



UPPSALA  
UNIVERSITET

# . NET-programmering

Namespace

Assembly



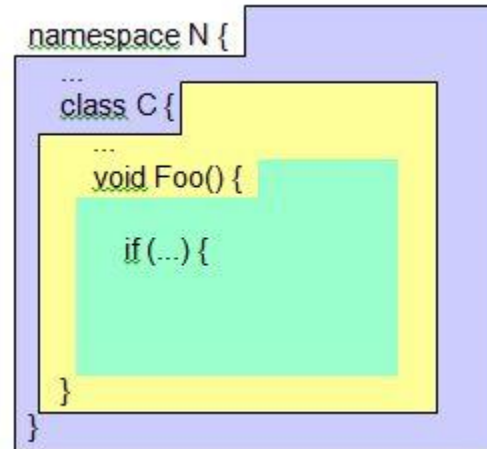
# Synlighet

- **public**
  - synlig där deklarerande namnutrymme är känt
- **private**
  - synlig i deklarerande klass
- **internal**
  - synlig i deklarerande assembly
- **protected**
  - synlig i deklarerande klass och dess underklasser
- **protected internal**
  - synlig i deklarerande assembly samt i underklasser till deklarerande klass som ligger i andra assemblies



# Deklaration

- Namespace
- Klasser, strukturer och interface
- Enumerationer
- Block





# Namespace

- Gruppera sammanhörande deklARATIONER
- Samma namn kan finnas i olika namespaces
- Hierarki av namespaces
- Nyckelordet "using" för åtkomst av klasser i andra namespace

`using System;`

```
namespace A
{
    klasser
    interface
    strukturer
    enumerationer
    delegater
    namespace B { ... }
}
```

# Skilnader Namespace – Java packages

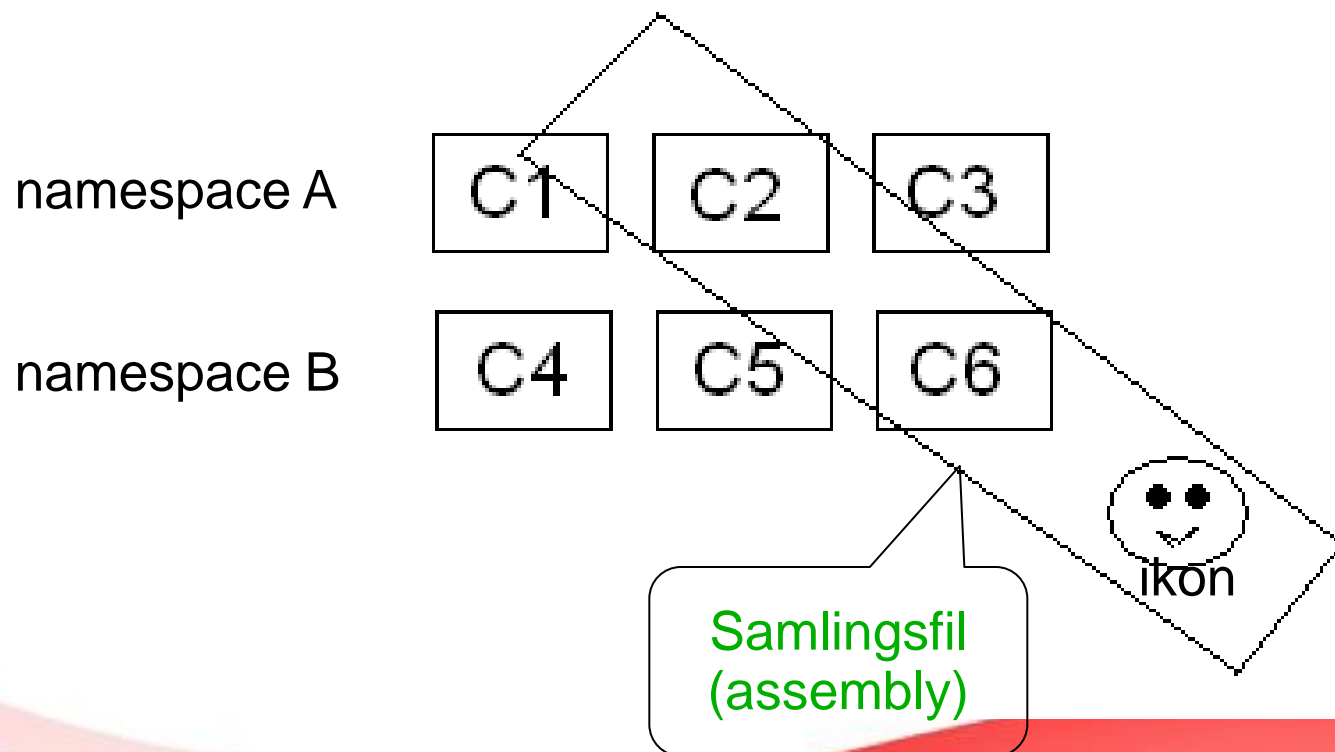
## C#

- Importerar namnutrymmen
- En cs-fil kan innehålla flera namnutrymmen
- Namnutrymmen och klasser mappas ej mot kataloger och filer
- Alias för namnutrymmen:  
`using D = System.Diagnostics;`

## Java

- Importerar klasser
- En java-fil kan bara innehålla ett namnutrymme
- Namnutrymmen och klasser mappas mot kataloger och filer
- Inga alias för namnutrymmen

# Assembly = Samlingsfiler





# Samlingsfiler (2)

- Den minsta enhet som kan driftsättas
- Består av typer (klasser) och eventuella resurser
- Påminner om JAR-filer i Java-världen
- Ofta gäller att  
1 samlingsfil = 1 namnutrymme = 1 program
- Notera dock att
  - en samlingsfil kan innehålla flera namnutrymmen
  - ett namnutrymme kan finnas i flera samlingsfiler
  - en samlingsfil kan bestå av flera filer
- Har manifest som är en innehållsförteckning