

C# Advanced Classes

Koen Bloemen



**DE HOGESCHOOL
MET HET NETWERK**

Elfde-Liniestraat 24, 3500 Hasselt, www.pxl.be





- 3.5 Constructor chaining
- 3.6 Static methods & properties
- 3.7 Partial classes
- 3.8 Access modifiers
- 3.9 Garbage collection
- 3.10 Class designer

Constructor chaining

- Constructors (vorige week)
 - Zonder parameters (lege default constructor)
 - Constructor met parameters (zelf gemaakt)
- Constructor chaining
 - Doorlinken naar andere ctor

```
// == Constructor chaining: constructor met parameters wordt opgeroepen. ==  
public Persoon() : this(null, "Onbekend")  
{ }  
  
// ==== Constructor met parameters ====  
public Persoon(string firstname, string name)  
{  
    Naam = name;  
    Voornaam = firstname;  
}
```


Constructor chaining

```
Public Persoon
{
    private int id;
    private string naam;

    public Persoon() : this(0, "Onbekend")
    {
    }

    // ==== Constructor met parameters ====
    public Persoon(int nr, string persoonsNaam)
    {
        id = nr;
        naam = persoonsNaam;
    }

    public Persoon(int nr) : this(nr, "")
    {
    }

    public Persoon(string naam) : this(0, naam)
    {
    }
}
```

Static methods

- Method die op klassenniveau wordt gebruikt en niet op een object
- Kan NIET gebruik maken van membervariabelen of properties, want die horen bij een object. We kunnen enkel lokale variabelen of static variabelen gebruiken hierin.
- Voorbeelden: `Math.Sqrt()`, `Convert.ToString()`, `Console.WriteLine()`, `String.Format()`
- Zelf een static method maken

```
public class Student
{
    public static Student MaakNieuweStudent()
    {
        return new Student();
    }
}
```

- Gebruiken

```
Student stud = Student.MaakNieuweStudent();
```

Partial class

- Een class opsplitsen over meerdere files
- Gebruik keyword **partial**
- Handig wanneer class te groot wordt

Contact.cs

```
public partial class Contact
{
    public string Naam { get; private set; }
    public string Adres { get; private set; }
    public Contact(string contactNaam, string contactAdres)
    {
        Naam = contactNaam;
        Adres = contactAdres;
    }
}
```

ContactPart2.cs

```
public partial class Contact
{
    // In dit deel kunnen enkel zelfstandige
    // methods staan want properties van
    // deel 1 worden hier niet herkend!
    public static Contact CreateContact(string name, string
        address)
    {
        return new Contact(name, address);
    }
}
```

Access modifiers

De access modifier bepaalt de toegankelijkheid (van properties, methods, ...)

- **public**
 - binnen de assembly en de assembly die ernaar verwijst
- **private**
 - binnen dezelfde class of struct
- **protected**
 - binnen de klasse of afgeleide klasse (zie volgend hoofdstuk over inheritance)
- **internal**
 - enkel binnen de assembly
- **protected internal**
 - binnen dezelfde assembly of binnen een afgeleide klasse

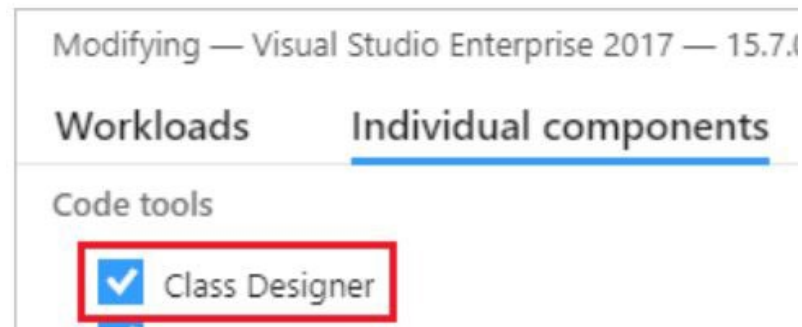
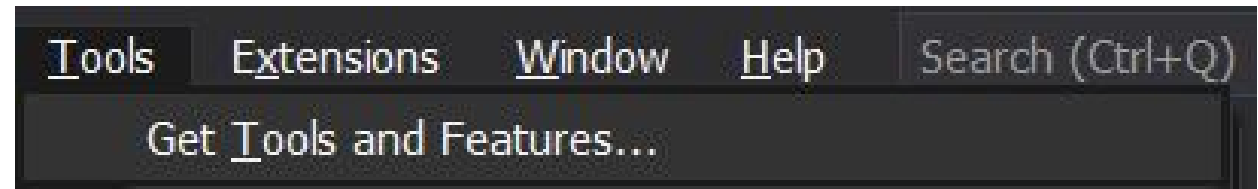
Vernietigen van objecten

- Wanneer programma stopt
- Als je het object gelijk stelt aan null
 - `stud = null;`
 - er is dan geen referentie meer naar het bestaand object
 - garbage collector ruimt object op.
- Wanneer een object uit scope gaat
 - stel dat het object aangemaakt wordt binnen een method
 - method afgelopen => object bestaat niet meer

Class designer

Class Designer installeren

- Open Visual Studio
- Klik bovenaan op Tools. Daarna op “Get Tools and Features...”
- Het kan zijn dat Visual Studio even moet updaten
 - Gewoon laten gebeuren
 - Kan ook zijn dat je moet herstarten
- In de Installer klik op Individual Components, kijk onder Code tools en vink aan Class Designer



Class designer

Class Designer gebruiken

- Nieuwe class maken via Class Designer
 - Open een solution met je project in.
 - In de Solution Explorer: rechterklik op project.
 - Add > New Item...
 - General items > General > Class Diagram
 - Via Toolbox kan je vormen en lijnen naar het diagram slepen
- Bestaande class bewerken via Class Designer
 - In de Solution Explorer: rechterklik op .cs file
 - Klik op View Class Diagram
- Volledige project tonen in Class Designer
 - In de Solution Explorer: rechterklik op project
 - Klik op View > View Class Diagram

