

Konto- und Bank-Beispiel Vererbung

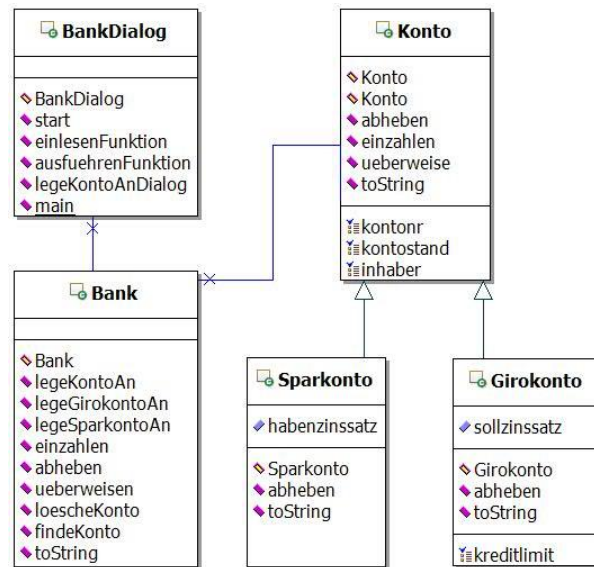
Prof. Dr. Helmut G. Folz



Zusätzliche Klassen einführen

- Von der Klasse Konto sollen zwei Klassen abgeleitet werden:
- **Sparkonto**
Zusätzliche Eigenschaften:
 - ⇒ Habenzinssatz
- **Girokonto**
Zusätzliche Eigenschaften:
 - ⇒ Kreditlimit
 - ⇒ Sollzinssatz

Neue Klassenstruktur



Änderungen in Klasse Konto (1)

```

public abstract class Konto {
    private int kontonr; // muss 4-stellig sein
    private String inhaber; // darf nicht leer sein
    private double kontostand;

    public Konto(int kontonr, String inhaber, double kontostand) {
        check(kontonr >= MIN_KONTONR
            && kontonr <= MAX_KONTONR, MSG_KONTONR);
        check((inhaber != null && inhaber.length() > 0),
            MSG_INHABER);
        check((kontostand >= 0.0), MSG_KONTOSTAND);
        this.kontonr = kontonr;
        this.inhaber = inhaber;
        this.kontostand = kontostand;
    }
}

```

Änderungen in Klasse Konto (2)

```
public String toString() {  
    return String.format(  
        "%-20s Typ: %-10s Kontonr: %4d Kontostand: %10.2f",  
        inhaber, getClass().getName(),  
        kontonr, kontostand);  
}
```

Klasse Sparkonto (1)

```
public class Sparkonto extends Konto {  
    public static final String MSG_HABENZINSSATZ =  
        "Habenzinssatz muss > 0.0% sein";  
    public static final String MSG_KONTOSTAND_KLEINER_0 =  
        "Kontostand darf nicht kleiner als 0.0 sein!";  
  
    private double habenzinssatz; // muss > 0.0 sein  
  
    public Sparkonto(int kontonr, String inhaber,  
        double habenzinssatz) {  
        super(kontonr, inhaber, 0.0);  
        check((habenzinssatz > 0.0), MSG_HABENZINSSATZ);  
        this.habenzinssatz = habenzinssatz;  
    }  
}
```

Klasse Sparkonto (2)

```
/**
 * abheben: Das Kreditlimit muss beachtet werden
 */
public void abheben(double betrag) {
    check((getKontostand() >= betrag),
          MSG_KONTOSTAND_KLEINER_0);
    super.abheben(betrag);
}

public String toString() {
    return String.format(
        "%s\n%20s Habenzinssatz: %10.2f",
        super.toString(), " ", habenzinssatz);
}
```

Klasse Girokonto (1)

```
public class Girokonto extends Konto {
    public static final String MSG_SOLLZINSSATZ =
        "Sollzinssatz nicht > 10.0%";
    public static final String MSG_KREDITLIMIT =
        "Kreditlimit muss >= 0 sein!";
    public static final String MSG_KREDITLIMIT_UEBERSCHRITTEN =
        "Kreditlimit ueberschritten!";

    private double kreditlimit; // muss >= 0 sein
    private double sollzinssatz; // muss > 10% sein

    public Girokonto(int kontonr, String inhaber,
                     double sollzinssatz) {
        super(kontonr, inhaber, 0.0);
        check((sollzinssatz > 10.0), MSG_SOLLZINSSATZ);
        kreditlimit = 0.0;
        this.sollzinssatz = sollzinssatz;
    }
}
```

Klasse Girokonto (2)

```
public double getKreditlimit() {
    return kreditlimit;
}

public void setKreditlimit(double kreditlimit) {
    check((kreditlimit >= 0.0), MSG_KREDITLIMIT);
    this.kreditlimit = kreditlimit;
}
```

Klasse Girokonto (3)

```
/**
 * abheben: Das Kreditlimit muss beachtet werden
 */
public void abheben(double betrag) {
    check((getKontostand() - betrag >= -kreditlimit),
        MSG_KREDITLIMIT_UEBERSCHRITTEN);
    super.abheben(betrag);
}

public String toString() {
    return String.format(
        "%s\n%20s Kreditlimit: %10.2f Sollzinssatz: %10.2f",
        super.toString(), " ", kreditlimit, sollzinssatz);
}
}
```

Änderungen in Klasse BankDialog (1)

```
public class BankDialog {  
    ...  
  
    private void ausfuehrenFunktion(Funktion funktion) {  
        int kontonr, nachKontonr;  
        double betrag;  
  
        switch (funktion) {  
            case ANLEGEN:  
                legeKontoAnDialog();  
                break;  
        }  
    }  
    ...  
}
```

Änderungen in Klasse BankDialog (2)

```
private void legeKontoAnDialog() {  
    final int GIROKONTO = 1;  
    final int SPARKONTO = 2;  
    int kontonr;  
    String inhaber;  
    System.out.print("Kontoart? (1 = Girokonto; 2 = Sparkonto) ");  
    int kontoart = input.nextInt();  
    kontonr = einlesenKontonr();  
    inhaber = einlesenInhaber();  
  
    if (kontoart == GIROKONTO) {  
        System.out.print("Kreditlimit = ");  
        double kreditlimit = input.nextDouble();  
        System.out.print("Sollzinssatz = ");  
        double sollzinssatz = input.nextDouble();  
        bank1.legeGirokontoAn(kontonr, inhaber, kreditlimit,  
                               sollzinssatz);  
    }  
}
```

Änderungen in Klasse BankDialog (3)

```
else if (kontoart == SPARKONTO) {
    System.out.print("Habenzinssatz = ");
    double habenzinssatz = input.nextDouble();
    bank1.legeSparkontoAn(kontonr, inhaber, habenzinssatz);
} else {
    System.out.println("Falsche Kontoart eingegeben");
}
}
```

Änderungen in Klasse Bank (1)

```
/**
 * Anlegen eines Kontos. Es ist zu pruefen, ob die
 * KontoNr bereits vergeben wurde und ob die
 * Kontotabelle bereits voll ist.
 *
 * @param k bereits erzeugtes Konto-Objekt
 * @param inhaber Name des Kontoinhabers
 */
public void legeKontoAn(Konto k) {
    Konto.check(findeKonto(k.getKontonr()) < 0,
                MSG_KONTO_VORHANDEN);
    Konto.check(anzKonten < kontoTab.Length,
                MSG_KONTO_ANLEGEN);

    kontoTab[anzKonten] = k;
    anzKonten++;
}
```

Änderungen in Klasse Bank (2)

```
/**
 * Fuege ein Girokonto hinzu
 *
 * @param kontonr Kontonummer muss 4-stellig sein
 * @param inhaber nichtleerer Kontoinhaber
 * @param kreditlimit zu setzendes Kreditlimit
 * @param sollzinssatz zu setzender Sollzinssatz
 */
public void legeGirokontoAn(int kontonr, String inhaber,
                             double kreditlimit,
                             double sollzinssatz) {
    Girokonto k = new Girokonto(kontonr, inhaber,
                                  sollzinssatz);
    k.setKreditlimit(kreditlimit);
    legeKontoAn(k);
}
```

Änderungen in Klasse Bank (3)

```
/**
 * Fuege ein Sparkonto hinzu
 *
 * @param kontonr Kontonummer muss 4-stellig sein
 * @param inhaber nichtleerer Kontoinhaber
 * @param habenzinssatz zu setzender habenzinssatz
 */
public void legeSparkontoAn(int kontonr, String inhaber,
                             double habenzinssatz) {
    Sparkonto k = new Sparkonto(kontonr, inhaber,
                                  habenzinssatz);
    legeKontoAn(k);
}
```