

If statements	Voorbeeld
if	<pre>if (test==true) { // }</pre>
If else	<pre>if (test==true) { // } else { // }</pre>
If else if	<pre>if (getal <3) { // } else if (getal <6) { / }</pre>
If else if else	<pre>if (getal <3) { // } else if (getal <6) { // } else { // }</pre>

Shorthand operator	Equivalent
getal++;	getal = getal + 1;
getal--;	getal = getal - 1;
getal += 7;	getal = getal + 7;
getal -= 7;	getal = getal - 7;
getal *= 7;	getal = getal * 7;
getal /= 7;	getal = getal / 7;

Switch voorbeeld	If equivalent
<pre>switch (getal) { case 0: Console.WriteLine("getal is 0"); break; case 3: Console.WriteLine("getal is 3"); Console.WriteLine("dus niet 0 of 10"); break; case 10: Console.WriteLine("getal is 10"); break; default: Console.WriteLine("getal is niet 0,3 of 10"); break; }</pre>	<pre>if (getal == 0) Console.WriteLine("getal is 0"); else if (getal == 3) { Console.WriteLine("getal is 3"); Console.WriteLine("dus niet 0 of 10"); } else if (getal == 10) Console.WriteLine("getal is 10"); else Console.WriteLine("getal is niet 0,3 of 10");</pre>

Comparison operator examples
<pre>if(getala > getalb) Console.WriteLine("getala is groter");</pre>
<pre>if(getala < getalb) Console.WriteLine("getala is kleiner");</pre>
<pre>if(getala == getalb) Console.WriteLine("getala is gelijk aan getalb");</pre>
<pre>if(getala != getalb) Console.WriteLine("getala is niet gelijk aan getalb");</pre>
<pre>if(!(getala==getalb)) Console.WriteLine("getala is niet gelijk aan getalb");</pre>
<pre>if(getala ==4 && getalb <5) Console.WriteLine("getala = 4 en getalb is kleiner dan 5");</pre>
<pre>if(getala ==4 getalb <5) Console.WriteLine("getala = 4 of getalb is kleiner dan 5");</pre>
<pre>if((getala ==4 getalb <5) && getalc!=10) Console.WriteLine("getala = 4 of getalb is kleiner dan 5, EN getalc is niet 10");</pre>

For syntax	While equivalent
<pre>int teller = 0; while (teller < 5) { Console.WriteLine(teller); teller++; }</pre>	<pre>for (int teller = 0; teller < 5; teller++) { Console.WriteLine(teller); }</pre>

Variabelen declareren en waarde geven (assignment)
<pre>int myage = 30;</pre>
<pre>bool truth = true;</pre>
<pre>double pi = 3.14;</pre>
<pre>decimal kommagetal = 1.89m;</pre>
<pre>float floatGetal = 1.98f;</pre>
<pre>char karakter = 'a'; char eenspatie = ' '; char eenEnter = '\n';</pre>
<pre>string zin = "hallo ik ben tim"; string eenKorteZin = "a"; string eenZinMetEnterenTab = "Hallo\nIk ben\tTim";</pre>

Loop	Output
<pre>int teller = 0; while (teller < 5) { Console.WriteLine(teller); teller++; }</pre>	0 1 2 3 4
<pre>int teller = 0; do { Console.WriteLine(teller); teller++; } while (teller < 5);</pre>	0 1 2 3 4
<pre>int teller1 = 0; int teller2 = 0; while (teller1 < 2) { teller1++; Console.WriteLine("teller1="+ teller1); teller2 = 0; while (teller2<3) { teller2++; Console.WriteLine(teller2); } }</pre>	teller1 = 1 1 2 3 teller2 = 2 1 2 3

Methode	Voorbeeld van gebruik
<pre>static void ToonNaam() { Console.WriteLine("Tim Dams"); }</pre>	<pre>ToonNaam();</pre>
<pre>static void ToonNaam(string name) { Console.WriteLine(name); }</pre>	<pre>ToonNaam("Tim Dams");</pre>
<pre>static void ToonNaamEnKlas(string name, string klas) { Console.WriteLine(name+ " "+ klas); }</pre>	<pre>ToonNaamEnKlas("Tim Dams", "2EA");</pre>
<pre>static void ToonGetal(int getal) { Console.WriteLine(getal); }</pre>	<pre>ToonGetal(6);</pre>
<pre>static int VerdubbelGetal(int getal) { int dubbel = getal*2; return dubbel; }</pre>	<pre>int resultaat = VerdubbelGetal(6);</pre>
<pre>static double Gemiddelde(int getal1, int getal2) { double gemiddelde = (getal1 + getal2)/2.0; return gemiddelde; }</pre>	<pre>double resultaat = Gemiddelde(6,9);</pre>
<pre>static bool IsGelijk(int getal1, int getal2) { if (getal1 == getal2) return true; else return false; }</pre>	<pre>if (IsGelijk(6, 8)) { //...</pre>
<pre>static bool CooleNaam(string naam) { bool result; if (naam == "Tim") result = true; else result = false; return result; }</pre>	<pre>bool ikbencool = CooleNaam("Jos");</pre>

Arrays van objecten	Voorbeeld
Declareren:	<pre>Student[] studenten = new Student[100]; //Zal NullReferenceException geven studenten[5].Leeftijd = 10;</pre>
Volledig instantiëren:	<pre>Student[] studenten = new Student[100]; for (int i = 0; i < studenten.Length; i++) { studenten[i] = new Student(); }</pre>
Enkel arrayelement aanpassen/uitlezen:	<pre>//Aanpassen studenten[6].Leeftijd = 5; //Uitlezen int leeftijd = studenten[8].Leeftijd;</pre>

C# cheatsheet

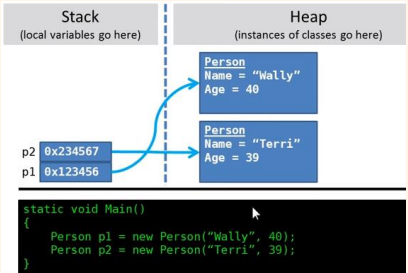
Klassen en objecten	Voorbeeld
Klasse definiëren:	<pre>BankRekening eenRekening = new BankRekening(); class BankRekening { }</pre>
Private variabelen en methoden:	<pre>Kan enkel binnen de klasse benaderd worden class BankRekening { private int saldo; }</pre>
Default constructor:	<pre>BankRekening eenRekening = new BankRekening(); class BankRekening { public BankRekening() { saldo = 0; } private int saldo; }</pre>
Overloaded constructor:	<pre>BankRekening eenRekening = new BankRekening(2000); class BankRekening { public BankRekening(int inSal) { saldo = inSal; } private int saldo; }</pre>
Publieke methode zonder output:	<pre>eenRekening.SaldoVerdubbelen(); class BankRekening { private int saldo; public void SaldoVerdubbelen() { saldo *= 2; } }</pre>
Publieke methode met output:	<pre>int dubbel = eenRekening.GeefDubbeleSaldo(); class BankRekening { private int saldo; public int GeefDubbeleSaldo() { return saldo * 2; } }</pre>

Arrays	Voorbeeld
Aanmaken 1D:	<pre>int[] getallen = { 2, 5, 7, 8 }; float[] kommagetallen = new float[10];</pre>
Aanmaken 2D:	<pre>int[,] getallen2d = { { 2, 5, 7, 8 }, { 6, 7, 4, 3 } }; float[,] kommagetallen2d = new float[10,20];</pre>
Lengte 1D:	<pre>getallen.Length</pre>
Lengte 2D:	<pre>kommagetallen.GetLength(0); //Dim 1 lengte kommagetallen.GetLength(1); //Dim 2 lengte</pre>

Properties	Voorbeeld
Autoproperty (prop [tab][tab]):	<pre>Student eenStudent = new Student(); //Instellen eenStudent.Leeftijd = 20; //Uitlezen int leeftijd = eenStudent.Leeftijd;</pre>
Full property (propfull [tab][tab]):	<pre>Student eenStudent = new Student(); //Instellen eenStudent.Leeftijd = 20; //Uitlezen int leeftijd = eenStudent.Leeftijd;</pre>
Full property met controle:	<pre>//Instellen, zal niet aangepast worden eenStudent.Leeftijd = -10;</pre>
Read-only property, met private set (kan enkel intern aangepast worden):	<pre>//Instellen zal niet werken eenStudent.Leeftijd = 20;</pre>
Read-only property:	<pre>string email = eenStudent.Email;</pre>

By reference/By value

Objecten steeds by reference doorgegeven.
Valuetypes (int,float,etc)worden standaard by value doorgegeven.
Goede uitleg: <https://www.youtube.com/watch?v=cIOudVDDzIM>



By Tim Dams