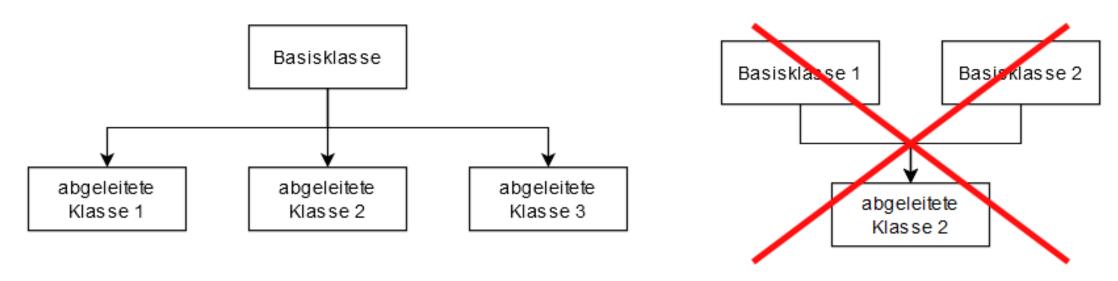


Vererbung und Zugriffsmodifizierer



Vererbung (Ableitung)

- Klassen können voneinander erben
- übernehmen Eigenschaften und Methoden der Basisklasse*
- Es kann an beliebig viele Klassen vererbt werden
- Es kann nur von jeweils einer Klasse geerbt werden



^{*}nicht alle Member werden übernommen(bspw. kein Konstruktor/Destruktor)



Konstruktoren in abgeleiteten Klassen

- beim Initialisieren einer neuen Instanz wird immer erst der Konstruktor der Basisklasse ausgeführt
- wenn der Konstruktor der Basisklasse Parameter enthält, muss dieser explizit mit ": base(*Parameter*)" angegeben werden

```
public class Lebewesen
    1-Verweis
    public int Alter { get; set; }
    public Lebewesen(int alter)
        this.Alter = alter;
public class Mensch : Lebewesen
    1-Verweis
    public string Wohnort { get; set; }
    0 Verweise
    public Mensch(int alter, string wohnort) : base(alter)
        this.Wohnort = wohnort;
```



virtuelle Member

- verhalten sich wie normale Member und können auch so aufgerufen werden
- können von einer abgeleiteten Klasse mit dem Schlüsselwort "override" überschrieben werden
- mit Schlüsselwort "base" kann auf den Basisklassenmember zugegriffen werden

```
public class Lebewesen
    public virtual string WasBinIch()
        return "Ich bin ein Lebewesen";
public class Mensch : Lebewesen
    public override string WasBinIch()
        string wasBinIch = base.WasBinIch();
        wasBinIch += " auf zwei Beinen und kann sprechen";
        return wasBinIch;
meinMensch.WasBinIch();
//"Ich bin ein Lebewesen auf zwei Beinen und kann sprechen"
```



Überschreibung verhindern

- man kann durchgehend virtuelle Member überschreiben, auch wenn eine Klasse von einer schon abgeleiteten Klasse abgeleitet ist
- dies kann man verbinden indem man das Schlüsselwort "sealed" verwendet und somit den neu ableitenden Klassen das Überschreiben unterbindet

```
public class Lebewesen
   public virtual string WasBinIch()
        return "Ich bin ein Lebewesen";
public class Mensch : Lebewesen
   public sealed override string WasBinIch()
        return "Ich bin ein Mensch";
public class Kind: Mensch
    //kann nicht überschrieben werden
   public override string WasBinIch()
        return "Ich bin ein Kind";
```



Zugriffsmodifizierer (Modifier)

 können für Felder, Eigenschaften, Methoden und ganze Klassen verwendet werden

Modifier	Zugriff
public	von außerhalb der Klasse
private	nur innerhalb der Klasse
protected	innerhalb der Klasse und in allen abgeleiteten
	Klassen
internal	nur die aktuellen Assembly
protected internal	nur die aktuellen Assembly oder abgeleitete Klasse
private protected	nur abgeleitete Klassen innerhalb der Assembly