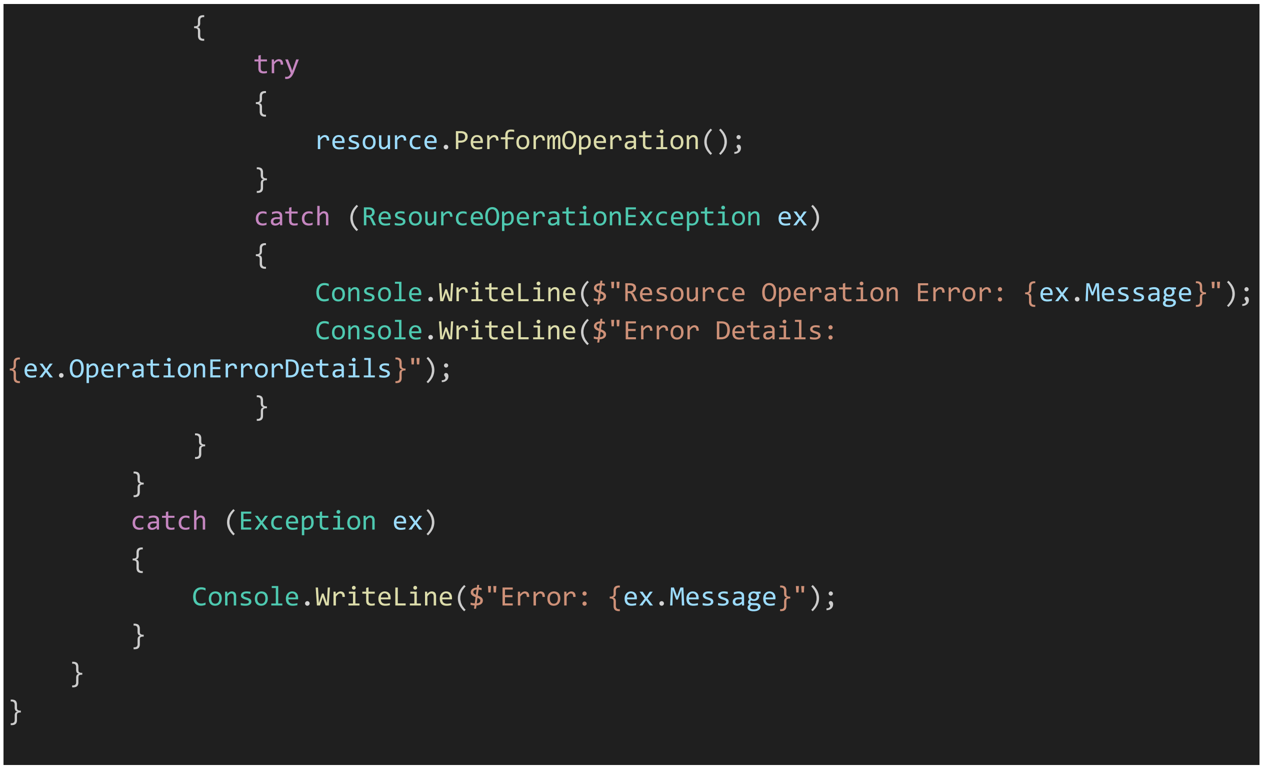
GARBAGE COLLECTION

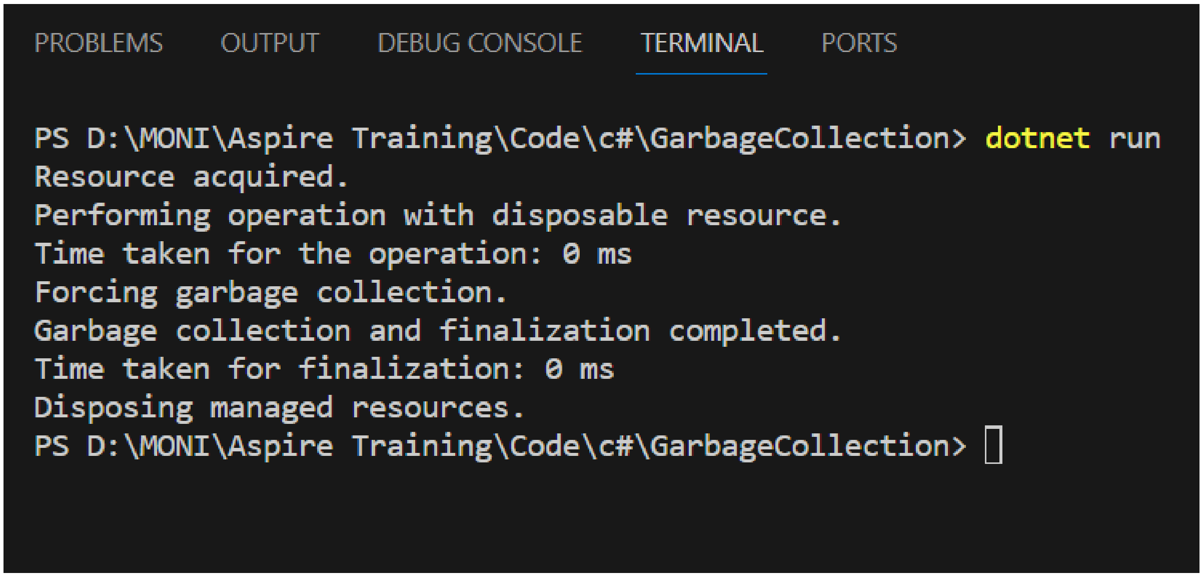
Code:

```
//Title: Garbage Collection
//Author: Monisha S
//Created on: 08/01/2024
//Updated on: 08/01/2024
//Reviewed by:
//Reviewed on:

using System;

public class ResourceOperationException : Exception
{
    public string OperationErrorDetails { get; }

    public ResourceOperationException(string message, string
operationErrorDetails)
        : base(message)
    {
        OperationErrorDetails = operationErrorDetails;
    }
}

public class DisposableResource : IDisposable
{
    private bool disposed = false;

    public DisposableResource()
    {
        Console.WriteLine("Resource acquired.");
    }

    public void PerformOperation()
    {
        if (disposed)
        {
            throw new ObjectDisposedException(nameof(DisposableResource),
"Resource has been disposed.");
        }

        Console.WriteLine("Performing operation with disposable resource.");
        // Simulate an error condition (replace this with your specific error
condition)
        if (DateTime.Now.Second % 2 == 0)
        {
            string errorDetails = "Additional details about the error.";
            throw new ResourceOperationException("Error during the operation with
disposable resource.", errorDetails);
        }
    }

    protected virtual void Dispose(bool disposing)
    {
        if (!disposed)
        {
            if (disposing)
            {
                // Dispose managed resources here
                Console.WriteLine("Disposing managed resources.");
            }

            // Dispose unmanaged resources here (if any)

            disposed = true;
        }
    }

    public void Dispose()
    {
        Dispose(true);
        GC.SuppressFinalize(this);
    }

    ~DisposableResource()
    {
        Dispose(false);
    }
}

class Program
{
    static void Main()
    {
        try
        {
            using (DisposableResource resource = new DisposableResource())
```



```
            {
                try
                {
                    resource.PerformOperation();
                }
                catch (ResourceOperationException ex)
                {
                    Console.WriteLine($"Resource Operation Error: {ex.Message}");
                    Console.WriteLine($"Error Details:
{ex.OperationErrorDetails}");
                }
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error: {ex.Message}");
        }
    }
}
```

Output:



```
PS D:\MONI\Aspire Training\Code\c#\GarbageCollection> dotnet run
Resource acquired.
Performing operation with disposable resource.
Time taken for the operation: 0 ms
Forcing garbage collection.
Garbage collection and finalization completed.
Time taken for finalization: 0 ms
Disposing managed resources.
PS D:\MONI\Aspire Training\Code\c#\GarbageCollection>
```