

# Interfaces

# Interface (Schnittstelle)

- definiert Member, i.d.R. ohne diese zu implementieren
  - alle Member sind standardmäßig public und abstract
  - Zugriffsmodifizierer können nicht explizit benutzt werden
  - können keine Felder definieren
- 
- eine Klasse kann von einem oder mehreren Interfaces erben und muss alle Member implementieren
  - es kann kein Objekt von einem Interface erstellt werden
  - ein Objekt kann jedoch in den Datentyp des Interfaces konvertiert werden

# Interface (Schnittstelle)

- wird durch Verwendung des Schlüsselwortes interface erstellt
- Eigenschaften werden nicht automatisch implementiert (auch wenn es der gleiche Aufbau ist)

```
interface IFahrzeug
{
    string Bezeichnung { get; set; }
    int Baujahr { get; set; }
    void Losfahren();
}

class Auto : IFahrzeug
{
    public string Bezeichnung { get; set; }
    public int Baujahr { get; set; }
    public void Losfahren()
    {
        Console.WriteLine("Das Auto faehrt los!");
    }
}
```

# Methoden und statische Properties

- Interfaces können statische Member haben
  - sind nur über das Interface aufrufbar
- Statische Properties und alle Methoden können eine Standardimplementierung besitzen können überschrieben werden

```
interface IFahrzeug
{
    public static string Test = "Ein String!";

    string Bezeichnung { get; set; }

    public string Info()
    {
        return "Ich bin ein " + Bezeichnung;
    }
}

class Auto : IFahrzeug
{
    public string Bezeichnung { get; set; }

    public Auto(string bezeichnung)
    {
        this.Bezeichnung = bezeichnung;
    }
}
```

# implizierte Implementierung

- Zugriff auf Member direkt über den Objekttyp

```
interface IFahrzeug
{
    void Losfahren();
}

class Auto : IFahrzeug
{
    public void Losfahren()
    {
        Console.WriteLine("Das Auto faehrt los!");
    }
}

Auto meinAuto = new Auto();
meinAuto.Losfahren();
```

# explizierte Implementierung

- die Member können keine Zugriffsmodifizierer verwenden
- Zugriff nur über Konvertierung in den Typ des Interfaces

```
interface IFahrzeug
{
    void Losfahren();
}

class Auto : IFahrzeug
{
    void IFahrzeug.Losfahren()
    {
        Console.WriteLine("Das Auto faehrt los!");
    }
}

Auto meinAuto = new Auto();
meinAuto.Losfahren();

IFahrzeug autoAlsFahrzeug = (IFahrzeug)meinAuto;
autoAlsFahrzeug.Losfahren();
```