

```

using System;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            using (SomePublisher publisher = new SomePublisher())
            {
                for (int i = 0; i < 10; i++)
                {
                    SomeSubscriber subscriber = new SomeSubscriber(publisher);
                    subscriber = null;
                }

                GC.Collect();
                GC.WaitForPendingFinalizers();

                Console.WriteLine(SomeSubscriber.Count.ToString());
            }
        }
    }

    public class SomePublisher: IDisposable
    {
        public event EventHandler SomeEvent;

        bool disposed = false;
        public void Dispose() {
            Dispose(true);
            GC.SuppressFinalize(this);
        }

        protected virtual void Dispose(bool disposing)
        {
            if (disposed)
                return;

            if (disposing)
            {
                SomeEvent = null;
            }

            disposed = true;
        }
    }
}

```

```
}

public class SomeSubscriber
{
    public static int Count;

    public SomeSubscriber(SomePublisher publisher)
    {
        publisher.SomeEvent += new EventHandler(publisher_SomeEvent);
    }

    ~SomeSubscriber()
    {
        SomeSubscriber.Count++;
    }

    private void publisher_SomeEvent(object sender, EventArgs e)
    {
        //TODO: something
    }
}

}
```