. var dataContainer = (IDataContainer)obj;
B. dynamic dataContainer = obj;
C. var dataContainer = obj is IDataContainer;
D. var dataContainer = obj as IDataContainer;

Answer: D

Explanation:
As - The as operator is like a cast operation. However, if the conversion isn't possible, as returns null instead of raising an exception.
http://msdn.microsoft.com/en-us/library/cscsdfbt(v=vs.110).aspx

Question: 10

You are creating an application that manages information about zoo animals. The application includes a class named Animal and a method named Save.
The Save() method must be strongly typed. It must allow only types inherited from the Animal class that uses a constructor that accepts no parameters.
You need to implement the Save() method.
Which code segment should you use?

A.

```
public static void Save<T>(T target) where T : new(), Animal
{
   ...
}
```

B.

```
public static void Save<T>(T target) where T : Animal
{
   ...
}
```

C.

```
public static void Save<T>(T target) where T : Animal, new()
{
   ...
}
```

D.

```
public static void Save(Animal target)
{
   ...
}
```

Answer: C

Explanation:
When you define a generic class, you can apply restrictions to the kinds of types that client code can use for type arguments when it instantiates your class. If client code tries to instantiate your class by using a type that is not allowed by a constraint, the result is a compile-time error. These restrictions are called constraints. Constraints are specified by using the where contextual keyword.
http://msdn.microsoft.com/en-us/library/d5x73970.aspx

Question: 11

You are developing an application. The application includes classes named Employee and Person and an interface named IPerson.
The Employee class must meet the following requirements:
It must either inherit from the Person class or implement the IPerson interface.
It must be inheritable by other classes in the application.
You need to ensure that the Employee class meets the requirements.
Which two code segments can you use to achieve this goal? (Each correct answer presents a complete solution. Choose two.)
A.

```
sealed class Employee : Person
{
   ...
}
```

B.

```
abstract class Employee : Person
{
   ...
}
```

C.

```
sealed class Employee : IPerson
{
   ...
}
```

D.

```
abstract class Employee : IPerson
{
   ...
}
```

Answer: B, D

Explanation:
Sealed - When applied to a class, the sealed modifier prevents other classes from inheriting from it. http://msdn.microsoft.com/en-us/library/88c54tsw(v=vs.110).aspx

Question: 12

You are developing an application that will convert data into multiple output formats.
The application includes the following code. (Line numbers are included for reference only.)

```
01 public class TabDelimitedFormatter : IOutputFormatter<string>
02 {
03    readonly Func<int, char> suffix = col => col % 2 == 0 ? '\n' : '\t';
04    public string GetOutput(IEnumerator<string> iterator, int recordSize)
05    {
06
07    }
08 }
```

You are developing a code segment that will produce tab-delimited output. All output routines implement the following interface:

```
public interface IOutputFormatter<T>
{
  string GetOutput(IEnumerator<T> iterator, int recordSize);
}
```

You need to minimize the completion time of the GetOutput() method.
Which code segment should you insert at line 06?

A.

```
string output = null;
for (int i = 1; iterator.MoveNext(); i++)
{
  output = string.Concat(output, iterator.Current, suffix(i));
}
return output;
```

B.

```
var output = new StringBuilder();
for (int i = 1; iterator.MoveNext(); i++)
{
  output.Append(iterator.Current);
  output.Append(suffix(i));
}
return output.ToString();
```

C.

```
string output = null;
for (int i = 1; iterator.MoveNext(); i++)
{
  output = output + iterator.Current + suffix(i);
}
return output;
```

D.

```
string output = null;
for (int i = 1; iterator.MoveNext(); i++)
{
  output += iterator.Current + suffix(i);
}
return output;
```

Answer: B

Explanation:
A String object concatenation operation always creates a new object from the existing string and the new data.
A StringBuilder object maintains a buffer to accommodate the concatenation of new data. New data is appended to the buffer if room is available; otherwise, a new, larger buffer is allocated, data from the original buffer is copied to the new buffer, and the new data is then appended to the new buffer.
The performance of a concatenation operation for a String or StringBuilder object depends on the frequency of memory allocations. A String concatenation operation always allocates memory, whereas a

StringBuilder concatenation operation allocates memory only if the StringBuilder object buffer is too small to accommodate the new data. Use the String class if you are concatenating a fixed number of String objects. In that case, the compiler may even combine individual concatenation operations into a single operation. Use a StringBuilder object if you are concatenating an arbitrary number of strings; for example, if you're using a loop to concatenate a random number of strings of user input.
http://msdn.microsoft.com/en-us/library/system.text.stringbuilder(v=vs.110).aspx

Question: 13

You are developing an application by using C#.
The application includes an object that performs a long running process.
You need to ensure that the garbage collector does not release the object's resources until the process completes.
Which garbage collector method should you use?

A. ReRegisterForFinalize()
B. SuppressFinalize()
C. Collect()
D. WaitForFullGCApproach()

Answer: B

Question: 14

You are creating a class named Employee. The class exposes a string property named EmployeeType. The following code segment defines the Employee class. (Line numbers are included for reference only.)

```
01 public class Employee
02 {
03   internal string EmployeeType
04   {
05     get;
06     set;
07   }
08 }
```

The EmployeeType property value must be accessed and modified only by code within the Employee class or within a class derived from the Employee class.
You need to ensure that the implementation of the EmployeeType property meets the requirements.
Which two actions should you perform? (Each correct answer represents part of the complete solution. Choose two.)

A. Replace line 05 with the following code segment:
protected get;
B. Replace line 06 with the following code segment:
private set;
C. Replace line 03 with the following code segment:
public string EmployeeType
D. Replace line 05 with the following code segment:
private get;

E. Replace line 03 with the following code segment:
protected string EmployeeType
F. Replace line 06 with the following code segment:
protected set;

Answer: B, E

Question: 15

You are implementing a method named Calculate that performs conversions between value types and reference types. The following code segment implements the method. (Line numbers are included for reference only.)

```
01 public static void Calculate(float amount)
02 {
03   object amountRef = amount;
04
05   Console.WriteLine(balance);
06 }
```

You need to ensure that the application does not throw exceptions on invalid conversions.
Which code segment should you insert at line 04?

A. int balance = (int) (float)amountRef;
B. int balance = (int)amountRef;
C. int balance = amountRef;
D. int balance = (int) (double) amountRef;

Answer: A