Vendor: Microsoft

Exam Code: 70-483

Exam Name: Microsoft Programming in C#

Questions and Answers No.: 171-180 (231Q&As)

☆ 100% Pass Guaranteed Or Full Money Back!

☆ Instant Download Access After Payment!

☆ One Year Free Updation!

☆ Well Formated: PDF,VCE,Exam Software!

☆ Multi-Platform capabilities – Windows, Laptop, Mac, Android, iPhone, iPod, iPad.

☆ Pass any exams at the FIRST try!

QUESTION 171
You are developing an application that will transmit large amounts of data between a client computer and a server.
You need to ensure the validity of the data by using a cryptographic hashing algorithm.
Which algorithm should you use?

A. ECDsa
B. RNGCryptoServiceProvider
C. Rfc2898DeriveBytes
D. HMACSHA512

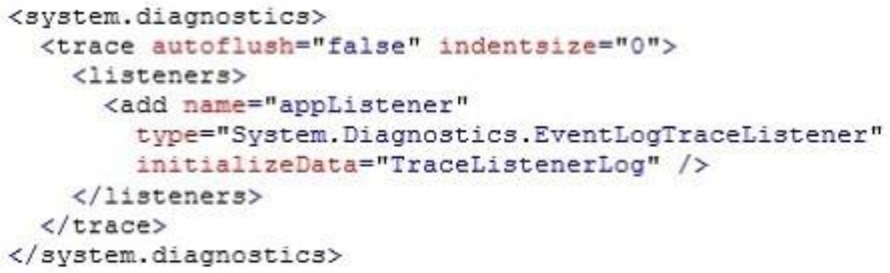
Answer: D

QUESTION 172
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. RemoveMemoryPressure()
B. ReRegisterForFinalize()
C. WaitForFullGCComplete()
D. KeepAlive()

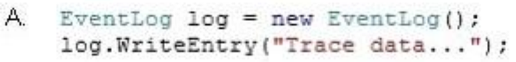
Answer: D

QUESTION 173
You are developing an application that uses a .config file.
The relevant portion of the .config file is shown as follows:



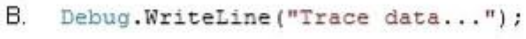
```
<system.diagnostics>
  <trace autoflush="false" indentsize="0">
    <listeners>
      <add name="appListener"
        type="System.Diagnostics.EventLogTraceListener"
        initializeData="TraceListenerLog" />
    </listeners>
  </trace>
</system.diagnostics>
```

You need to ensure that diagnostic data for the application writes to the event tog by using the configuration specified in the .config file.
What should you include in the application code?



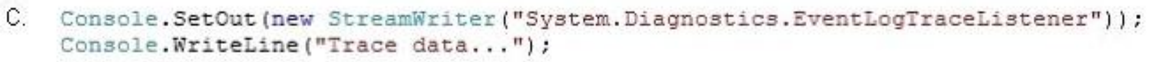
A.

```
EventLog log = new EventLog();
log.WriteEntry("Trace data...");
```



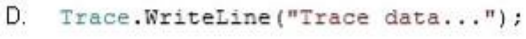
B.

```
Debug.WriteLine("Trace data...");
```



C.

```
Console.SetOut(new StreamWriter("System.Diagnostics.EventLogTraceListener"));
Console.WriteLine("Trace data...");
```



D.

```
Trace.WriteLine("Trace data...");
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: D
Explanation:
Debug.Writeline() statements will not be included in the Release compilation by default, whereas Trace.Writeline statements will be included.

QUESTION 174
You use the Task.Run() method to launch a long-running data processing operation.
The data processing operation often fails in times of heavy network congestion.
If the data processing operation fails, a second operation must clean up any results of the first operation.
You need to ensure that the second operation is invoked only if the data processing operation throws an unhandled exception.
What should you do?

A. Create a task within the operation, and set the Task.StartOnError property to true.
B. Create a TaskFactory object and call the ContinueWhenAll() method of the object.
C. Create a task by calling the Task.ContinueWith() method.
D. Use the TaskScheduler class to create a task and call the TryExecuteTask() method on the class.

Answer: C

QUESTION 175
You are implementing a method named ProcessReports that performs a long-running task.
The ProcessReports() method has the following method signature:

```
public void ProcessReports(List<decimal> values,CancellationTokenSource
cts, CancellationToken ct)
```

If the calling code requests cancellation, the method must perform the following actions:

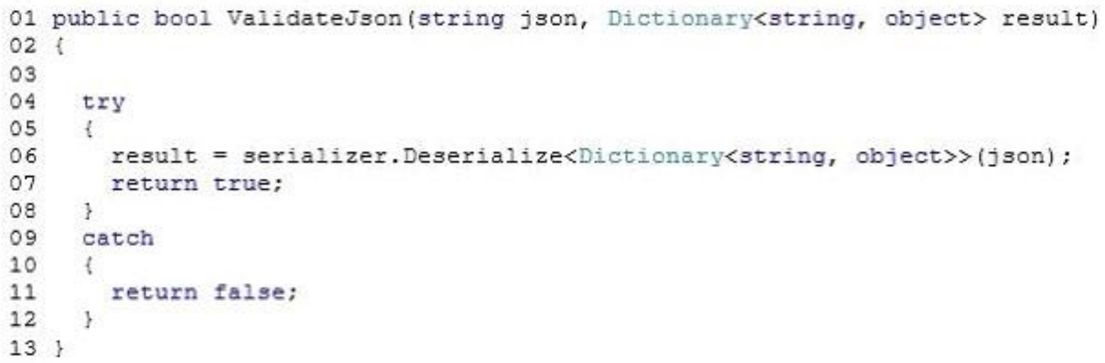
```
- Cancel the long-running task.
- Set the task status to TaskStatus.Canceled.
```

You need to ensure that the ProcessReports() method performs the required actions.
Which code segment should you use in the method body?

A. `if (ct.IsCancellationRequested)`
   `return;`
B. `ct.ThrowIfCancellationRequested() ;`
C. `cts.Cancel();`
D. `throw new AggregateException();`

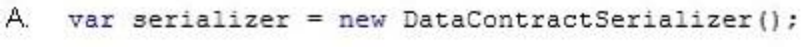
Answer: B

QUESTION 176
You are creating a console application named App1.
App1 retrieves data from the Internet by using JavaScript Object Notation (JSON).
You are developing the following code segment (line numbers are included for reference only):

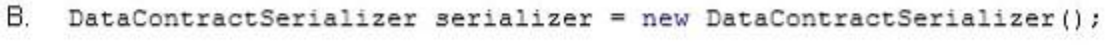


```
01 public bool ValidateJson(string json, Dictionary<string, object> result)
02 {
03
04   try
05   {
06     result = serializer.Deserialize<Dictionary<string, object>>(json);
07     return true;
08   }
09   catch
10   {
11     return false;
12   }
13 }
```

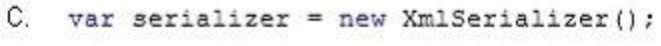
You need to ensure that the code validates the JSON string.
Which code should you insert at line 03?



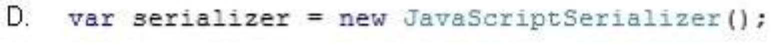
```
A.  var serializer = new DataContractSerializer();
```



```
B.  DataContractSerializer serializer = new DataContractSerializer();
```



```
C.  var serializer = new XmlSerializer();
```



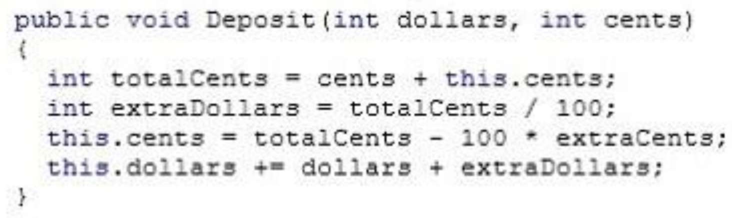
```
D.  var serializer = new JavaScriptSerializer();
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: D
Explanation:
The JavaScriptSerializer Class Provides serialization and deserialization functionality for AJAX-enabled applications.
The JavaScriptSerializer class is used internally by the asynchronous communication layer to serialize and deserialize the data that is passed between the browser and the Web server.
You cannot access that instance of the serializer. However, this class exposes a public API.

Therefore, you can use the class when you want to work with JavaScript Object Notation (JSON) in managed code.

QUESTION 177
You are developing code for a class named Account.
The Account class includes the following method:
You need to ensure that overflow exceptions are thrown when there is an error.
Which type of block should you use?

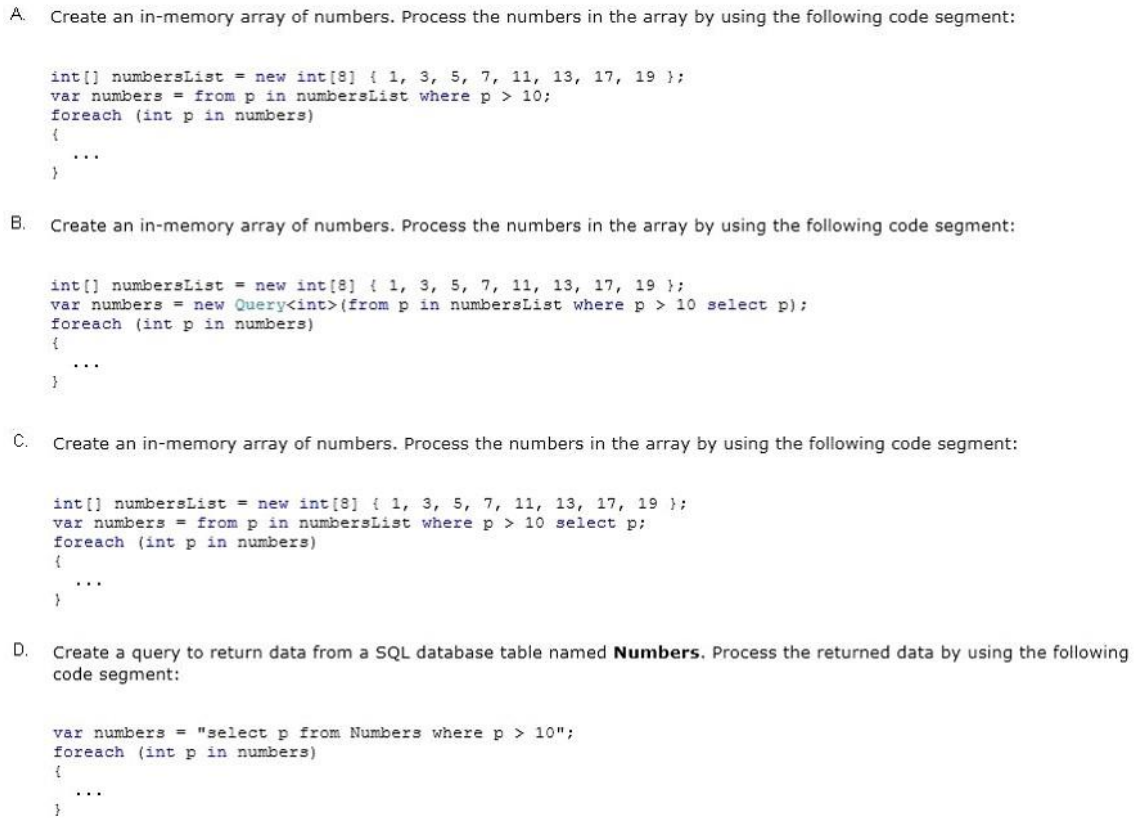


```
public void Deposit(int dollars, int cents)
{
    int totalCents = cents + this.cents;
    int extraDollars = totalCents / 100;
    this.cents = totalCents - 100 * extraCents;
    this.dollars += dollars + extraDollars;
}
```

A. checked
B. try
C. using
D. unchecked

Answer: A

QUESTION 178
You are creating an application that processes a list of numbers.
The application must define a method that queries the list and displays a subset of the numbers to the user. The method must not update the list.
You need to create an extendable query by using LINQ.
What should you do?



A. Create an in-memory array of numbers. Process the numbers in the array by using the following code segment:

```
int[] numbersList = new int[8] { 1, 3, 5, 7, 11, 13, 17, 19 };
var numbers = from p in numbersList where p > 10;
foreach (int p in numbers)
{
  ...
}
```

B. Create an in-memory array of numbers. Process the numbers in the array by using the following code segment:

```
int[] numbersList = new int[8] { 1, 3, 5, 7, 11, 13, 17, 19 };
var numbers = new Query<int>(from p in numbersList where p > 10 select p);
foreach (int p in numbers)
{
  ...
}
```

C. Create an in-memory array of numbers. Process the numbers in the array by using the following code segment:

```
int[] numbersList = new int[8] { 1, 3, 5, 7, 11, 13, 17, 19 };
var numbers = from p in numbersList where p > 10 select p;
foreach (int p in numbers)
{
  ...
}
```

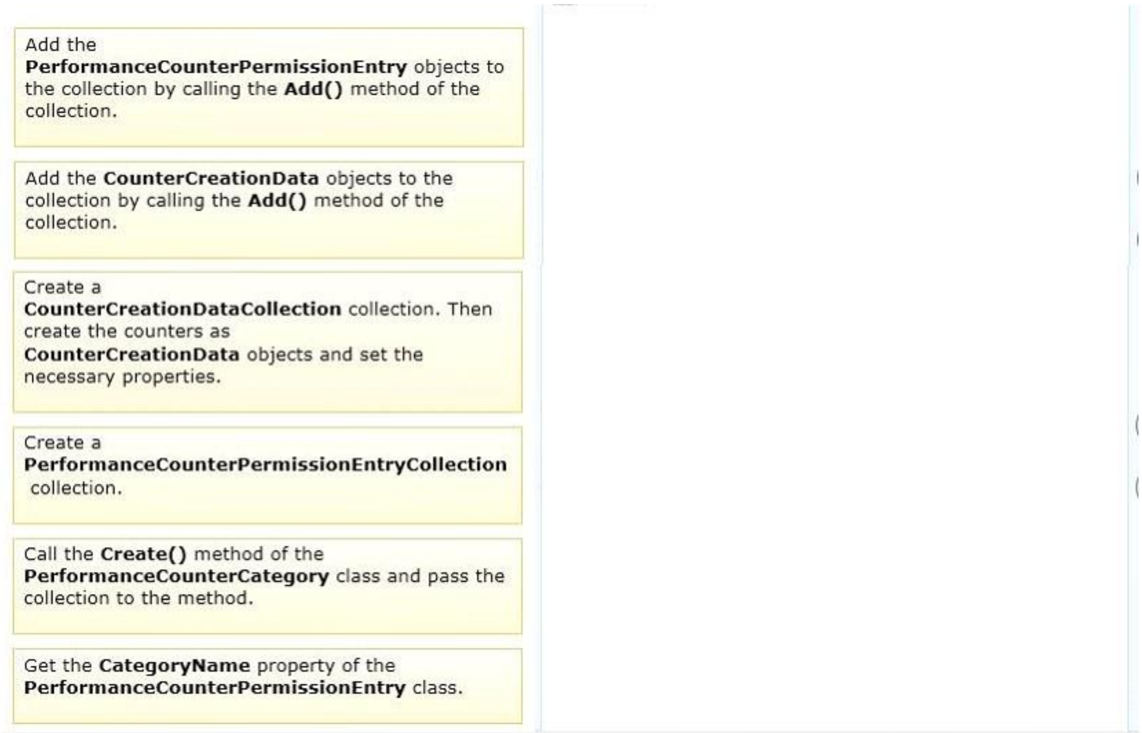
D. Create a query to return data from a SQL database table named **Numbers**. Process the returned data by using the following code segment:

```
var numbers = "select p from Numbers where p > 10";
foreach (int p in numbers)
{
  ...
}
```

A. Option A
B. Option B
C. Option C
D. Option D

Answer: C

QUESTION 179
Drag and Drop Question
You are developing an application by using C#.
The application will process several objects per second.
You need to create a performance counter to analyze the object processing.
Which three actions should you perform in sequence? (To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.)



Add the **PerformanceCounterPermissionEntry** objects to the collection by calling the **Add()** method of the collection.

Add the **CounterCreationData** objects to the collection by calling the **Add()** method of the collection.

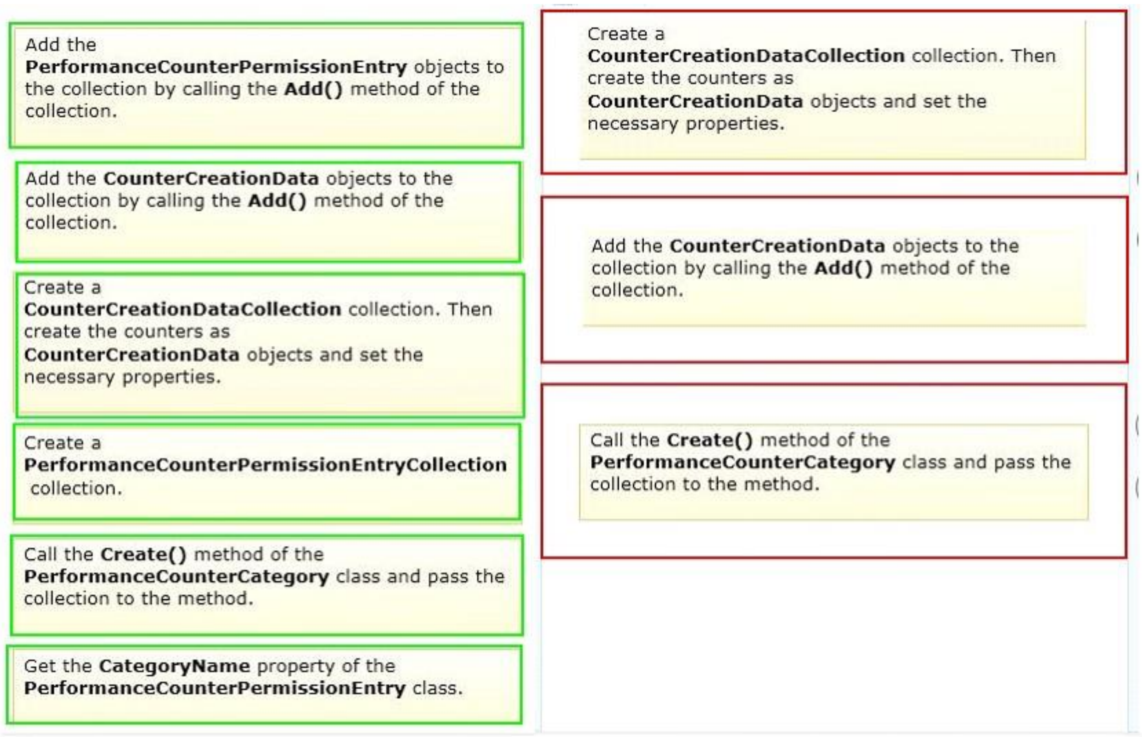
Create a **CounterCreationDataCollection** collection. Then create the counters as **CounterCreationData** objects and set the necessary properties.

Create a **PerformanceCounterPermissionEntryCollection** collection.

Call the **Create()** method of the **PerformanceCounterCategory** class and pass the collection to the method.

Get the **CategoryName** property of the **PerformanceCounterPermissionEntry** class.

Answer:



Add the **PerformanceCounterPermissionEntry** objects to the collection by calling the **Add()** method of the collection.

Add the **CounterCreationData** objects to the collection by calling the **Add()** method of the collection.

Create a **CounterCreationDataCollection** collection. Then create the counters as **CounterCreationData** objects and set the necessary properties.

Create a **PerformanceCounterPermissionEntryCollection** collection.

Call the **Create()** method of the **PerformanceCounterCategory** class and pass the collection to the method.

Get the **CategoryName** property of the **PerformanceCounterPermissionEntry** class.

Create a **CounterCreationDataCollection** collection. Then create the counters as **CounterCreationData** objects and set the necessary properties.

Add the **CounterCreationData** objects to the collection by calling the **Add()** method of the collection.

Call the **Create()** method of the **PerformanceCounterCategory** class and pass the collection to the method.

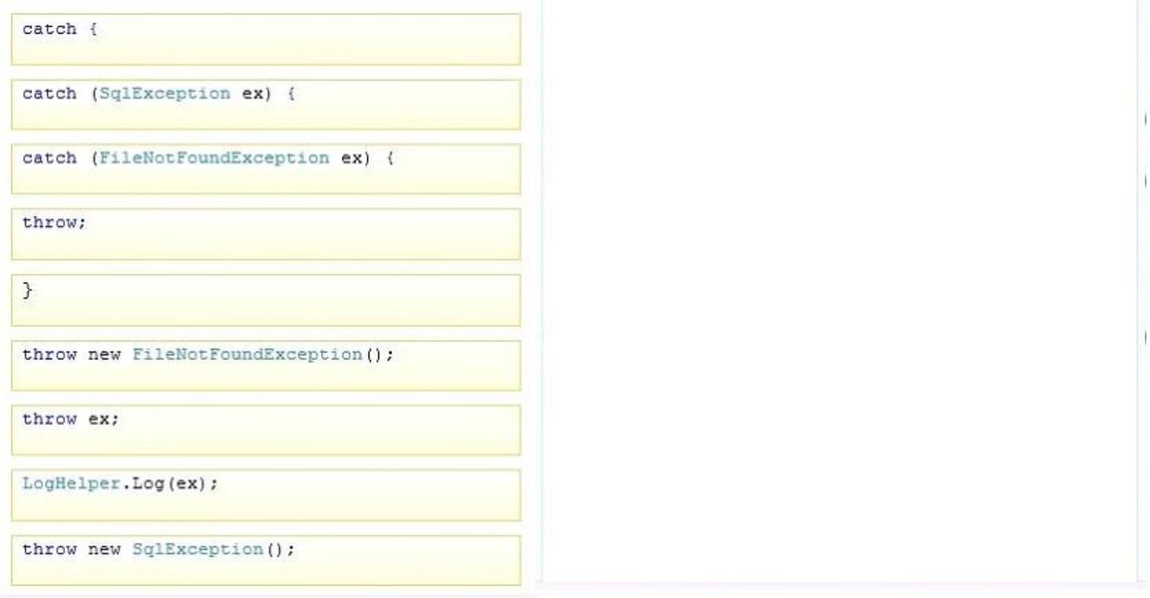
QUESTION 180

Drag and Drop Question

You are creating a method that saves information to a database.

You have a static class named LogHelper.

LogHelper has a method named Log to log the exception.
You need to use the LogHelper Log method to log the exception raised by the database server.
The solution must ensure that the exception can be caught by the calling method, while preserving the original stack trace.
How should you write the catch block? (Develop the solution by selecting and ordering the required code snippets. You may not need all of the code snippets.)



```
catch {
```

```
catch (SqlException ex) {
```

```
catch (FileNotFoundException ex) {
```

```
throw;
```

```
}
```

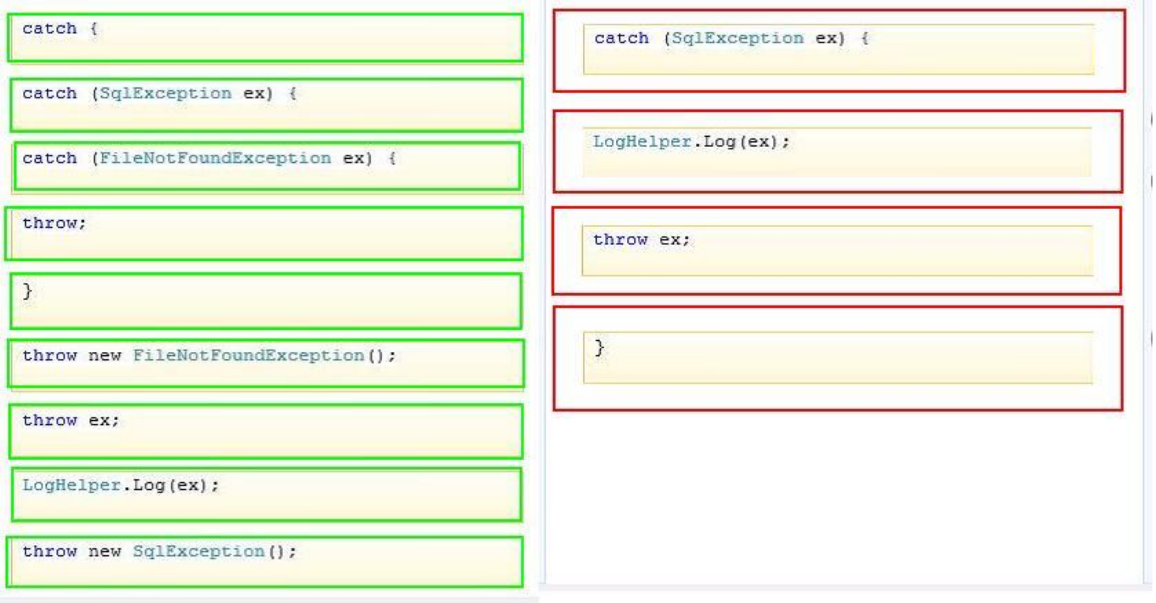
```
throw new FileNotFoundException();
```

```
throw ex;
```

```
LogHelper.Log(ex);
```

```
throw new SqlException();
```

Answer:



```
catch {
```

```
catch (SqlException ex) {
```

```
catch (FileNotFoundException ex) {
```

```
throw;
```

```
}
```

```
throw new FileNotFoundException();
```

```
throw ex;
```

```
LogHelper.Log(ex);
```

```
throw new SqlException();
```

```
catch (SqlException ex) {
```

```
LogHelper.Log(ex);
```

```
throw ex;
```

```
}
```