



UNIVERSITÄT
LEIPZIG

Einführung in die Objekt-Orientierte Modellierung und Programmierung

Wintersemester 2025/2026

Dirk Zeckzer

Institut für Informatik



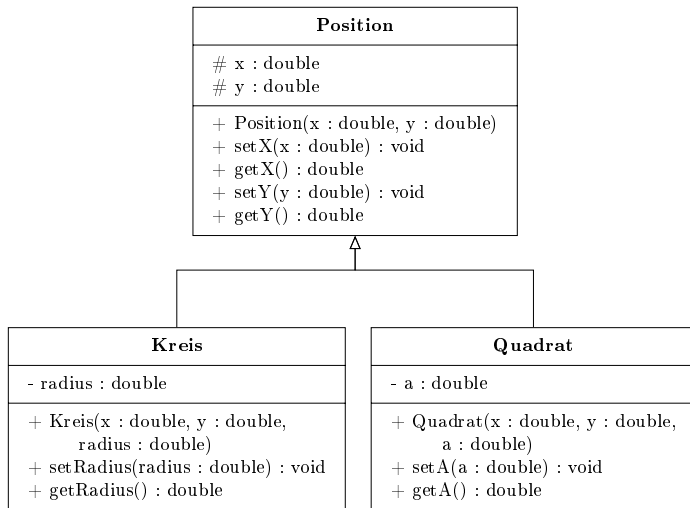
Teil XIII

Modellierung: Erweiterung

Erweiterung

Kreis
<ul style="list-style-type: none">- x : double- y : double- radius : double
<ul style="list-style-type: none">+ Kreis(x : double, y : double, radius : double)+ setX(x : double) : void+ getX() : double+ setY(y : double) : void+ getY() : double+ setRadius(radius : double) : void+ getRadius() : double

Quadrat
<ul style="list-style-type: none">- x : double- y : double- a : double
<ul style="list-style-type: none">+ Quadrat(x : double, y : double, a : double)+ setX(x : double) : void+ getX() : double+ setY(y : double) : void+ getY() : double+ setA(a : double) : void+ getA() : double



Erweiterung

```
1  package eoomp;
2
3  public class Position {
4
5      protected double x;
6      protected double y;
7
8      public Position(
9          final double x,
10         final double y
11     ) {
12         this.x = x;
13         this.y = y;
14     }
```

```
1  public void setX (
2      final double x
3  ) {
4      this.x = x;
5  }
6
7  public double getX () {
8      return x;
9  }
10
11 public void setY (
12     final double y
13 ) {
14     this.y = y;
15 }
16
17 public double getY () {
18     return y;
19 }
20 }
```

Erweiterung

```
1  package eomp;
2
3  public class Kreis
4      extends Position {
5
6      private double radius;
7
8      public Kreis(
9          final double x,
10         final double y,
11         final double radius
12     ) {
13         super(x, y);
14         setRadius(radius);
15     }
16
17     public void setRadius(
18         final double radius
19     ) {
20         this.radius = radius;
21     }
22
23     public double getRadius () {
24         return radius;
25     }
26 }
```

```
1  package eomp;
2
3  public class Quadrat
4      extends Position {
5
6      private double a;
7
8      public Quadrat (
9          final double x,
10         final double y,
11         final double a
12     ) {
13         super(x, y);
14         setA(a);
15     }
16
17     public void setA(
18         final double a
19     ) {
20         this.a = a;
21     }
22
23     public double getA () {
24         return a;
25     }
26 }
```

Erweiterung: instanceof

instanceof

```
1 Position position = new Kreis(1.0, 1.0, 1.0);
2
3 boolean isKreis = position instanceof Kreis;
4 // Ergebnis: true
5
6 boolean isQuadrat = position instanceof Quadrat;
7 // Ergebnis: false
```

Erweiterung: instanceof

instanceof

```
1  if (position instanceof Kreis) {
2      Kreis kreis = (Kreis) position;
3      System.out.println("Radius: " + kreis.getRadius());
4  } else if (position instanceof Quadrat) {
5      Quadrat quadrat = (Quadrat) position;
6      System.out.println("Seitenlaenge: " + quadrat.getA());
7  } else {
8      ...
9  }
```

```
1  if (position instanceof Kreis kreis) {
2      System.out.println("Radius: " + kreis.getRadius());
3  } else if (position instanceof Quadrat quadrat) {
4      System.out.println("Seitenlaenge: " + quadrat.getA());
5  } else {
6      ...
7  }
```


Modifikator: final

final

- ▶ Methoden mit dem Modifizierer **final** können in erweiternden Klassen nicht überschrieben werden
- ▶ Klassen mit dem Modifizierer **final** können nicht durch andere Klassen erweitert werden