Yesterday!

1. Why .NET.
2. What is .NET Flw.
3. Diff. standard .NET vs .NET Core.
4. FCL, BCL, CoreFx, CLR, JIT, MSIL, Assembly
5. JIT – 1) Standard JIT Compiler
   2) Pre-JIT Compiler ✓
6. Diff. .exe and .dll → ildasm
7. Private assembly – default
   Shared assembly – e.g flw
   } .dll
8. Flw versions & CLR versions
9. .sln files and .csproj files
10. Visual Studio 2022 templates

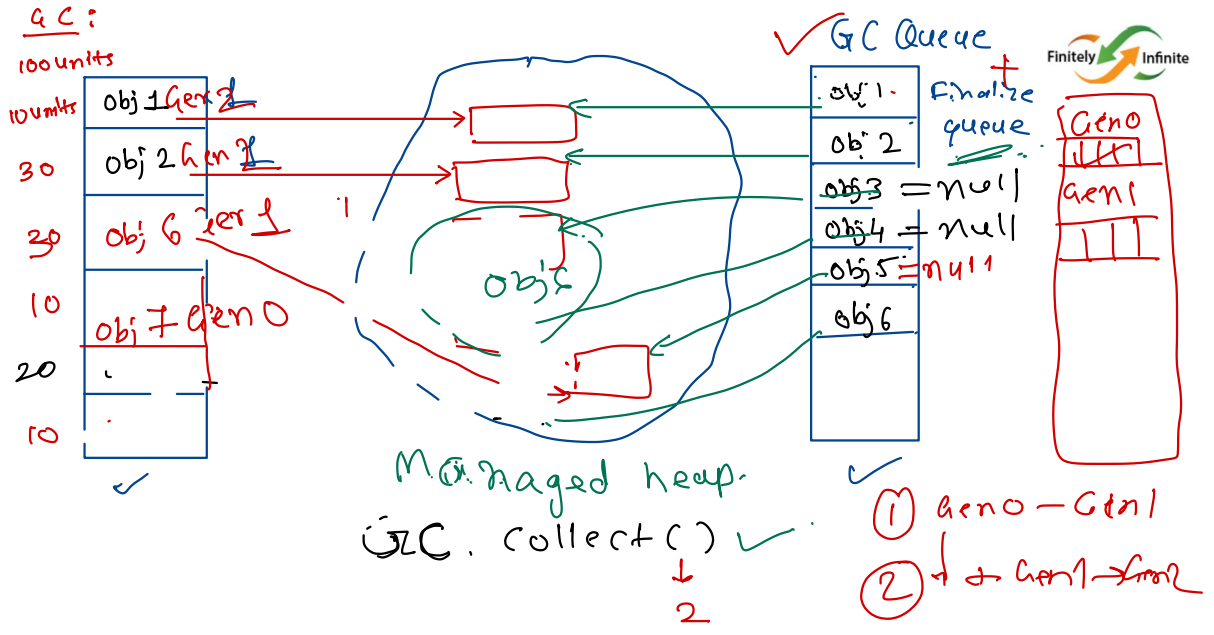
11. CTS, GAC, CLS → use "new" keyword to create new objects

12. CLR features :→

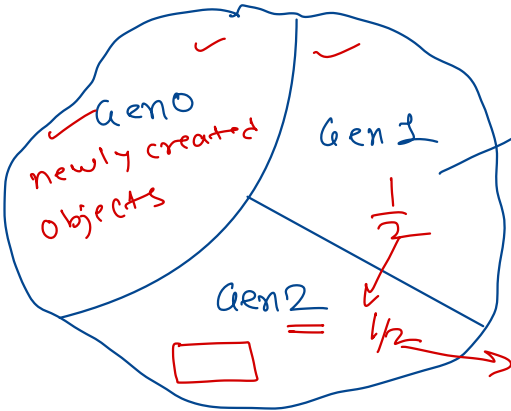
1. load the .NET assemblies
2. allocate memory to managed code.
3. Execute managed code
4. JIT compilation.
5. Security.
6. Exception Handling.
7. Execution of assembly → feed the mic code to CPU
8. Garbage Collecter — GC

8275006278 mugdha@fispl.co.in

GC:


100 units
10 units
30
30
10
20
10
obj 1 Gen 2
obj 2 Gen 2
obj 6 Gen 1
obj 7 Gen 0
GC Queue
obj 1
obj 2
obj 3 = null
obj 4 = null
obj 5 = null
obj 6
Finalize queue
Gen 0
Gen 1
obj 5
Managed heap.
GC.collect()
2
1 Gen 0 – Gen 1
2 Gen 1 → Gen 2

managed heap



survived objects after GC happens.

survived obj.

obj → 85,000 + Gen2.

Using System;

Using System.IO;

declaration of namespace, → logical Grouping of Types e.g:— class, interface, delegate, abstract static

System.Console.WriteLine( ); → o/p

Console.ReadLine( ); → i/p

Console → static

System.Console.ReadLine( );

System.IO → namespace

File.GetDir( ) → class, method

1. Features of CLR ✓
2. GC ✓
3. namespace, using keyword. ✓
4. Datatypes ✓
5. i/n / o/p ✓
6. OOP → inheritance, overloading, virtual, overriding, new, this, sealed classes, sealed methods, base, :, constructor overloading, mapping, class, methods,
7. Debugging.
8. Boxing, Unboxing, type casting
9. Exception class
10. loops