Memory

Professional C# 2021 – Chapter 13

https://csharp.christiannagel.com

Memory

Stack

- Freed on method end

Managed Heap

- Managed by the Garbage Collector (GC)
- Generations 0 – 1 – 2

Native Heap

- You’re responsibility
- Native Code

GC - LatencyMode

- Batch
  - disable concurrency settings, maximum throughput for GC
- Interactive
  - default for workstation
  - Concurrency, balances throughput and responsiveness
- LowLatency
  - conservative GC
  - full collections only with memory pressure
- NoGCRegion
  - TryStartNoGCRegion, EndNoGCRegion)
  - Temporary stop GC runs

WeakReference

- Compare to strong references
- GC can collect
- Use for caches

Release Resources

- Destructor (Finalizer)
- IDisposable

Release

- Call Dispose within a try/finally block
- using statement
- using declaration

Guidelines Resources

- If your class defines a member that is IDisposable, implement IDisposable
- Implementing IDisposable does not necessary require a finalizer
- Implementing a finalizer requires IDisposable
- Within finalization code, don’t access other objects that might have been finalized already
- Invoke Dipose if the object is no longer needed

Summary

- Garbage collector
- Finalizer
- IDisposable
- using