

Bankkonten als Objekte in Java:

```
class Konto {
    int kontostand;                // lokaler Zustand

    Konto(int i) { kontostand = i; } // Konstruktor

    public int kontostand() {      // Methode
        return kontostand;
    }

    public void abheben(int betrag) {
        if (betrag > kontostand)
            System.out.println("Konto nicht gedeckt!");
        else
            kontostand -= betrag;
    }

    public void einzahlen(int betrag) {
        kontostand += betrag;
    }

    public static void main(String[] args) {
        Konto k = new Konto(100);
        System.out.println("Kontostand: " + k.kontostand());
        k.einzahlen(10);
        System.out.println("Kontostand: " + k.kontostand());
        k.abheben(130);
        System.out.println("Kontostand: " + k.kontostand());
    }
}
```

Definition von Untertypen:

Privatkonto **ist ein** Bankkonto und hat *zusätzliche* Methode zum Einziehen von Kontoführungsgebühren (und Attribut zur Gebührenhöhe):

(Beachte: Privatkonto *erbt* automatisch alle Komponenten und Methoden von Bankkonto!)

```
class Privatkonto extends Konto {  
  
    int gebuehren = 10;  
  
    Privatkonto(int i) { super(i); }  
  
    public void gebuehrenEinziehen() {  
        kontostand -= gebuehren;  
    }  
}
```

Redefinition von Methoden:

Ausbildungskonto **ist ein** Privatkonto, wobei aber die Kontoführung gebührenfrei ist:

```
class Ausbildungskonto extends Privatkonto {  
  
    Ausbildungskonto(int i) { super(i); }  
  
    public void gebuehrenEinziehen() { }  
}
```