bib International College

C#

Vererbung

Vererbung

bib International College

```
class Basisklasse {}
class Subklasse : Basisklasse {}
```

- keine Mehrfachvererbung
- Basisklasse: public
  => Subklasse: public oder internal
- Basisklasse: internal
  => Subklasse: internal
- Konstruktoren und Destruktoren werden nicht geerbt

```
class Person {
   public string Name {set;get;}

   public Person (string name){
      Name = name;
}
```

Konstruktoraufruf der Basisklasse

```
class Mitarbeiter : Person
{
  private float gehalt;

  public Mitarbeiter(string name, float gehalt): base(name)
  {
    this.gehalt = gehalt;
  }

  public Mitarbeiter(): this("", 0.0F) {}

}
```

Überladen

bib International College

Geerbte Methoden können in der abgeleiteten Klasse problemlos überladen werden. Man muss nur darauf achten, dass die Methoden gleichen Namens unterschiedliche Parameterlisten haben.

Überschreiben

bib International College

```
class Person {
  public string Name;
  public virtual void MeineMethode(){
   System.Console.WriteLine("Person: "+ Name);}
}
class Mitarbeiter: Person {
  public override void MeineMethode(){
   System.Console.WriteLine("Mitarb: "+ Name);}
}
Person p1 = new Person("Moni");
p1.MeineMethode(); => Person: Moni
Mitarbeiter m = new Mitarbeiter((„Anna");
m.MeineMethode(); => Mitarb: Anna
Person m1 = new Mitarbeiter("Toni");
m1.MeineMethode(); => Mitarb: Toni
```

gewünschte Ausgabe (Polymorphie)

Abstrakte Klassen

bib International College

- Eine abstrakte Klasse kann man niemals instantiieren.
- Abstrakte Klassen können abstrakte Member deklarieren, die sich wie virtuelle Member verhalten, aber keine Standardimplementierung bereitstellen.
- Die Implementierung muss in der Subklasse mit *override* vorgenommen werden.

Funktionen und Klassen versiegeln

- Ein Member kann seine Implementierung mithilfe von *sealed* versiegeln, um zu verhindern, dass er durch weitere Subklassen überschrieben wird.
- Wird eine Klasse selbst versiegelt, werden alle virtuellen Member implizit versiegelt.
- Das Versiegeln einer Klasse kommt häufiger vor als das Versiegeln eines Members