



UPPSALA
UNIVERSITET

. NET-programmering

Interface



Interface (1)

```
public interface IList : ICollection, IEnumerable
{
    int Add(object value);
    bool Contains(object value);           // Metoder
    ...
    bool IsReadOnly { get; }             // Egenskap
    ...
    object this [int index] { get; set; } // Indexerare
}
```

- Fullständigt abstrakta klasser, ingen implementation
- Kan innehålla metoder, egenskaper, indexerare och händelser
- Medlemmar är public abstract (virtuella), får ej vara static
- Kan utöka andra interface



Interface (2)

- Klass som implementerar ett interface måste implementera samtliga medlemmar
- Implementerade medlemmar anges ej som override
- Implementerade medlemmar kan anges som abstract (d v s kan implementeras även i abstrakta klasser)
- Skall en metod kunna omdefinieras i en subklass, måste den anges som virtual



Använda interface

- Tilldelning:

```
MyClass c = new MyClass();  
IList list = c;
```

- Metodanrop:

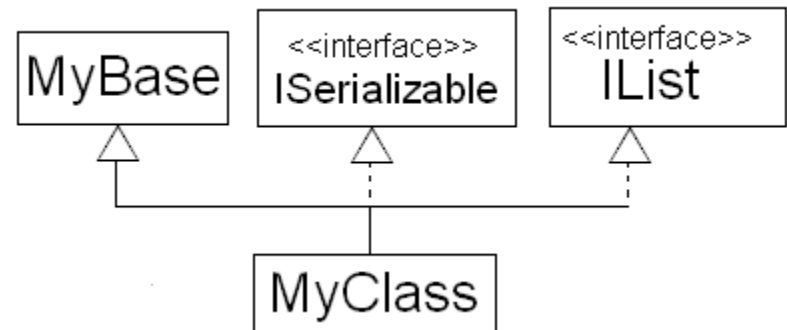
```
list.Add("Kalle");
```

- Typkontroll:

```
if (list is MyClass)    // sant  
if (list is ISerializable) // sant
```

- Typkastning:

```
c = list as MyClass;  
c = (MyClass) list;
```



C# - Preprocessordirektiv

- Instruktioner till kompilatorn
- Tolkas innan kompilering
- Finns inte kvar i den kompilerade koden
- Används bl.a. för villkorlig kompilering

C# - Preprocessordirektiv

Några exempel:

- `#define`
- `#if`
- `#endif`
- `#region`
- `#endregion`

<https://docs.microsoft.com/en-us/dotnet/articles/csharp/language-reference/preprocessor-directives/preprocessor-if>



Anonyma typer

```
var myObject =  
    new {Name = "Apple", Weight = 5};  
  
// Skriver ut "Apple"  
Console.WriteLine(myObject.Name);  
  
// Skriver ut...  
Console.WriteLine(myObject.GetType());
```



Anonyma typer

- Ärver klassen Object
- Är read-only
- Tilldelas ett "hemligt" namn utav kompilatorn, därför måste nyckelordet *var* användas vid deklaration.
- Två anonyma typer som innehåller likadana typer med samma namn i samma ordning ses som samma typ.