# C# Fundamentals for Absolute beginners

Mine egne kommentarer til kurset.

Link: <https://mva.microsoft.com/en-US/training-courses/c-fundamentals-for-absolute-beginners-16169?l=p90QdGQIC_7106218949>

Host: Bob Tabor

### 03. Creating your first C# program

* HelloWorld program
* End all commands using semicolon
* Mark strings “
* C# is case sensitive

### 04. Understanding you first C# program

* How to write code
  + Learn syntax
  + Learn what is already made, prebuild available functionality; .NET framework.
* Class Library
* Common Language Runtime [CLR]
  + Developer may focus on what the application should do, The CLR takes care of the what’s going on behind the scene.
  + Hardware, memory, …
  + Protection for the user. The app is running inside a “bubble”
* Linefeed, whitespace don’t bother in C#. ONLY “;” will terminate a command

### 05. Working with Code Files, Projects and Solutions

* Code files are organized into projects
* Project are compiled into single assembly
* A solution may contain of several assemblies

### 06. Understanding Datatypes and Variables

* Use camelCasing for increased readability
* Refactoring by [ctrl] + [.]
* C# is CaseSensitive!
* Intellisense may automatically pop up
  + Manual: [ctrl] + [space]
  + may autocomplete using marked alternative when typing [=] or [.] or [enter]
* Declare variables as you need then
  + As opposed to declare in separate line, and opposed to declaring in the top of code
  + Opt. give them initial values

### 07. the IF decision statement

* [==] means to evaluate if something is the same
* If, if else, else
* A variable is only available inside current scope {}
* One-line-code-block does not need curly braces.
* Var += something; is short for: var = var + something;
* If(var == value) ? result-if-yes; result-if-false;
* Writeline(“You won a {0}.”, var); //replace the {0} with the value of var.
* Console.WriteLine("You entered: {0}, therefor you won a {1}.", userValue, message); //works with several replacements like this

### 08. Operators, Expressions and Statements

* Basic building blocks:
  + Statements. Complete thoughts, like a sentence.
  + Statements are made of expressions
  + Expressions are made of operators and operands
  + Operands are the nouns. The things, the objects [classnames, ..]
    - Developer give these names
  + Operators are the verbs. Performs action [=, +=]
    - Developer need to learn these
* See project: “OperatorsExpressionsStatements” for code examples/ documentations

### 09. Iteration Statement