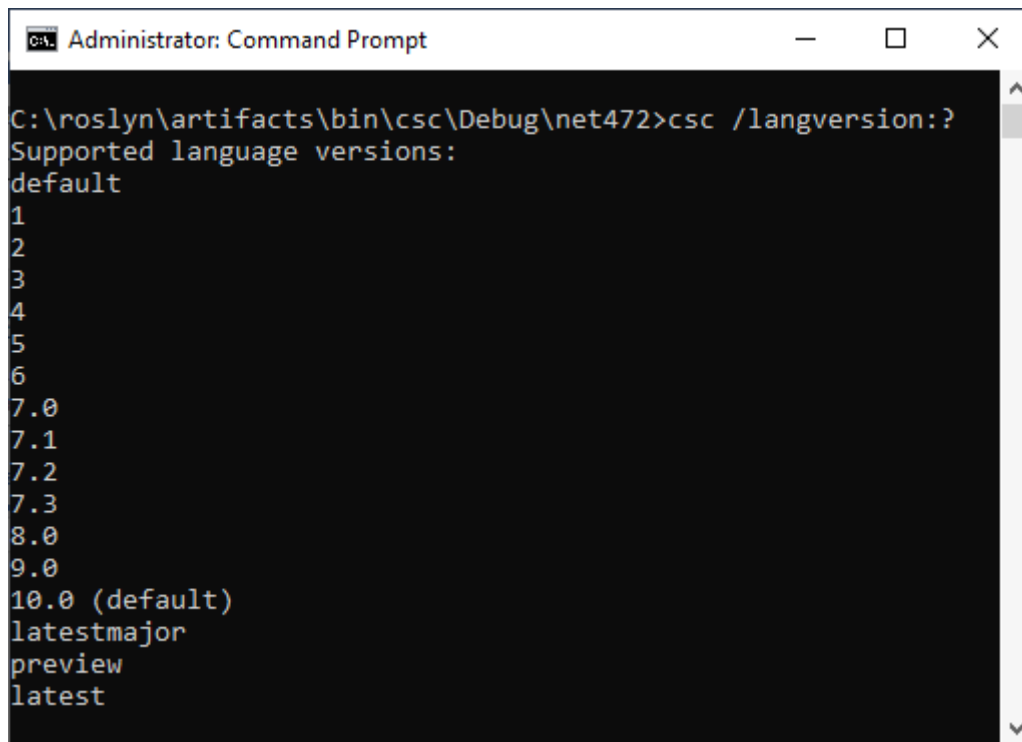
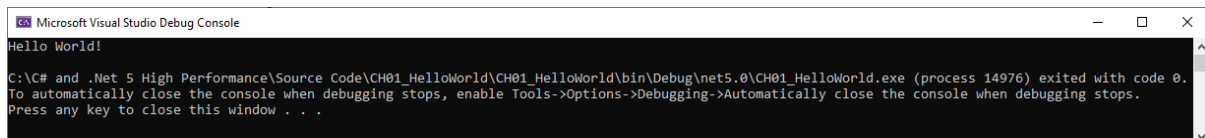


Chapter 1: Introducing C# 10.0 and .NET 6



```
Administrator: Command Prompt

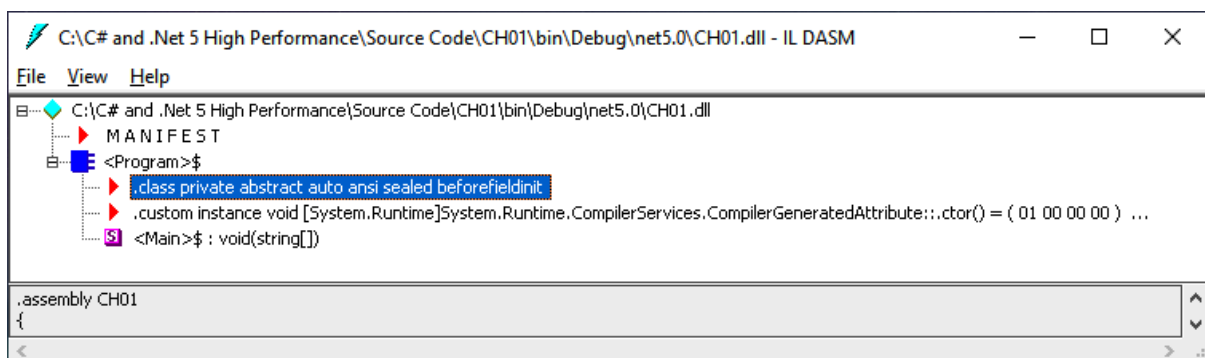
C:\roslyn\artifacts\bin\csc\Debug\net472>csc /langversion:?
Supported language versions:
default
1
2
3
4
5
6
7.0
7.1
7.2
7.3
8.0
9.0
10.0 (default)
latestmajor
preview
latest
```



```
Microsoft Visual Studio Debug Console

Hello World!

C:\C# and .Net 5 High Performance\Source Code\CH01_HelloWorld\CH01_HelloWorld\bin\Debug\net5.0\CH01_HelloWorld.exe (process 14976) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```



```
C:\C# and .Net 5 High Performance\Source Code\CH01\bin\Debug\net5.0\CH01.dll - IL DASM
File View Help

C:\C# and .Net 5 High Performance\Source Code\CH01\bin\Debug\net5.0\CH01.dll
├── MANIFEST
├── <Program>$
│   ├── .class private abstract auto ansi sealed beforefieldinit
│   ├── .custom instance void [System.Runtime]System.Runtime.CompilerServices.CompilerGeneratedAttribute::.ctor() = ( 01 00 00 00 ) ...
│   └── <Main>$ : void(string[])
└── .assembly CH01
    {
```

```
Microsoft Visual Studio Debug Console
Made up book name is written by Made Up Author. Well worth reading!
C:\_book\C-9-and-.NET-5-High-Performance\CH01\CH01_Books\CH01_Books\bin\Debug\net5.0\CH01_Books.exe (process 25852) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console
Some of Made Up Author's books include:
- Made Up Book
- And Another Made Up Book
- Yet Another Made Up Book
- And Yet Another Made Up Book: Part 1
- And Yet Another Made Up Book: Part 2
- And Yet Another Made Up Book: Part 3
My favourite book by Made Up Author is Made Up Book.
C:\_book\C-9-and-.NET-5-High-Performance\CH01\CH01_Records\CH01_Records\bin\Debug\net5.0\CH01_Records.exe (process 23440) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console

Some of Made Up Author's books include:

- Made Up Book
- And Another Made Up Book
- Yet Another Made Up Book
- And Yet Another Made Up Book: Part 1
- And Yet Another Made Up Book: Part 2
- And Yet Another Made Up Book: Part 3

My favourite book by Made Up Author is Made Up Book.
These books were originally published by Made Up Publisher Ltd..

C:\_book\C-9-and-.NET-5-High-Performance\CH01\CH01_Records\CH01_Records\bin\Debug\net5.0\CH01_Records.exe
(process 25628) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console

Some of Made Up Author's books include:

- Made Up Book
- And Another Made Up Book
- Yet Another Made Up Book
- And Yet Another Made Up Book: Part 1
- And Yet Another Made Up Book: Part 2
- And Yet Another Made Up Book: Part 3

My favourite book by Made Up Author is Made Up Book.
These books were originally published by Made Up Publisher Ltd..
Are Made Up Book and Made Up Book equal? Yes

C:\_book\C-9-and-.NET-5-High-Performance\CH01\CH01_Records\CH01_Records\bin\Debug\net5.0\CH01_Records.exe
(process 11332) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console

Some of Made Up Author's books include:

- Made Up Book
- And Another Made Up Book
- Yet Another Made Up Book
- And Yet Another Made Up Book: Part 1
- And Yet Another Made Up Book: Part 2
- And Yet Another Made Up Book: Part 3

My favourite book by Made Up Author is Made Up Book.
These books were originally published by Made Up Publisher Ltd..
Are Made Up Book and Made Up Book equal? Yes
The product called Awesome-X is an Advanced Multi-Language IDE.

C:\_book\C-9-and-.NET-5-High-Performance\CH01\CH01_Records\CH01_Records\bin\Debug\net5.0\CH01_Records.exe (process 16456) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console

The discount for Order One is 10%.
The discount for Order Two is 20%.
The discount for Order Three is 30%.
The discount for Order One is 10%.
The discount for Order Two is 20%.
The discount for Order Three is 30%.

C:\C# and .Net 5 High Performance\Source Code\CH01_PatternMatching\CH01_PatternMatching\bin\Debug\net5.0\CH01_PatternMatching.exe (process 18992) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console

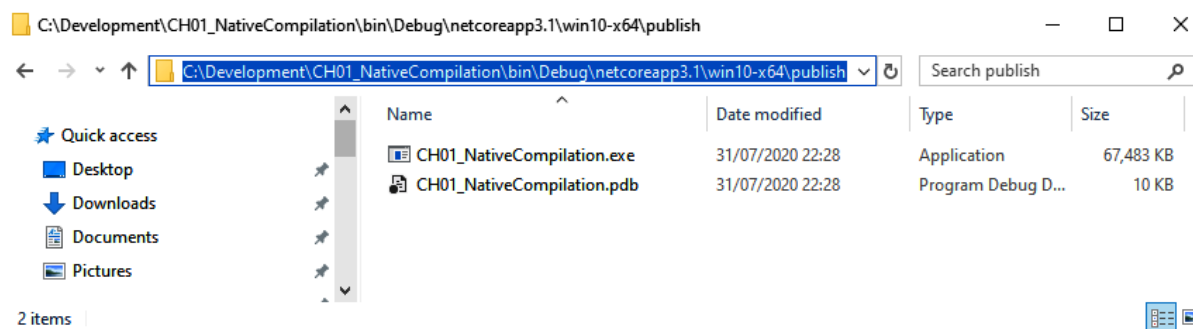
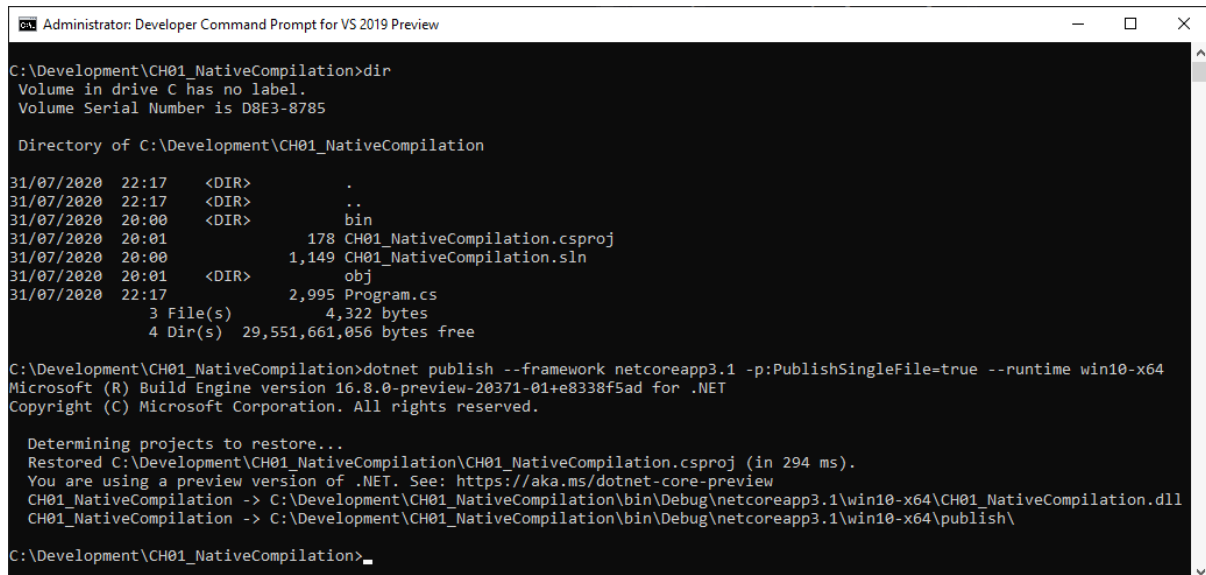
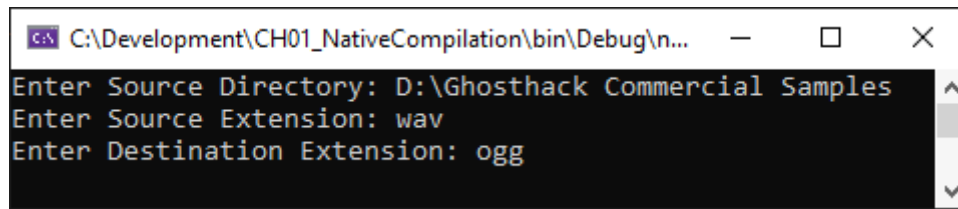
The discount for Order One is 10%.
The discount for Order Two is 20%.
The discount for Order Three is 30%.
The discount for Order One is 10%.
The discount for Order Two is 20%.
The discount for Order Three is 30%.
The discount for Order One is 10%.
The discount for Order Two is 20%.
The discount for Order Three is 30%.

C:\C# and .Net 5 High Performance\Source Code\CH01_PatternMatching\CH01_PatternMatching\bin\Debug\net5.0\CH01_PatternMatching.exe (process 21756) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

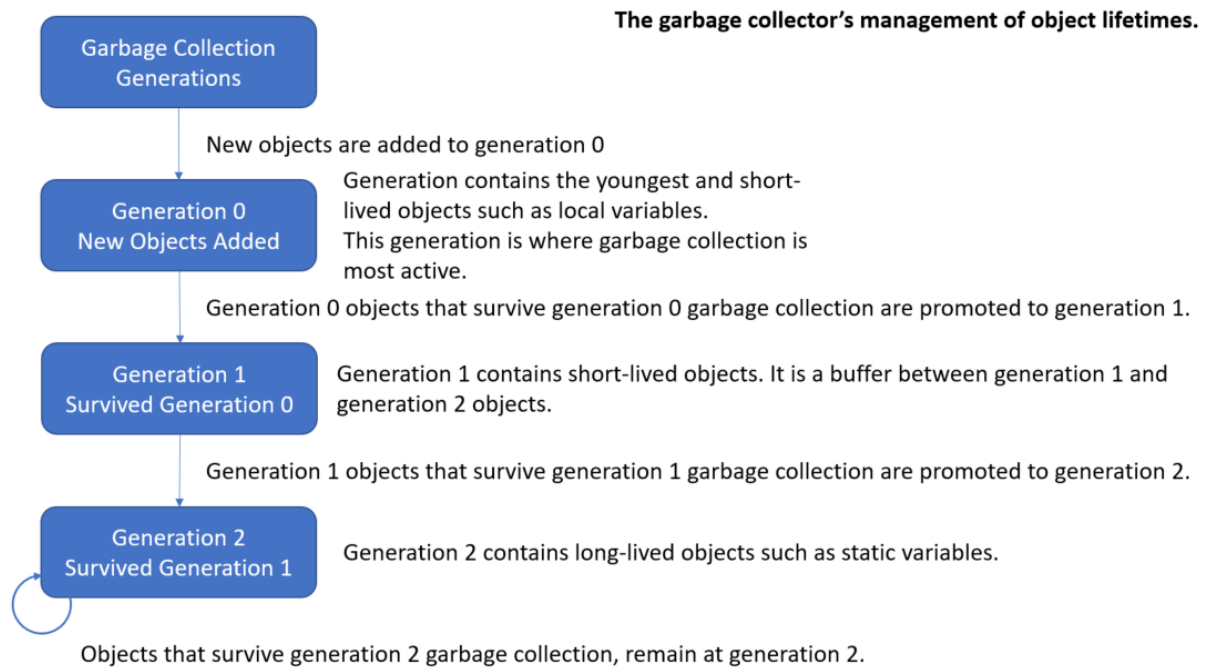
```
Microsoft Visual Studio Debug Console

Student: Albright, Jennifer
Student: Braithwait, Lydia

C:\C# and .Net 5 High Performance\Source Code\CH01_TypeTypeExpressions\CH01_TypeTypeExpressions\bin\Debug\net5.0\CH01_TypeTypeExpressions.exe (process 13052) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```



Chapter 2: Implementing C# Interoperability



Administrator: Command Prompt

```
C:\WINDOWS\system32>"C:\Program Files (x86)\Microsoft Visual
Studio\2019\Preview\VC\Tools\MSVC\14.28.29115\bin\Hostx64\x64
\dumpbin.exe" /exports user32.dll
Microsoft (R) COFF/PE Dumper Version 14.28.29115.0
Copyright (C) Microsoft Corporation. All rights reserved.
```

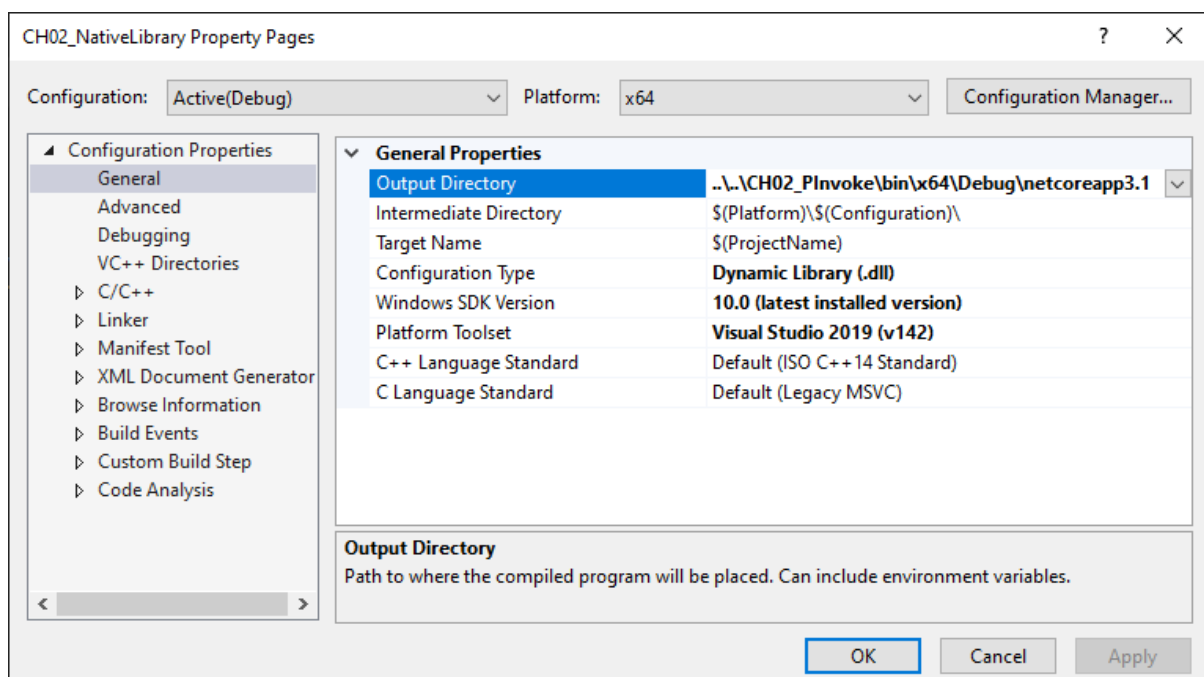
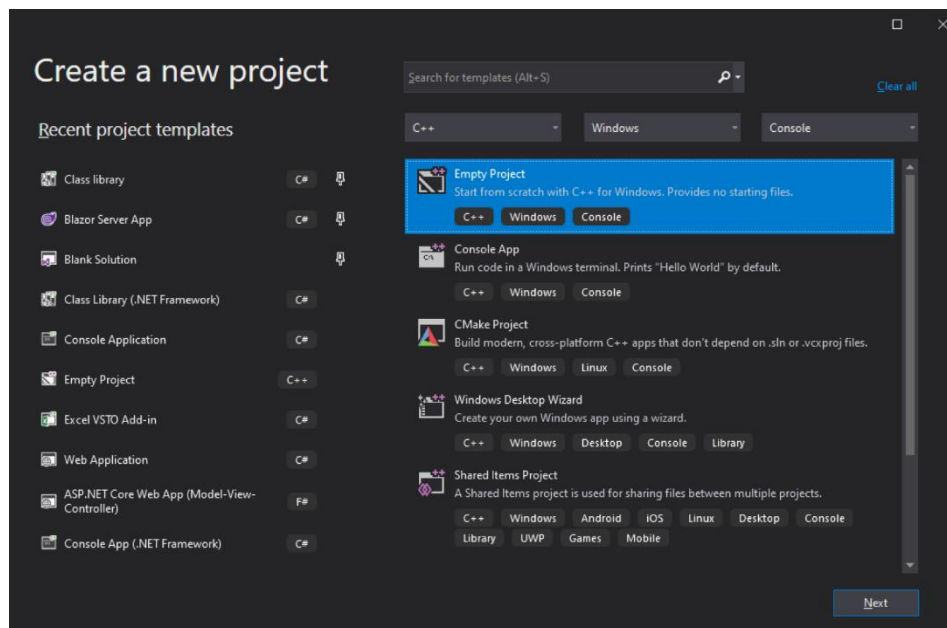
Dump of file user32.dll

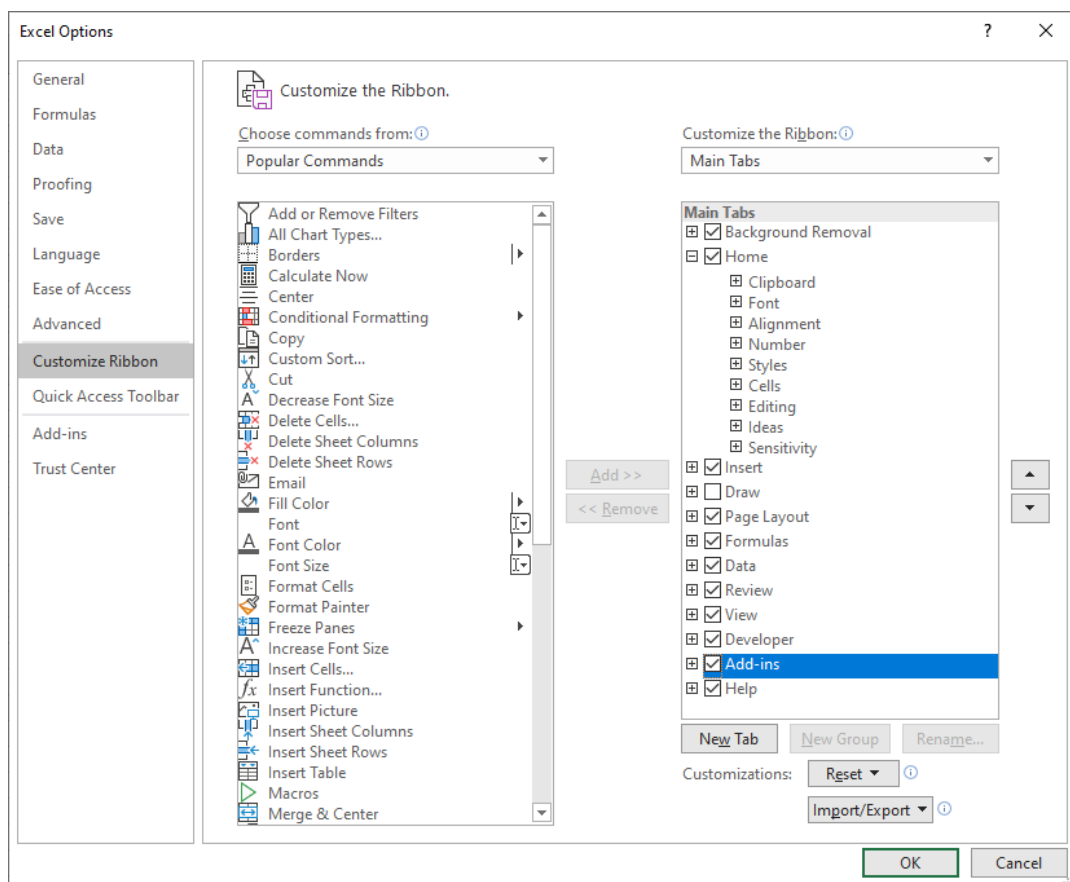
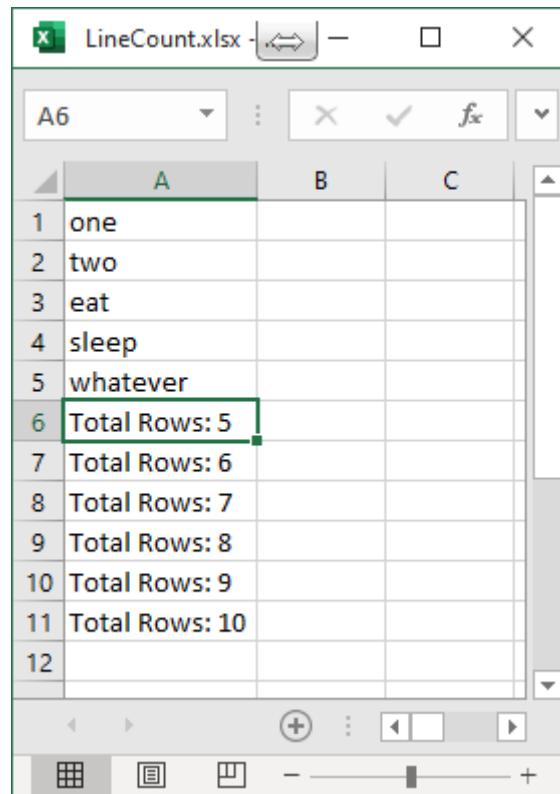
File Type: DLL

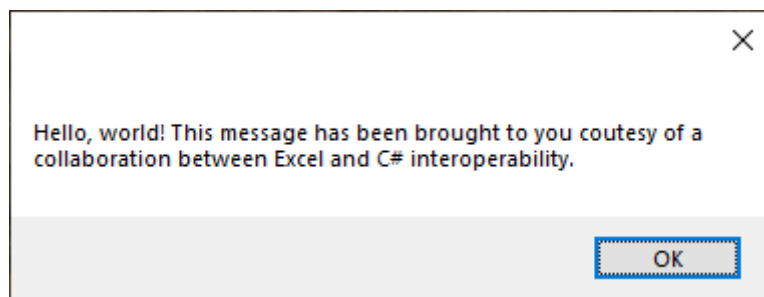
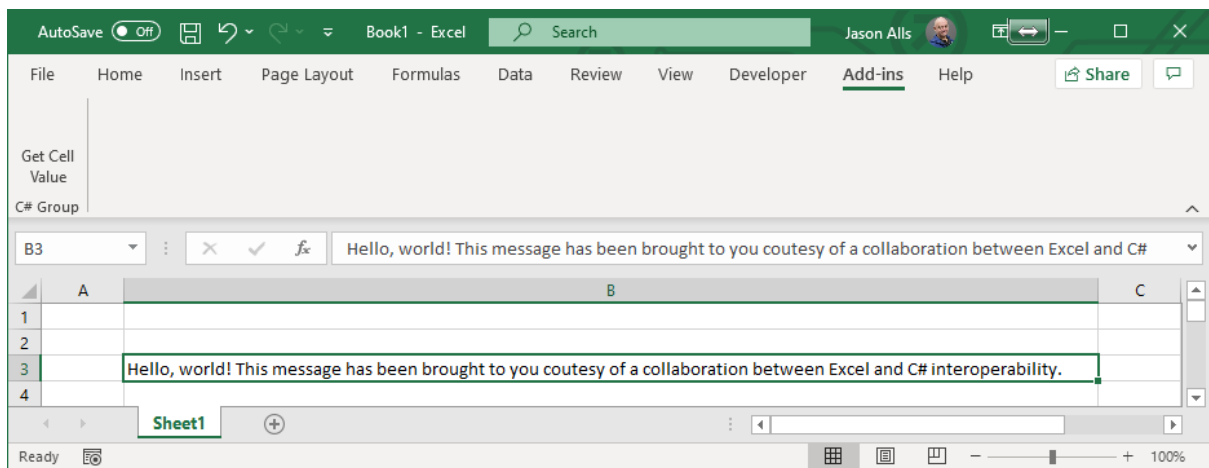
Section contains the following exports for USER32.dll

```
00000000 characteristics
1B20969D time date stamp
0.00 version
1502 ordinal base
1215 number of functions
1004 number of names
```

ordinal	hint	RVA	name
1504	0	0002C0C0	ActivateKeyboardLayout
1505	1	0002C8A0	AddClipboardFormatListener
1506	2	000330F0	AddVisualIdentifier
1507	3	00087900	AdjustWindowRect
1508	4	000160B0	AdjustWindowRectEx
1509	5	0000FEA0	AdjustWindowRectExForDpi
1510	6	0008D880	AlignRects
1511	7	00087970	AllowForegroundActivation
1512	8	0002B130	AllowSetForegroundWindow
1513	9	000822E0	AnimateWindow







Chapter 3: Predefined Data Types and Memory Allocations

```
Select Microsoft Visual Studio Debug Console

Chapter 3: Strings are immutable
greeting1=Hello, world!
greeting2=Hello, world!
greeting1=Hello, world! Isn't life grand!
greeting1=Hello, world!
```

```
Select Administrator: Command Prompt

// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19041.450 (2004/7/20H1)
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.100-rc.1.20452.10
[Host] : .NET Core 5.0.0 (CoreCLR 5.0.20.45114, CoreFX 5.0.20.45114), x64 RyuJIT
DefaultJob : .NET Core 5.0.0 (CoreCLR 5.0.20.45114, CoreFX 5.0.20.45114), x64 RyuJIT

-----
| Method | Mean | Error | StdDev | Median |
-----|-----|-----|-----|-----|
| MeasureVarUsage | 0.0601 ns | 0.0361 ns | 0.0964 ns | 0.0000 ns |
| MeasureVarDynamicUsage | 0.0206 ns | 0.0212 ns | 0.0508 ns | 0.0000 ns |
| MeasureTypeDynamicUsage | 14.5733 ns | 0.2884 ns | 0.7288 ns | 14.4971 ns |
| MeasureTypeTypeUsage | 0.0918 ns | 0.0608 ns | 0.1763 ns | 0.0000 ns |
-----

// * Warnings *
ZeroMeasurement
BenchmarkTests.MeasureVarUsage: Default -> The method duration is indistinguishable from the empty method duration
BenchmarkTests.MeasureVarDynamicUsage: Default -> The method duration is indistinguishable from the empty method duration
BenchmarkTests.MeasureTypeTypeUsage: Default -> The method duration is indistinguishable from the empty method duration

// * Hints *
Outliers
BenchmarkTests.MeasureVarUsage: Default -> 2 outliers were removed (3.28 ns..3.60 ns)
BenchmarkTests.MeasureVarDynamicUsage: Default -> 7 outliers were removed (3.20 ns..3.52 ns)
BenchmarkTests.MeasureTypeDynamicUsage: Default -> 6 outliers were removed (17.07 ns..19.36 ns)
BenchmarkTests.MeasureTypeTypeUsage: Default -> 3 outliers were removed (3.78 ns..4.06 ns)

// * Legends *
Mean : Arithmetic mean of all measurements
Error : Half of 99.9% confidence interval
StdDev : Standard deviation of all measurements
Median : Value separating the higher half of all measurements (50th percentile)
1 ns : 1 Nanosecond (0.000000001 sec)

// ***** BenchmarkRunner: End *****
// ** Remained 0 benchmark(s) to run **
Run time: 00:06:18 (378.83 sec), executed benchmarks: 4
Global total time: 00:06:26 (386.47 sec), executed benchmarks: 4
// * Artifacts cleanup *
```

C:\Users\maranf\Downloads\High-Pe...

File View Help

C:\Users\maranf\Downloads\High-Performance\High-Performance-Prog

MANIFEST

CH03_DynamicPerformance

CH03_DynamicPerformance.BenchmarkTests

CH03_DynamicPerformance.Program

.class private auto ansi beforefieldinit

.dynamicType : private object

.ctor : void()

Main : void(string[])

Items added to the stack grow upwards

Application starting point memory address

Items added to the heap grow downwards

The Heap

assembly CH03_DynamicPerformance

{

.ver 1:0:0:0

The Stack	
Main(string[] _) The active method is placed in a stack frame on the stack	
<div> <div>data</div> <div>Id</div> <div>Quantity</div> <div>PurchaseDate</div> <div>greeting</div> <div>Id</div> <div>20</div> </div>	<div> static void Main(string[] _) { int x = 47; string greeting = "Hello, world!"; var data = new Data(); } Internal struct Data { public int Id; public double Quantity; public DateTime PurchaseDate } </div>
Memory Address allocated at start up time. Everything below this address belongs to the heap. And everything above this address is the stack	
0x01576649	Hello, world!
The Heap	

The Stack			
Array:			
Object 1 Memory Address	Object 2 Memory Address	Object 3 Memory Address	Object 4 Memory Address
Object 1			
Object 2			
Object 3			
Object 4			

```

Microsoft Visual Studio Debug Console
----- Histogram -----
(43.543 ns ; 44.496 ns) | @
(44.496 ns ; 45.885 ns) | @
(45.885 ns ; 47.739 ns) | @
-----

// * Summary *
BenchmarkDotNet=v0.13.0, OS=Windows 10.0.19043.1826 (21H1/May2021Update)
Intel Core i5-8365U CPU 1.60GHz (Coffee Lake), 1 CPU, 8 logical and 4 physical cores
.NET SDK=6.0.302
[Host] : .NET 6.0.7 (6.0.722.32202), X64 RyuJIT [AttachedDebugger]
DefaultJob : .NET 6.0.7 (6.0.722.32202), X64 RyuJIT

| Method | Mean | Error | StdDev |
|-----|-----|-----|-----|
| ProcessClassNoReferences | 41.41 ns | 0.840 ns | 0.967 ns |
| ProcessStructNoReferences | 42.42 ns | 0.860 ns | 1.088 ns |
| ProcessClassWithReferences | 51.11 ns | 0.866 ns | 0.768 ns |
| ProcessStructWithReferences | 45.64 ns | 0.909 ns | 0.933 ns |

// * Warnings *
Environment
Summary -> Benchmark was executed with attached debugger

// * Hints *
Outliers
BenchmarkTests.ProcessStructNoReferences: Default -> 1 outlier was removed (48.48 ns)
BenchmarkTests.ProcessClassWithReferences: Default -> 1 outlier was removed (55.65 ns)

// * Legends *
Mean : Arithmetic mean of all measurements
Error : Half of 99.0% confidence interval
StdDev : Standard deviation of all measurements
1 ns : 1 Nanosecond (0.000000001 sec)

// ***** BenchmarkRunner: End *****
// ** Remained 0 benchmark(s) to run **
Run time: 00:01:51 (111.03 sec), executed benchmarks: 4
Global total time: 00:02:11 (131.95 sec), executed benchmarks: 4
// * Artifacts cleanup *

```

```

Microsoft Visual Studio Debug Console
Chapter 3: Pass by value and reference
=====
int x = 0;
    AddByValue(x): 0
AddByReference(x): 1

```

Select Microsoft Visual Studio Debug Console

```
Chapter 3: Pass by value and reference
=====
int x = 0;
    AddByValue(x): 0
AddByReference(x): 1
13
The value of x is: 123.
```

Microsoft Visual Studio Debug Console

```
// * Summary *
BenchmarkDotNet=v0.13.0, OS=Windows 10.0.19043.1826 (21H1/May2021Update)
Intel Core i5-8365U CPU 1.60GHz (Coffee Lake), 1 CPU, 8 logical and 4 physical cores
.NET SDK=6.0.302
[Host] : .NET 6.0.7 (6.0.722.32202), X64 RyuJIT [AttachedDebugger]
DefaultJob : .NET 6.0.7 (6.0.722.32202), X64 RyuJIT

|-----| Method | Mean | Error | StdDev |
|-----|-----|-----|-----|-----|
| NonBoxingUnboxingTest | 0.0851 ns | 0.0319 ns | 0.0299 ns |
| BoxingUnboxingTest | 7.2714 ns | 0.2403 ns | 0.6738 ns |

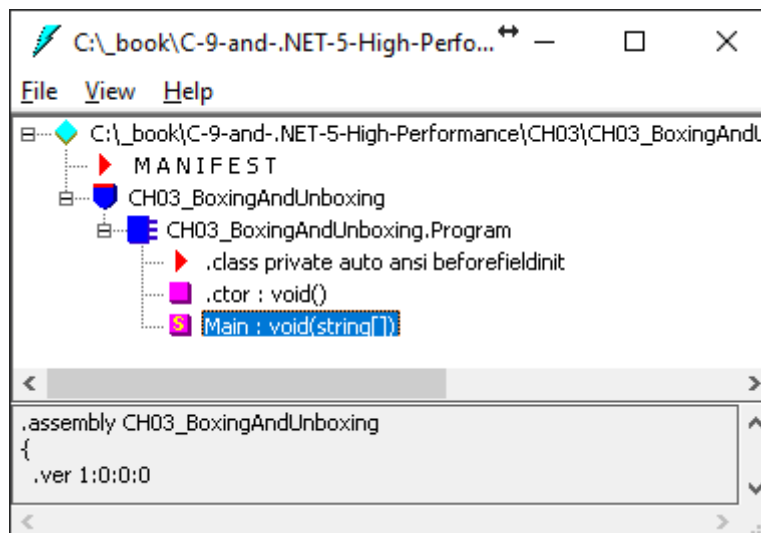
// * Warnings *
Environment
Summary -> Benchmark was executed with attached debugger

// * Hints *
Outliers
BoxingAndUnboxingBenchmarkTests.NonBoxingUnboxingTest: Default -> 2 outliers were removed (1.81 ns, 1.84 ns)
BoxingAndUnboxingBenchmarkTests.BoxingUnboxingTest: Default -> 9 outliers were removed (11.03 ns..14.06 ns)

// * Legends *
Mean : Arithmetic mean of all measurements
Error : Half of 99.9% confidence interval
StdDev : Standard deviation of all measurements
1 ns : 1 Nanosecond (0.000000001 sec)

// ***** BenchmarkRunner: End *****
// ** Remained 0 benchmark(s) to run **
Run time: 00:01:58 (118.88 sec), executed benchmarks: 2
Global total time: 00:02:18 (138.15 sec), executed benchmarks: 2
// * Artifacts cleanup *
```

If you open the Developer Command Prompt for Visual Studio (VS) 2019 and



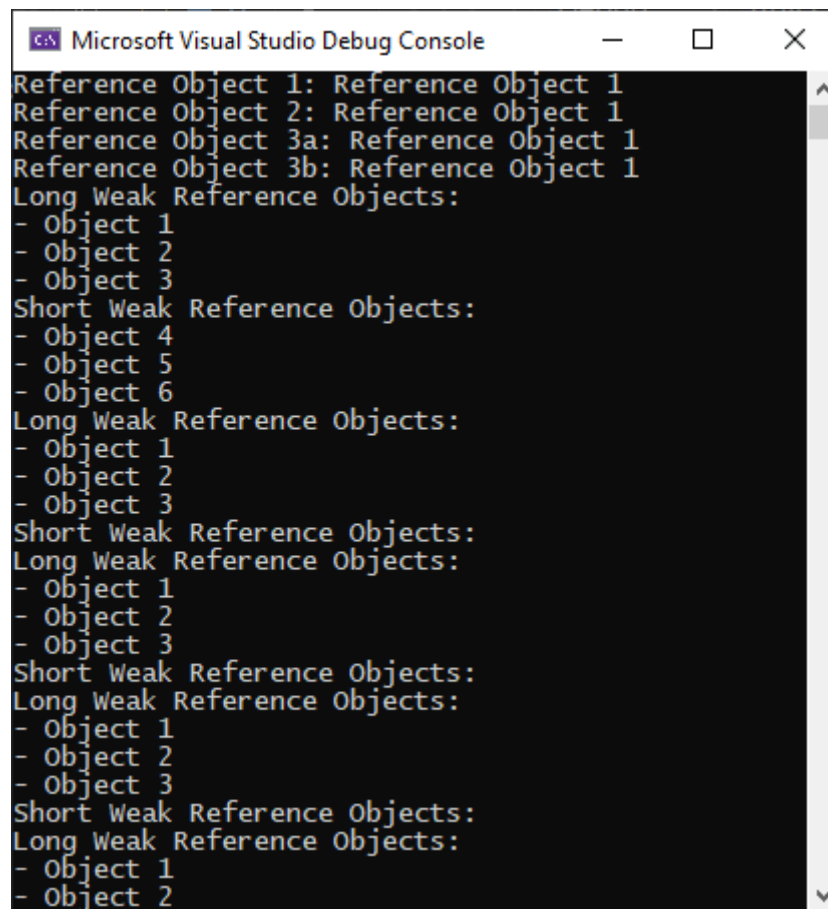
CH03_BoxingAndUnboxing.Program::Main : void(string[])

Find Find Next

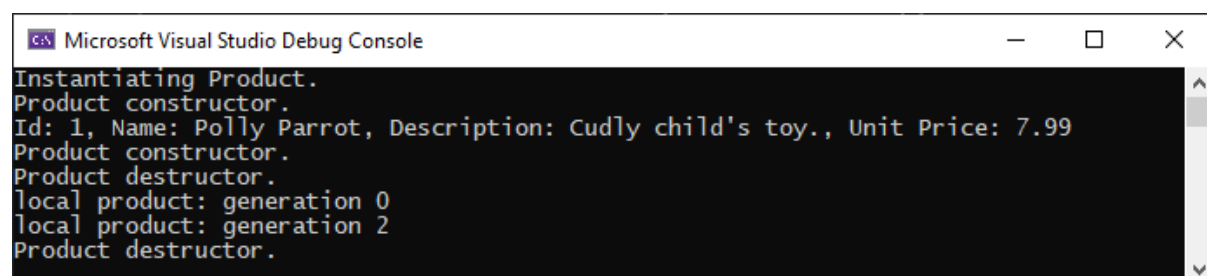
```
.method private hidebysig static void Main(string[] args) cil managed
{
    .entrypoint
    // Code size          265 (0x109)
    .maxstack 5
    .locals init (int32 V_0,
        int32 V_1,
        object V_2,
        object V_3,
        int32 V_4,
        class [System.Runtime]System.Diagnostics.Stopwatch V_5,
        int32 V_6,
        bool V_7)
    IL_0000: nop
    IL_0001: ldstr      "Chapter 3: Boxing and unboxing example."
    IL_0006: call       void [System.Console]System.Console::WriteLine(string)
    IL_000b: nop
    IL_000c: ldc.i4.1
    IL_000d: stloc.0
    IL_000e: ldc.i4.2
    IL_000f: stloc.1
    IL_0010: ldc.i4.4
    IL_0011: box        [System.Runtime]System.Int32
    IL_0016: stloc.2
    IL_0017: ldc.i4.4
    IL_0018: box        [System.Runtime]System.Int32
    IL_001d: stloc.3
    IL_001e: newobj     instance void [System.Runtime]System.Diagnostics.Stopwatc
    IL_0023: stloc.s   V_5
}
```

Name	Size in bytes	Nullable	Default Value	Min Value	Max Value	Notes
bool	1	Yes	false			true/false
byte	1	Yes	0	0	255	
char	2	Yes	0 (\u0000)	0 (0\uFFFF)	65535 (\uFFFF)	
DateTime	8	Yes	01/01/0001 00:00:00	01/01/0001 00:00:00	31/12/9999 23:59:59 (3155378975599999999 ticks)	
decimal	16	Yes	0	79228162514264337593543950335	79228162514264337593543950335	
double	8	Yes	0	-1.7976931348623157E+308	1.7976931348623157E+308	
enum	4	Yes	0			Grows
float	4	Yes	0	-3.4028235E+38	3.4028235E+38	
int	4	Yes	0	-2147483648	2147483647	
long	8	Yes	0	-9223372036854775808	9223372036854775807	
sbyte	1	Yes	0	-128	127	
short	2	Yes	0	-32768	32767	
struct	Variable					
value tuple	1					Grows
uint	4	Yes	0	0	4294967295	
ulong	8	Yes	0	0	18446744073709551615	

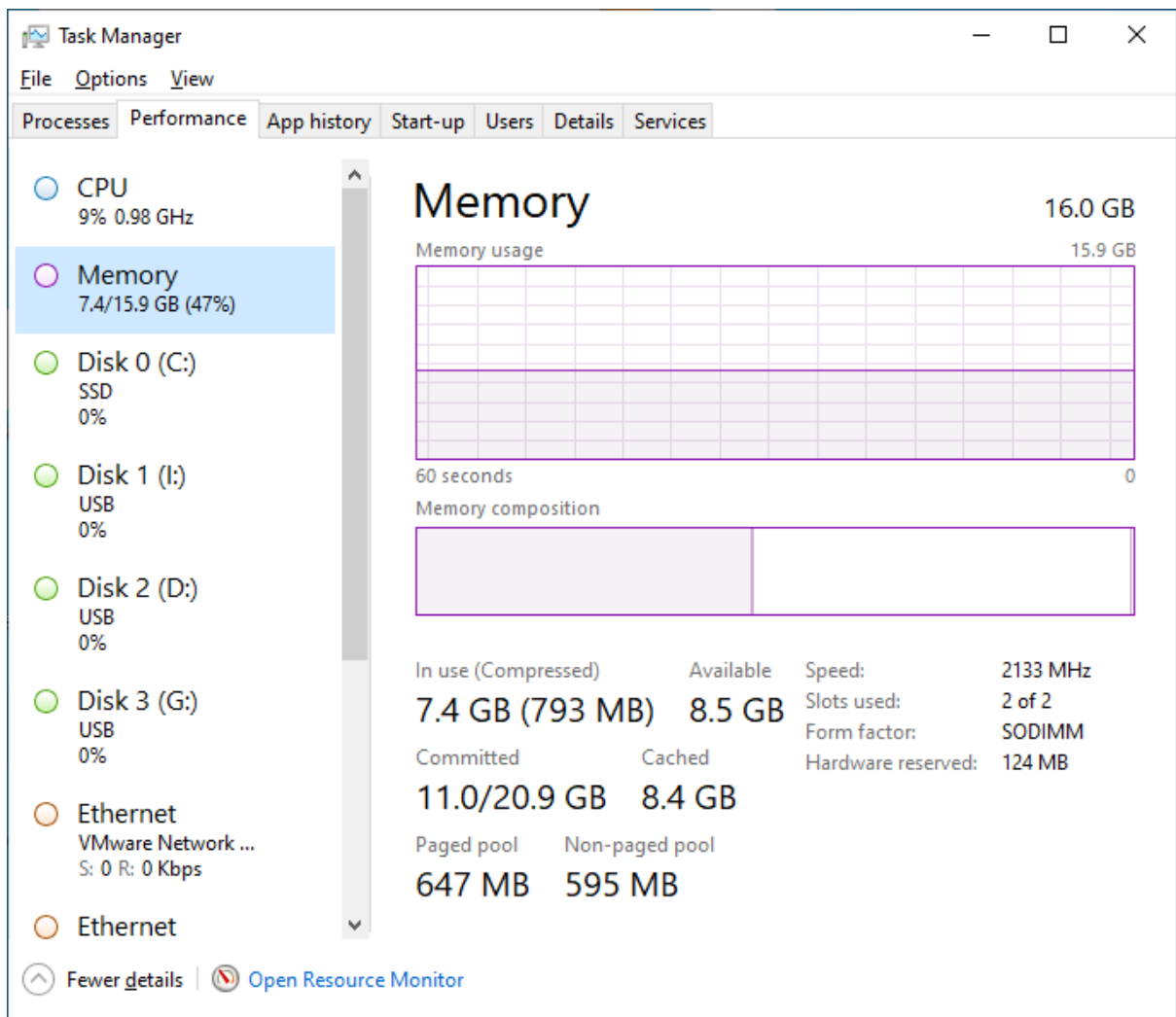
Chapter 4: Memory Management

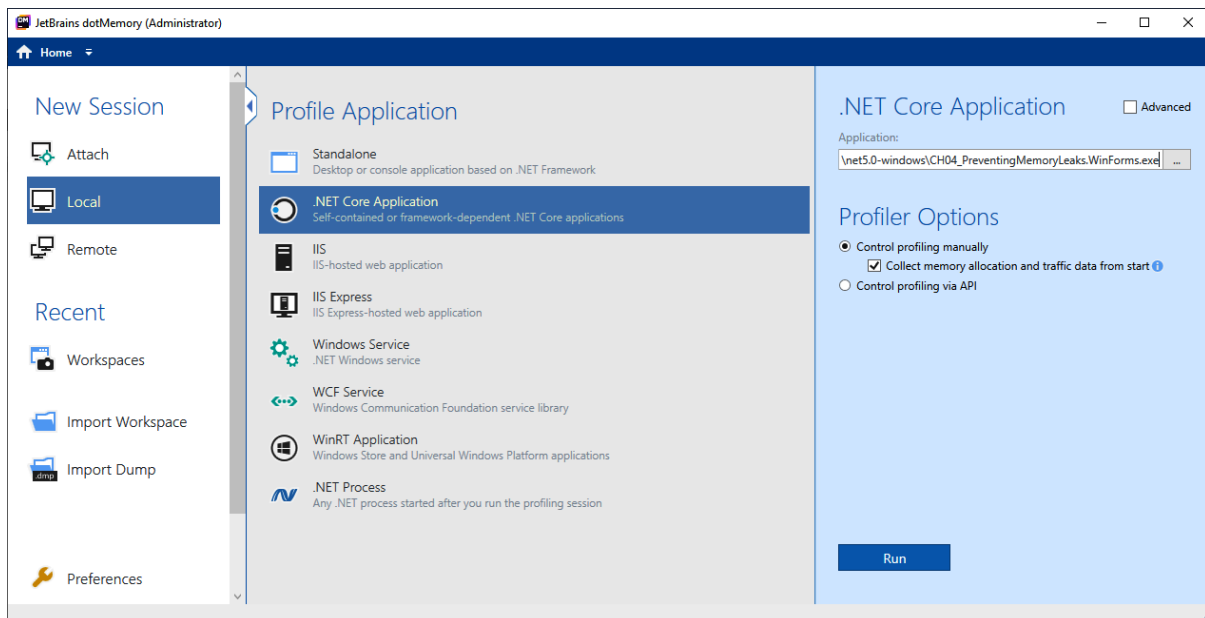
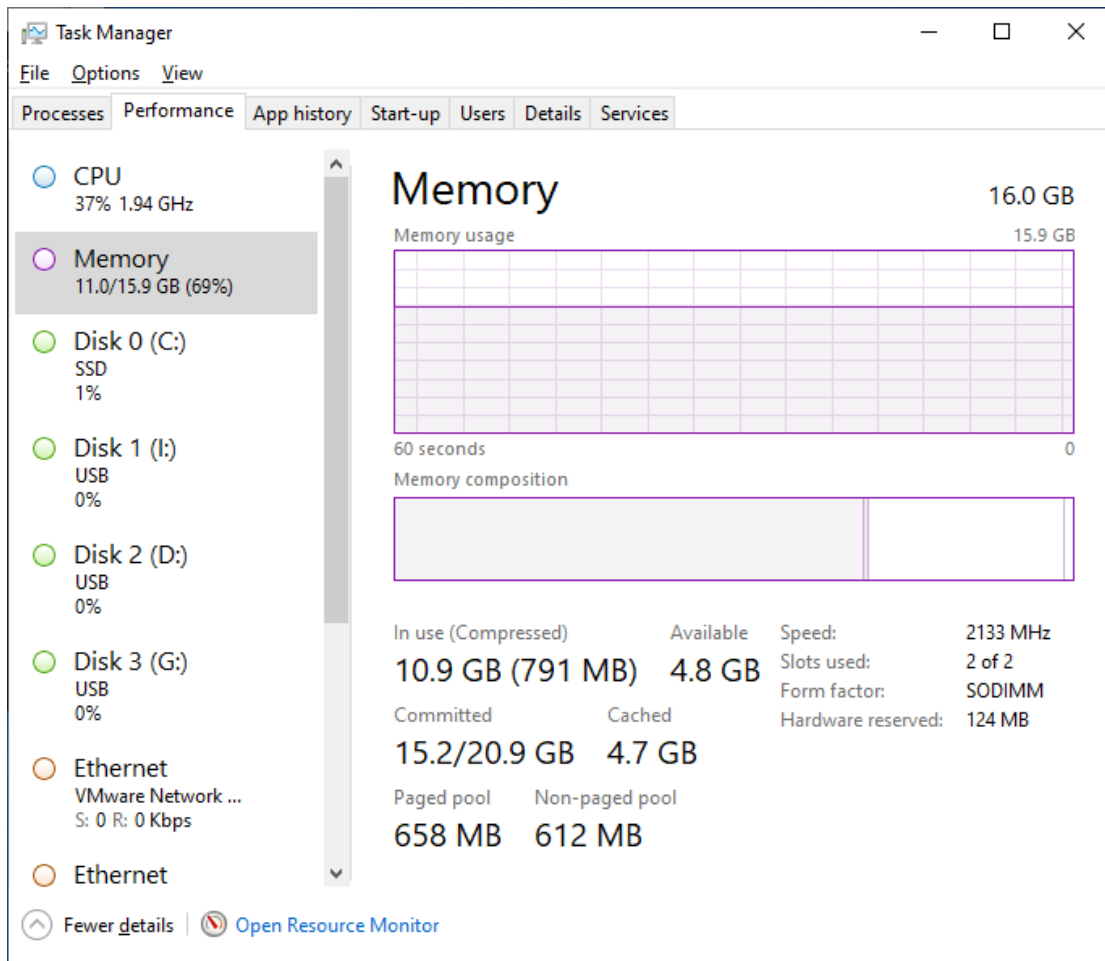


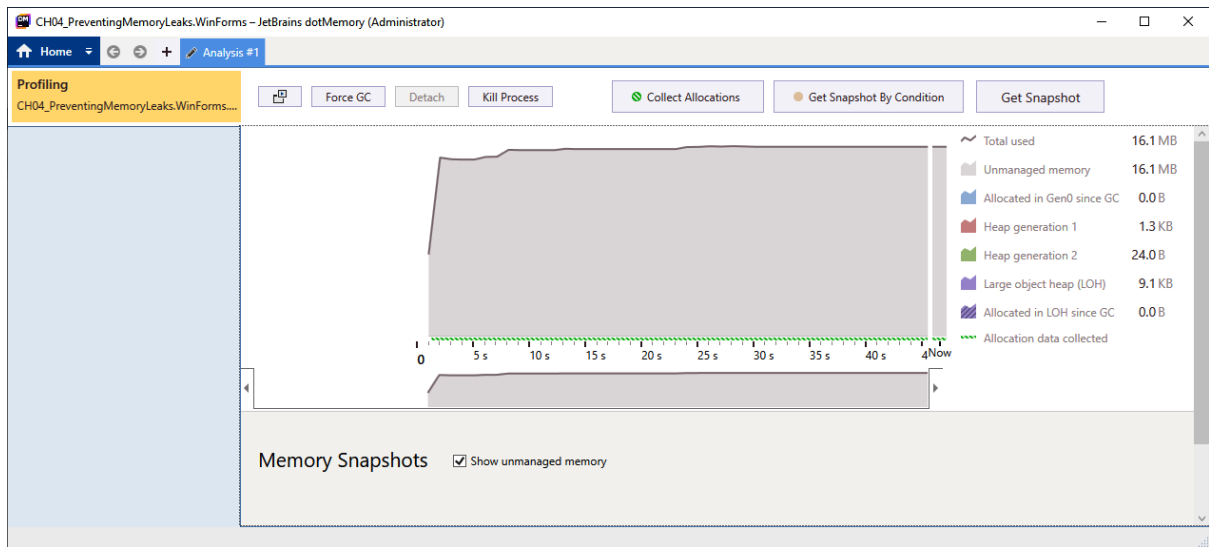
```
Microsoft Visual Studio Debug Console
Reference Object 1: Reference Object 1
Reference Object 2: Reference Object 1
Reference Object 3a: Reference Object 1
Reference Object 3b: Reference Object 1
Long Weak Reference Objects:
- Object 1
- Object 2
- Object 3
Short Weak Reference Objects:
- Object 4
- Object 5
- Object 6
Long Weak Reference Objects:
- Object 1
- Object 2
- Object 3
Short Weak Reference Objects:
Long Weak Reference Objects:
- Object 1
- Object 2
- Object 3
Short Weak Reference Objects:
Long Weak Reference Objects:
- Object 1
- Object 2
- Object 3
Short Weak Reference Objects:
Long Weak Reference Objects:
- Object 1
- Object 2
```



```
Microsoft Visual Studio Debug Console
Instantiating Product.
Product constructor.
Id: 1, Name: Polly Parrot, Description: Cudly child's toy., Unit Price: 7.99
Product constructor.
Product destructor.
local product: generation 0
local product: generation 2
Product destructor.
```





Form1

Information

Raise Events

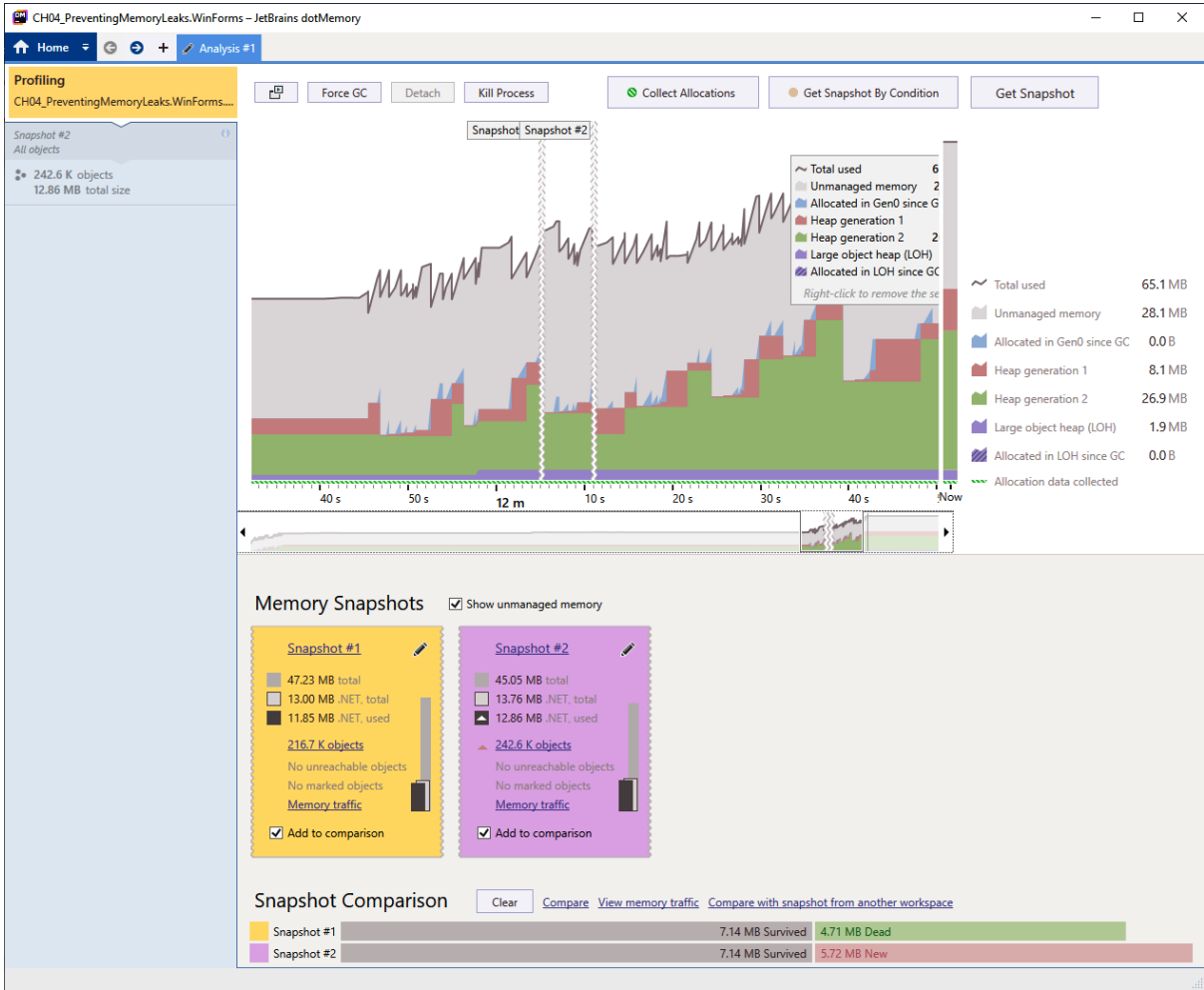
Progress:

50000/0 - 50000/50000

Raised Events (No Memory Leak): 50000, Alive Events: 0
Raised Events (Memory Leak): 50000, Alive Events: 50000

Raise Events

Event Raise: 09/12/2020 14:12:46



CH04_PreventingMemoryLeaks.WinForms - JetBrains dotMemory

Home Analysis #1

Profiling
CH04_PreventingMemoryLeaks.WinForms...

Snapshot #3
All objects
182.4 K objects
9.45 MB total size

Inspections Group by: Types Dominators Similar Retention Instances Call Tree Call Tree (Icicle Chart) Back Traces
Generations Shortest Paths

Finalizable objects 82,489 objects of 2 type(s) found

	Queued	Finalized
EventOne	0	19576
Gen2GcCallback	2	0

Objects that were queued for finalization or already finalized since the previous snapshot. It might be more effective if these objects were disposed via the IDisposable interface.

Event handlers leak 5.04 MB retained by 60,000 objects

	Retained Size	Count
EventTwo	5.04 MB	60,000

Objects subscribed to an event of another object and never unsubscribed from this event.

WPF binding leak No issues found.

Dependency property leak No issues found.

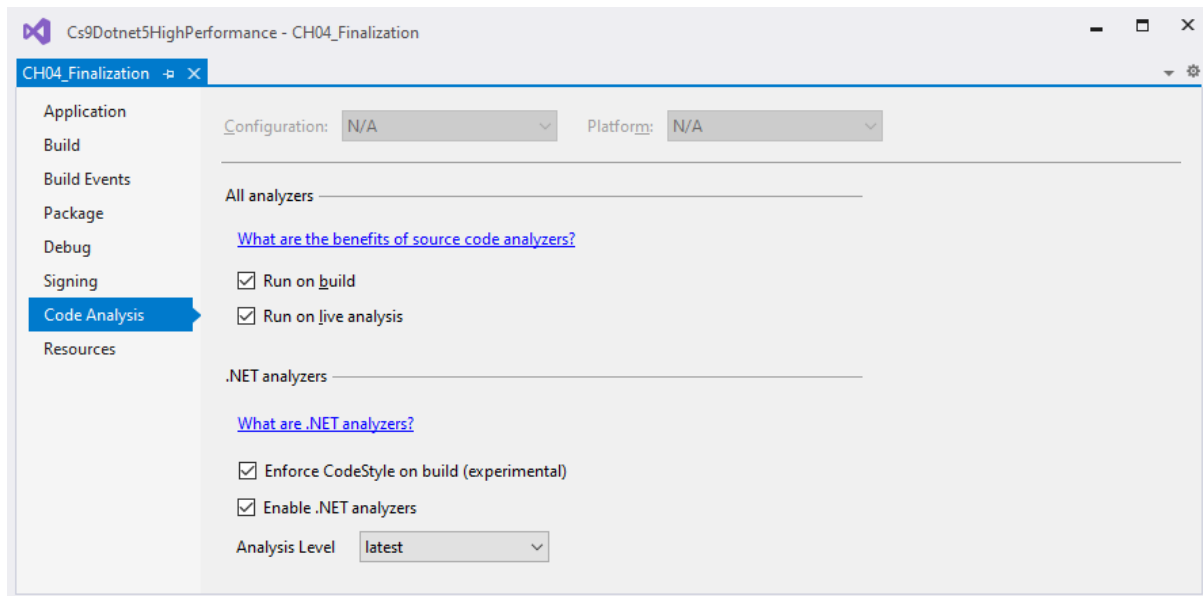
WPF collection binding leak No issues found.

x:Name WPF leak No issues found.

Immediate Window

```
Anonymous logout event handler using a local method.  
Anonymous login event handler using a local method.  
Anonymous logout event handler using a local method.  
Anonymous login event handler using a local method.
```

Chapter 5: Application Profiling and Tracing



Code Metrics Results						
Filter:	None	Min:		Max:		
Hierarchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Source code	Lines of Executable code

Code Metrics Results						
Filter:	None	Min:		Max:		
Hierarchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Source code	Lines of Executable code
CH04\CH04_WeakReferences (Debug)	82	20	1	13	193	67
CH04_WeakReferences	82	20	1	13	193	67
Instrumentation	69	3	1	5	39	12
Measure(string, int, Action) : void	61	2		5	23	11
WarmUp(Action) : void	100	1		1	9	1
LongWeakReferenceObjectManager	92	3	1	3	16	4
Objects : List<ReferenceObject>	93	0		2	1	1
Add(ReferenceObject) : void	96	1		2	4	1
ListObjects() : void	84	2		3	6	2
Program	68	6	1	8	89	45
StrongReferences : LongWeakReferenc	93	0		1	1	1
WeakReferences : ShortWeakReferenc	93	0		1	1	1
Main(string[]) : void	77	1		2	9	6
TestLongWeakReference() : void	58	1		4	33	15
TestStrongReferences() : void	67	1		2	10	6
TestWeakReferences() : void	63	1		2	14	9
ProcessReferences() : void	70	2		4	13	7
ReferenceObject	100	4	1	0	5	0
Id : int	100	2		0	1	0
Name : string	100	2		0	1	0
ShortWeakReferenceObjectManager	83	4	1	4	20	6
Objects : List<WeakReference<Referer	93	0		3	1	1
Add(ReferenceObject) : void	93	1		3	4	1
ListObjects() : void	75	3		4	10	4

Error List						
Entire Solution		0 Errors	4 Warnings	62 Messages	Build + IntelliSense	Search Error List
	Code	Description	Project	File	Line	Suppression State
▶	CA1822	Member 'GenericListIntsBenchmarkTest' does not access instance data and can be marked as static	CH06_Collections	CollectionPerformanceBe...	37	Active
▶	CA1822	Member 'NativeArrayIntsBenchmarkTest' does not access instance data and can be marked as static	CH06_Collections	CollectionPerformanceBe...	47	Active
▶	CA1822	Member 'ArrayListObjectsBenchmarkTest' does not access instance data and can be marked as static	CH06_Collections	CollectionPerformanceBe...	57	Active

Cs9Dotnet5HighPerformance - CH04_WeakReferences.dmp

CH04_WeakReferences.dmp

Minidump File Summary

13/04/2021 21:07:55

^ Dump Summary

Dump File	CH04_WeakReferences.dmp : C:\Temp\CH04_WeakReferences.dmp
Last Write Time	13/04/2021 21:07:55
Process Name	CH04_WeakReferences.exe : G:_book\C-9-and-.NET-5-High-Perfomance
Process Architecture	x64
Exception Code	0x80000004
Exception Information	A trace trap or other single-instruction mechanism signaled that one i
Heap Information	Present
Error Information	

^ Actions

- ▶ Debug with Managed Only
- ▶ Debug with Mixed
- ▶ Debug with Native Only
- 🔍 Run Diagnostic Analysis
- 📁 Set symbol paths
- 📋 Copy all to clipboard

^ System Information

OS Version	10.0.19042
CLR Version(s)	6.0.21.15406



















^ Modules

Search

Module Name	Module Version	Module Path
CH04_WeakReferences.exe	1.0.0.0	G:_book\C-9-and-.NET-5-High-Performance\CH0...
ntdll.dll	10.0.19041.844	C:\Windows\System32\ntdll.dll
kernel32.dll	10.0.19041.804	C:\Windows\System32\kernel32.dll
KERNELBASE.dll	10.0.19041.804	C:\Windows\System32\KERNELBASE.dll
apphelp.dll	10.0.19041.572	C:\Windows\System32\apphelp.dll
user32.dll	10.0.19041.746	C:\Windows\System32\user32.dll
win32u.dll	10.0.19041.867	C:\Windows\System32\win32u.dll
gdi32.dll	10.0.19041.746	C:\Windows\System32\gdi32.dll
gdi32full.dll	10.0.19041.746	C:\Windows\System32\gdi32full.dll
msvc_p_win.dll	10.0.19041.789	C:\Windows\System32\msvc_p_win.dll
ucrtbase.dll	10.0.19041.789	C:\Windows\System32\ucrtbase.dll
shell32.dll	10.0.19041.844	C:\Windows\System32\shell32.dll
advapi32.dll	10.0.19041.610	C:\Windows\System32\advapi32.dll

	Immediate	Ctrl+G
	Locals	
	Breakpoints	
	Exception Settings	Ctrl+Alt+E
	Output	
	XAML Binding Failures	
	Autos	
	Call Stack	Ctrl+L
	Threads	
	Show Diagnostic Tools	Ctrl+Alt+F2
	GPU Threads	
	Tasks	Ctrl+Shift+D, K
	Parallel Stacks	Ctrl+Shift+D, S
	Parallel Watch	▶
	Watch	▶
	DOM Explorer	▶
	Live Visual Tree	
	Live Property Explorer	
	Modules	
	Processes	Ctrl+Alt+Z
	Diagnostic Analysis	Ctrl+Shift+Alt+D

Modules									
Search									
	Process	Name	AppDomain	Address	Path	Optimized	User Code	Order ▲	Version
	[3280] CH04_WeakRe...	System.Private.CoreLib.dll	[1] clrhost	00007FFC7C36000...	C:\Progra...	Yes	No	1	6.00.21.154...
	[3280] CH04_WeakRe...	CH04_WeakReferences.dll	[1] clrhost	0000021EA710000...	G:_book\...	No	Yes	2	1.00.0.0
	[3280] CH04_WeakRe...	System.Runtime.dll	[1] clrhost	0000021EA711000...	C:\Progra...	Yes	No	3	6.00.21.154...
	[3280] CH04_WeakRe...	System.Console.dll	[1] clrhost	00007FFCE619000...	C:\Progra...	Yes	No	4	6.00.21.154...


	Immediate	Ctrl+G
	Locals	
	Breakpoints	
	Exception Settings	Ctrl+Alt+E
	Output	
	XAML Binding Failures	
	Autos	
	Call Stack	Ctrl+L
	Threads	
	Show Diagnostic Tools	Ctrl+Alt+F2
	GPU Threads	
	Tasks	Ctrl+Shift+D, K
	Parallel Stacks	Ctrl+Shift+D, S
	Parallel Watch	▶
	Watch	▶
	DOM Explorer	▶
	Live Visual Tree	
	Live Property Explorer	
	Modules	
	Processes	Ctrl+Alt+Z
	Diagnostic Analysis	Ctrl+Shift+Alt+D

Cs9Dotnet5HighPerformance - Report20210413-2135.diagsession

Report20210413-2135.diagsession

>

Analysis Target



Startup Project

CH04_Finalization


Change Target

Available Tools

Show all tools

☐ .NET Async

Tool to investigate async/await usage in .NET applications

☒ .NET Object Allocation Tracking 

See where .NET Objects are allocated and when they are reclaimed by the GC

☐ Database

Examine when queries were executed and measure how long they take

☐ Instrumentation

Instrument your application to investigate exact call counts and call times

☐ .NET Counters

Tool to visualize performance counters in .NET applications

☐ CPU Usage

See where the CPU is spending time executing your code. Useful when the CPU is the performance bottleneck

☐ Events Viewer

See the events (ETW or NetTrace) that occurred during the session, such as log messages, exceptions and HTTP requests

☐ Memory Usage

Investigate application memory to find issues such as memory leaks

Start

Cs9Dotnet5HighPerformance - Report20210413-2135.diagsession*

Report20210413-2135.diagsession*

Output

Zoom In

Reset Zoom

Clear Selection

Diagnostics session: 11.314 seconds

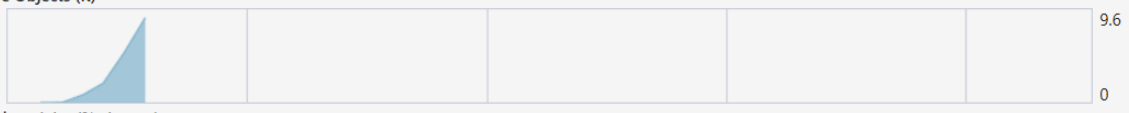
2.5s

5s


7.5s

10s

Live Objects (K)



Object delta (% change)



Allocations

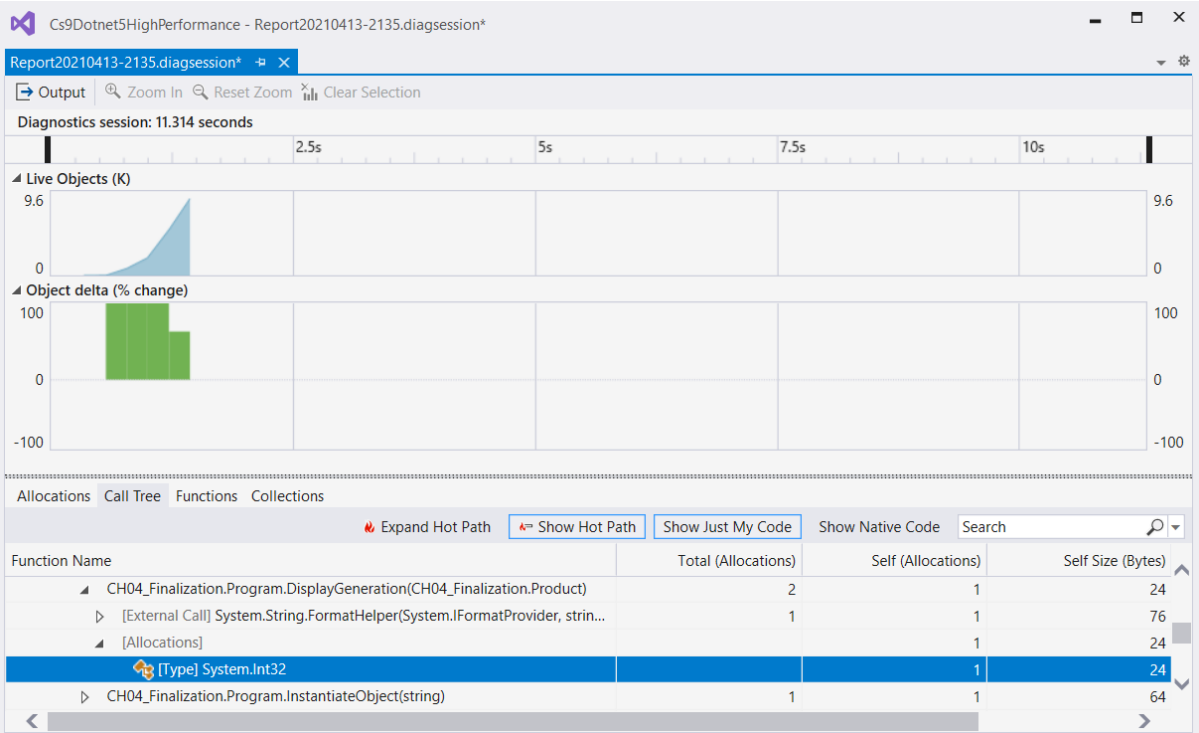
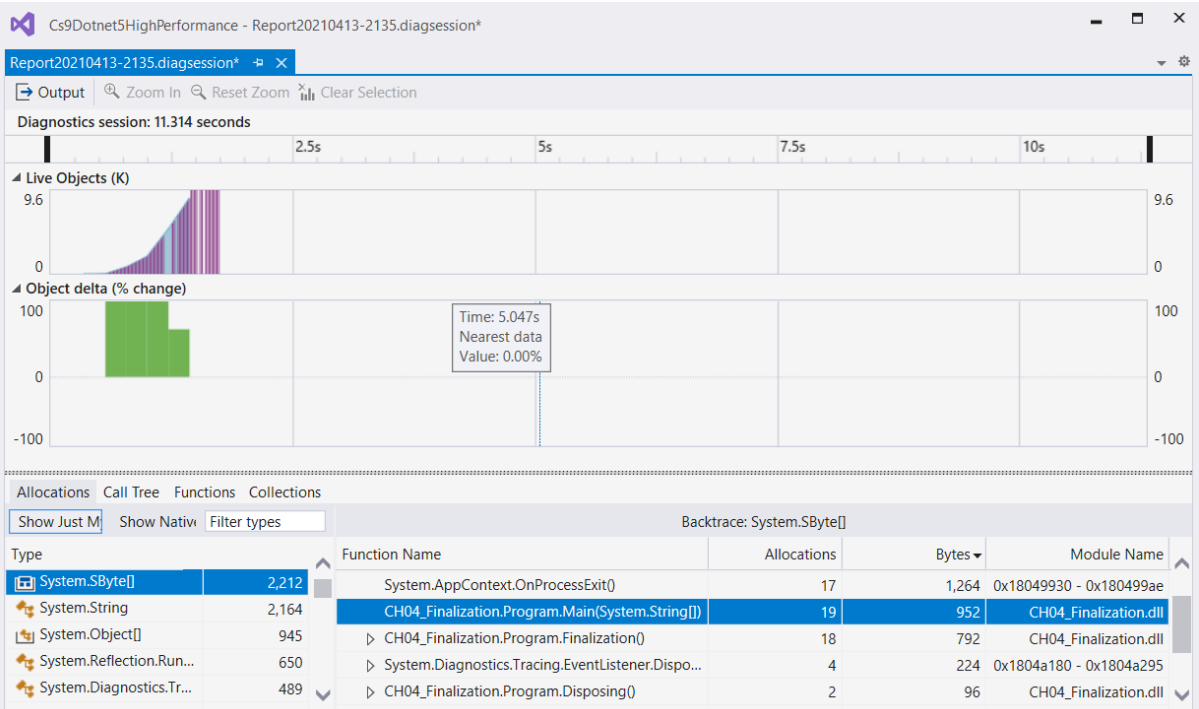
Call Tree

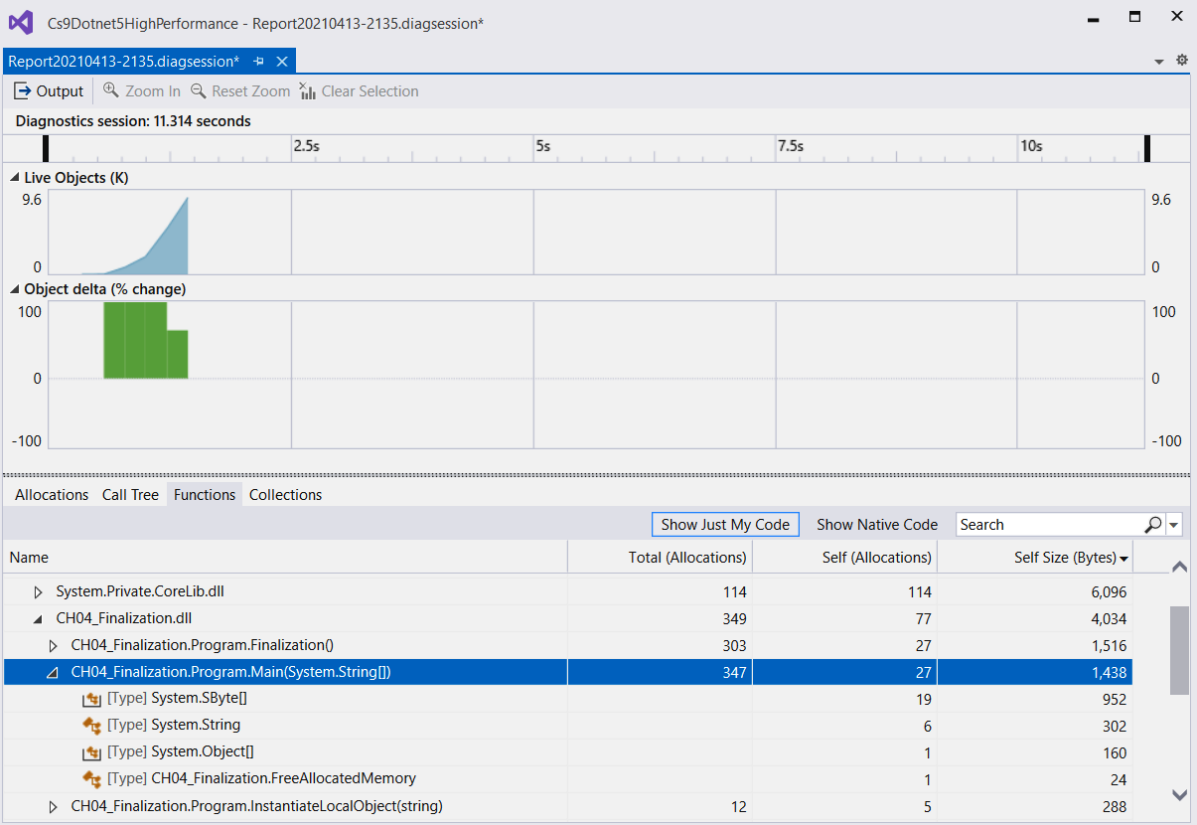
Functions

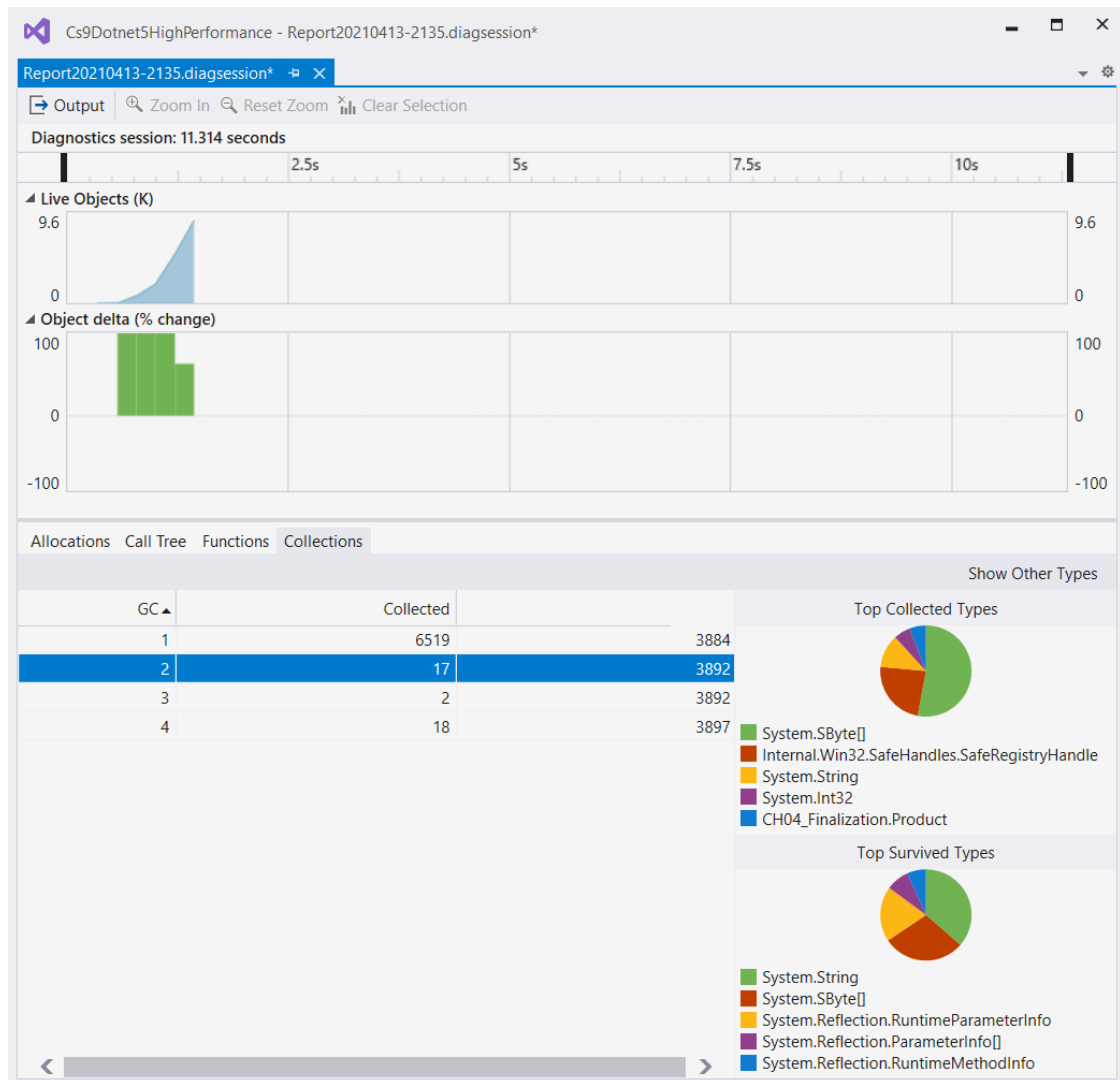
Collections

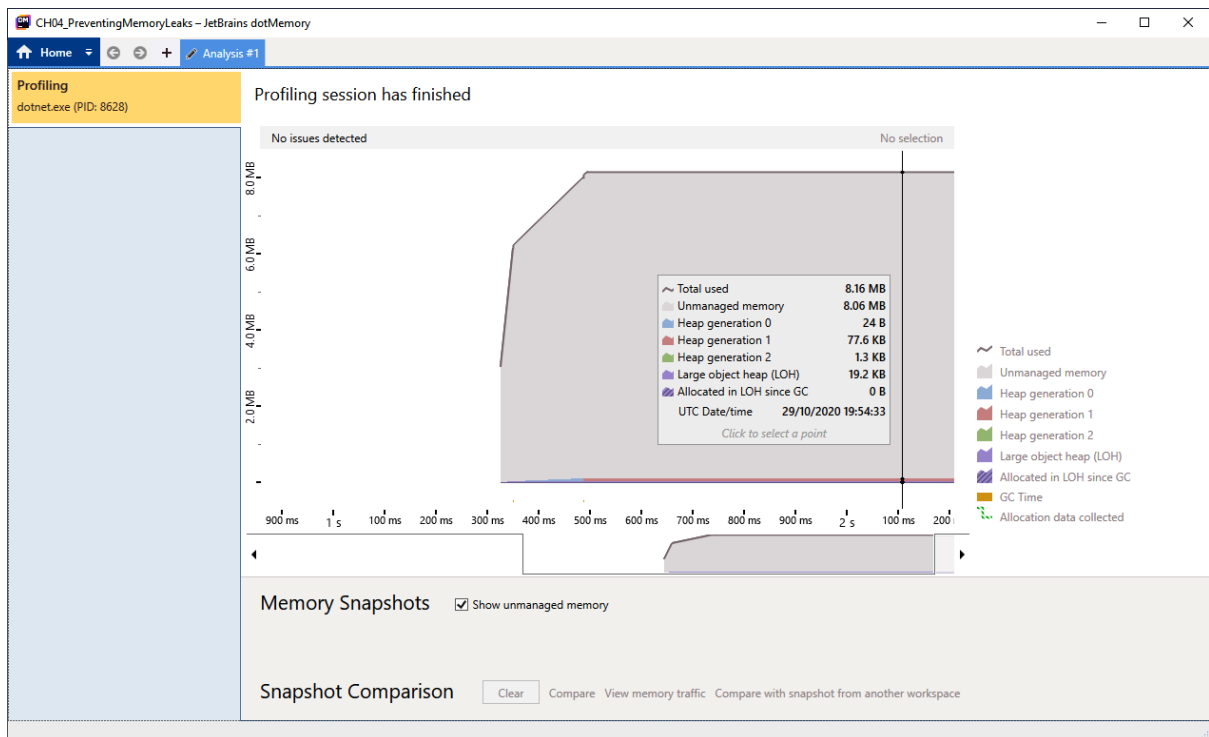
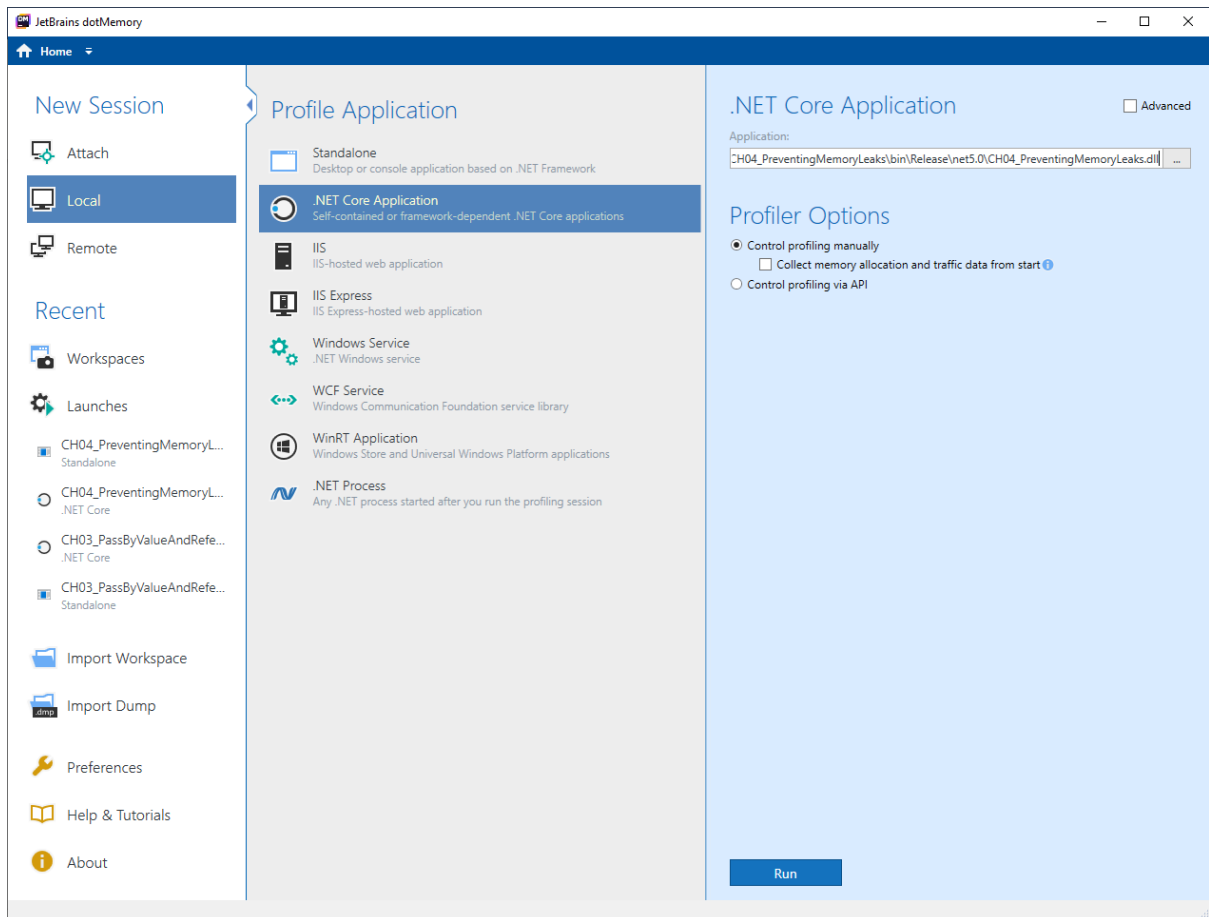
Show Other Types

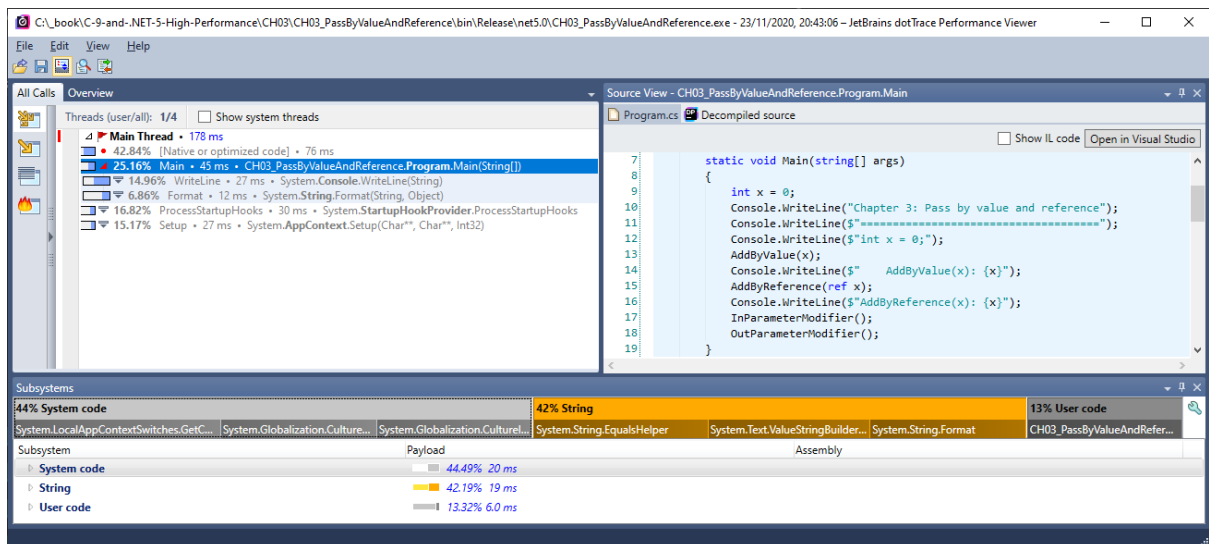
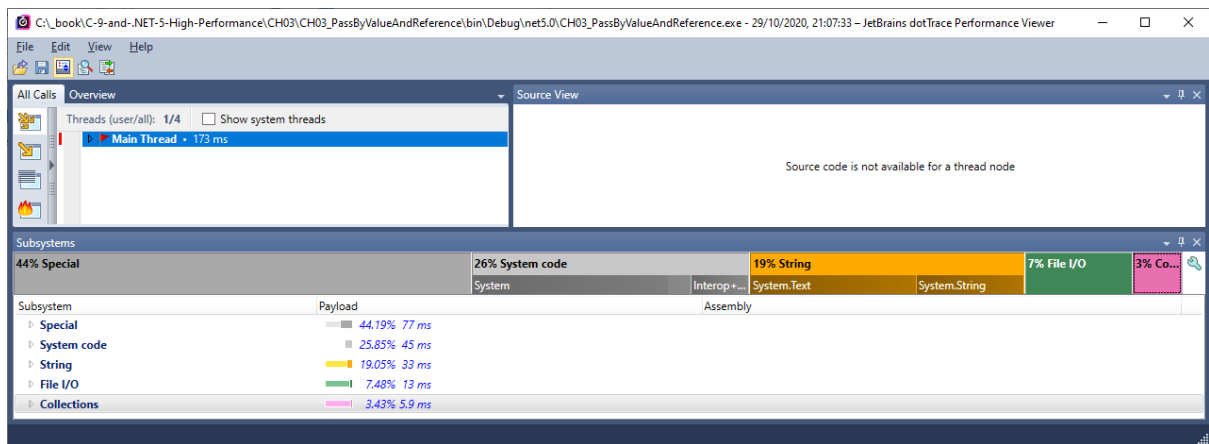
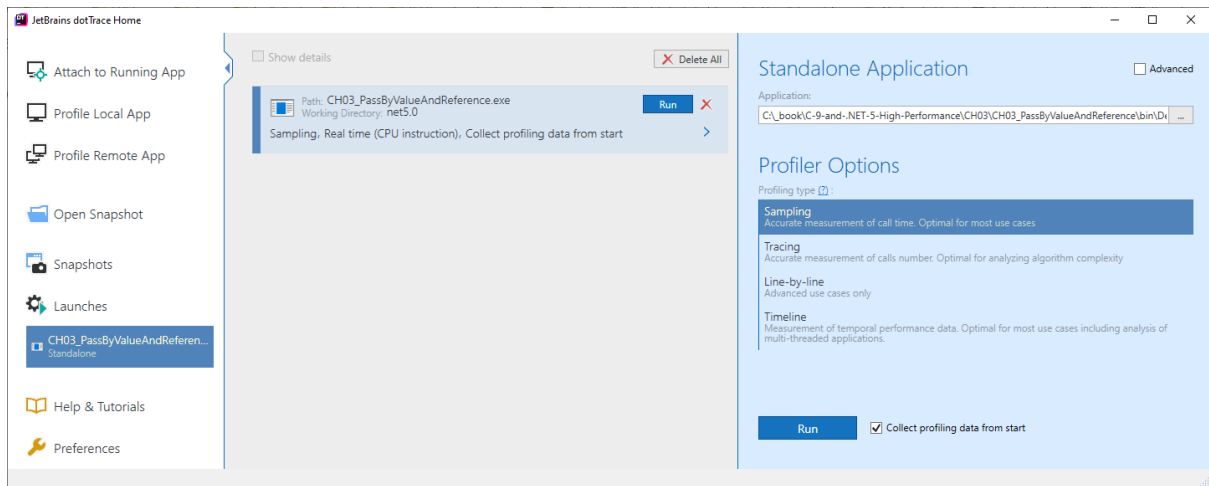
GC	Collected	Survived	Top Collected Types	Top Survived Types
1	6519	3884		
2	17	3892		
3	2	3892		
4	18	3897		











```
Administrator: Developer Command Prompt for VS 2019

C:\Windows\System32>dotnet tool install --global dotnet-counters --version 3.1.141901
You can invoke the tool using the following command: dotnet-counters
Tool 'dotnet-counters' (version '3.1.141901') was successfully installed.

C:\Windows\System32>
```

```
Administrator: Developer Command Prompt for VS 2019

*****
** Visual Studio 2019 Developer Command Prompt v16.7.7
** Copyright (c) 2020 Microsoft Corporation
*****

C:\Windows\System32>cd C:\Temp

C:\Temp>dotnet-counters ps
17652 CH04_WeakReferences C:\_book\C-9-and-.NET-5-High-Performance\CH04\CH04_WeakReferences\bin\Debug\net5.0\CH04_WeakReferences.exe

C:\Temp>dotnet-counters collect --process-id 17652
counter_list is unspecified. Monitoring all counters by default.
[Warning] counter.csv already exists. This file will be overwritten.
Starting a counter session. Press Q to quit.
File saved to counter.csv

C:\Temp>
```

	A	B	C	D	E	F
1	Timestamp	Provider	Counter Name	Counter Type	Mean/Increment	
2	22/11/2020 20:38	System.Runtime	CPU Usage (%)	Metric	0	
3	22/11/2020 20:38	System.Runtime	Working Set (MB)	Metric	20	
4	22/11/2020 20:38	System.Runtime	GC Heap Size (MB)	Metric	0	
5	22/11/2020 20:38	System.Runtime	Gen 0 GC Count (Count / 1 sec)	Rate	1	
6	22/11/2020 20:38	System.Runtime	Gen 1 GC Count (Count / 1 sec)	Rate	1	
7	22/11/2020 20:38	System.Runtime	Gen 2 GC Count (Count / 1 sec)	Rate	1	
8	22/11/2020 20:38	System.Runtime	ThreadPool Thread Count	Metric	2	
9	22/11/2020 20:38	System.Runtime	Monitor Lock Contention Count (Count / 1 sec)	Rate	1	
10	22/11/2020 20:38	System.Runtime	ThreadPool Queue Length	Metric	0	
11	22/11/2020 20:38	System.Runtime	ThreadPool Completed Work Item Count (Count / 1 sec)	Rate	2	
12	22/11/2020 20:38	System.Runtime	Allocation Rate (B / 1 sec)	Rate	50496	
13	22/11/2020 20:38	System.Runtime	Number of Active Timers	Metric	0	
14	22/11/2020 20:38	System.Runtime	GC Fragmentation (%)	Metric	41.08559694	
15	22/11/2020 20:38	System.Runtime	Exception Count (Count / 1 sec)	Rate	0	
16	22/11/2020 20:38	System.Runtime	% Time in GC since last GC (%)	Metric	0	
17	22/11/2020 20:38	System.Runtime	Gen 0 Size (B)	Metric	24	
18	22/11/2020 20:38	System.Runtime	Gen 1 Size (B)	Metric	71392	
19	22/11/2020 20:38	System.Runtime	Gen 2 Size (B)	Metric	80328	

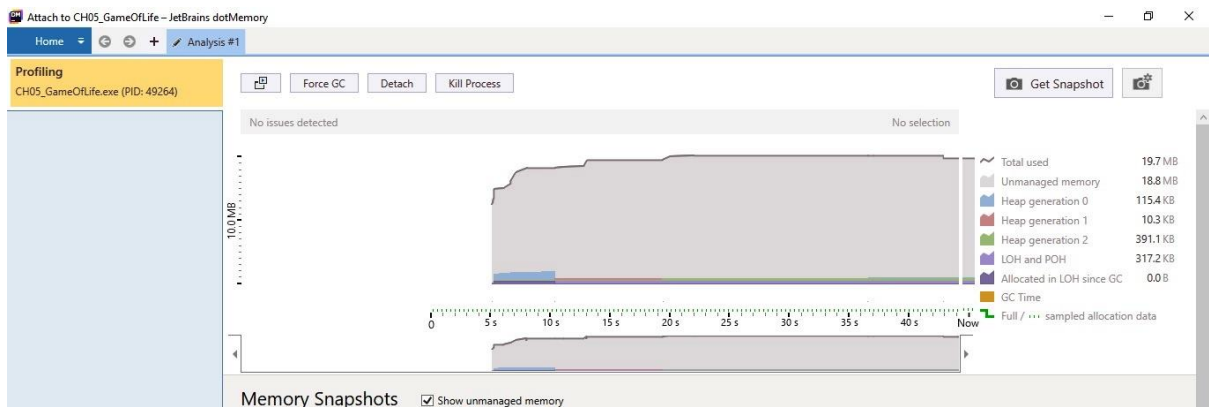
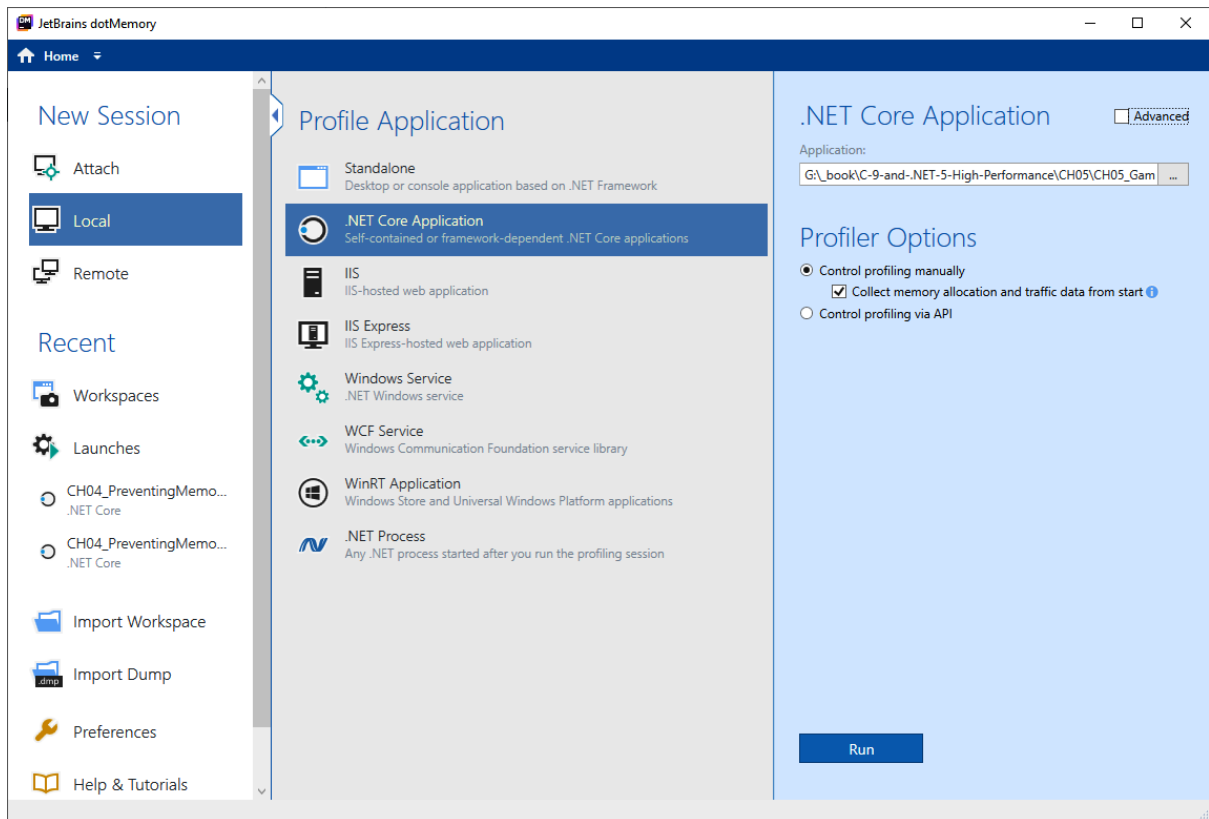
```
Select Administrator: Command Prompt
Microsoft Windows [Version 10.0.19041.572]
(c) 2020 Microsoft Corporation. All rights reserved.

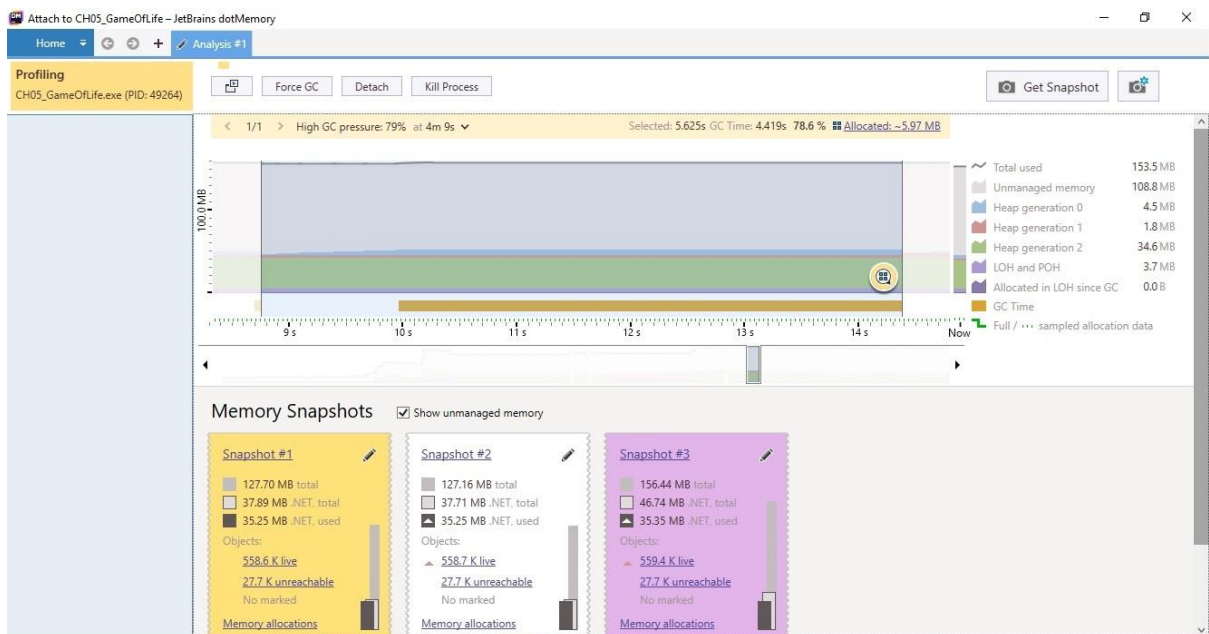
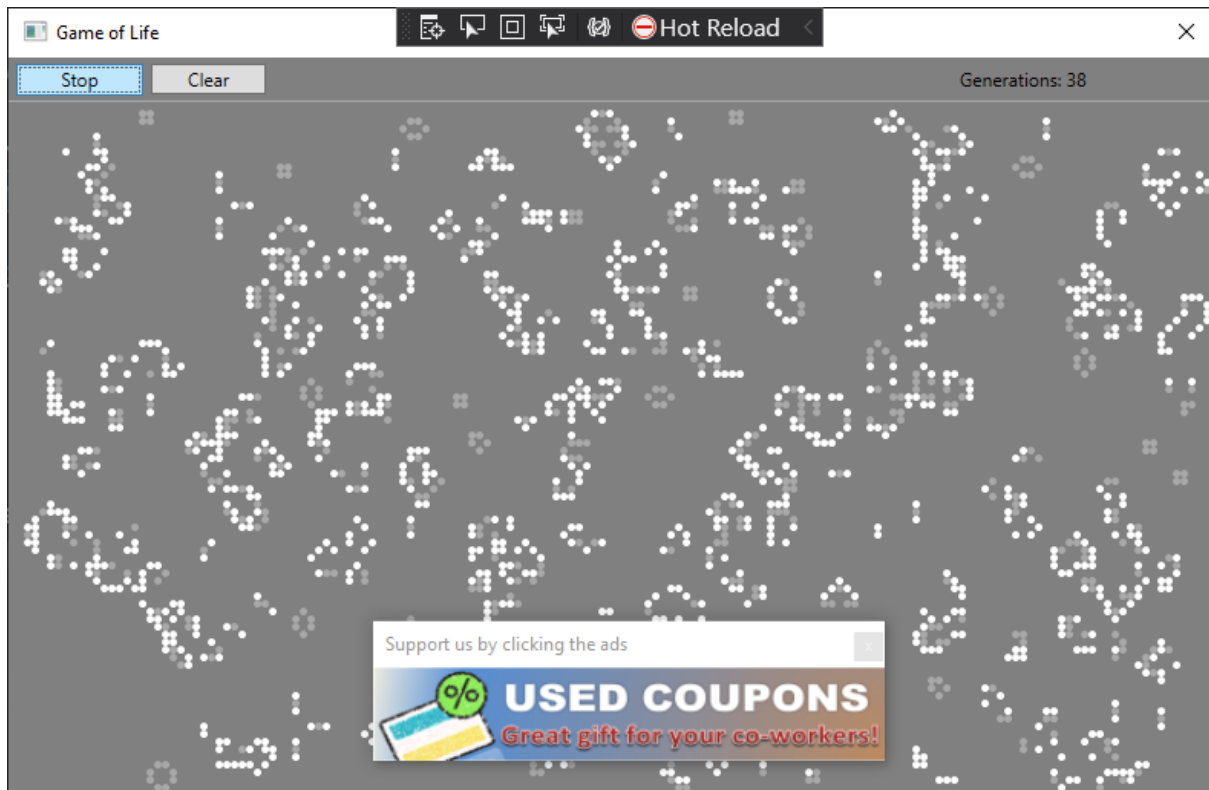
C:\WINDOWS\system32>dotnet-counters ps
5364 CH04_WeakReferences C:\_book\C-9-and-.NET-5-High-Performance\
CH04\CH04_WeakReferences\bin\Release\net5.0\CH04_WeakReferences.exe

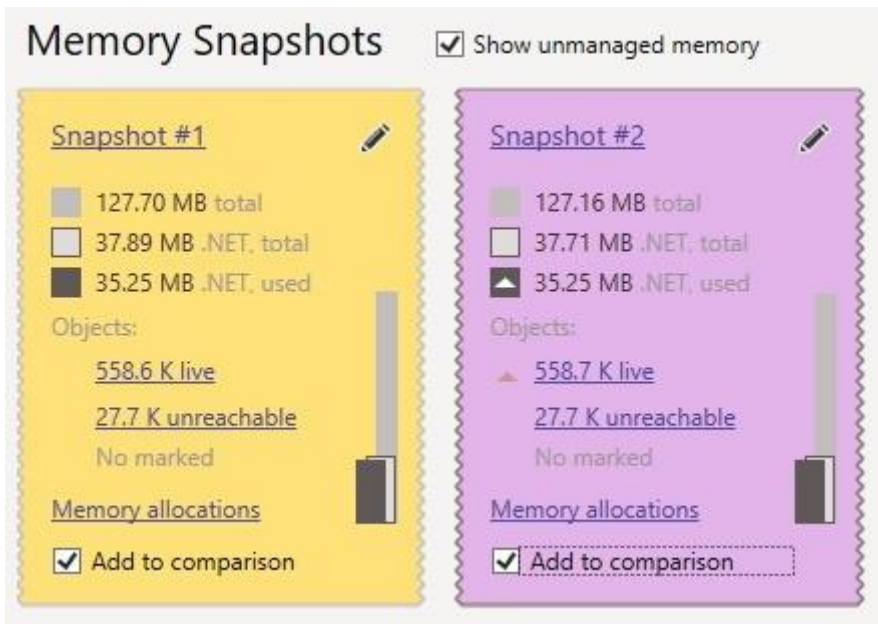
C:\WINDOWS\system32> █
```

```
Administrator: Developer Command Prompt for VS 2019 - dotnet-counters monit...
Press p to pause, r to resume, q to quit.
Status: Running

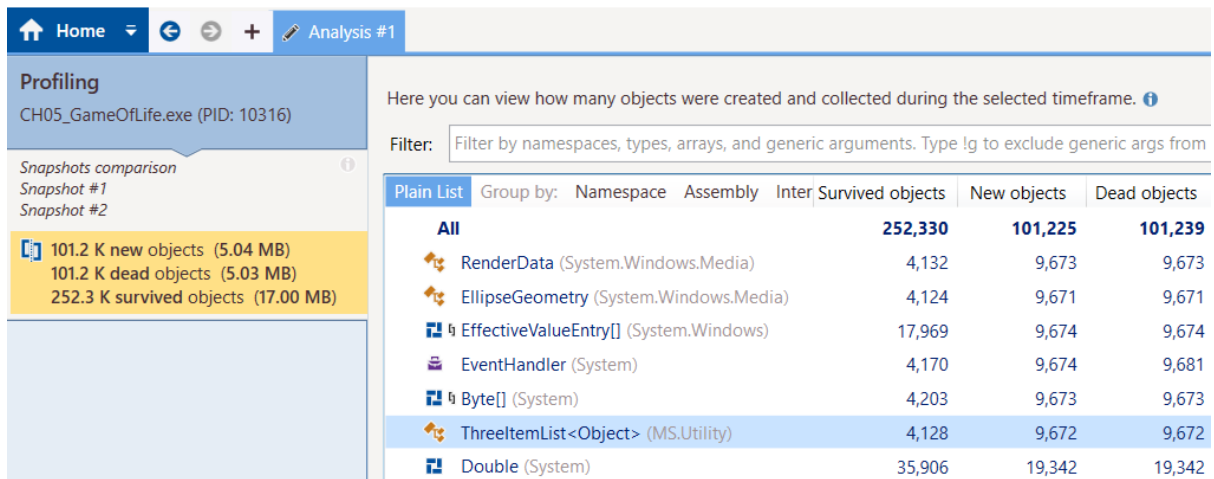
[System.Runtime]
% Time in GC since last GC (%) 0
Allocation Rate (B / 1 sec) 6,056
CPU Usage (%) 0
Exception Count (Count / 1 sec) 0
GC Fragmentation (%) 19.042
GC Heap Size (MB) 0
Gen 0 GC Count (Count / 1 sec) 0
Gen 0 Size (B) 24
Gen 1 GC Count (Count / 1 sec) 0
Gen 1 Size (B) 24
Gen 2 GC Count (Count / 1 sec) 0
Gen 2 Size (B) 80,864
IL Bytes Jitted (B) 26,569
LOH Size (B) 19,640
Monitor Lock Contention Count (Count / 1 sec) 0
Number of Active Timers 0
Number of Assemblies Loaded 9
Number of Methods Jitted 243
POH (Pinned Object Heap) Size (B) 24
ThreadPool Completed Work Item Count (Count / 1 sec) 0
ThreadPool Queue Length 0
ThreadPool Thread Count 0
Working Set (MB) 21
```







CH05_GameOfLife – JetBrains dotMemory



Plain List	Group by:	Namespace	Assembly	Interface	Survived objects	New objects	Dead objects	Objects delta	Survived bytes	New bytes	Dead bytes	Bytes delta
All					249,621	103,951	103,902	49	17,650,871	5,460,726	5,424,282	36,444
System					161,748	80,132	80,060	72	13,977,999	4,535,530	4,500,458	35,072
MS					73,930	10,024	10,047	-23	3,004,360	483,756	482,384	1,372
CH05_GameOfLife					13,801	13,795	13,795		663,904	441,440	441,440	
Cell (CH05_GameOfLife)					13,795	13,795	13,795		441,440	441,440	441,440	
App (CH05_GameOfLife)					1				208			
MainWindow (CH05_GameOfLife)					1				720			
Grid (CH05_GameOfLife)					1				56			
Cell[] (CH05_GameOfLife)					2				220,800			
AdWindow (CH05_GameOfLife)					1				680			

dotMemory 2020.3.4



- ☐ Open "Survived Objects" in the older snapshot
- ☒ Open "Survived Objects" in the newer snapshot

OK

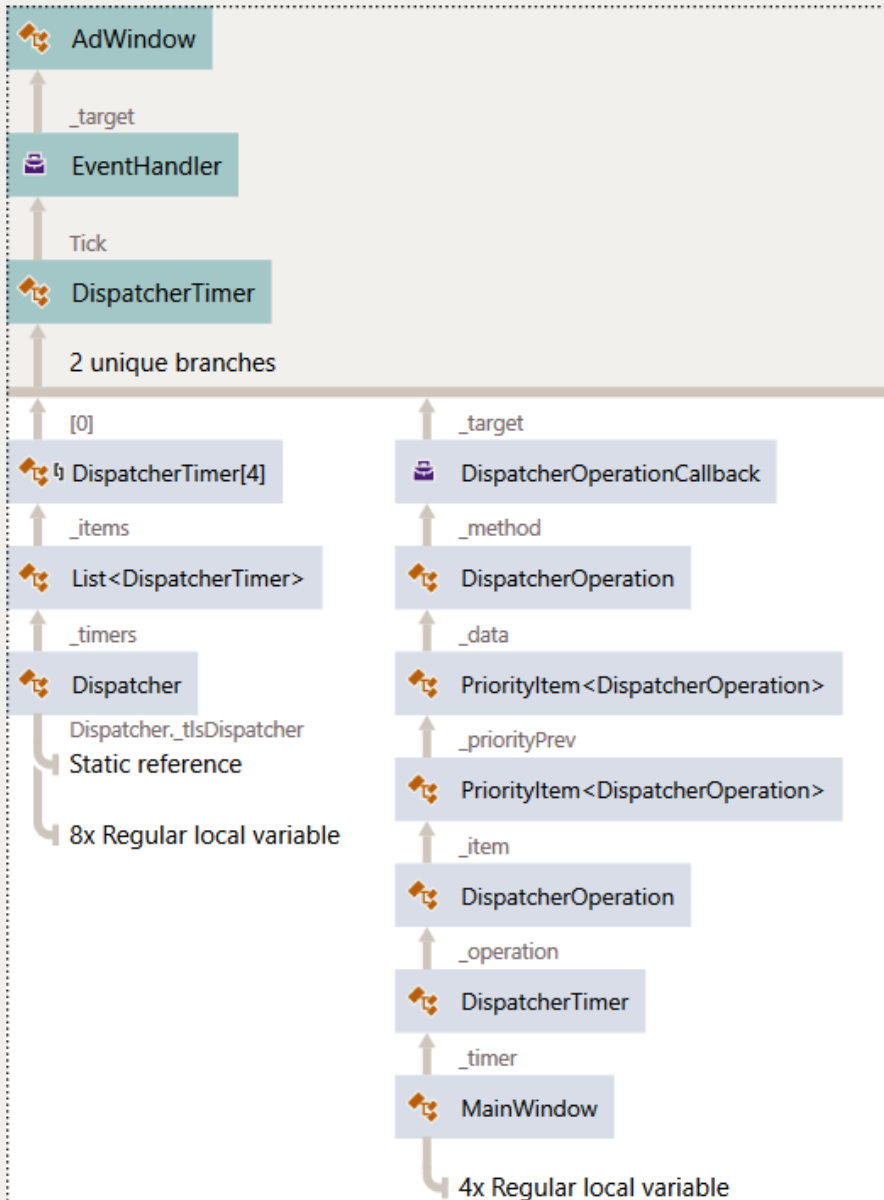
Cancel

Outgoing References

Key Retention Paths

Incoming Referen

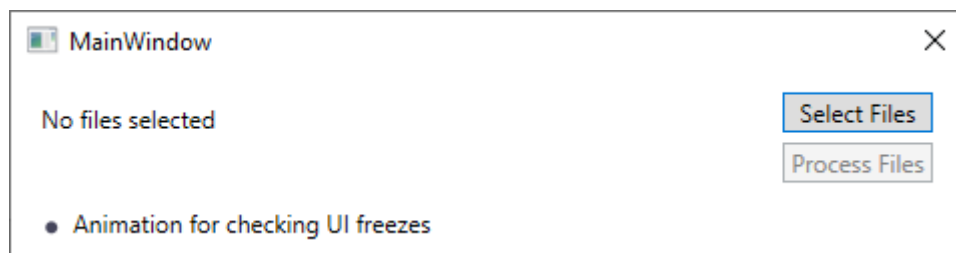
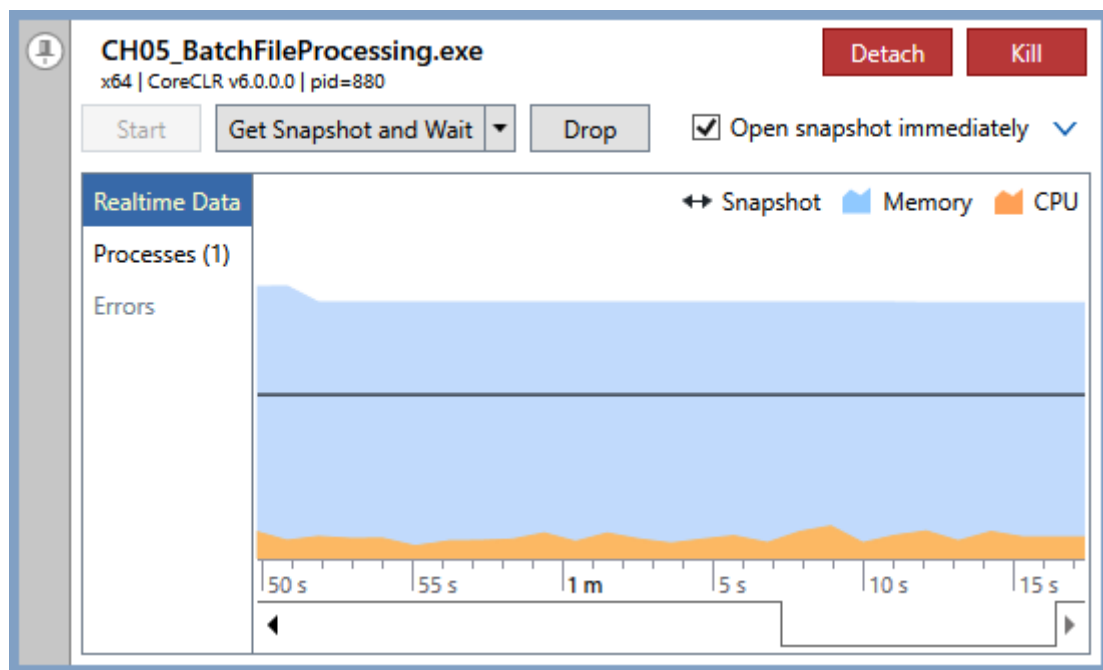
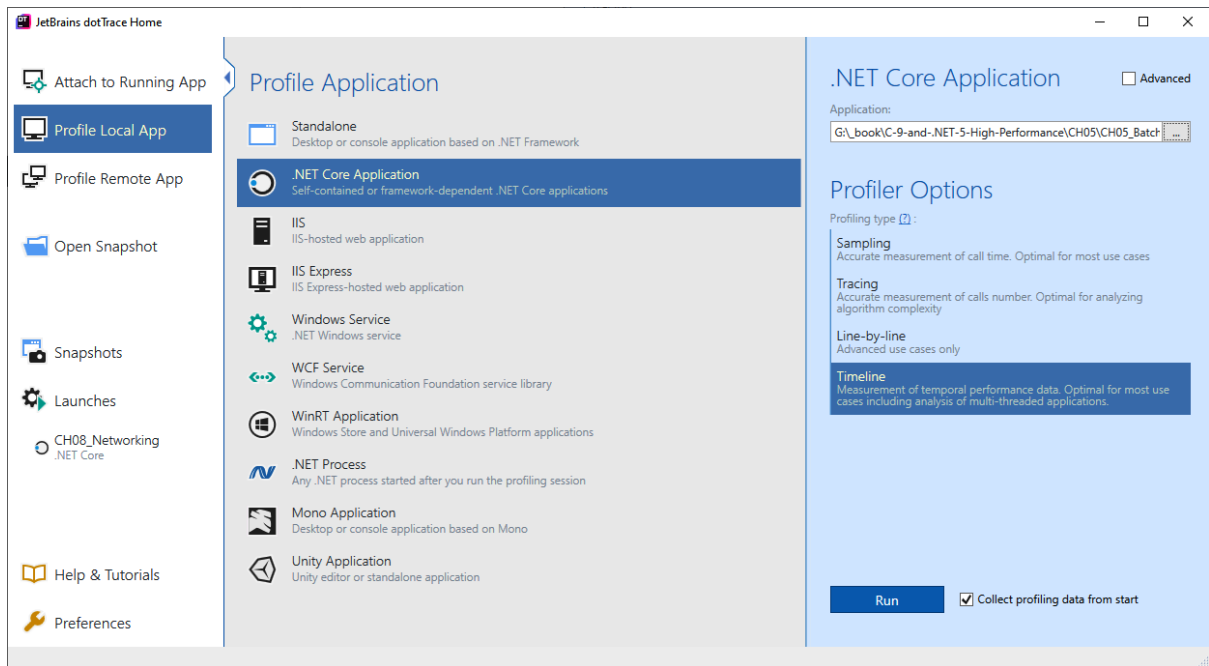
Here you can view key paths to roots for the instance. The graph shows only

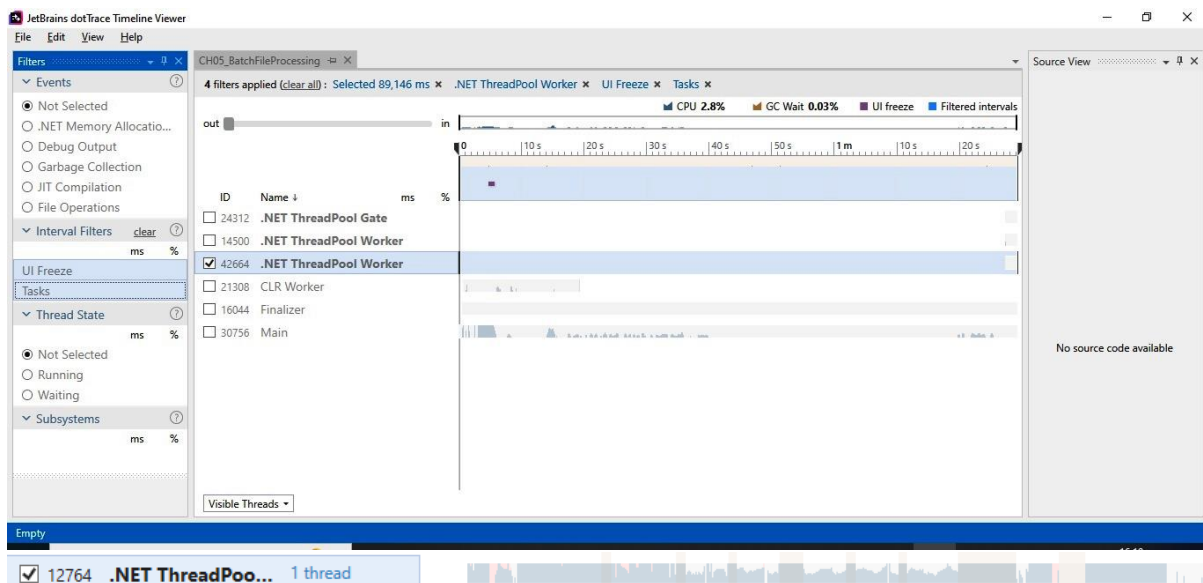
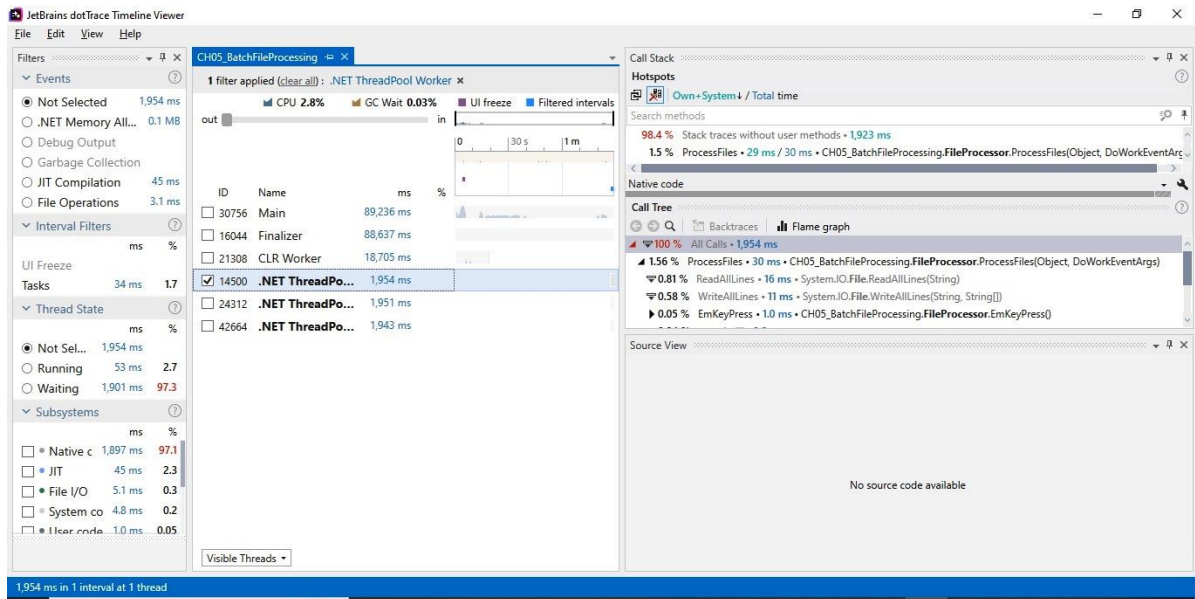


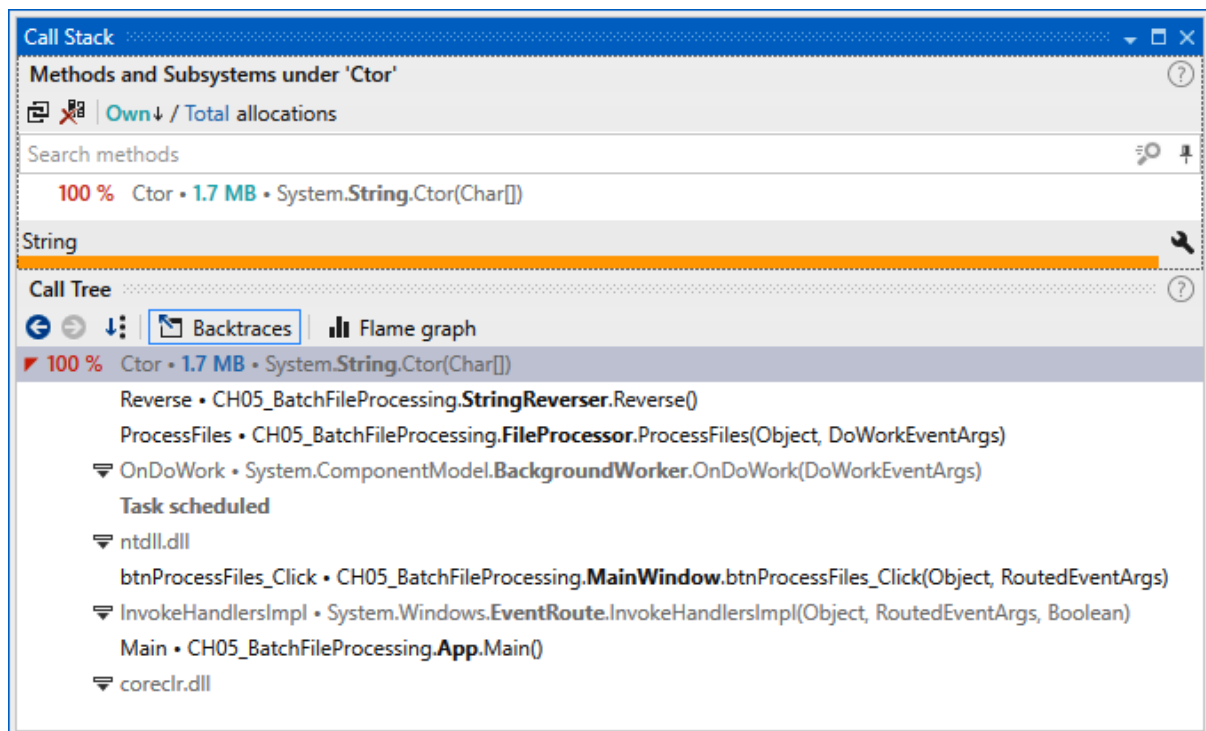
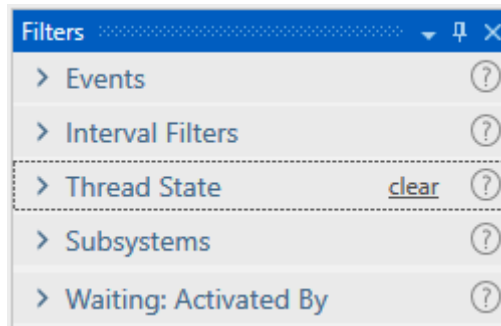
Outgoing References	Key Retention Paths	Incoming References	Creation Stack Trace	Shortest Paths	Merged Shortest Paths
Here you can view what objects are referenced from the instance as well as values of instance's fields. ⓘ					
Type	Address	References count	Bytes	Retained bytes	
DispatcherTimer (System.Windows.Threading)	0000017e2f807188	4	80	6,176	
Fields					
Tick EventHandler (System)	0000017e2f8071f0	1	64	6,072	
Fields					
_methodPtr Int64 140,715,014,550,256 = 0x7FFAC474F2F0					
_methodPtrAux Int64 0 = 0x00					
_invocationCount Int64 0 = 0x00					
_target AdWindow (CH05_GameOfLife)	0000017e2f7fb0d8	11	680	6,008	
Fields					
_dispatcher Dispatcher (System.Windows.Threading)	0000017e2e8fe4c0	15	232	1,904	
_dType DependencyObjectType (System.Windows)	0000017e2f7fb400	2	40	40	
_effectiveValues EffectiveValueEntry[24] (System.Windows)	0000017e2fbf5af0	19	408	848	

Outgoing References	Key Retention Paths	Incoming References	Creation Stack Trace	Shortest Paths	Merged Shortest Paths
Here you can view the stack trace responsible for creating the instance. Calls are shown starting from the last call that directly created the instance descending to the first call in the stack. ⓘ					
Function	Namespace				
AdWindow.ctor(Window)	CH05_GameOfLife				
MainWindow.StartAd()	CH05_GameOfLife				
MainWindow.Button_OnClick(Object, EventArgs)	CH05_GameOfLife				
EventRoute.InvokeHandlersImpl(Object, RoutedEventArgs, Boolean)	System.Windows				
UIElement.RaiseEventImpl(DependencyObject, RoutedEventArgs)	System.Windows				
ButtonBase.OnClick()	System.Windows.Controls.Primitives				
Button.OnClick()	System.Windows.Controls				
ButtonBase.OnMouseLeftButtonUp(MouseButtonEventArgs)	System.Windows.Controls.Primitives				
UIElement.OnMouseLeftButtonUpThunk(Object, MouseButtonEventArgs)	System.Windows				
MouseButtonEventArgs.InvokeEventHandler(Delegate, Object)	System.Windows.Input				
RoutedEventArgs.InvokeHandler(Delegate, Object)	System.Windows				
EventRoute.InvokeHandlersImpl(Object, RoutedEventArgs, Boolean)	System.Windows				

Plain List	Group by: Namespace Assembly Interface	Objects	Bytes	Minimum retained bytes
System		4,32,581	3,18,33,342	3,59,25,178
MS		98,791	40,97,180	1,12,77,070
CH05_GameOfLife		27,596	11,05,344	12,21,078
Cell CH05_GameOfLife		27,590	8,82,880	8,82,880
Cell[,] CH05_GameOfLife		2	2,20,800	11,03,680
AdWindow CH05_GameOfLife		1	680	2,544
App CH05_GameOfLife		1	208	1,560
Grid CH05_GameOfLife		1	56	12,14,136
MainWindow CH05_GameOfLife		1	720	2,838
Microsoft		280	27,192	27,50,790
<CppImplementationDetails>		113	3,288	3,288
global::		16	952	952
Internal		4	112	112
<CrtImplementationDetails>		1	32	200







Source View - System.String.Ctor

Error Decompiled source

☒ Show IL code [Open in Visual Studio](#)

```
701
702 [DynamicDependency("Ctor(System.Char[])")]
703 [MethodImpl(MethodImplOptions.InternalCall)]
704 public extern String(char[]? value);
705
706 private
707 #nullable disable
708 string Ctor(char[] value)
709 {
710     if (value == null || value.Length == 0)
711         return string.Empty;
712     string str = string.FastAllocateString(value.Length);
713 }
```

IL_0000: ldarg.1 // 'value'

IL_0001: brfalse.s IL_0007

IL_0003: ldarg.1 // 'value'

IL_0004: ldlen

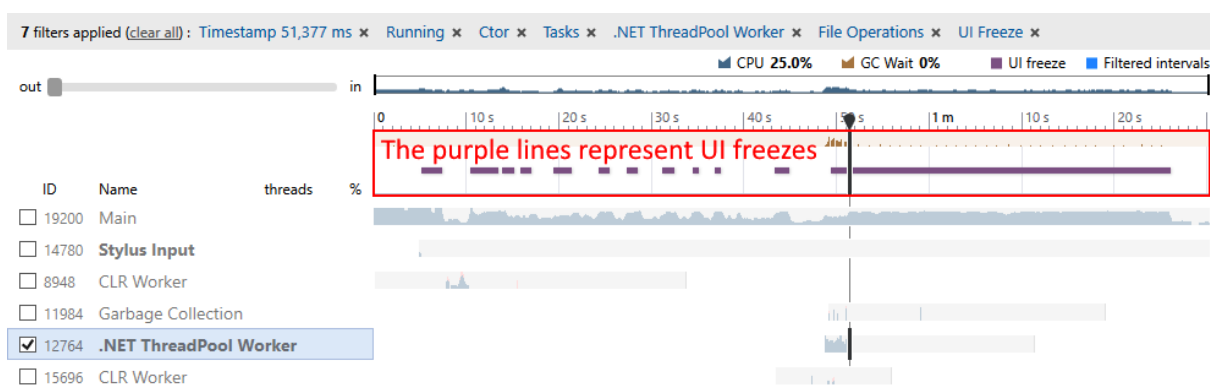
IL_0005: brtrue.s IL_000d

IL_0007: ldsfld string System.String::Empty

IL_000c: ret

IL_000d: ldarg.1 // 'value'

IL_000e: ldlen



Call Stack

Methods and Subsystems

Own / Total allocations

Search methods

- 100 % ProcessInProgress • 9.6 MB • CH05_BatchFileProcessing.MainWindow.ProcessInProgress(Object, ProgressChangedEventArgs)
 - 0 % InternalRealCall • 0 / 9.6 MB • System.Windows.Threading.ExceptionWrapper.InternalRealCall(Delegate, Object, Int32)
 - 0 % TryCatchWhen • 0 / 9.6 MB • System.Windows.Threading.ExceptionWrapper.TryCatchWhen(Object, Delegate, Object, Int32, Delegate)
 - 0 % InvokeImpl • 0 / 9.6 MB • System.Windows.Threading.DispatcherOperation.InvokeImpl()
 - 0 % CallbackWrapper • 0 / 9.6 MB • MS.Internal.CulturePreservingExecutionContext.CallbackWrapper(Object)
 - 0 % RunInternal • 0 / 9.6 MB • System.Threading.ExecutionContext.RunInternal(ExecutionContext, ContextCallback, Object)
 - 0 % Run • 0 / 9.6 MB • MS.Internal.CulturePreservingExecutionContext.Run(CulturePreservingExecutionContext, ContextCallback, Object)
 - 0 % Invoke • 0 / 9.6 MB • System.Windows.Threading.DispatcherOperation.Invoke()
 - 0 % ProcessQueue • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.ProcessQueue()
 - 0 % WndProcHook • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.WndProcHook(IntPtr, Int32, IntPtr, IntPtr, ref Boolean)
 - 0 % WndProc • 0 / 9.6 MB • MS.Win32.HwndWrapper.WndProc(IntPtr, Int32, IntPtr, IntPtr, ref Boolean)
 - 0 % DispatcherCallbackOperation • 0 / 9.6 MB • MS.Win32.HwndSubclass.DispatcherCallbackOperation(Object)
 - 0 % LegacyInvokeImpl • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.LegacyInvokeImpl(DispatcherPriority, TimeSpan, Delegate, Object, Int32)
 - 0 % SubclassWndProc • 0 / 9.6 MB • MS.Win32.HwndSubclass.SubclassWndProc(IntPtr, Int32, IntPtr, IntPtr)
 - 0 % PushFrameImpl • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.PushFrameImpl(DispatcherFrame)
 - 0 % PushFrame • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.PushFrame(DispatcherFrame)
 - 0 % Run • 0 / 9.6 MB • System.Windows.Threading.Dispatcher.Run()
 - 0 % RunDispatcher • 0 / 9.6 MB • System.Windows.Application.RunDispatcher(Object)
 - 0 % RunInternal • 0 / 9.6 MB • System.Windows.Application.RunInternal(Window)
 - 0 % Run • 0 / 9.6 MB • System.Windows.Application.Run(Window)
 - 0 % Run • 0 / 9.6 MB • System.Windows.Application.Run()
 - 0 % Main • 0 / 9.6 MB • CH05_BatchFileProcessing.App.Main()

User code

Call Tree

Backtraces Flame graph

- 100 % All Calls • 9.6 MB
 - 100 % Main • 9.6 MB • CH05_BatchFileProcessing.App.Main()
 - 100 % Run • 9.6 MB • System.Windows.Application.Run()
 - 100 % ProcessInProgress • 9.6 MB • CH05_BatchFileProcessing.MainWindow.ProcessInProgress(Object, ProgressChangedEventArgs)
 - 100 % coreclr.dll • 9.6 MB

Call Stack

Methods and Subsystems

Own / Total allocations

Search methods

- 66.3 % Concat • 169 MB • System.String.Concat(String, String)
- 28.5 % Reverse • 73 MB / 252 MB • CH05_BatchFileProcessing.StringReverser.Reverse()
- 4.0 % ToCharArray • 10 MB • System.String.ToCharArray()
- 1.2 % ProcessFiles • 2.9 MB / 255 MB • CH05_BatchFileProcessing.FileProcessor.ProcessFiles(Object, DoWorkEventArgs)

User code

String

Call Tree

Backtraces Flame graph

- 100 % All Calls • 255 MB
 - 100 % ProcessFiles • 255 MB • CH05_BatchFileProcessing.FileProcessor.ProcessFiles(Object, DoWorkEventArgs)
 - 98.8 % Reverse • 252 MB • CH05_BatchFileProcessing.StringReverser.Reverse()
 - 1.16 % coreclr.dll • 2.9 MB
 - 0.04 % WriteAllLines • 0.1 MB • System.IO.File.WriteAllLines(String, IEnumerable)

Call Stack Source View - CH05_BatchFileProcessing.StringReverser.Reverse

Call Stack

Methods and Subsystems

Own / Total allocations

Search methods

26.6 % Ctor • 9.9 MB • System.String.Ctor(Char[], Int32, Int32)

19.7 % Ctor • 7.3 MB • System.String.Ctor(Char[])

10.1 % ProcessFiles • 3.8 MB / 37 MB • CH05_BatchFileProcessing.FileProcessor.ProcessFiles(Object, DoWorkEventArgs)

5.7 % set_Capacity • 2.1 MB • System.Collections.Generic.List`1.set_Capacity(Int32)

2.7 % ToArray • 1.0 MB • System.Collections.Generic.List`1.ToArray()

1.4 % AllocateUninitializedArray • 0.5 MB • System.GC.AllocateUninitializedArray(Int32, Boolean)

0.5 % StringBuilder..ctor • 0.2 MB / 0.7 MB • System.Text.StringBuilder..ctor(Int32, Int32)

0.3 % ReadLine • 0.1 MB / 11 MB • System.IO.StreamReader.ReadLine()

0.3 % ToString • 0.1 MB • System.Text.StringBuilder.ToString()

▼ 0 % Reverse • 0 / 19 MB • CH05_BatchFileProcessing.StringReverser.Reverse()

0 % AddWithResize • 0 / 2.1 MB • System.Collections.Generic.List`1.AddWithResize(T)

User code StringCollecti...

Call Tree

BacktracesFlame graph

▲ 100 % All Calls • 37 MB

▲ 100 % ProcessFiles • 37 MB • CH05_BatchFileProcessing.FileProcessor.ProcessFiles(Object, DoWorkEventArgs)

▶ 52.4 % Reverse • 19 MB • CH05_BatchFileProcessing.StringReverser.Reverse()

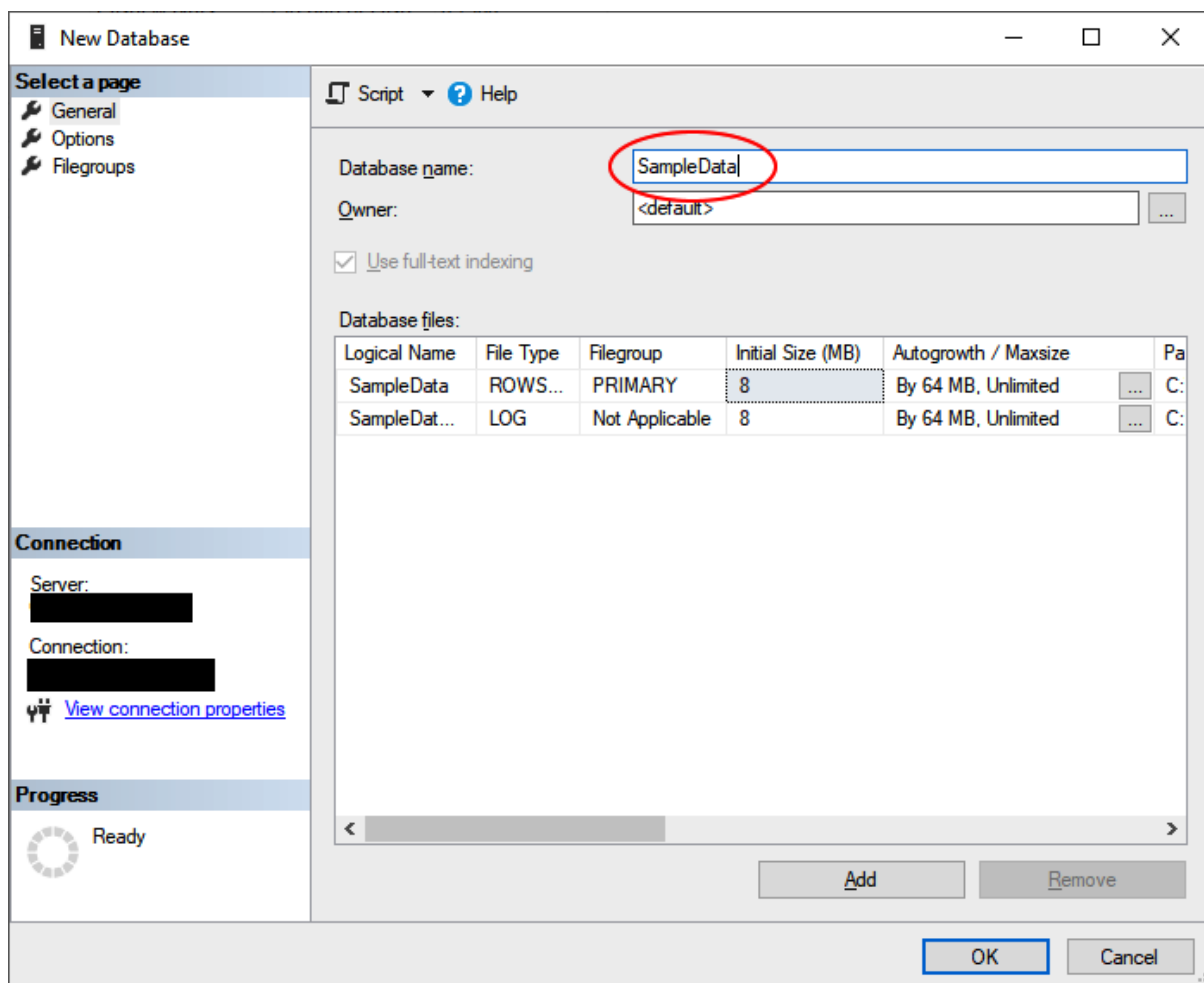
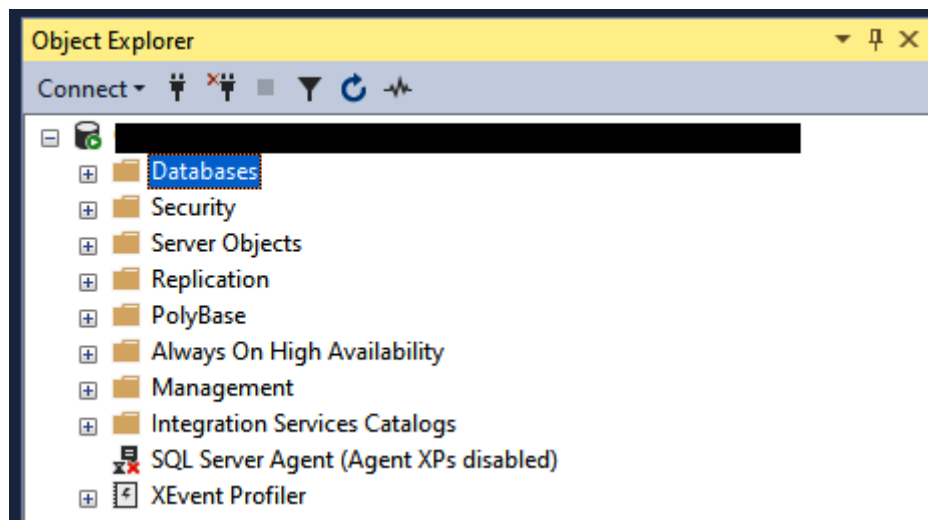
▼ 37.5 % ReadAllLines • 14 MB • System.IO.File.ReadAllLines(String)

10.1 % coreclr.dll • 3.8 MB

Call Stack

Source View - CH05_BatchFileProcessing.StringReverser.Reverse

Chapter 6: The .NET Collections



```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT

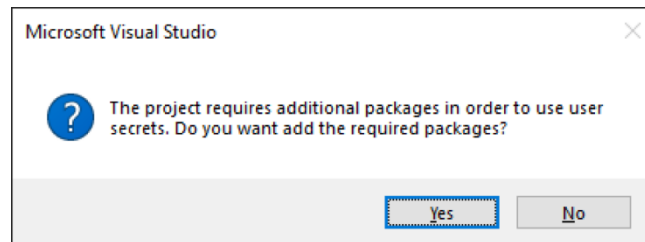
| Method | Mean | Error | StdDev | Median | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| BasicRateInterface | 7.938 ns | 0.4089 ns | 1.1666 ns | 7.494 ns | 1 | 0.0204 | - | - | 32 B |
| BasicRateConcrete | 8.110 ns | 0.2150 ns | 0.1795 ns | 8.127 ns | 2 | 0.0204 | - | - | 32 B |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT

| Method | Mean | Error | StdDev | Median | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| ArrayAdd1000Logic | 1,220.1710 ns | 23.2521 ns | 64.4315 ns | 1,189.0492 ns | 4 | 2.5635 | - | - | 4024 B |
| CollectionAdd1000Logic | 3,555.7861 ns | 34.5734 ns | 28.8703 ns | 3,552.7678 ns | 6 | 5.3596 | - | - | 8424 B |
| ArrayIterationLogic | 1,466.9898 ns | 15.4082 ns | 13.6590 ns | 1,468.7730 ns | 5 | - | - | - | - |
| CollectionIterationLogic | 840.3122 ns | 12.5182 ns | 11.0971 ns | 837.0914 ns | 3 | - | - | - | - |
| ArrayGetElement500Logic | 0.0550 ns | 0.0348 ns | 0.0957 ns | 0.0000 ns | 1 | - | - | - | - |
| CollectionGetElement500Logic | 0.6037 ns | 0.0490 ns | 0.0459 ns | 0.6024 ns | 2 | - | - | - | - |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| BenchmarkIEnumerable | 127.36 ns | 2.137 ns | 2.624 ns | 2 | 0.0253 | - | - | 40 B |
| BenchmarkIEnumerator | 58.78 ns | 0.960 ns | 1.577 ns | 1 | 0.0254 | - | - | 40 B |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
```

Method	Mean	Error	StdDev	Median	Rank	Gen 0	Gen 1	Gen 2	Allocated
QueryDb	700.6 us	39.67 us	113.81 us	657.2 us	2	-	-	-	16.67 KB
QueryDbAsList	691.4 us	33.00 us	94.16 us	676.8 us	2	-	-	-	12.26 KB
QueryDbAsIEnumerableable	695.0 us	37.09 us	104.63 us	671.8 us	2	-	-	-	12.22 KB
QueryDbAsIEnumerator	551.9 us	27.85 us	77.62 us	542.4 us	1	-	-	-	11.41 KB
QueryDbAsIQueryable	675.8 us	29.08 us	82.01 us	653.4 us	2	-	-	-	12.22 KB

Error List Developer Command Prompt Developer PowerShell

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
```

Method	Mean	Error	StdDev	Rank	Gen 0	Gen 1	Gen 2	Allocated
GetValuesBenchmark	13,102,611.27 ns	106,549.512 ns	94,453.367 ns	2	734.3750	734.3750	734.3750	16777666 B
GetValuesYieldBenchmark	14.50 ns	0.303 ns	0.237 ns	1	0.0255	-	-	40 B

Developer Command Prompt Developer PowerShell

```

Developer Command Prompt
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.102
[Host] : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT
DefaultJob : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT

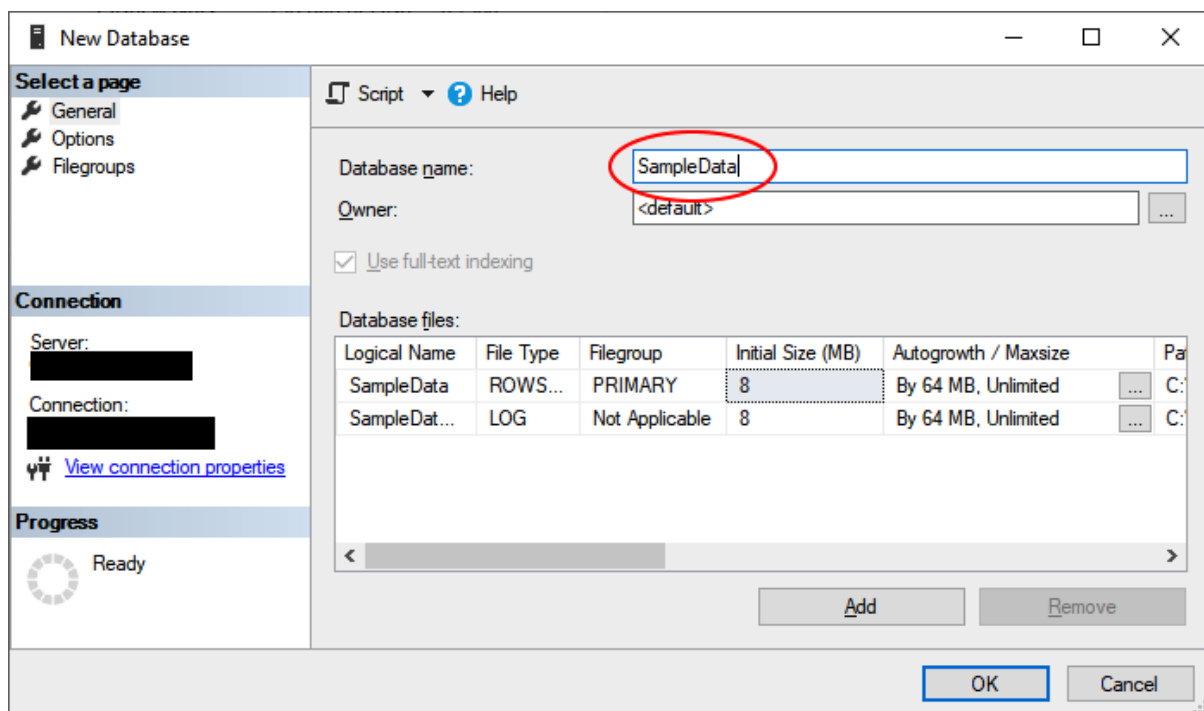
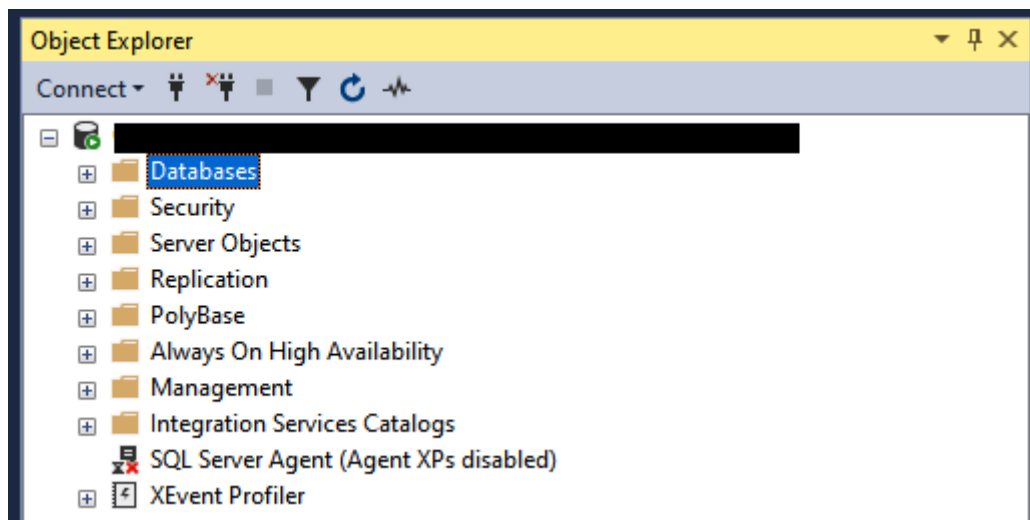
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Method | Mean | Error | StdDev | Median | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| ValueOperatorValue | 0.0354 ns | 0.0288 ns | 0.0384 ns | 0.0294 ns | 1 | - | - | - | - |
| ValueEqualsValue | 0.0553 ns | 0.0456 ns | 0.1323 ns | 0.0000 ns | 1 | - | - | - | - |
| ReferenceOperatorReference | 0.3477 ns | 0.0959 ns | 0.2783 ns | 0.3110 ns | 2 | - | - | - | - |
| ReferenceEqualsReference | 3.5910 ns | 0.2315 ns | 0.6717 ns | 3.4596 ns | 4 | - | - | - | - |
| StringOperatorString | 4.4132 ns | 0.2828 ns | 0.8069 ns | 4.2949 ns | 5 | - | - | - | - |
| StringEqualsString | 1.8127 ns | 0.1474 ns | 0.4205 ns | 1.7387 ns | 3 | - | - | - | - |

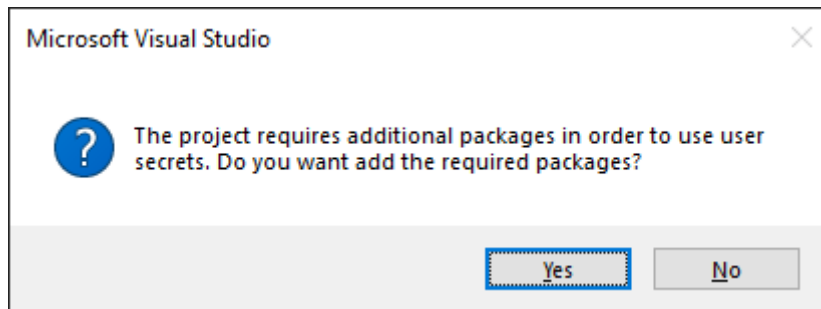
```

Name	Type	Null	Primary Key	Auto Increment
Id	int	No	Yes	Yes
Name	nvarchar(50)	No	No	No
Description	nvarchar(255)	No	No	No
UnitPrice	money	No	No	No

Id	Name	Description	UnitPrice
1	Roasted Peanuts	500g bag of dry roasted peanuts.	0.69
2	Cashew Nuts	75g bag of cashew nuts.	0.75
3	Milk (Whole)	2 liters of whole milk.	1.25
4	Bread (50/50)	50% white and 50% wholemeal bread.	1
5	Butter (Salted)	100g salted butter.	2.5
6	Roast Chicken	5kg frozen roast chicken.	4.99
7	Potatoes	5kg Maris variety potatoes.	1.75
8	Roasting Vegetables	1kg bag of frozen roasting vegetables.	1.5
9	Coffee	1kg of Arabic coffee.	2.99
10	Demera Sugar	1kg bag of Demera sugar.	1
11	Chicken Gravy	1 tub of chicken gravy granules.	0.89
12	Yorkshire Puddings	1 bag of 12 frozen Yorkshire puddings.	1.35
13	Sage and Onion Stuffing	1 box of sage and onion stuffing.	0.59

Chapter 7: LINQ Performance





```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=5.0.101
[Host] : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
DefaultJob : .NET Core 5.0.1 (CoreCLR 5.0.120.57516, CoreFX 5.0.120.57516), X64 RyuJIT
```

Method	Mean	Error	StdDev	Median	Rank	Gen 0	Gen 1	Gen 2	Allocated
QueryDb	700.6 us	39.67 us	113.81 us	657.2 us	2	-	-	-	16.67 KB
QueryDbAsList	691.4 us	33.00 us	94.16 us	676.8 us	2	-	-	-	12.26 KB
QueryDbAsIEnumerable	695.0 us	37.09 us	104.63 us	671.8 us	2	-	-	-	12.22 KB
QueryDbAsIEnumerator	551.9 us	27.85 us	77.62 us	542.4 us	1	-	-	-	11.41 KB
QueryDbAsIQueryable	675.8 us	29.08 us	82.01 us	653.4 us	2	-	-	-	12.22 KB

Error List Developer Command Prompt Developer PowerShell

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
```

Method	Mean	Error	StdDev	Rank	Gen 0	Gen 1	Gen 2	Allocated
GetLastPersonVersion2	0.3532 ns	0.0316 ns	0.0280 ns	1	-	-	-	-
GetLastPersonVersion1	24.6327 ns	0.3540 ns	0.4726 ns	2	-	-	-	-

```
Developer PowerShell
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| ReadingDataWithoutUsingLet | 33.46 ns | 0.676 ns | 0.779 ns | 1 | 0.0561 | - | - | 88 B |
| ReadingDataUsingLet | 194.45 ns | 3.451 ns | 6.396 ns | 2 | 0.2446 | - | - | 384 B |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| GroupByVersion3 | 1.961 us | 0.0270 us | 0.0252 us | 1 | 1.3390 | - | - | 2.05 KB |
| GroupByVersion2 | 2.011 us | 0.0265 us | 0.0248 us | 2 | 1.1826 | - | - | 1.81 KB |
| GroupByVersion1 | 2.204 us | 0.0351 us | 0.0595 us | 3 | 1.2817 | - | - | 1.97 KB |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| FilterGroupsVersion1 | 61.70 ns | 1.011 ns | 0.896 ns | 1 | 0.1169 | - | - | 184 B |
| FilterGroupsVersion2 | 158.77 ns | 2.680 ns | 2.507 ns | 2 | 0.1931 | - | - | 304 B |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| FilterGroupsVersion3 | 14.59 ns | 1.655 ns | 4.881 ns | 1 | 0.0204 | - | - | 32 B |
| FilterGroupsVersion1 | 61.86 ns | 1.090 ns | 1.664 ns | 2 | 0.1173 | - | - | 184 B |
| FilterGroupsVersion2 | 160.30 ns | 3.280 ns | 3.509 ns | 3 | 0.1938 | - | - | 304 B |
```

```
Developer Command Prompt
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| FilterGroupsVersion4 | 8.754 ns | 0.2678 ns | 0.2631 ns | 1 | 0.0204 | - | - | 32 B |
| FilterGroupsVersion3 | 8.921 ns | 0.2659 ns | 0.3265 ns | 1 | 0.0204 | - | - | 32 B |
| FilterGroupsVersion1 | 61.049 ns | 1.3271 ns | 1.9452 ns | 2 | 0.1173 | - | - | 184 B |
| FilterGroupsVersion2 | 160.216 ns | 3.2436 ns | 3.6053 ns | 3 | 0.1931 | - | - | 304 B |
```

```
Developer PowerShell
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
.NET Core SDK=6.0.100-preview.2.21155.3
[Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT
DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.15406, CoreFX 6.0.21.15406), X64 RyuJIT

| Method | Mean | Error | StdDev | Rank | Gen 0 | Gen 1 | Gen 2 | Allocated |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| LinqWithParameters | 771.5 ns | 14.39 ns | 13.46 ns | 1 | 1.2598 | - | - | 1.93 KB |
| LinqWithLocalVariable | 1,048.0 ns | 20.80 ns | 23.95 ns | 2 | 1.6975 | - | - | 2.6 KB |
```

Name	Type	Null	Primary Key	Auto Increment
Id	int	No	Yes	Yes
Name	nvarchar(50)	No	No	No
Description	nvarchar(255)	No	No	No
UnitPrice	money	No	No	No

Id	Name	Description	UnitPrice
1	Roasted Peanuts	500 g bag of dry roasted peanuts	0.69
2	Cashew Nuts	75 g bag of cashew nuts	0.75
3	Milk (Whole)	2 litres of whole milk	1.25
4	Bread (50/50)	50% white and 50% wholemeal bread	1
5	Butter (Salted)	100 g salted butter	2.5
6	Roast Chicken	5 kg frozen roast chicken	4.99
7	Potatoes	5 kg Maris Piper variety potatoes	1.75
8	Roasting Vegetables	1 kg bag of frozen roasting vegetables	1.5
9	Coffee	1 kg of Arabic coffee	2.99
10	Demerara Sugar	1 kg bag of Demerara sugar	1
11	Chicken Gravy	1 tub of chicken gravy granules	0.89
12	Yorkshire Puddings	1 bag of 12 frozen Yorkshire puddings	1.35
13	Sage and Onion Stuffing	1 box of sage and onion stuffing	0.59

Chapter 8: File and Stream I/O

Developer PowerShell

+ Developer PowerShell

```
// * Summary *  
  
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042  
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores  
.NET Core SDK=5.0.102  
[Host] : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT  
DefaultJob : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT
```

Method	Mean	Error	StdDev	Median	Rank	Gen 0	Gen 1	Gen 2	Allocated
FileCopy	99,829.85 us	6,040.520 us	16,433.681 us	91,663.30 us	3	-	-	-	72 B
FileInfoMoveTo	1,312.02 us	23.782 us	19.859 us	1,309.93 us	2	-	-	-	833 B
FileInfoReadCacheAndMoveTo	79.65 us	0.564 us	0.554 us	79.77 us	1	-	-	-	-

Data Tools Operations Output Error List Developer PowerShell

Developer PowerShell

+ Developer PowerShell

```
// * Summary *  
  
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042  
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores  
.NET Core SDK=5.0.102  
[Host] : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT  
DefaultJob : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT
```

Method	Mean	Error	StdDev	Rank	Gen 0	Gen 1	Gen 2	Allocated
GetDirectorySizeUsingGetFileSystemInfos	3.643 ms	0.0341 ms	0.0302 ms	1	304.6875	148.4375	-	1.2 MB
GetDirectorySizeUsingArrayAndFileInfo	143.327 ms	1.3808 ms	2.8515 ms	2	-	-	-	1.16 MB

Developer PowerShell

+ Developer PowerShell

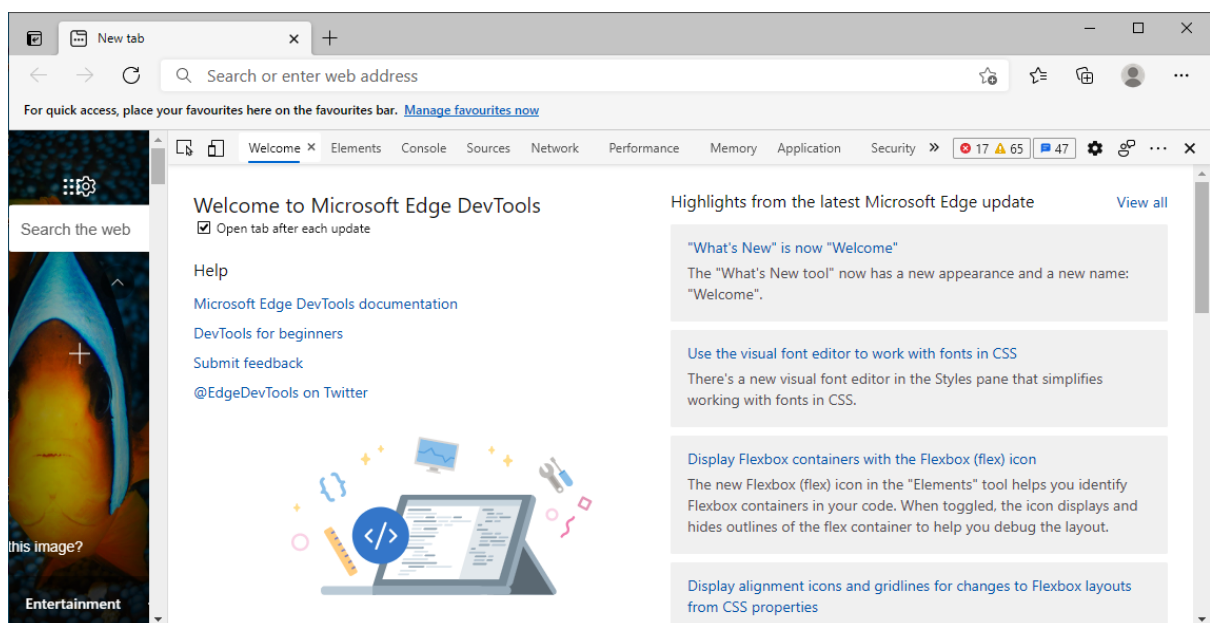
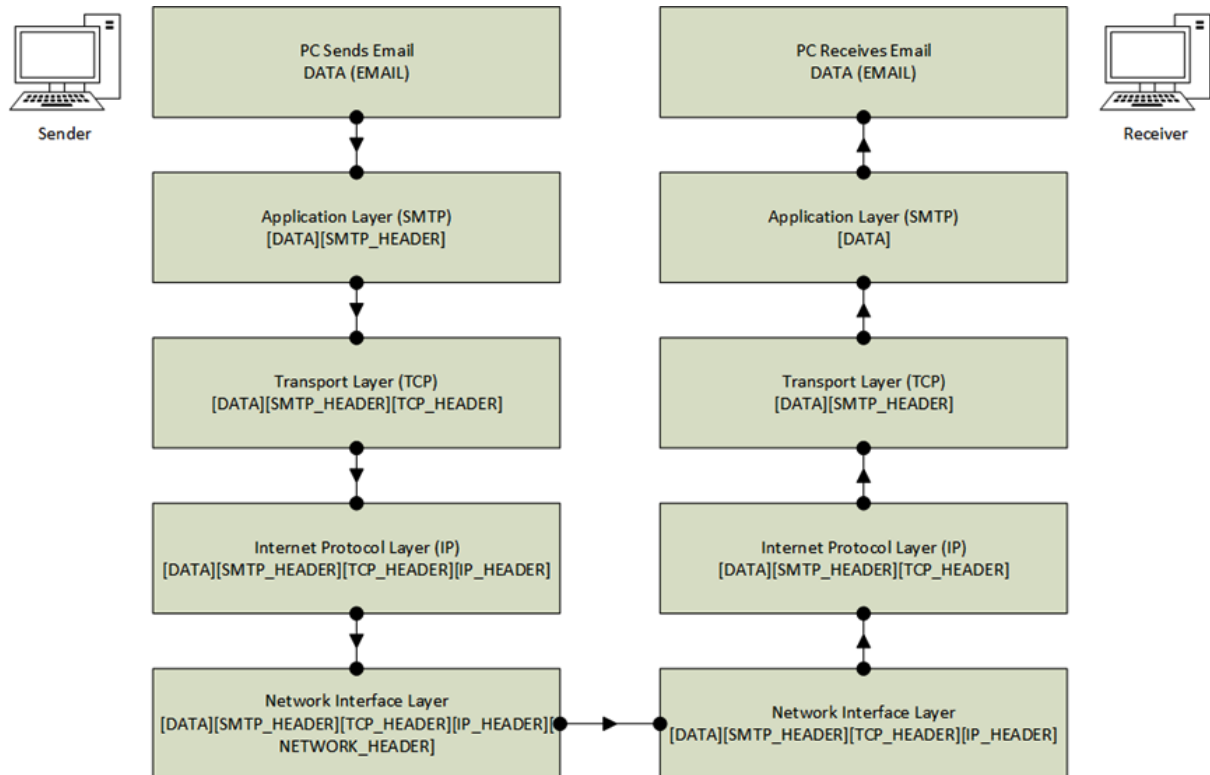
```
// * Summary *  
  
BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042  
Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores  
.NET Core SDK=5.0.102  
[Host] : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT  
DefaultJob : .NET Core 5.0.2 (CoreCLR 5.0.220.61120, CoreFX 5.0.220.61120), X64 RyuJIT
```

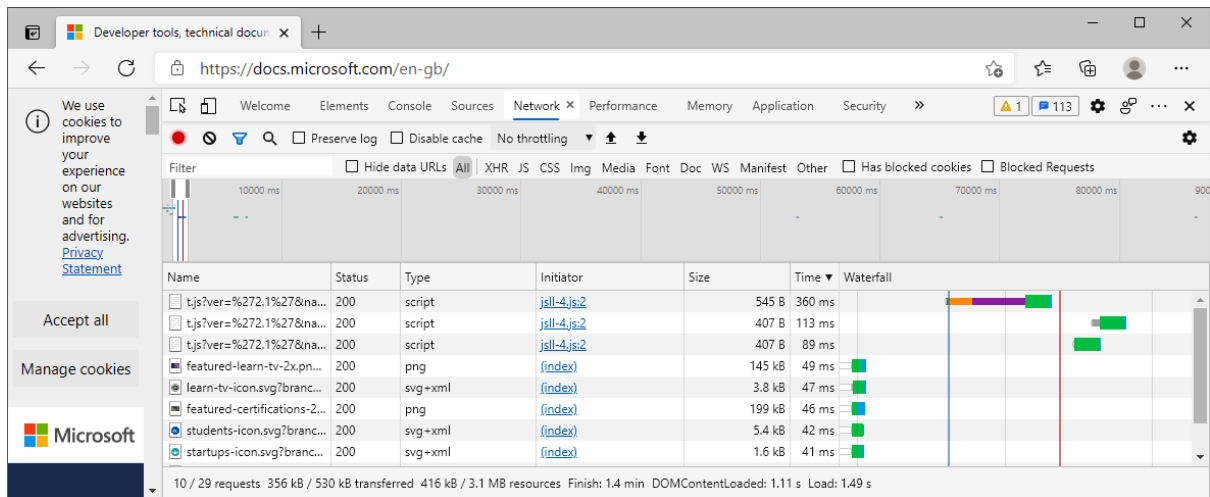
Method	Mean	Error	StdDev	Rank	Gen 0	Gen 1	Gen 2	Allocated
ReturnFormattedString	567.0 ns	10.79 ns	26.66 ns	2	0.1411	-	-	224 B
ReturnInterpolatedString	257.8 ns	5.25 ns	12.38 ns	1	0.1683	-	-	264 B

Path	Description
C:\Work\Databases\DevDbDataDictionary.xls	Absolute path. The drive being used is C:\.
\temp\training.txt	Absolute path. This is relative to the current location.
2021\expenses.docx	The relative path to the current directory.
..\temp\training.txt	The relative path up a level from the current directory.
\\Staff\SEN\2020\Yr7\StudentWellbeing.txt	UNC path to the shared network resource.
\\.\C:\Admin\log.txt	DOS device path.
\\?\C:\Admin\log.txt	DOS device path.
\\.\Volume{GUID}\Admin\log.txt	DOS device path.

Exception	Description
<code>IOException</code>	The base I/O exception that all the other I/O exceptions derive from.
<code>FileNotFoundException</code>	This exception is raised when an attempt to read a file fails.
<code>DirectoryNotFoundException</code>	This exception is raised when an attempt to read a directory fails.
<code>DriveNotFoundException</code>	This exception is raised when an attempt to read a drive fails.
<code>PathTooLongException</code>	This exception is raised when the file path exceeds the 256 Windows path length limit.
<code>OperationCanceledException</code>	This exception is raised when an I/O operation is canceled.
<code>UnauthorizedAccessException</code>	This exception is raised when read/write access has been attempted on a file or directory that the user does not have access to.

Chapter 9: Enhancing the Performance of Networked Applications





CH08_GrpcServiceClient

Connected Services

Publish

Service Dependencies

There are currently no service dependencies configured.

Add

Service References (OpenAPI, gRPC)

There are currently no service references configured.


Add


Other Services

Microsoft WCF Web Service Reference Provider
Add a WCF web service reference to your project.

Add service reference

Select a service reference to add to your application

**OpenAPI**
Consume web services which conform to the OpenAPI Specification

**gRPC**
Consume and produce web services which conform to the gRPC open source universal RPC framework

Back

Next

Finish

Cancel

Add new gRPC service reference

Select a file or URL

☐ File

ook\C-9-and-.NET-5-High-Performance\CH08\CH08_GrpcService\Protos\greet.proto

Browse...

☒ URL

Select the type of class to be generated

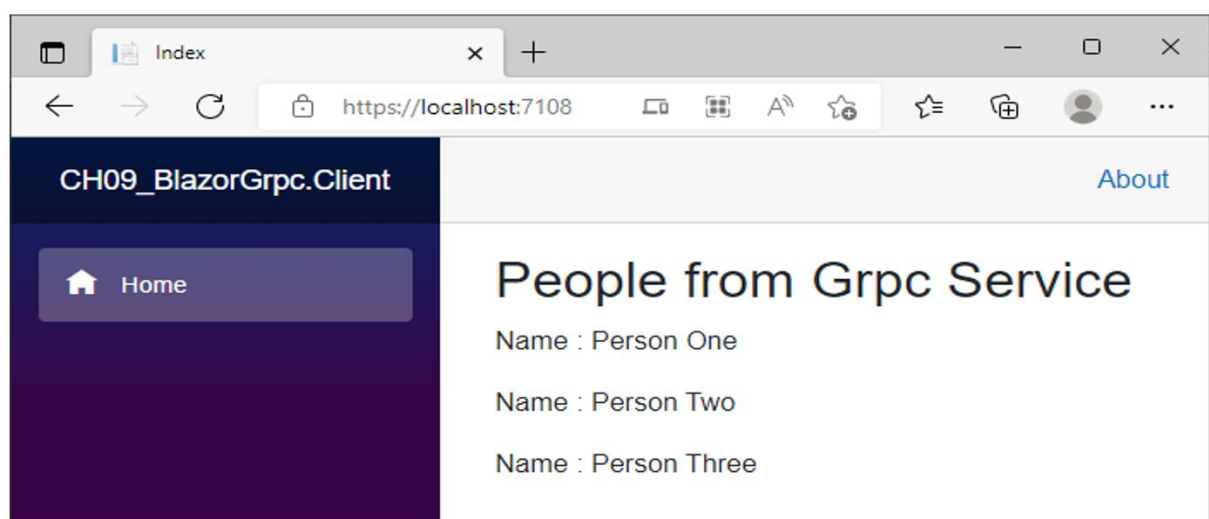
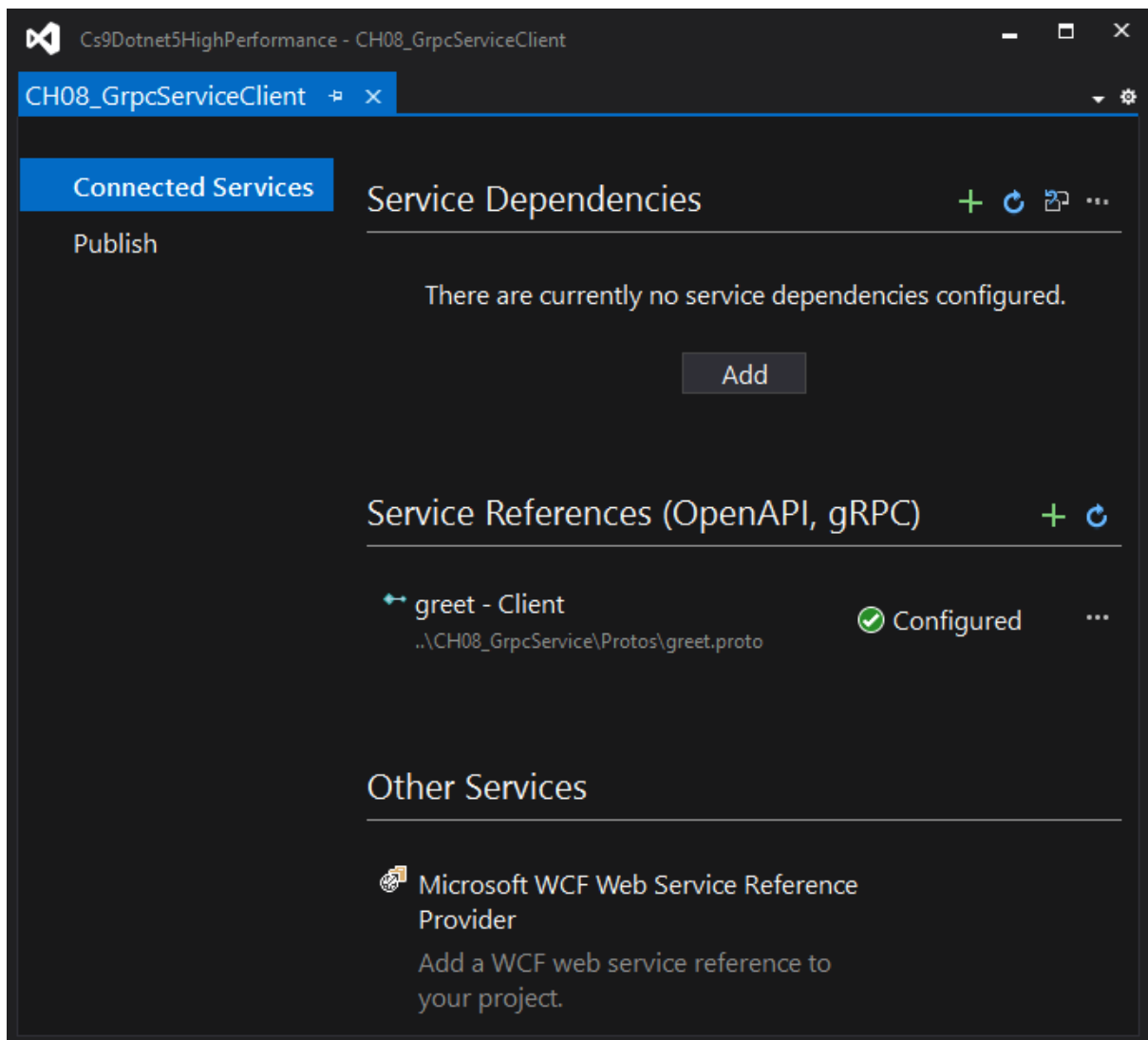
Client

Back

Next

Finish

Cancel



```
Developer PowerShell
+ Developer PowerShell
G:\_book\Source\Mastering-High-Performance-with-C-10.0-and-.NET-6\CH09\CH09_TcpServer\bin\Debug\net6.0>CH09_TcpServer.exe
Listening on port 7000
```

```
Developer PowerShell
+ Developer PowerShell
G:\_book\Source\Mastering-High-Performance-with-C-10.0-and-.NET-6\CH09\CH09_TcpClient\bin\Debug\net6.0>CH09_TcpClient.exe
Connecting to port 7000
```

```
Developer PowerShell
+ Developer PowerShell
G:\_book\Source\Mastering-High-Performance-with-C-10.0-and-.NET-6\CH09\CH09_TcpClient\bin\Debug\net6.0>CH09_TcpClient.exe
Connecting to port 7000
Hello, World!
```

```
Developer PowerShell
+ Developer PowerShell
G:\_book\Source\Mastering-High-Performance-with-C-10.0-and-.NET-6\CH09\CH09_TcpServer\bin\Debug\net6.0>CH09_TcpServer.exe
Listening on port 7000
[::ffff:127.0.0.1]:61193]: connected
Hello, World!
```

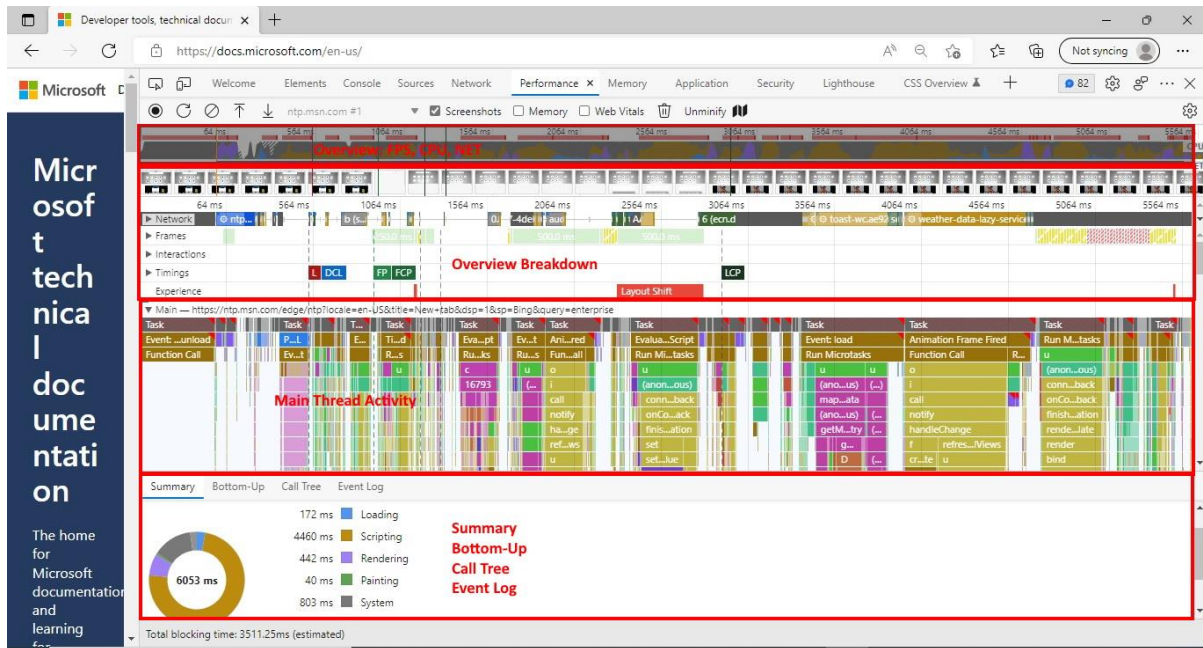
Index

When Cached: 04/08/2021 20:34:17;Current Time: 04/08/2021 20:34:25;

OSI Reference Model	Protocol Suite
Application	HTTP, HTTPS, SSL, FTP, TFTP, Telnet, NTP, NNTP
Presentation	Email: SMTP, POP, IMAP
Session	TCP, UDP
Transport	
Network	IP, IGMP, ICMP, ARP, RARP
Data Link	Ethernet, PPP
Physical	

TCP/IP Model	OSI Model
Application Layer	Application Layer
	Presentation Layer
	Session Layer
Transport Layer	Transport Layer
Internet Layer	Network Layer
Network Interface Layer	Data Link Layer
	Physical Layer

Language	Operating System	Compiler/SDK
C/C++	Linux, macOS	GCC 4.9+, Clang 3.4+
C/C++	Windows 7+	Visual Studio 2015+
C#	Linux, macOS	.NET Core, Mono 4+
C#	Windows 7+	.NET Core, .NET 4.5+
Dart	Windows, Linux, macOS	Dart 2.2+
Go	Windows, Linux, macOS	Go 1.13+
Java	Windows, Linux, macOS	JDK 8 recommended (Jelly Bean+ for Android)
Kotlin/JVM	Windows, Linux, macOS	Kotlin 1.3+
Node.js	Windows, Linux, macOS	Node v8+
Objective-C	macOS 10.10+, iOS 9.0+	Xcode 7.2+
PHP	Linux, macOS	PHP 7.0+
Python	Windows, Linux, macOS	Python 3.5+
Ruby	Windows, Linux, macOS	Ruby 2.3+



Chapter 10: Setting Up Our Database Project

Operation	Data Access Method	Per-Operation Execution Time (ms)	Operation Time (Processing Days)
3,456,000,000 Google searches per day	ReadFilteredProductsADNSP	1.078	43.12
3,456,000,000 Google searches per day	ReadFilteredProductsDDN	199.910	7,996.4
65,000,000,000 WhatsApp business messages per day	InsertProductDDNSP	1.841	1,385.01157
65,000,000,000 WhatsApp business Messages per day	InsertProductEFSP	396.509	298,299.595

Chapter 11: Benchmarking Relational Data Access Frameworks

Developer PowerShell

+ Developer PowerShell - [Icon] [Icon]

```
// * Summary *
```

BenchmarkDotNet=v0.12.1, OS=Windows 10.0.19042
 Intel Core i5-6300U CPU 2.40GHz (Skylake), 1 CPU, 4 logical and 2 physical cores
 .NET Core SDK=6.0.100-preview.3.21202.5
 [Host] : .NET Core 6.0.0 (CoreCLR 6.0.21.20104, CoreFX 6.0.21.20104), X64 RyuJIT
 DefaultJob : .NET Core 6.0.0 (CoreCLR 6.0.21.20104, CoreFX 6.0.21.20104), X64 RyuJIT

Method	Mean	Error	StdDev	Median	Rank	Gen 0	Gen 1	Gen 2	Allocated
InsertProductADN	2.092 ms	0.0418 ms	0.1150 ms	2.102 ms	9	37.1094	15.6250	-	71.31 KB
InsertProductADNSP	1.894 ms	0.0470 ms	0.1350 ms	1.866 ms	8	23.4375	5.8594	-	72.53 KB
ReadScalarProductADN	1.407 ms	0.0370 ms	0.1060 ms	1.399 ms	4	37.1094	13.6719	-	70.03 KB
ReadScalarProductADNSP	1.433 ms	0.0390 ms	0.1107 ms	1.416 ms	4	27.3438	7.8125	-	70.58 KB
ReadFilteredProductADN	1.084 ms	0.0400 ms	0.1153 ms	1.068 ms	1	35.1563	13.6719	-	68.04 KB
ReadFilteredProductADNSP	1.078 ms	0.0431 ms	0.1264 ms	1.067 ms	1	35.1563	9.7656	-	68.15 KB
UpdateProductADN	3.583 ms	0.2150 ms	0.6306 ms	3.475 ms	13	-	-	-	70.01 KB
UpdateProductADNSP	1.562 ms	0.0496 ms	0.1367 ms	1.541 ms	5	23.4375	5.8594	-	69.96 KB
DeleteProductADN	6.263 ms	0.0640 ms	2.3943 ms	5.775 ms	14	-	-	-	70.01 KB
DeleteProductADNSP	1.760 ms	0.1261 ms	0.3516 ms	1.588 ms	6	-	-	-	70.88 KB
InsertProductEF	2.516 ms	0.1014 ms	0.2942 ms	2.840 ms	12	62.5000	15.6250	-	124.29 KB
InsertProductEFSP	2.251 ms	0.0540 ms	0.1592 ms	2.261 ms	10	54.6875	15.6250	-	104.65 KB
ReadScalarProductEF	396.509 ms	23.6081 ms	68.8658 ms	360.339 ms	17	-	-	-	109.63 KB
ReadScalarProductEFSP	53.235 ms	1.6418 ms	4.6037 ms	51.883 ms	15	-	-	-	113.3 KB
ReadFilteredProductsEF	1.305 ms	0.0566 ms	0.1596 ms	1.294 ms	3	37.1094	9.7656	-	99.48 KB
ReadFilteredProductsEFSP	1.187 ms	0.0457 ms	0.1319 ms	1.171 ms	2	37.1094	9.7656	-	97.29 KB
UpdateProductEF	5,304.279 ms	51.0259 ms	45.2331 ms	5,309.797 ms	18	110000.0000	23000.0000	4000.0000	424807.66 KB
UpdateProductEFSP	1.964 ms	0.0952 ms	0.2605 ms	1.859 ms	8	-	-	-	104.26 KB
DeleteProductEF	386.716 ms	7.4865 ms	8.0105 ms	384.664 ms	17	-	-	-	129.72 KB
DeleteProductEFSP	2.012 ms	0.1439 ms	0.4174 ms	1.804 ms	8	-	-	-	103.23 KB
InsertProductDDN	2.058 ms	0.0466 ms	0.1374 ms	2.076 ms	9	37.1094	11.7188	-	71.42 KB
InsertProductDDNSP	1.841 ms	0.0442 ms	0.1289 ms	1.851 ms	7	39.0625	13.6719	-	75.24 KB
ReadScalarProductDDN	1.403 ms	0.0431 ms	0.1251 ms	1.381 ms	4	37.1094	9.7656	-	70.16 KB
ReadScalarProductDDNSP	1.514 ms	0.0528 ms	0.1497 ms	1.489 ms	5	37.1094	13.6719	-	71.84 KB
ReadFilteredProductDDN	199.910 ms	3.9077 ms	9.0305 ms	200.090 ms	16	9666.6667	3333.3333	1333.3333	18113.34 KB
ReadFilteredProductDDNSP	1.529 ms	0.0515 ms	0.1427 ms	1.510 ms	5	23.4375	5.8594	-	71.89 KB
UpdateProductDDN	2.297 ms	0.2973 ms	0.8719 ms	1.903 ms	11	-	-	-	70.13 KB
UpdateProductDDNSP	1.891 ms	0.1623 ms	0.4577 ms	1.714 ms	7	-	-	-	72.95 KB
DeleteProductDDN	2.522 ms	0.4970 ms	1.4497 ms	1.807 ms	11	-	-	-	70.3 KB
DeleteProductDDNSP	1.863 ms	0.1538 ms	0.4362 ms	1.763 ms	7	-	-	-	72.34 KB

Operation	Data Access Method	Per-Operation Execution Time (ms)	Operation Time (Processing Days)
3,456,000,000 Google searches per day	ReadFilteredProductsADNSP	1.078	43.12
3,456,000,000 Google searches per day	ReadFilteredProductsDDN	199.910	7,996.4
65,000,000,000 WhatsApp business messages per day	InsertProductDDNSP	1.841	1,385.01157
65,000,000,000 WhatsApp business Messages per day	InsertProductEFSP	396.509	298,299.595

Chapter 12: Responsive User Interfaces

Responsive WinForms Example

Progress Update: Performing load operation 82 of 100...

Form1

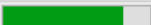
You have clicked the button 10 times.

Increment Count					
	Id	Name	Description	Price	Units
▶	91	Product 91	It is a long esta...	9.99	100
	92	Product 92	It is a long esta...	9.99	100
	93	Product 93	It is a long esta...	9.99	100
	94	Product 94	It is a long esta...	9.99	100
	95	Product 95	It is a long esta...	9.99	100
	96	Product 96	It is a long esta...	9.99	100
	97	Product 97	It is a long esta...	9.99	100
	98	Product 98	It is a long esta...	9.99	100
	99	Product 99	It is a long esta...	9.99	100
	100	Product 100	It is a long esta...	9.99	100

It is a long established fact that a reader will be distracted by the readable content of a page

|<< << Page 10 of 10 >> >>|

Cancel long running operation

Progress: 88% 

Responsive WPF Example

Progress Update: Performing load operation 96 of 100...

MainWindow

Hot Reload

You have clicked the button 102 times.

Increment Counter

Id	Name	Description	Price	Un
81	Product 81	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
82	Product 82	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
83	Product 83	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
84	Product 84	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
85	Product 85	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
86	Product 86	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
87	Product 87	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
88	Product 88	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
89	Product 89	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100
90	Product 90	It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout.	9.99	100

<

<<

Page 9 of 10

>>

>

Progress: 17%

Cancel Task

CH12_ResponsiveASPNET [Home](#) [Privacy](#)

Ajax Demonstration

Ajax Demonstration

© 2021 - CH12_ResponsiveASPNET - [Privacy](#)

CH12_ResponsiveASPNET [Home](#) [Privacy](#)

Ajax Demonstration

Ajax Demonstration

Ajax Demo Result

© 2021 - CH12_ResponsiveASPNET - [Privacy](#)

Click the following button to see the function in action

Display

[open] Connection established

Sending to server

[message] Data received from server: Hello from server, Guest!

[close] Connection closed cleanly, code=1000 reason=Bye!

© 2021 - CH12_ResponsiveASPNET - [Privacy](#)

Add Client-Side Library

Provider:

unpkg

Library:

@microsoft/signalr@latest

☒ Include all library files

☐ Choose specific files:

Files:

dist

browser

signalr.js

signalr.js.map

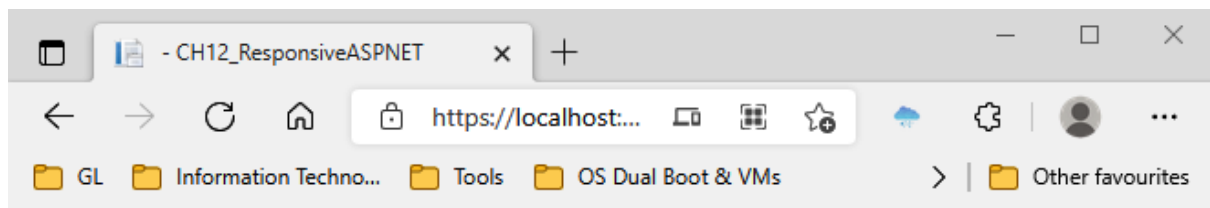
signalr.min.js

Target Location:

wwwroot/lib/microsoft/signalr/

Install

Cancel

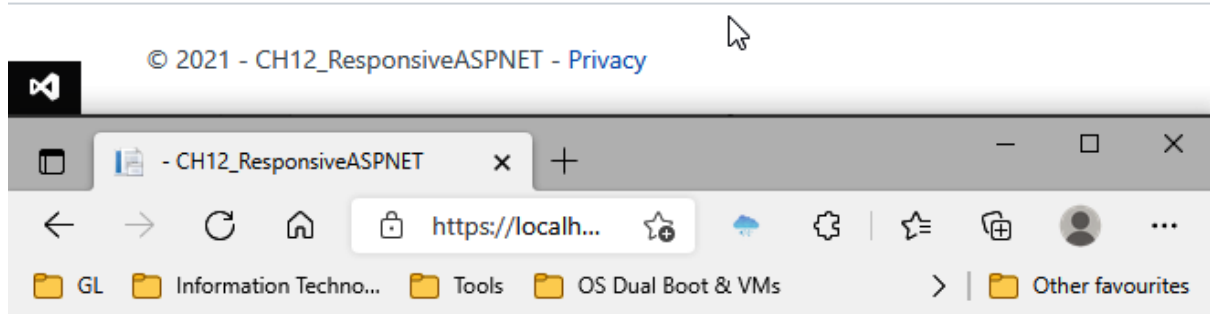


CH12_ResponsiveASPNET [Home](#) [Privacy](#)

User

Message

- User 1 says Hello, User 2!
- User 2 says Hello, User 1!

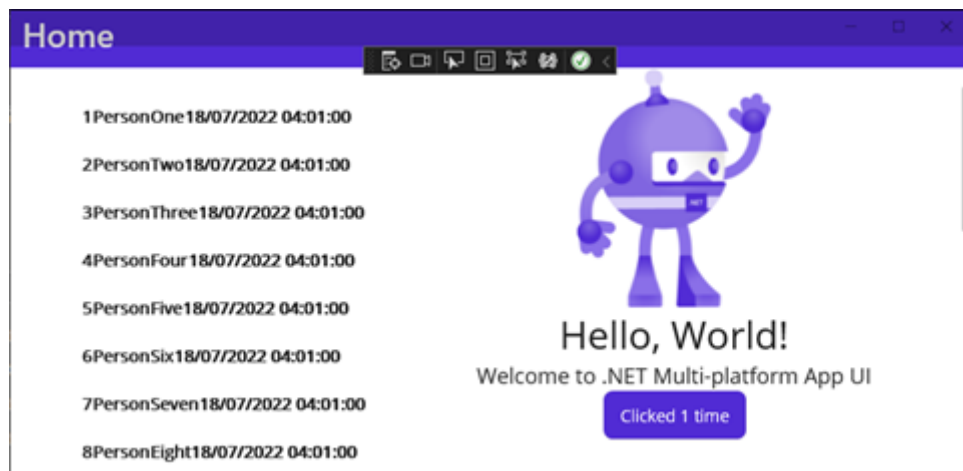
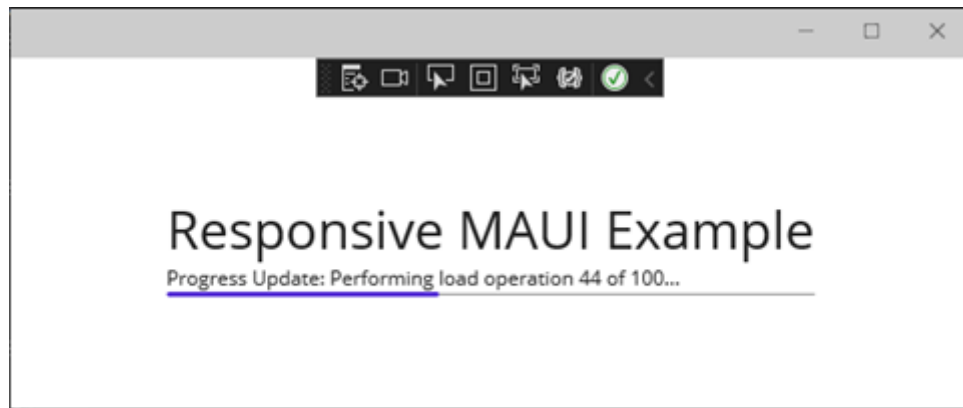


CH12_ResponsiveASPNET [Home](#) [Privacy](#)

User

Message

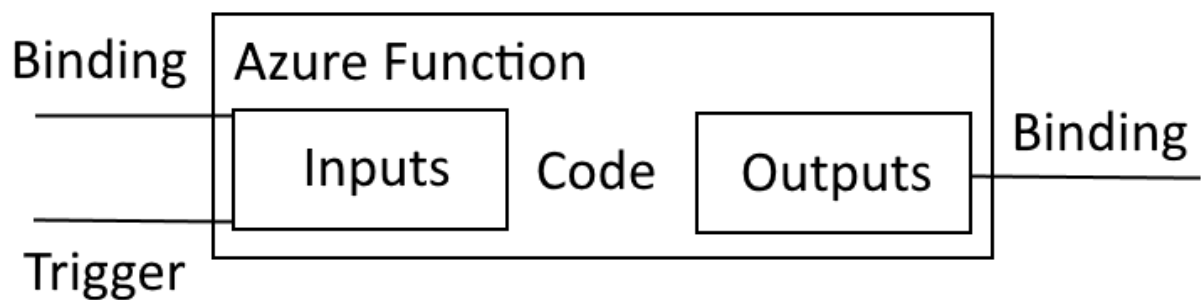
- User 1 says Hello, User 2!
- User 2 says Hello, User 1!



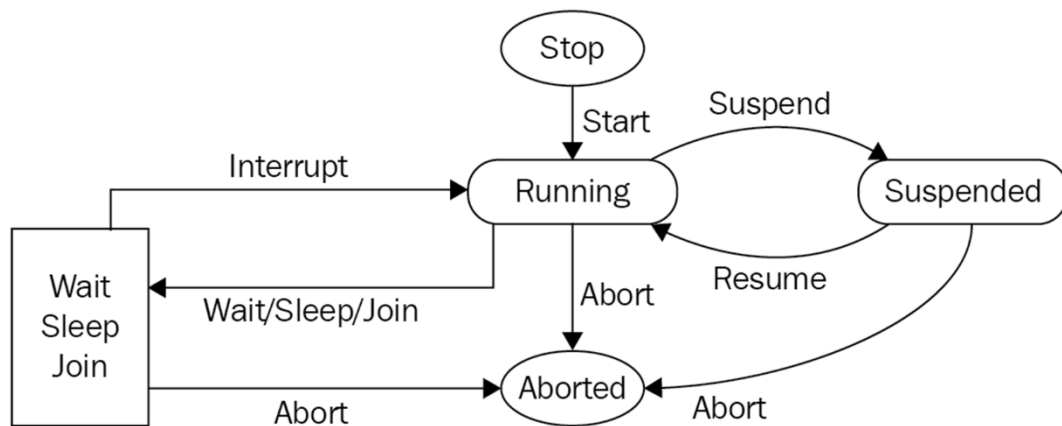
Chapter 13: Distributed Systems

```
Microsoft Visual Studio Debug Console
Dividend paid by Company Name to Customer Name on 02/03/2022 22:43:40 of £23.45.
InvalidDate: Payment cannot be made before the standing order month pay day.

C:\Work\CH13_EventSourcing\CH13_EventSourcing\bin\Debug\net6.0\CH13_EventSourcing.exe
(process 24008) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```



Chapter 14: Multi-Threaded Programming



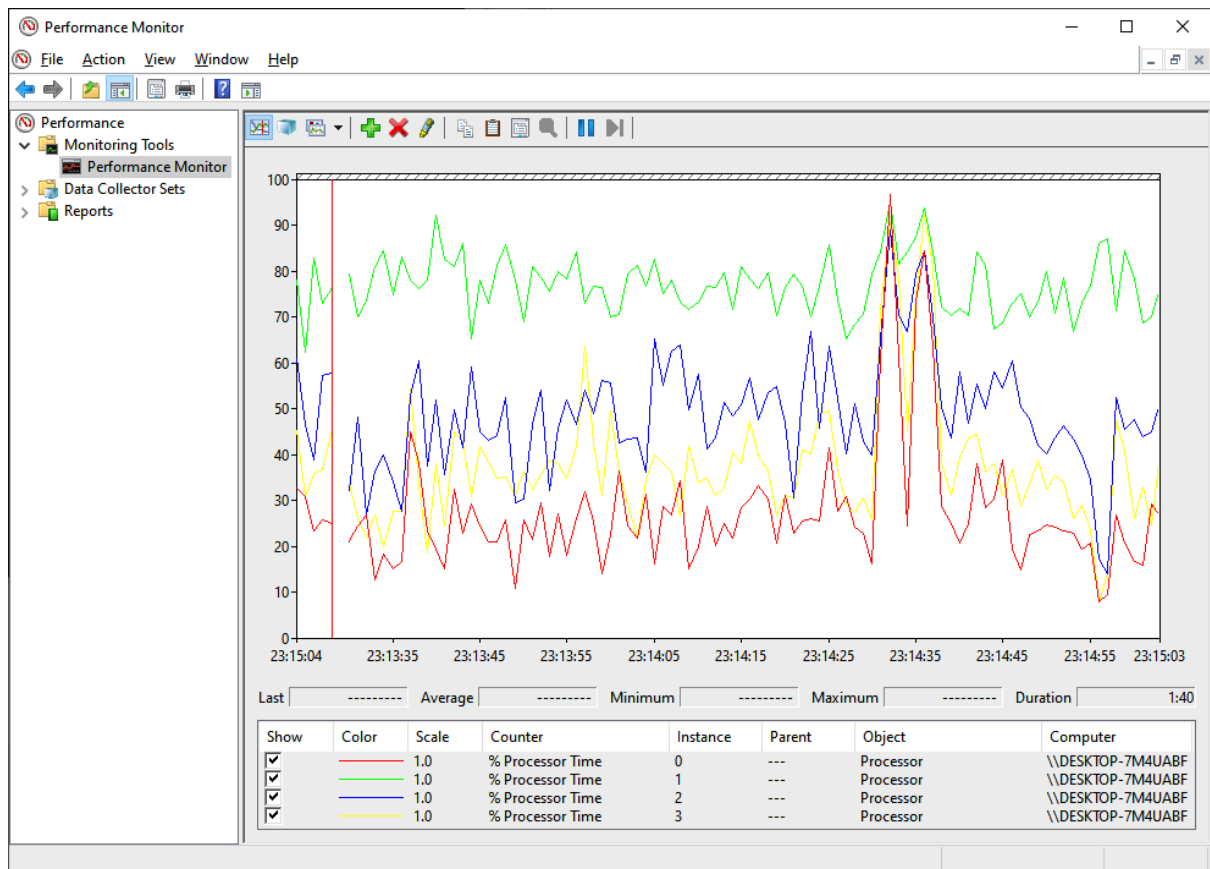
Thread Lifecycle

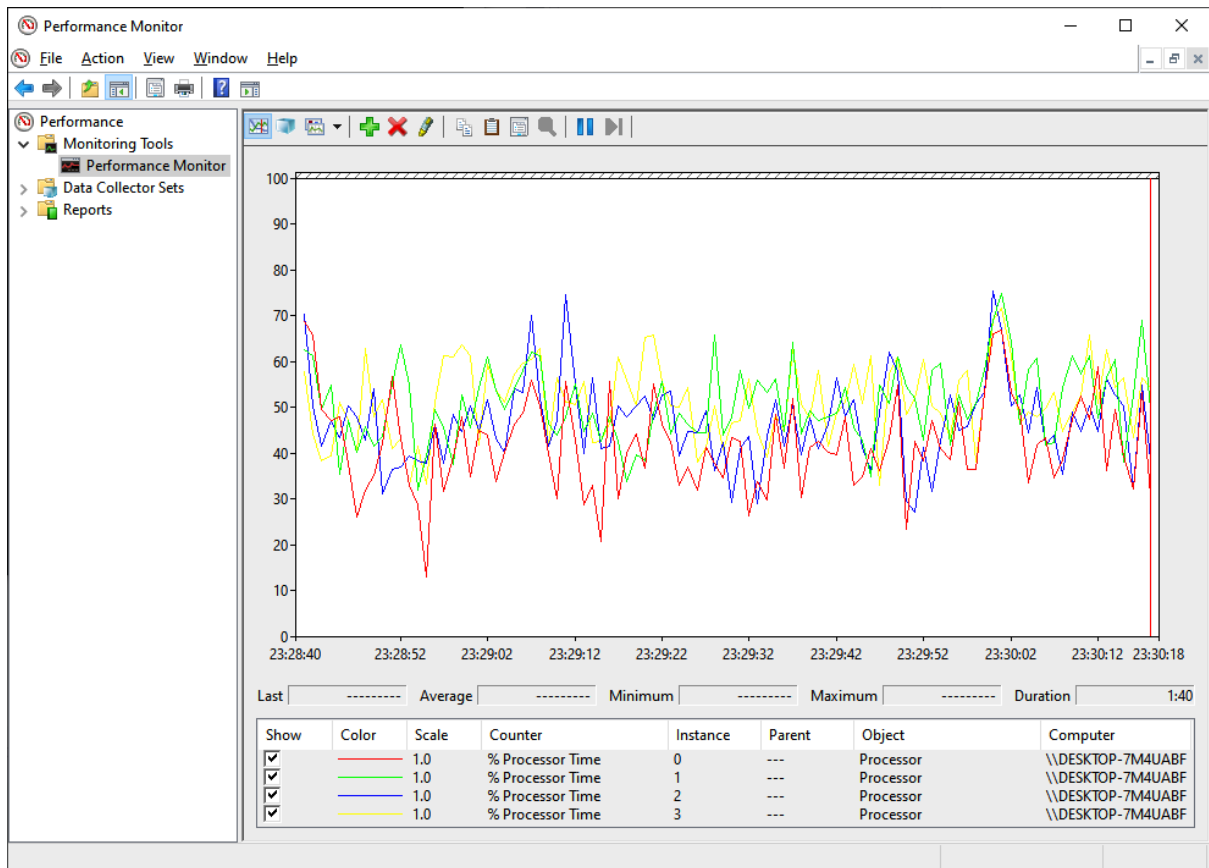
```
Microsoft Visual Studio Debug Console
SleepyMethod() Cancelled

C:\Work\CH14_Multithreading\CH14_Multithreading\bin\De
bug\net6.0\CH14_Multithreading.exe (process 17340) exi
ted with code 0.
To automatically close the console when debugging stop
s, enable Tools->Options->Debugging->Automatically clo
se the console when debugging stops.
Press any key to close this window . . .
```

```
Microsoft Visual Studio Debug Console
Execute() method execute.
Message: Hello, world!
Message: Goodbye, world!
```

Chapter 15: Parallel Programming



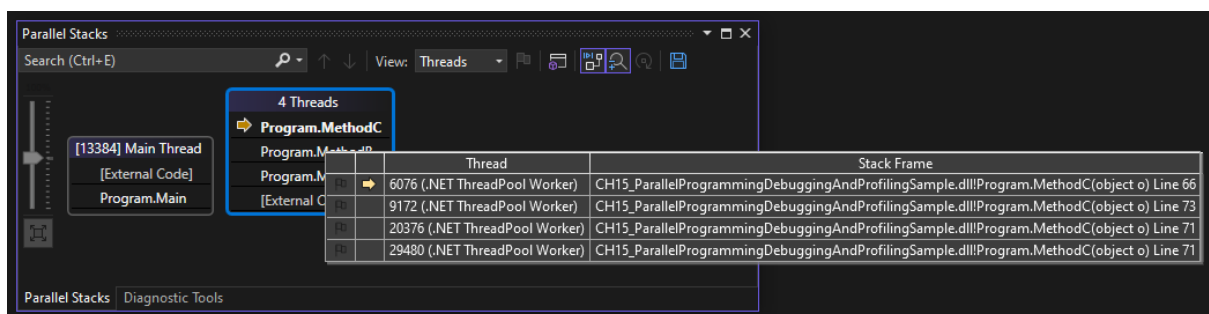
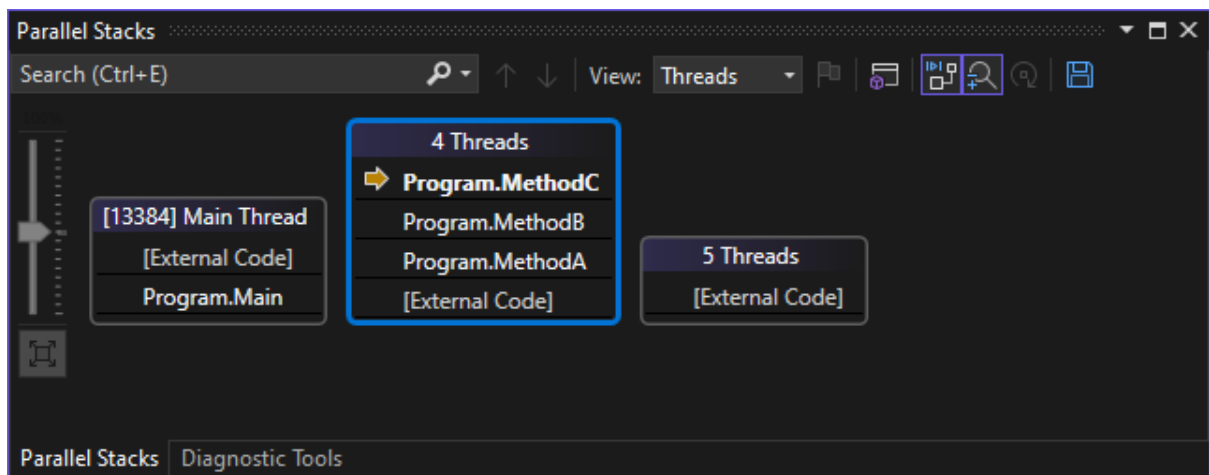


Select Microsoft Visual Studio Debug Console

```
// * Summary *
```

BenchmarkDotNet=v0.13.1, OS=Windows 10.0.19044.1766 (21H2)
11th Gen Intel Core i7-11700 2.50GHz, 1 CPU, 16 logical and 8 physical cores
.NET SDK=6.0.301
[Host] : .NET 6.0.6 (6.0.622.26707), X64 RyuJIT [AttachedDebugger]
DefaultJob : .NET 6.0.6 (6.0.622.26707), X64 RyuJIT

Method	Mean	Error	StdDev	Median
StandardForEachLoopExample	3,218.7 ms	168.00 ms	479.32 ms	3,194.3 ms
ParallelForEachLoopExample	3,512.7 ms	293.63 ms	823.37 ms	3,314.2 ms
DownloadWebsites1	548.4 ms	10.93 ms	27.02 ms	547.1 ms
DownloadWebsites2	228.6 ms	4.87 ms	13.58 ms	224.4 ms
DownloadWebsites3	551.7 ms	10.66 ms	28.64 ms	551.1 ms
DownloadWebsites4	222.9 ms	4.44 ms	11.05 ms	220.0 ms



	Thread	Stack Frame
	6076 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 1) Line 50
	9172 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 2) Line 50
	20376 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 3) Line 50
	29480 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 4) Line 50

Parallel Stacks

Search (Ctrl+E) View: Tasks

4 Async Logical Stacks

- Program.MethodC
- Program.MethodB
- Program.MethodA

1 Async Logical Stack

- Start<StartupHook.
- StartupHook.Initialize.AnonymousMethod_0
- Task.Run

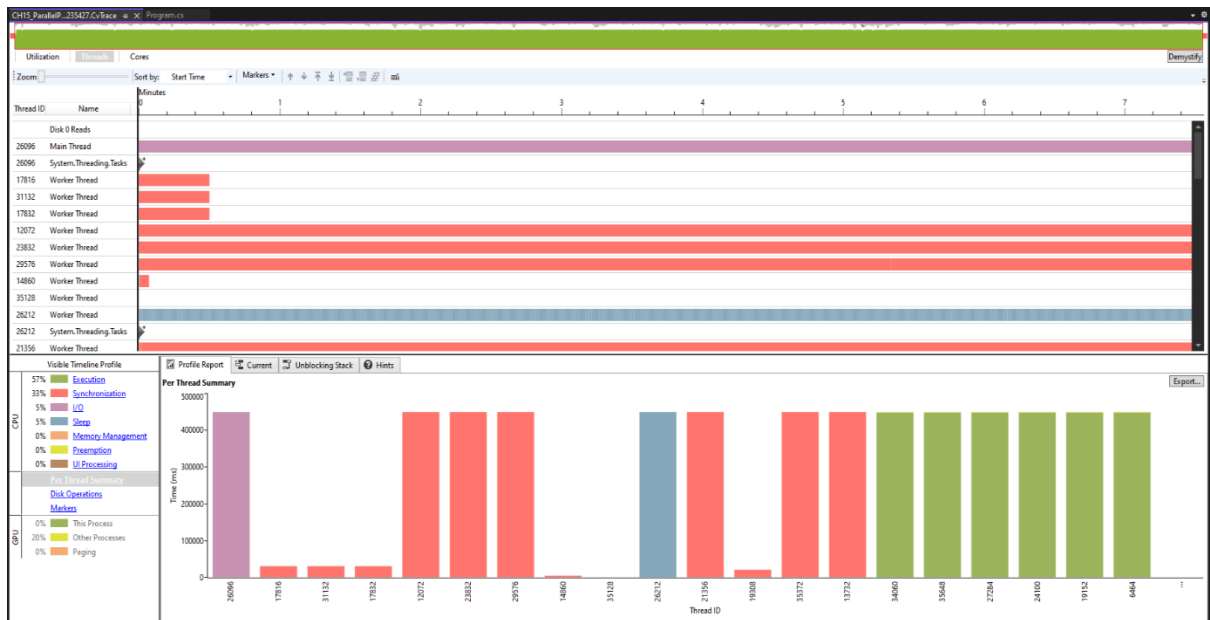
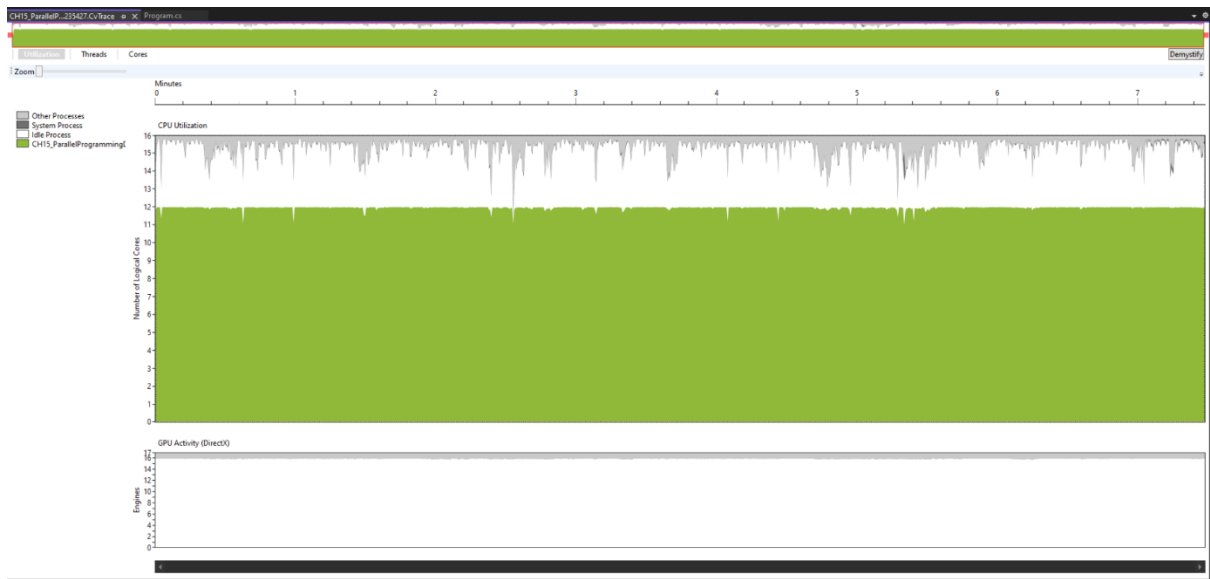
Parallel Stacks Diagnostic Tools

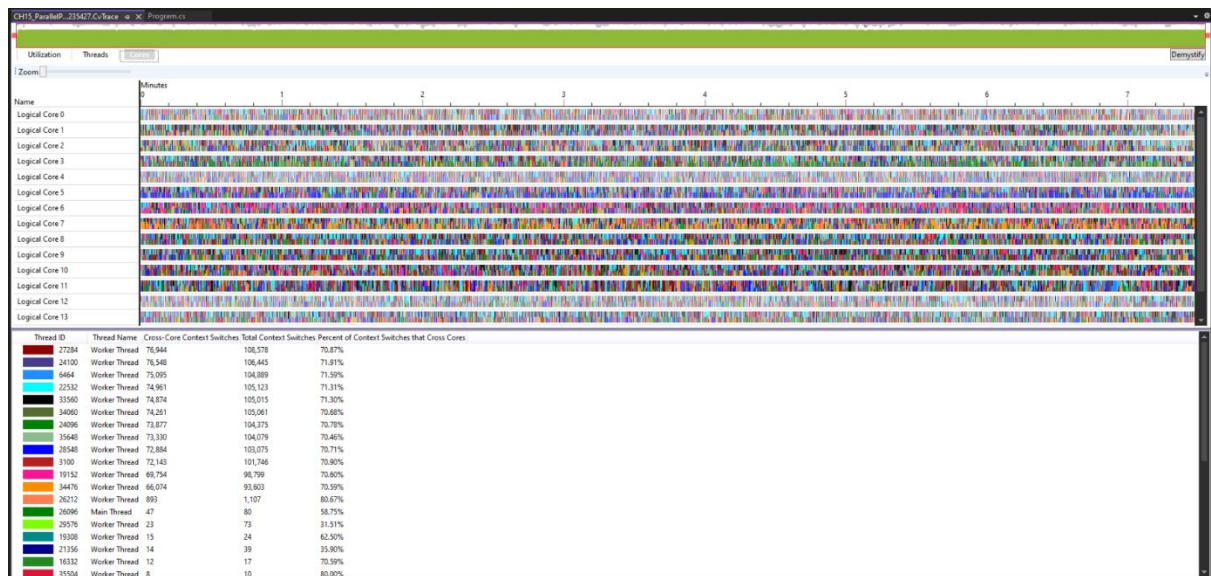
	Thread	Stack Frame
	6076 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 1) Line 50
	9172 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 2) Line 50
	20376 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 3) Line 50
	29480 (.NET ThreadPool Worker)	CH15_ParallelProgrammingDebuggingAndProfilingSample.dll!Program.MethodB(object obj = 4) Line 50

Tasks

	ID	Status	Start Time (sec)	Duration (sec)	Completi...	Location
	7	Awaiting	0.000	1457.285	0.000	System.Threading.Tasks.UnwrapPromise<TResult>.Invoke(completingTask)
	6	Active	0.000	1457.285	0.000	StartupHook.<>c__DisplayClass1_0.<<Initialize>b_0>d.MoveNext
	8	Active	0.000	1457.285	0.000	Program.MethodC
	9	Active	0.000	1457.285	0.000	Program.MethodC
	10	Active	0.000	1457.285	0.000	Program.MethodC
	11	Active	0.000	1457.285	0.000	Program.MethodC

Tasks Containers Immediate Window Output Error List Call Hierarchy





Chapter 16: Asynchronous Programming

```
Developer PowerShell
+ Developer PowerShell
Intel Core i5-3330 CPU 3.00GHz (Ivy Bridge), 1 CPU, 4 logical and 4 physical cores
.NET SDK=6.0.101
[Host] : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT
DefaultJob : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT

| Method | Mean | Error | StdDev |
|-----|-----|-----|-----|
| SynchronousMethod | 7.3220 ns | 0.0714 ns | 0.0668 ns |
| TaskMethod | 112.4494 ns | 1.3471 ns | 1.2601 ns |
| AsynchronousTaskMethod | 0.9982 ns | 0.0138 ns | 0.0122 ns |

// * Hints *
Outliers
  Benchmarks.AsynchronousTaskMethod: Default -> 1 outlier was removed (3.20 ns)

// * Legends *
Mean : Arithmetic mean of all measurements
Error : Half of 99.9% confidence interval
StdDev : Standard deviation of all measurements
1 ns : 1 Nanosecond (0.000000001 sec)

// ***** BenchmarkRunner: End *****
// ** Remained 0 benchmark(s) to run **
Run time: 00:01:19 (79.6 sec), executed benchmarks: 3

Developer PowerShell Output
```



```
Developer PowerShell
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.13.1, OS=Windows 10.0.19043.1415 (21H1/May2021Update)
Intel Core i5-3330 CPU 3.00GHz (Ivy Bridge), 1 CPU, 4 logical and 4 physical cores
.NET SDK=6.0.101
[Host] : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT
DefaultJob : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT

| Method | Mean | Error | StdDev |
|-----|-----|-----|-----|
| GetAwaiterGetResult | 822.3 ns | 16.28 ns | 21.17 ns |
| Result | 1,133.5 ns | 10.06 ns | 9.41 ns |
| Wait | 749.2 ns | 11.70 ns | 9.77 ns |
| GetAwaiter | 118.1 ns | 3.02 ns | 8.91 ns |
```

```
Developer PowerShell
+ Developer PowerShell
// * Summary *

BenchmarkDotNet=v0.13.1, OS=Windows 10.0.19043.1415 (21H1/May2021Update)
Intel Core i5-3330 CPU 3.00GHz (Ivy Bridge), 1 CPU, 4 logical and 4 physical cores
.NET SDK=6.0.101
[Host] : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT
DefaultJob : .NET 6.0.1 (6.0.121.56705), X64 RyuJIT

| Method | Mean | Error | StdDev |
|-----|-----|-----|-----|
| SynchronousAwait | 624.4 ms | 6.42 ms | 5.69 ms |
| AsynchynchronousWhenAll | 311.3 ms | 2.23 ms | 1.87 ms |
```

```
C:\Development\CH16_...
CH13 - Asynchronous Programming
Hello, World!
```