



Applikationsudvikling (01) – Intro

Tid	Aktivitet
08.30	Introduktion af mig og jer
08.50	Introduktion til faget, Visual Studio og C# (og live demo)
09.50	Opgaver i C# - ja, I skal kode en del i dag...
10.00	Pause
10.30	Fortsæt med Opgaver i C#.
11.40	Lidt historie om C# og .Net arkitektur generelt
11.50	Opsamling og afslutning

NB: Det er altid ca. tider – vi tager det som det kommer😊

Dagens plan



Vi tager lige en kort præsentationsrunde

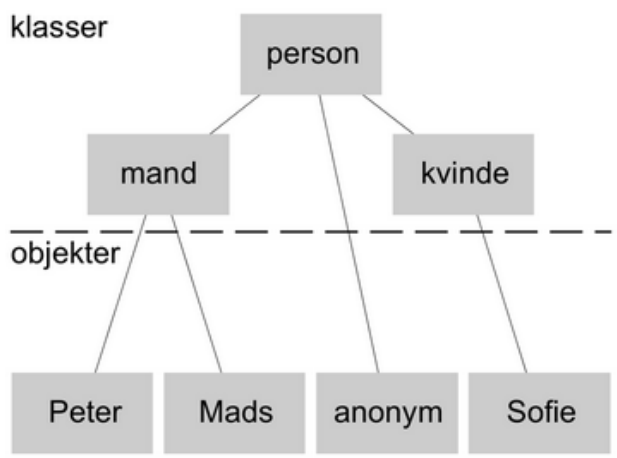
Man siger sit navn, **et** "højdepunkt" fra sommerferien og om man har erfaring med objekt orienteret programmering.

Så f.eks: "Lasse, Roskilde festival, nej ingen erfaring med objekt orienteret programmering".

Dit navn, 1 højdepunkt fra sommerferien

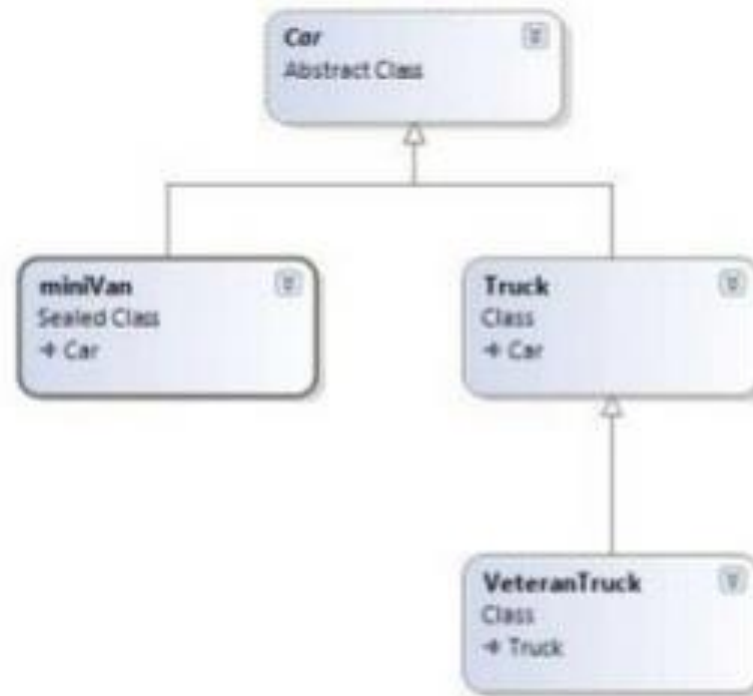


- To vigtigste ting i kurset:
 - Objekt-orienteret programmering
 - En web applikation (web-app) der kan noget (brug af Blazor)



Applikationsudvikling

- UML modellering (også i Kells fag)
- Klasser
- Nedarvning
- Polymorfi
- Objekter
- Interfaces



Objekt-orienteret programmering (senere)

En app der kan noget



Grundlæggende
programmering øves og
udvides:

If, for, Array, metoder
Fejlhåndtering, klasser, algoritmer



Blazor

Citat bogen: "Kig på Blazor hvis du
skal skabe en ny webapplikation"
Nyeste framework til webudvikling
med C#

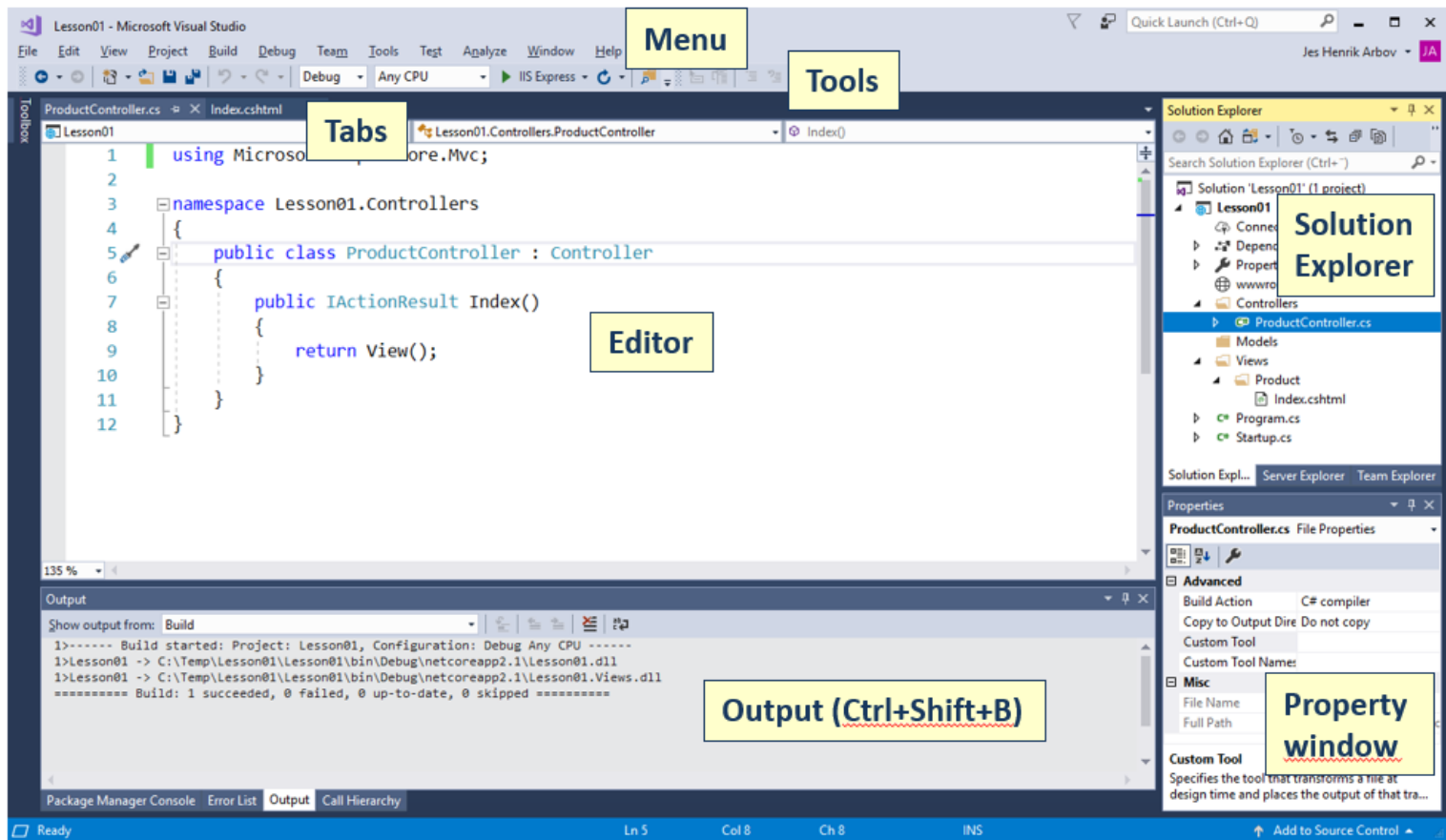


Visual Studio IDE

- IDE står for "Integrated Development Environment" og er et udviklingsværktøj
- En IDE indeholder
 - En editor til programmeringskode med farvefremhævning, code completion ("IntelliSense"), information omkring fejl, linjenumre, mulighed for at expande og collapse kode osv.
 - Værktøjer til automatisk kodegenerering
 - En debugger
- Vi bruger **Visual Studio 2022 Community Edition** (IKKE Visual Studio Code)



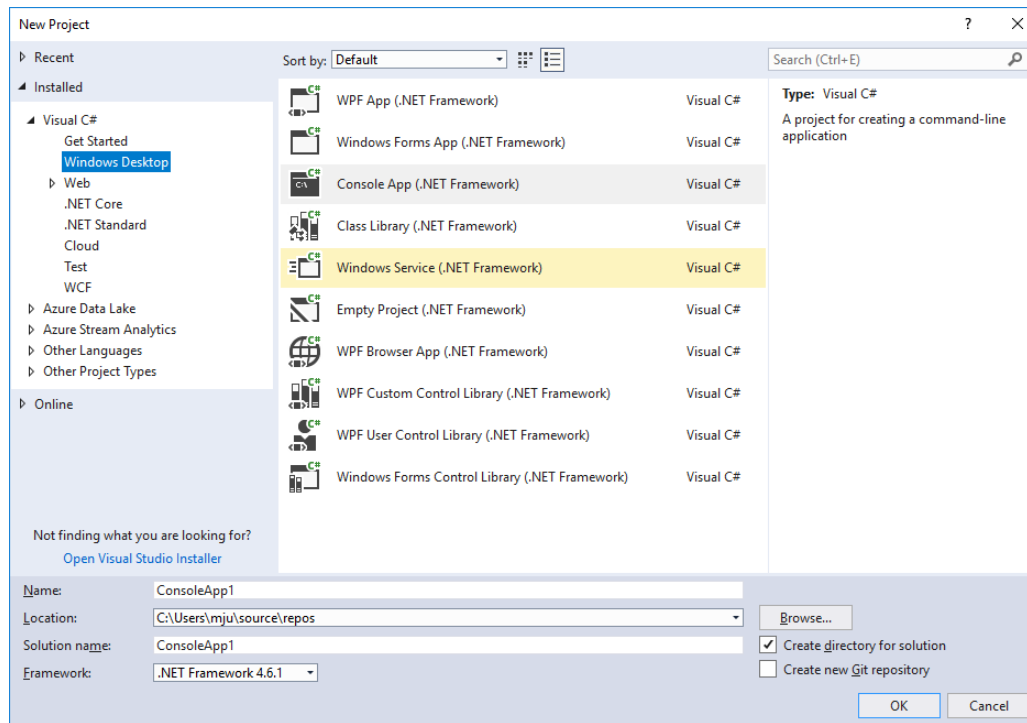
Visual Studio



Visual Studio

Projects: **Console App**, Class Library, WPF App,

...

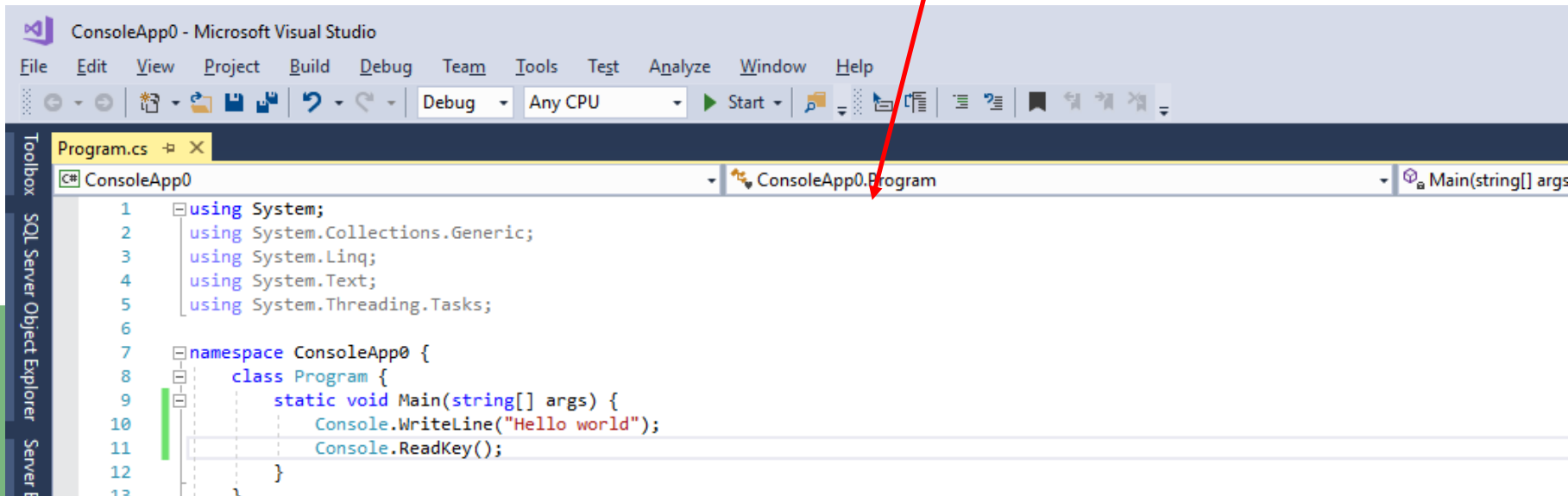


Console App

Vælg console application (hedder bare console app, **ikke noget med "console (.Net framework)"**)

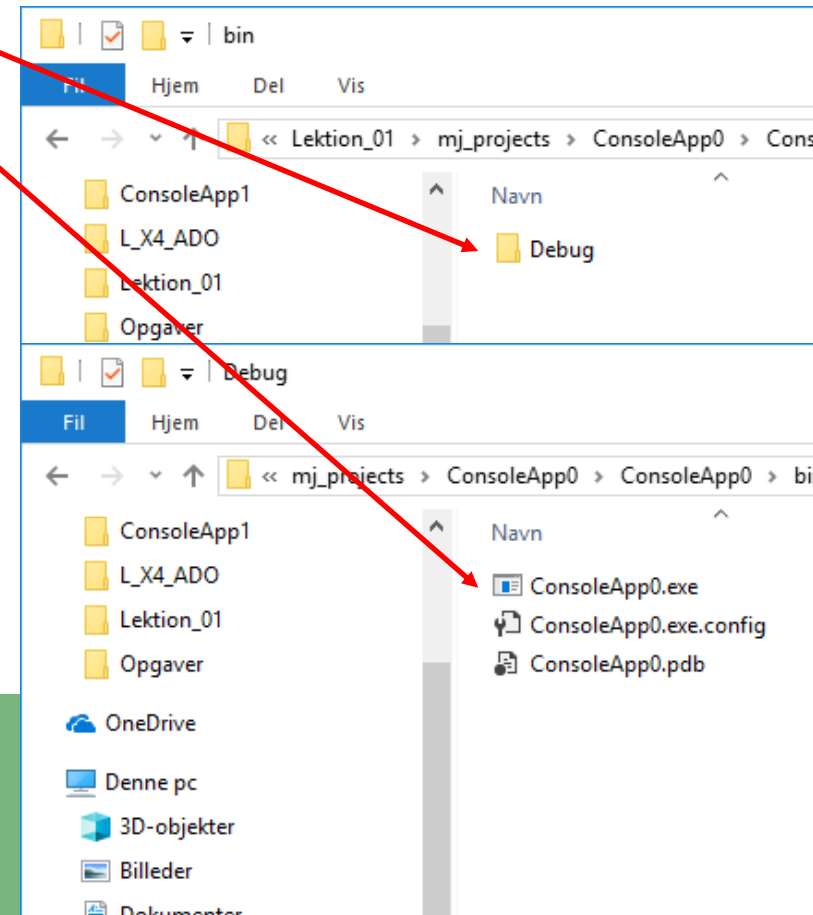
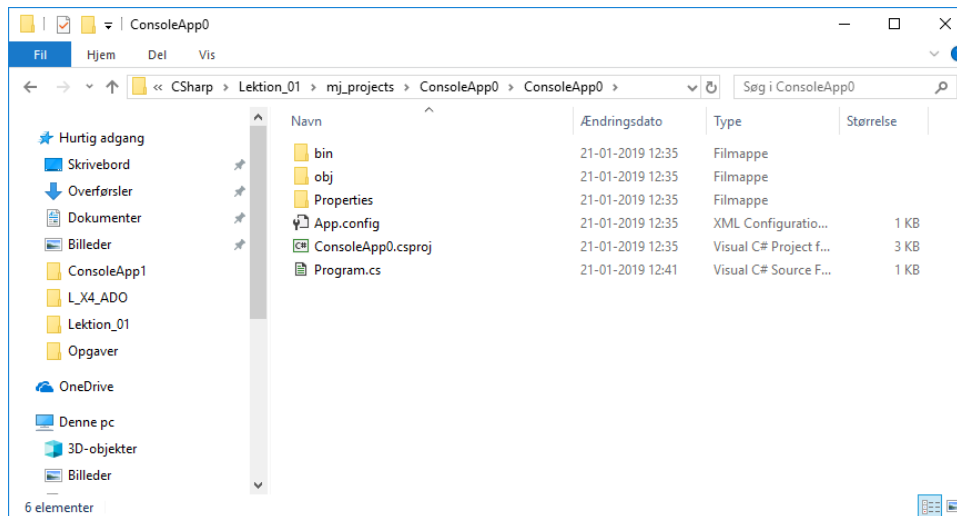
Skriv hello world to console som vist (og Readkey)

Run application ved at trykke Ctrl-F5, or Start



Kør fra command line

- Open Console window (cmd.exe or power-shell)
- cd bin
- cd Debug
- C:\path\path\bin\Debug>ConsoleApp.exe
- Så skulle "Hello world" skrives til console



Valg af .net version – skal defineres når man laver et nyt projekt. Vælg .Net 6.0

Additional information

Console App C# Linux macOS Windows Console

Framework ⓘ

.NET 6.0 (Long-term support)	▼
.NET Core 2.1 (Out of support)	
.NET Core 3.1 (Long-term support)	
.NET 5.0 (Out of support)	
.NET 6.0 (Long-term support)	

Bemærk: Lidt forskel på .net 5 og .net 6 (bogen viser .net 5)

.net 5:

```
namespace Lesson01
{
    class Program
    {
        static void Main(string[] args)
        {
            //her starter kode
            Console.WriteLine("Hello World!");
        }
    }
}
```

.net 6

```
Console.WriteLine("Hello  
World!");
```



Start af program fra VS og brug af debugger (=fejlfinder)

I VS kan du starte programmet med debuggeren med F5 eller ved at klikke på den grønne startknap.

Indsætte Breakpoints ved at bruge F9, eller fra Debug menu, or eller højre klik på en linje og indsætte et breakpoint.

Du kan også starte et program UDEN debuggeren tændt – brug ctrl-F5 i stedet for blot F5. **(vi kommer tilbage til debugging senere også)**



Basic C# syntaks

- Syntaksen er ligesom i C. Sproget "ligner" JavaScript og PHP en del
- Case sensitiv (forskel på stor/lille bogstav)
- Alle kommandoer slutter med ;
- Kode blokke i {}
- Betingelser i ()
- Loops (for, while, foreach)
- Operatorer (=, ==, !=, <, <=, >=, >, &&, ||)

I C# skal alle variabler være **deklareret** med en **type**

```

Type      Navn
↓         ↓
string name;           // declaration, erklæring
name = "Brian Wilson"; // assignment, tildeling
           ↑
        Value eller værdi

// Initialization: combined declaration and assignment
string name = "Brian Wilson";
  ↑         ↑           _____ ↑
  Type      Name       Initializer

name = 200; // Hvad er problemet her ...?
//
```

Visual Studio Tip: Ctrl-K efterfulgt af Ctrl-D vil autoformatere din kode.

Opgaver



Pause



Lidt historie om C#

- Et “nyt” (fra 2001) programmeringssprog designet specifikt til .NET

- 100% Objekt Orienteret Sprog

- Inspired fra Java og C++

- Der kommer hele tiden nye versioner af C#

- Anders Hejlsberg – er en af hoveddesignerne af C# (dansker!)

(som arbejder for Microsoft)

Men C# er et åbent sprog administreret af en ”forening”



Lidt historie om C#

Inspireret af Java fra 1995, så lanceres C# 2001 som en ”forbedring”.
Understøtter Object Oriented Programming : OOP (det snakker vi meget mere om senere)

Det skal være et komplet udviklingsmiljø, som kan bruges til flere platforme – f.eks :

- Desktop applikationer

- Web applikationer (på flere måder faktisk)

- Mobil applikationer (xamarin f.eks.)

- Og andre...

Vi vil bruge det til webapplikationer i dette kursus (og lidt konsol applikationer i starten som i dag).



.Net generelt

C# er et muligt valg, til at programmere .NET

Syntax meget tæt på Java

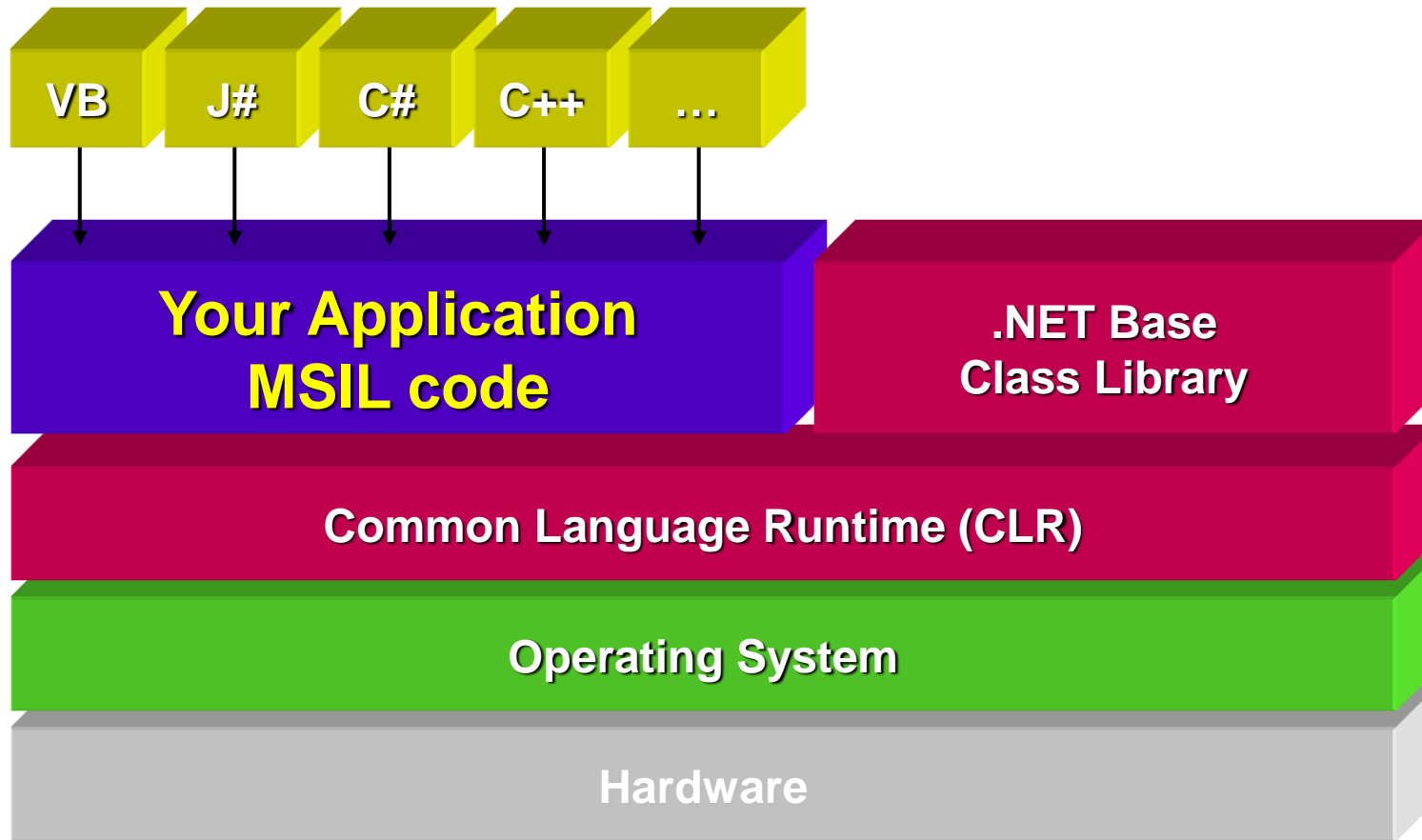
Rimelig "simpelt" at komme i gang når man kender Java, selvom selvfølgelig der også er nye ting.

Andre valg er f.eks. J#, F#, VB, MS-C++

Vi vil kun bruge C# i dette kursus (og lidt css/html og lidt javascript, men de ting kender I jo)



.Net arkitektur (multiplatform)



- Lært at bruge Visual Studio til grundlæggende konsoludvikling
- Forståelse for hvordan man skriver noget til konsollen (output)
- Forståelse for kommunikation mellem konsol og bruger (input)
- Kunne oprette variabler og bruge dem i C#

Opsummering af i dag



- Fortsætte med at programmere i C# (**lav dagens øvelser færdig inden næste gang!!**)
- Mere flow i programmerne med if og for
- Medbring papir og blyant/kuglepen – der vil muligvis komme nogle opgaver, hvor man skal forstå noget ved at ”tegne” sig frem til det.
- Litteratur
 - Cronberg s. 60-119 (men meget er ligesom i javascript, så meget burde være genkendeligt – dog lidt anderledes syntax)
 - Simple variabler
 - Tekster (strings)
 - Konstanter
 - Program flow (især loops / løkker)

Hvad vi skal lave næste gang
