

Interfaces

Freitag, 13. Januar 2017 14:47

Interfaces (Schnittstellen)

- Vertragsvereinbarung für Klassen, welche Pflichten sie erfüllen müssen (welche Member sie implementieren müssen)
- Klassen dürfen in C# nur von einer Basis-Klasse erben, aber beliebig viele Schnittstellen implementieren
- Schnittstellen sind wie abstrakte Klassen, definieren nur Member aber spezifizieren keine Werte / Implementierung
- erlaubte Member:
 - Methoden
 - Properties
 - Events

Beispiel:

```
public interface ICopy
{
    string Caption {get; set;}
    void Copy();
}
class Document : BaseDocument, ICopy {
    public void Copy() {
        [...]
    }
    public string Caption {
        get{ [...] }
        set{ [...] }
    }
    [...]
}
```

Nutzen: Ersatz für fehlende Mehrfachvererbung

Beispiel:

- Sie schreiben ein Grafikprogramm und haben viele Objekte die alle eine Methode Draw() implementieren sollen, die das Objekt auf dem Bildschirm zeichnet.
- Sie haben alle instanziierten Objekte in einem Array gespeichert
- Dieses Array soll durchlaufen werden um alle Objekte zu zeichnen

Wie muss der folgende Code ergänzt werden, damit man ein Array der mal-baren Objekte durchlaufen und dabei Draw() ausführen kann ?

```
public class Rectangle : GeometricObject
{
    public void Draw() { ... }
}
public class Cylinder : GeometricObject
{
    public void Draw() { ... }
}
public class Transporter : Car
{
    public void Draw() { ... }
}
```

```
_____ [] graphicObjects = new _____[3];
graphicObjects[0] = new Rectangle();
graphicObjects[1] = new Cylinder();
graphicObjects[2] = new Transporter();
foreach (_____ item in graphicObjects) item.Draw();
```