

Læseplan for ”Avanceret C#”

Indhold

Kurset er et avanceret kursus, der har til formål at give deltagerne detaljeret viden om både C# 7.x svarende til en overbygning på en grundlæggende viden om og erfaring med C# og Visual Studio 2017.

Kurset baseres på Andrew Troelsens bog [1]. Vi vil fokusere næsten udelukkende på de avancerede dele af Part II, III og IV af bogen, da disse dele beskriver C#-sproget i detaljer. Vi vil dog også berøre en del af Part V samt de senere kapitler, der omhandler selve .NET 4.6 frameworket, eksempelvis Entity Framework og LINQ to XML.

Vi vil ofte præsentere stoffet en del anderledes og i lidt mere dybde end bogen gør det – forhåbentligt med et større udbytte for kursisterne til følge.

Vi vil i dette kursus (dog med nogle undtagelser) koncentrere os om at introducere C#-sproget ”så rent som muligt” gennem konsol-applikationer. Vi beskæftiger os derfor ikke meget med de forskellige typer af brugerfladekomponenter, som eksempelvis WinForms, WPF eller ASP.NET. Men vi introducerer løbende features, tips og tricks til at benytte værktøjet Visual Studio 2015, når chancerne byder sig.

Tidsplan

Kursus strækker sig over 4 dage efter følgende forventede tidsplan.

Dag 1+2	Indledning Module 1: ”Advanced Types and Methods” Module 2: ”Delegates, Events, and Lambdas” Module 3: ”LINQ”
Dag 3+4	Module 4: ”Operator Overloading” Module 5: ”Object Lifetime” Module 6: ”Dynamic Types” Module 7: ”Asynchronous Programming” Module 8: ”New Features in C# 6.0” Module 9: ”An Introduction to C# 7” Module 10: ”What’s New in C# 7.x?” Afslutning

Kurset indeholder både instruktorgennemgang af stof samt praktiske øvelser, der relaterer sig til stoffet. Fordelingen mellem gennemgang og øvelser er ca. 75:25, dog med variationer fra modul til modul.

Hvis man har lyst, kunne det være en mulighed at have skimmet stoffet inden instruktørgennemgangen, da det så bliver lettere at få et udbytte af gennemgangen. Endvidere vil det være **optimalt**, hvis perioden efter kursusdagene benyttes til at få en smule mere hands-on øvelse med de avancerede dele af C# gennem det stof, som er blevet gennemgået. Den bedste måde at lære avancerede konstruktioner på, er at ”få det ind gennem fingrene”. ☺

Beskrivelse af de enkelte moduler følger nedenfor.

Module 1: “Advanced Types and Methods”

Synopsis

Vi repeterer først de grundlæggende data-typer, og opdelingen mellem værdi- og reference-typer beskrives. Derefter genopfriskes generics og collections, hvorefter vi ser på de såkaldte iteratorer, som collections ofte implementerer vha. en særlig syntaks. Vi lægger så vægt på anonyme typer samt andre variationer over typer generelt.

Herefter repeteres metoder med vægt på de avancerede aspekter af parameteroverførsel inden vi slutter med at introducere extension-metoder.

Læsestof

- [Chapter 4, s. 109 – 119] udgør det grundlæggende stof omkring metoder.
- [Chapter 4, s. 136 – 143] sammenligner værdi- og reference-typer og diskuterer blandt andet, hvordan værdi- og reference-typer hhv. opfører sig som parametre til metoder.
- [Chapter 8, s. 297 – 302] gennemgår iteratorer.
- [Chapter 9, s. 313 – 353] beskriver collections og generics i stor detalje.
- [Chapter 11, s. 418 – 423] introducerer extension-metoder.
- [Chapter 11, s. 423 – 429] omhandler anonyme typer.
- <http://msdn.microsoft.com/en-us/vcsharp/ee957397.aspx> forklarer tupler.

Module 2: “Delegates, Events, and Lambdas”

Synopsis

Indledningsvis undersøges delegates, der i sin simpleste form blot er en reference til en metode. Vi ser hvordan delegates virker, samt hvordan man selv kan implementere sine egne delegate-typer. Herefter beskæftiger vi os med events som en indkapsling af delegates og ser eksempler på brugen af events til løsning af almindelige programmeringsopgaver. Dette leder over til at kigge på syntaksen for anonyme metoder og senere de såkaldte lambda-udtryk, der basalt set blot er en præcis notation for anonyme metoder. Vi gennemgår de indbyggede delegate-typer og bruger lidt tid på de avancerede emner som type-variants for delegates og interfaces.

Læsestof

- [Chapter 10, s. 355 – 397] udgør hoveddelen af stoffet
- <http://blogs.msdn.com/b/csharpfaq/archive/2010/02/16/covariance-and-contravariance-faq.aspx> gennemgår typevariants-principperne

Module 3: “LINQ”

Synopsis

Vi introducerer LINQ, der er en kraftfuld mekanisme til at forespørge og søge på data i forskellig form. Vi koncentrerer os indledende om den del af LINQ, der er ren C# – dvs. uden eksempelvis komponenterne, der

tilgår data fra eksempelvis SQL og XML. Vi præsenterer de praktiske forudsætninger man skal igennem for at kunne benytte LINQ, og gennemgår herefter alle LINQs query keywords samt en del af query-operatorerne. Vi ser disse benyttet på collections og arrays repræsenteret som objekter. Vi bevæger os derefter til at beskrive principperne bag Entity Framework og et par eksempler på, hvordan LINQ to Entities kan benyttes til at tilgå data i databaser. Vi beskriver så, hvordan LINQ-teknikkerne kan benyttes til at processere XML-data gennem LINQ to XML. Til sidst gennemgår vi kort, hvordan flere af disse teknikker er realiseret vha. Expression Trees.

Læsestof

- [Chapter 12, s. 439 – 471] udgør stoffet omkring grundlæggende LINQ.
- <http://msdn.microsoft.com/en-us/library/bb310804.aspx> beskriver LINQ's query keywords.
- [Chapter 23, s. 929 – 999] introducerer Entity Framework.
- [Chapter 24, s. 1001 – 1019] beskriver LINQ to XML.
- <http://msdn.microsoft.com/en-us/library/bb397951.aspx> beskriver Expression Trees.

Module 4: “Operator Overloading”

Synopsis

Vi definerer indexere, der er en særlig type properties, som kan benyttes til at give klasser ”array-lignende” egenskaber, og ser eksempler på at konstruere disse. Herefter vises, hvordan man kan overskrive de velkendte operatoren på egne typer, og eksplicitte og implicitte konverteringer for disse gennemgås.

Læsestof

- [Chapter 12, s. 399 – 417] udgør stoffet.

Module 5: “Object Lifetime”

Synopsis

Vi introducerer IDisposable som et vigtigt interface .NET til deterministisk oprydning af objekter. For at forstå detaljerne, gennemgår vi oprydning af objekter og taler om, hvordan garbage collection og finalizers virker i .NET – herunder C#'s destructors.

Læsestof

- [Chapter 13, s. 473 – 494] beskriver objekters levetid i detaljer. Den resterende del af kapitlet ser vi bort fra.

Module 6: “Dynamic Types”

Synopsis

Vi gennemgår vi de dynamiske typer tilføjet i C# 4.0 via det nye dynamic keyword. Vi beskriver, hvordan brugen af dynamic udsætter type-checket til runtime, hvilket tillader C# at interface til dynamiske sprog. Vi

viser eksempler på, hvordan denne meget moderne konstruktion kan benyttes til på en let måde at gøre visse aspekter utroligt lette og elegante.

Læsestof

- [Chapter 16, s. 599 – 607] udgør det stof, som vi løseligt baserer gennemgangen på.

Module 7: “Asynchronous Programming”

Synopsis

Vi introducerer de forskellige muligheder og teknikker for asynkron programmering i C# gennem Task Parallel Library, der blev introduceret i .NET 4.0 og yderligere udvidet med .NET 4.5 og C# 5.0. Vi har hovedvægten på de såkaldte tasks og beskriver derefter i detaljer, hvordan disse instantieres, kombineres og håndteres. Til sidst berører vi trådsikkerhed gennem trådsikre collections og lock-keywordet.

Læsestof

- [Chapter 19, s. 695 – 747] beskriver multi-trådet programmering generelt.
- <http://msdn.microsoft.com/en-us/library/system.collections.concurrent.aspx> forklarer concurrent collections.

Module 8: “New Features in C# 6.0”

Synopsis

C# 6.0 udkom i sommeren 2015 sammen med Visual Studio 2015. Vi gennemgår de forskellige nye operatoren, keywords og anden syntaks, der giver udviklere gode muligheder for at skrive samme features hurtigere, pænere og med færre linier kode.

Læsestof

- <https://msdn.microsoft.com/en-us/magazine/dn879355.aspx> er en opdateret artikel af Mark Michaelis, der beskriver de nye features i C# 6.0, der er implementeret i den endelige version.

Module 9: “An Introduction to C# 7”

- C# Language Version History
 - <https://github.com/dotnet/csharplang/blob/master/Language-Version-History.md>
- Tuples
 - <https://csharp.christiannagel.com/2016/10/11/tuples/>
 - <https://docs.microsoft.com/en-us/dotnet/csharp/deconstruct>
- Pattern Matching
 - <https://docs.microsoft.com/en-us/dotnet/csharp/pattern-matching>
 - <https://csharp.christiannagel.com/2017/02/15/patternmatching/>
- Out Vars and Ref Returns

- <https://csharp.christiannagel.com/2017/02/01/refreturns/>
- Expression-bodied Members
 - <https://csharp.christiannagel.com/2017/01/25/expressionbodiedmembers/>
- Literals and Separators
 - <https://csharp.christiannagel.com/2016/10/06/literals/>
- Discards
 - <https://docs.microsoft.com/en-us/dotnet/csharp/discards>

Module 10: “What’s New in C# 7.x?”

- C# Language Version History
 - <https://github.com/dotnet/csharplang/blob/master/Language-Version-History.md>
- Span<T>
 - <https://blogs.msdn.microsoft.com/dotnet/2017/11/15/welcome-to-c-7-2-and-span/>
 - <https://channel9.msdn.com/Events/Connect/2017/T125>
- Microsoft Build 2018
 - <https://channel9.msdn.com/Events/Build/2018/BRK2155>

Referencer

- [1] Andrew Troelsen og Philip Japikse: *“C# 6.0 and the .NET 4.6 Framework”* (Seventh Edition). Apress (2016)