



Folge 1.2

Wir erstellen ein kleines Bild aus Objekten

Folge 1.2

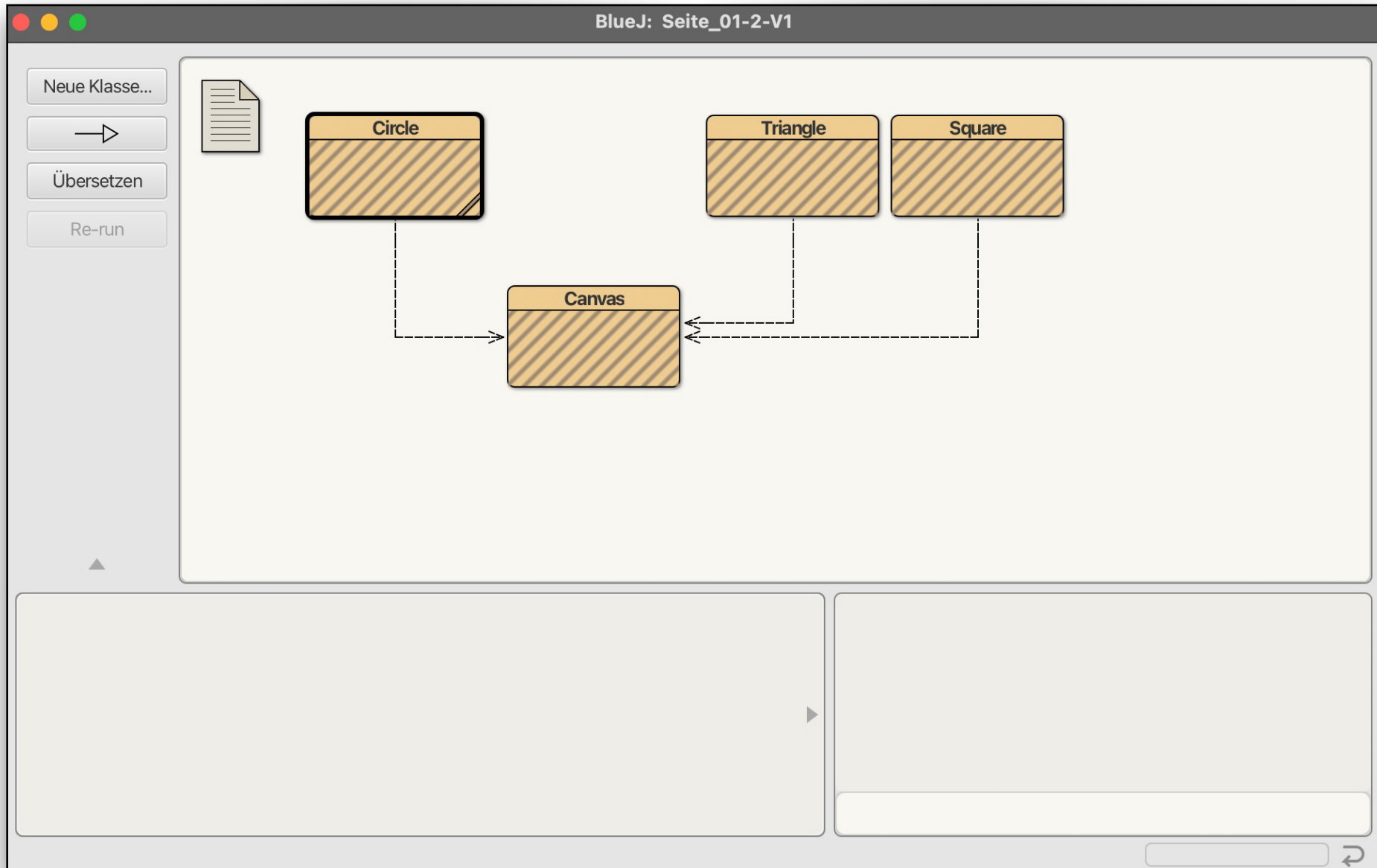
Wir erstellen ein kleines Bild aus Objekten

1.2.1 Das Shapes Projekt

1.2.2 Arbeiten mit Methoden

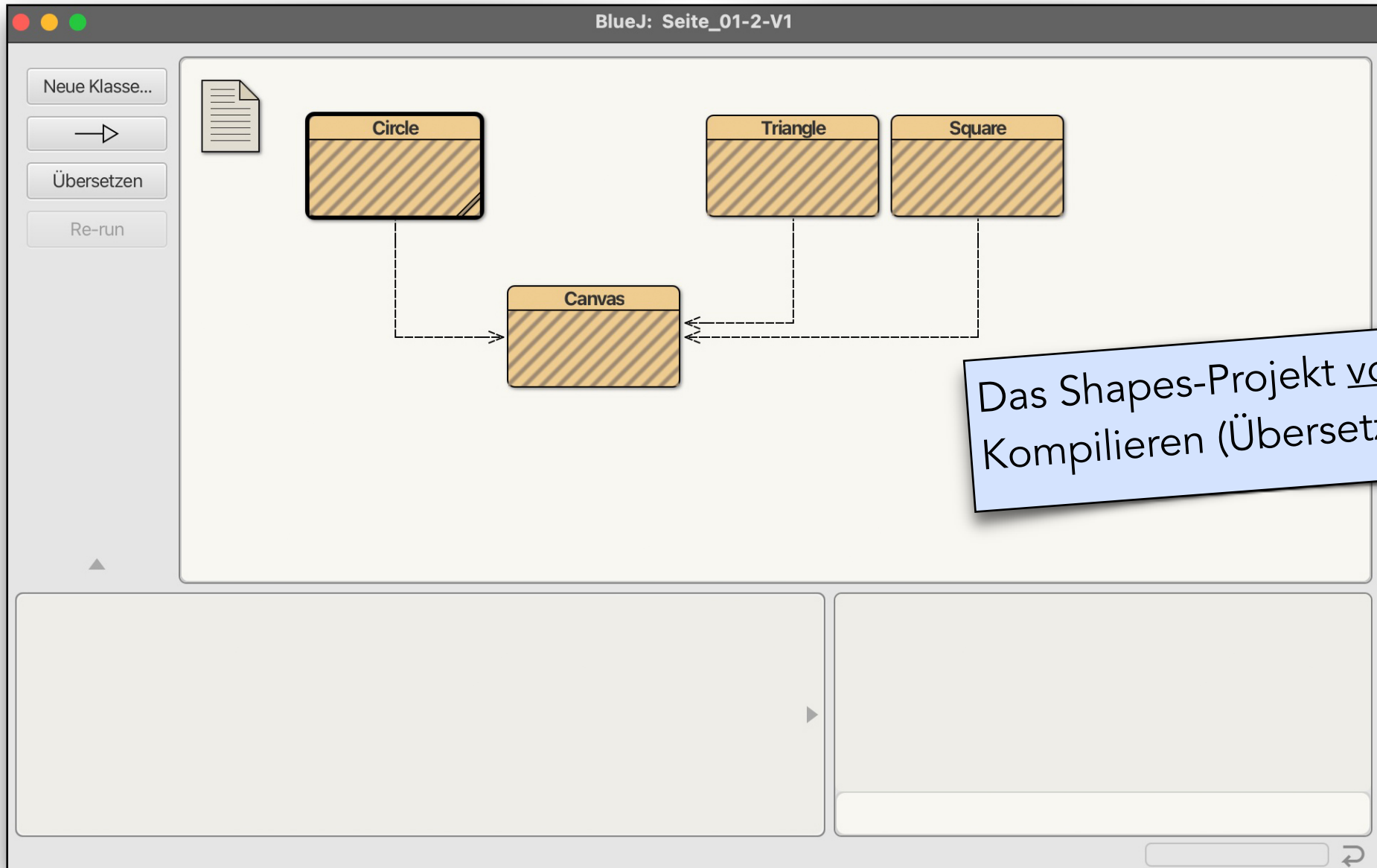
Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt



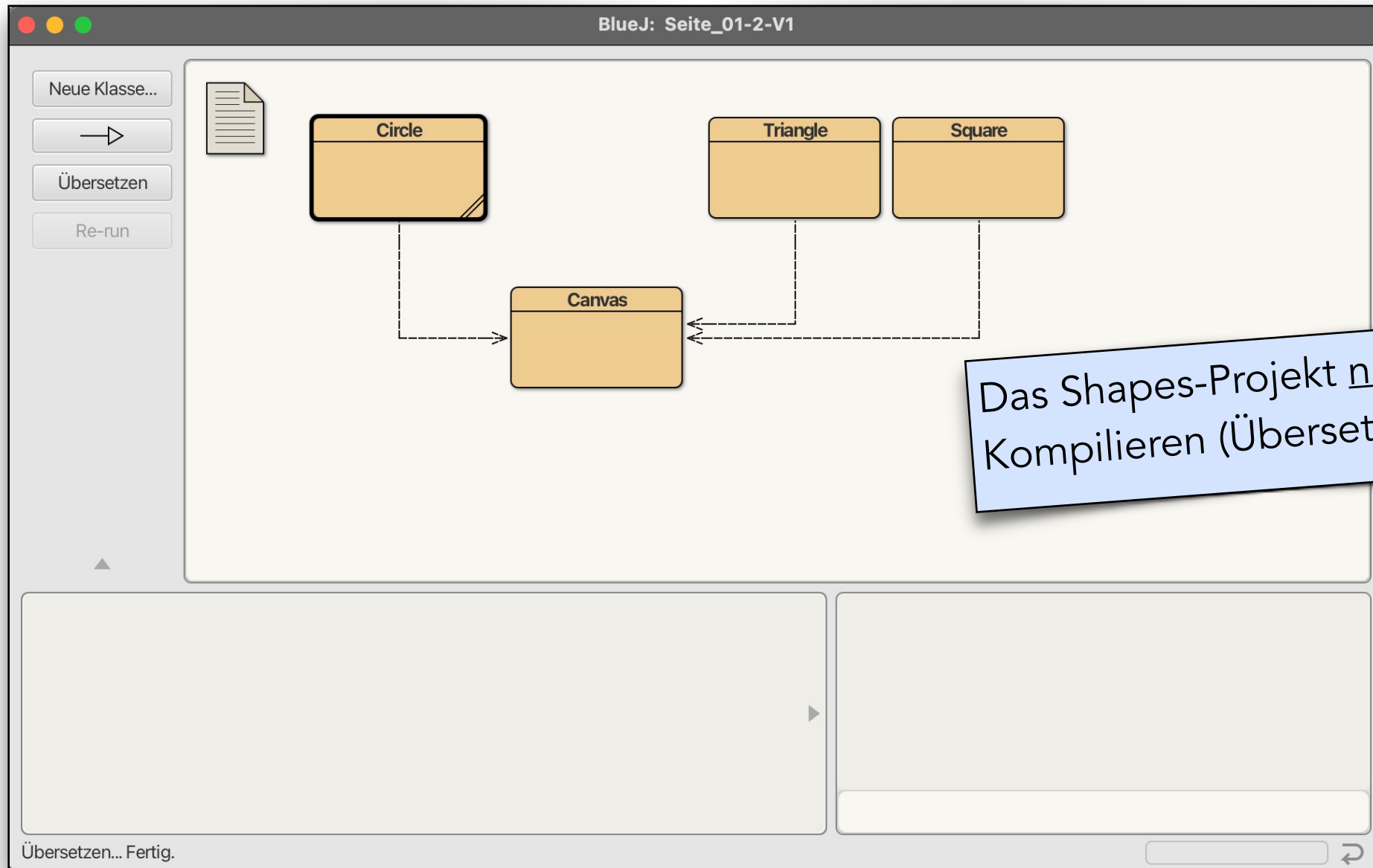
Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt



Einführung in die Objektorientierte Programmierung (OOP)

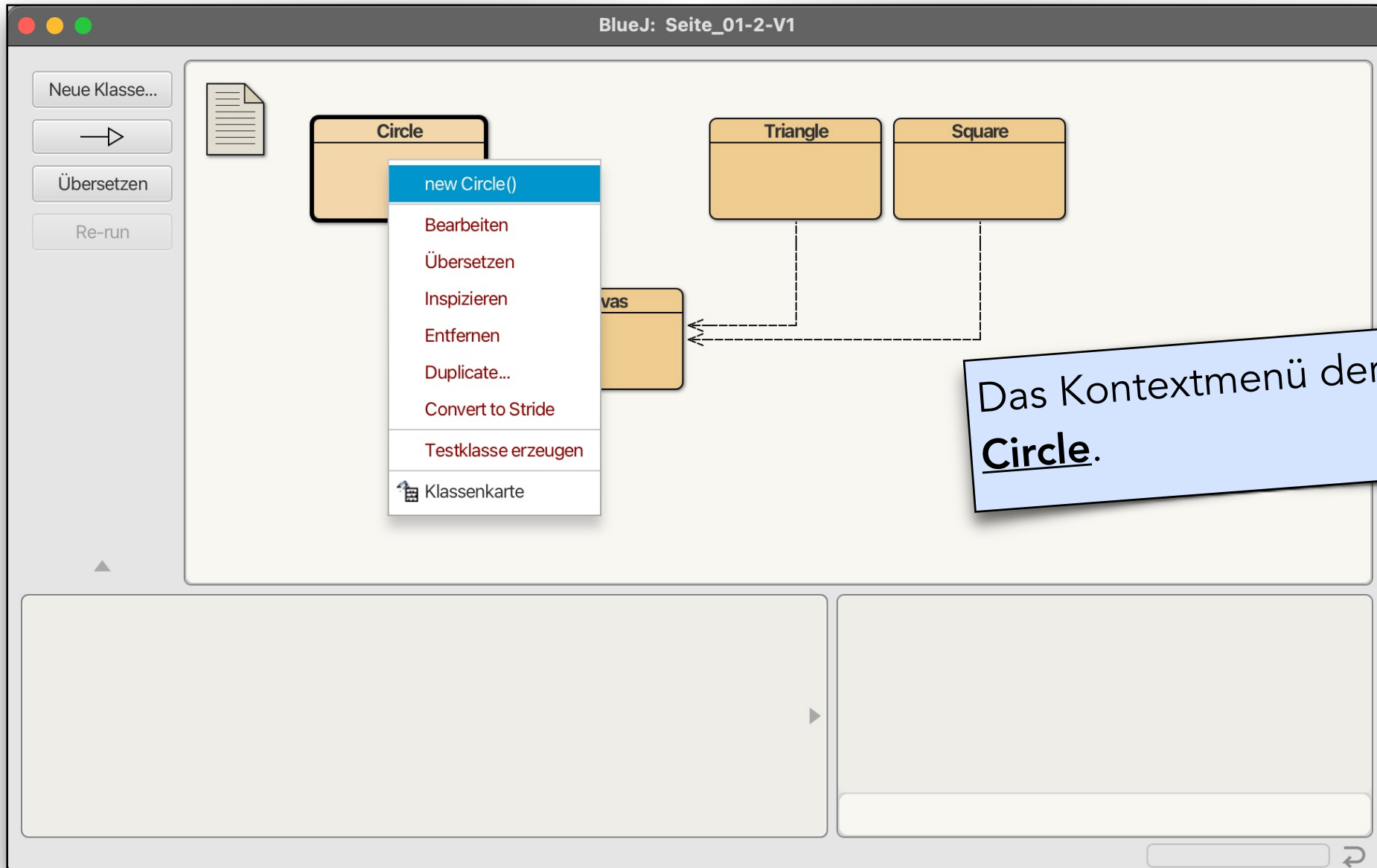
1.2.1 Das Shapes-Projekt



Das Shapes-Projekt nach dem Kompilieren (Übersetzen).

Einführung in die Objektorientierte Programmierung (OOP)

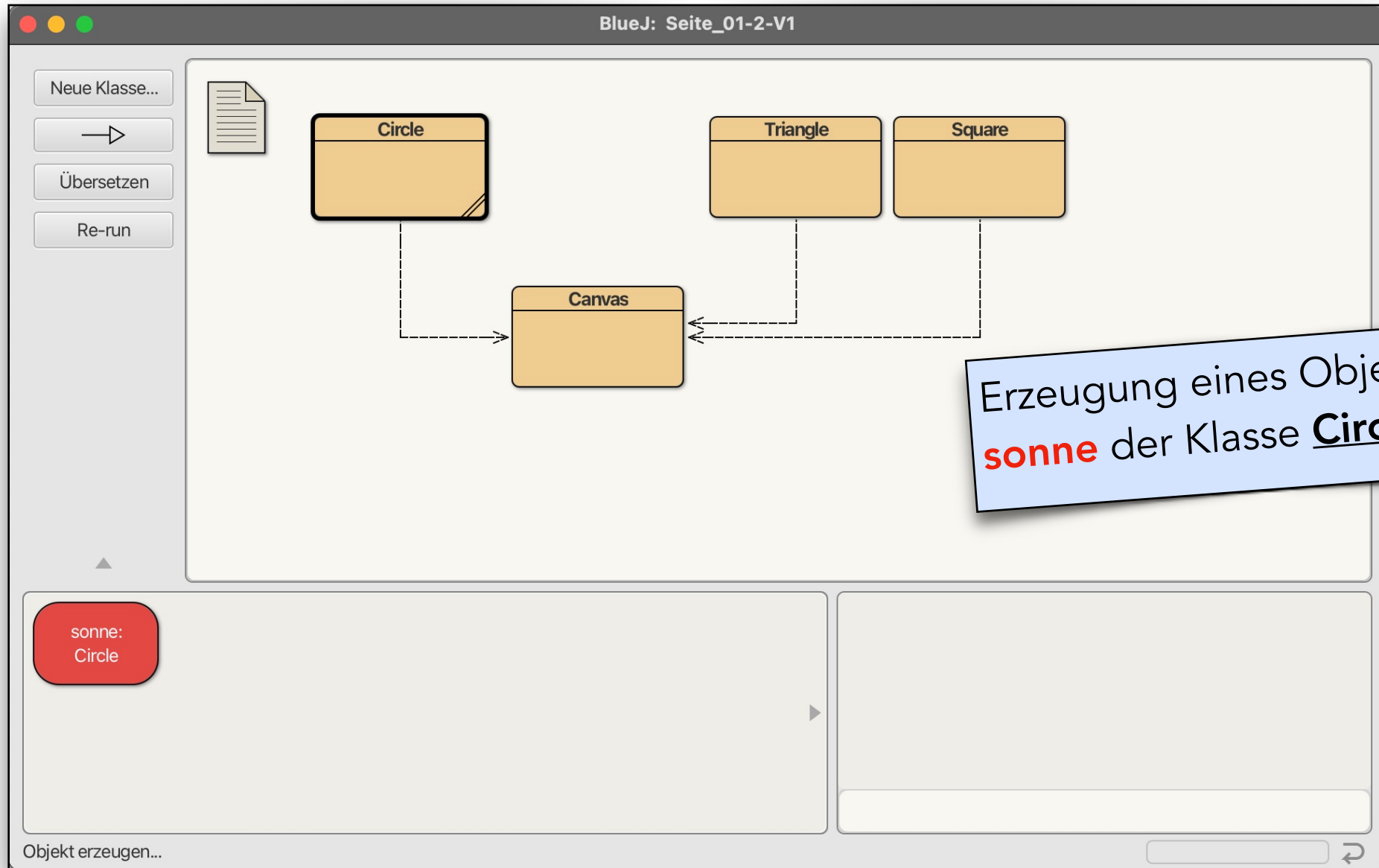
1.2.1 Das Shapes-Projekt



Das Kontextmenü der Klasse **Circle**.

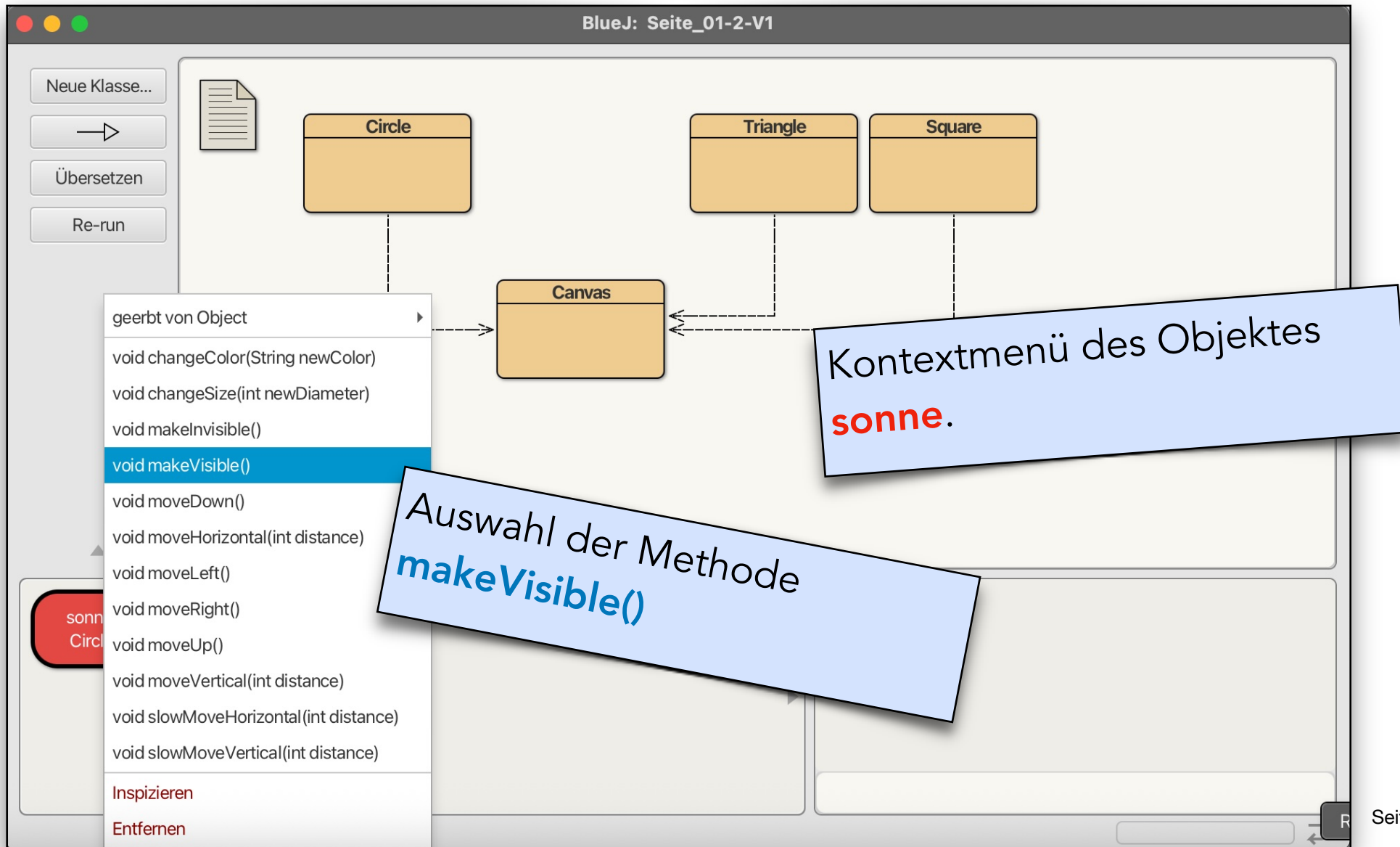
Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt



Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt



Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt

BlueJ: Seite_01-2-V1

Neue Klasse...

→

Übersetzen

Re-run

Circle

Triangle

Square

Canvas

Zeichenfläche mit einem sichtbaren blauen Kreis nach Auswahl der Methode **makeVisible()**

BlueJ Shapes Demo

sonne: Circle

sonne : Circle

Einführung in die Objektorientierte Programmierung (OOP)

1.2.1 Das Shapes-Projekt

Zusammenfassung der Schritte

1. Starten von BlueJ
2. Öffnen des Shapes-Projekts
3. Kompilieren der vier Klassen
4. Erzeugen eines Objekts **sonne** der Klasse **Circle**
5. Aufruf der Methode **makeVisible()** des Objekts **sonne**

sonne:
Circle

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



BlueJ: Seite_01-2-V2

Neue Klasse...
→
Übersetzen
Re-run

Circle Triangle Square

Beide **Circle**-Objekte **sonne** und **mond** wurden mit **makeVisible()** sichtbar gemacht.

sonne: Circle mond: Circle

BlueJ Shapes Demo

mond : Circle

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

The screenshot shows the BlueJ IDE interface. At the top, the title bar reads "BlueJ: Seite_01-2-V2". On the left, there is a sidebar with buttons: "Neue Klasse...", a right-pointing arrow, "Übersetzen", and "Re-run". The main workspace displays a class hierarchy with three classes: "Circle", "Triangle", and "Square". "Circle" is the superclass, and "Triangle" and "Square" are subclasses, indicated by dashed lines. Below the classes, there are two red buttons labeled "sonne: Circle" and "mond: Circle". A blue callout box points to these buttons with the text: "Beide **Circle**-Objekte **sonne** und **mond** wurden mit **makeVisible()** sichtbar gemacht." Another blue callout box points to the "mond: Circle" button with the text: "Dennoch ist nur ein einziger Kreis zu sehen." In the bottom right corner, there is a window titled "BlueJ Shapes Demo" which displays a single blue circle. The status bar at the bottom left shows "mond : Circle".

BlueJ: Seite_01-2-V2

Neue Klasse...

→

Übersetzen

Re-run

Circle

Triangle

Square

sonne: Circle

mond: Circle

BlueJ Shapes Demo

mond : Circle

Beide **Circle**-Objekte **sonne** und **mond** wurden mit **makeVisible()** sichtbar gemacht.

Dennoch ist nur ein einziger Kreis zu sehen.

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

BlueJ: Seite_01-2-V2

Neue Klasse...

→

Circle Triangle Square

sonne : Circle

private int diameter	30
private int xPosition	20
private int yPosition	60
private String color	"blue"
private boolean isVisible	false

Inspiziere

Zeige statische Variablen

mond : Circle

private int diameter	30
private int xPosition	20
private int yPosition	60
private String color	"blue"
private boolean isVisible	false

Schließen

sonne: Circle mond: Circle

mond : Circle

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

The screenshot shows the BlueJ IDE interface. At the top, the title bar reads "BlueJ: Seite_01-2-V2". Below the title bar, there are buttons for "Neue Klasse..." and a right-pointing arrow. In the center, there are three class boxes: "Circle", "Triangle", and "Square". Below these, two object inspectors are open. The first inspector is for an object named "sonne : Circle". It displays a list of private variables: "private int diameter" (value 30), "private int xPosion" (value 20), "private int yPosion" (value 60), "private String color" (value "blue"), and "private boolean isVisible" (value false). The "private int diameter" row is highlighted in yellow. Below the list is a button labeled "Zeige statische Variablen". The second inspector is for an object named "mond : Circle". It displays the same list of private variables with the same values. The "private int diameter" row is also highlighted in yellow. To the right of the "mond : Circle" inspector are three buttons: "Inspiziere", "Hole", and "Schließen". At the bottom left, there is a text box with the text "mond : Circle".

Aufzug des Objektinspektors für beide Objekte.

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

BlueJ: Seite_01-2-V2

Neue Klasse...

Circle

sonne : Circle

private int diameter	30
private int xPos	20
private int yPos	60
private String color	"blue"
private boolean isVisible	false

Zeige statische Variablen

mond : Circle

private int diameter	30
private int xPos	20
private int yPos	60
private String color	"blue"
private boolean isVisible	false

Inspiziere

Hole

Schließen

Aufruf des Objektinspektors für beide Objekte.

Alle fünf Instanzvariablen haben die gleichen Werte!

mond : Circle

Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

BlueJ: Seite_01-2-V2

Neue Klasse...

Circle

sonne : Circle

private int diameter 30

private int xPosition 20

private int yPosition

private String color

private boolean isVisible

Zeige statische Variablen

mond : Circle

private boolean isVisible false

Inspiziere

Hole

Schließen

Alle fünf Instanzvariablen haben die gleichen Werte!

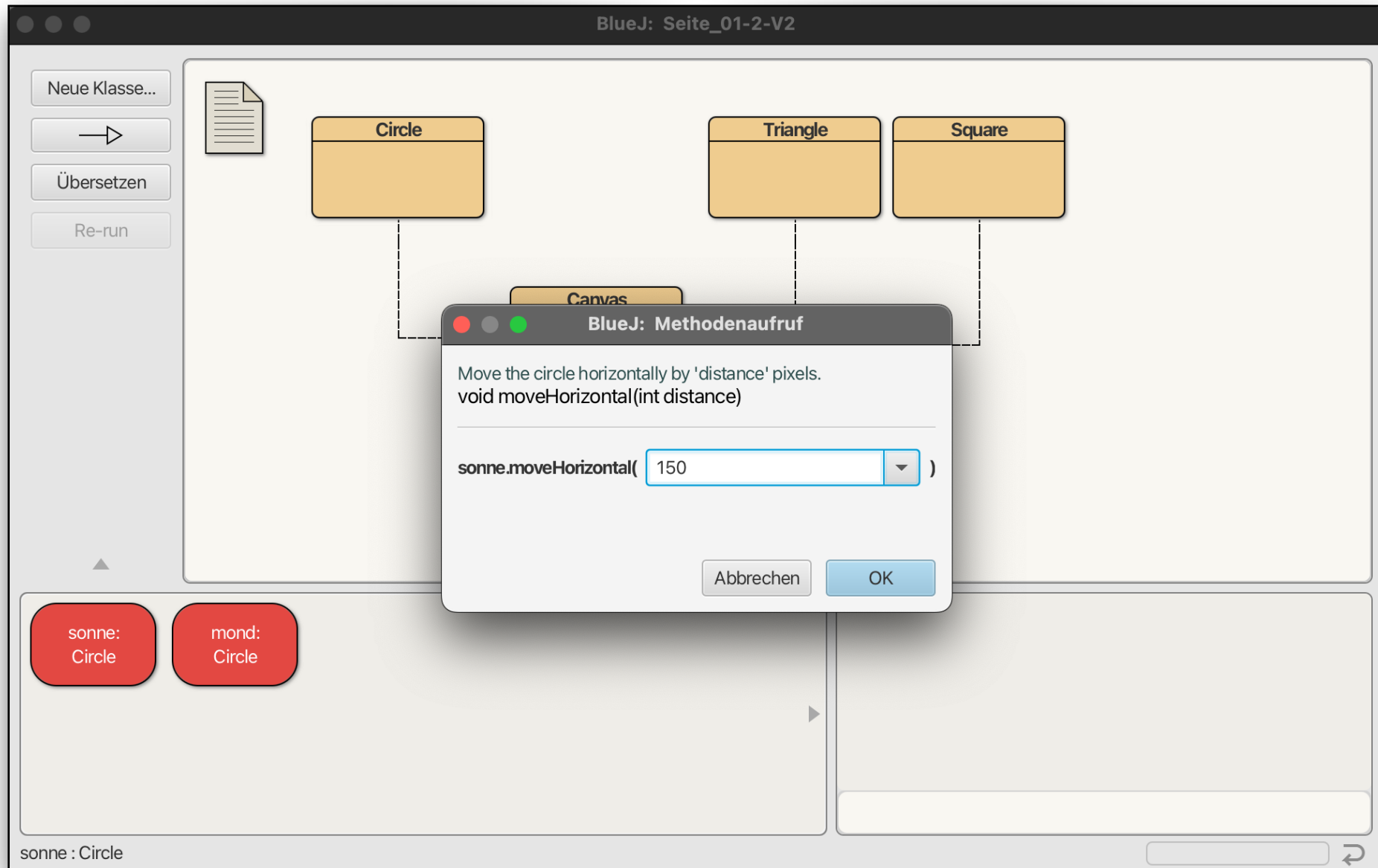
Durch gezielten Einsatz der Methoden können wir das Objekt **sonne** in eine andere Position bringen.

Aufruf des Objektinspektors für beide Objekte.

mond : Circle

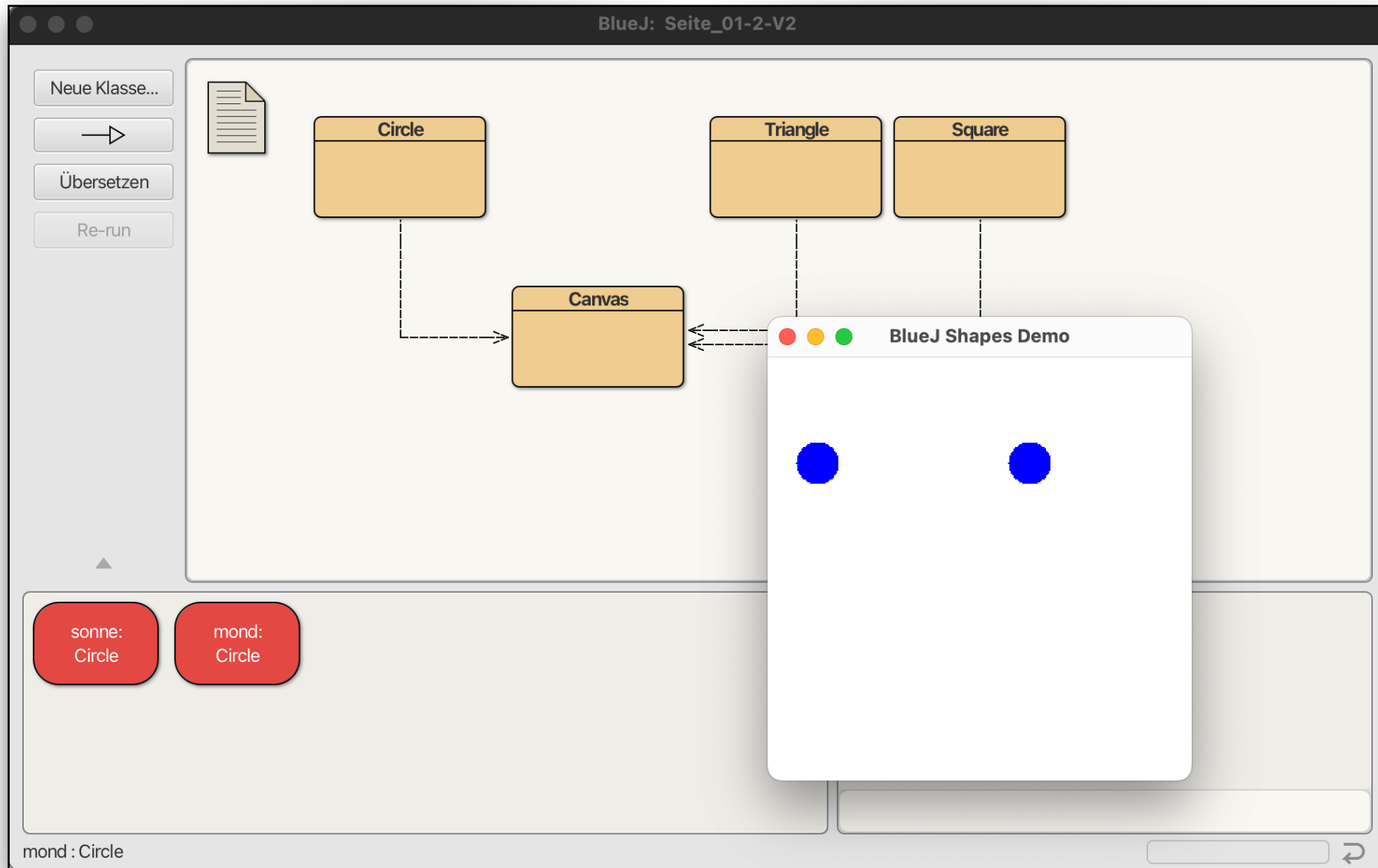
Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



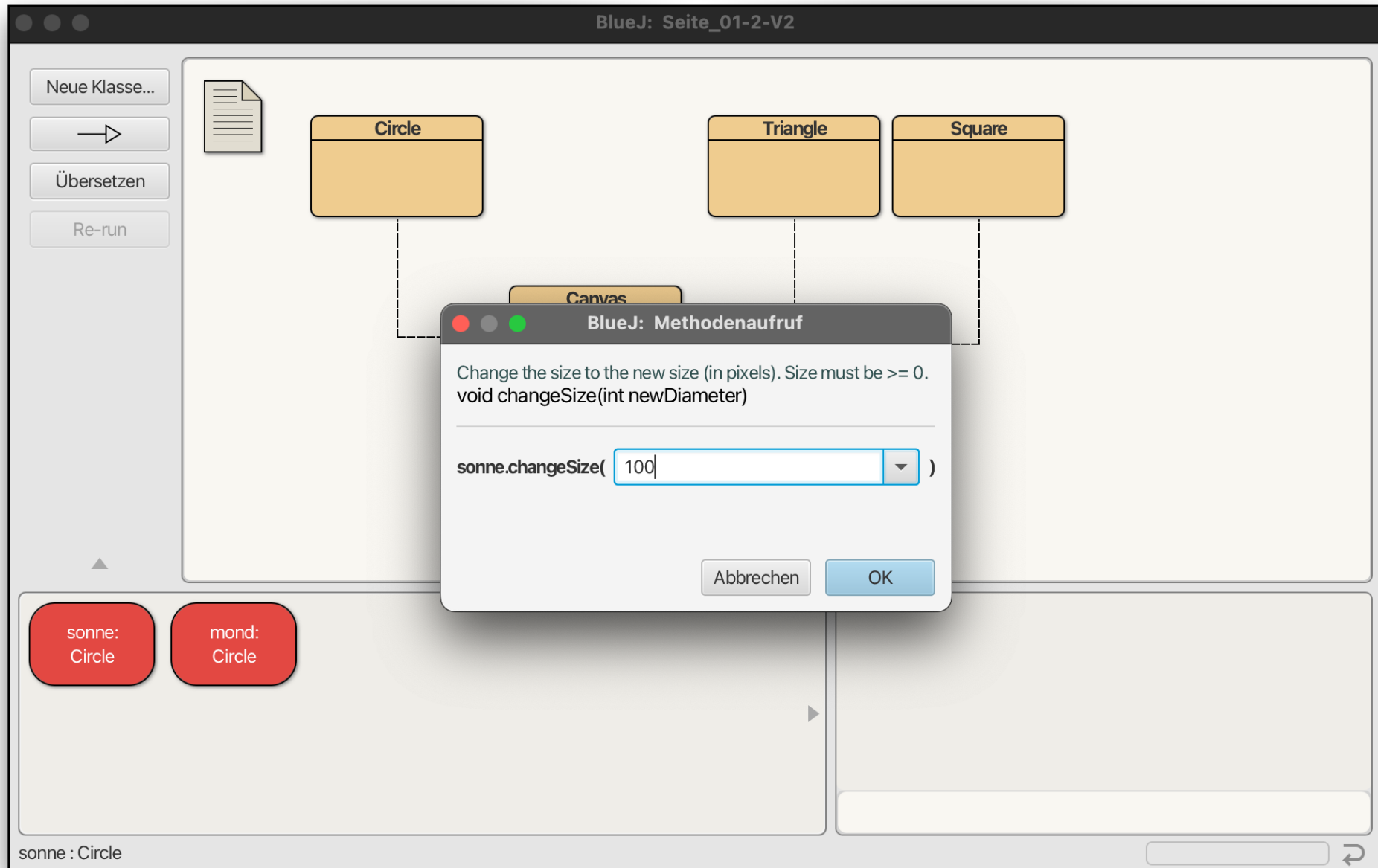
Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



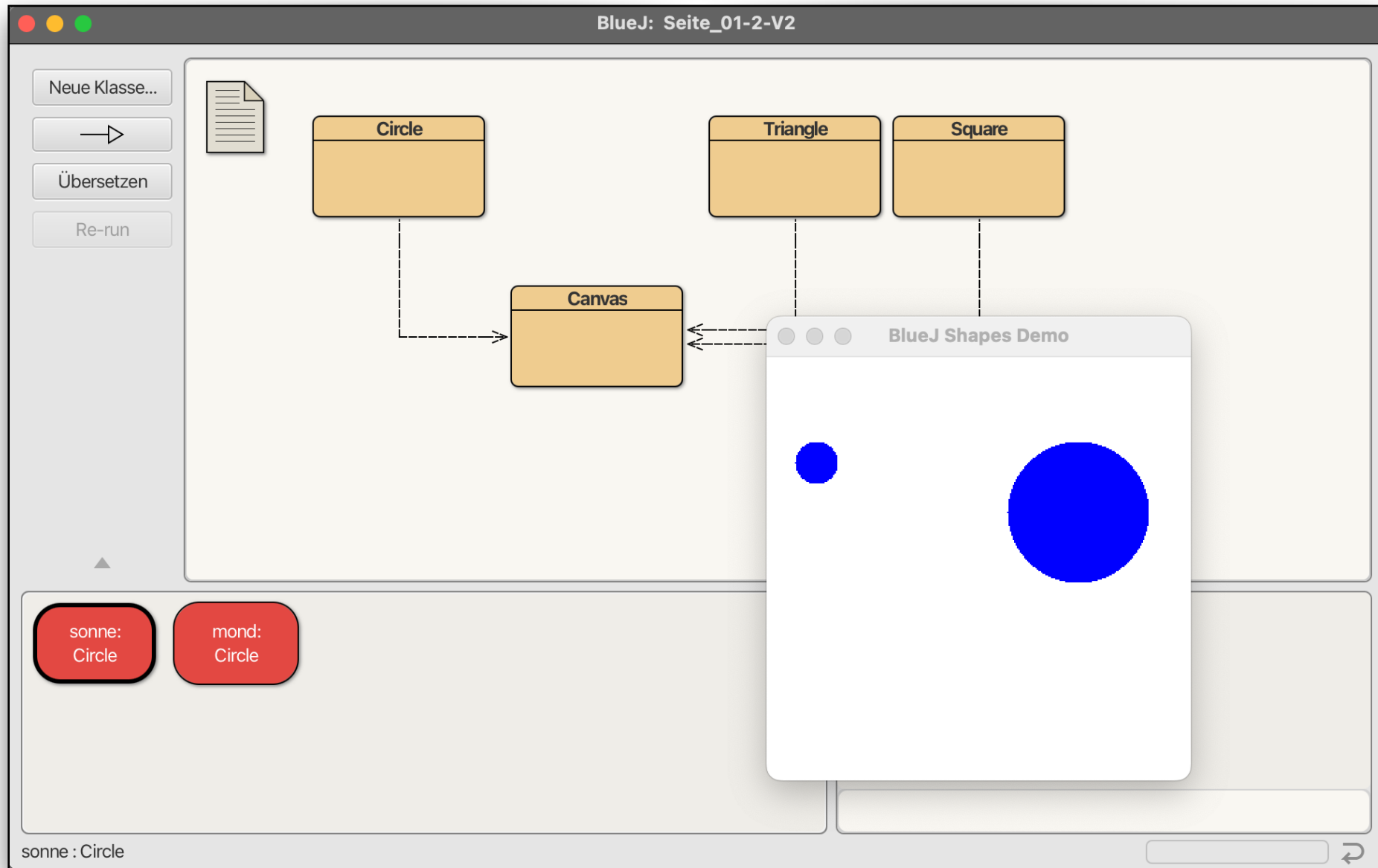
Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



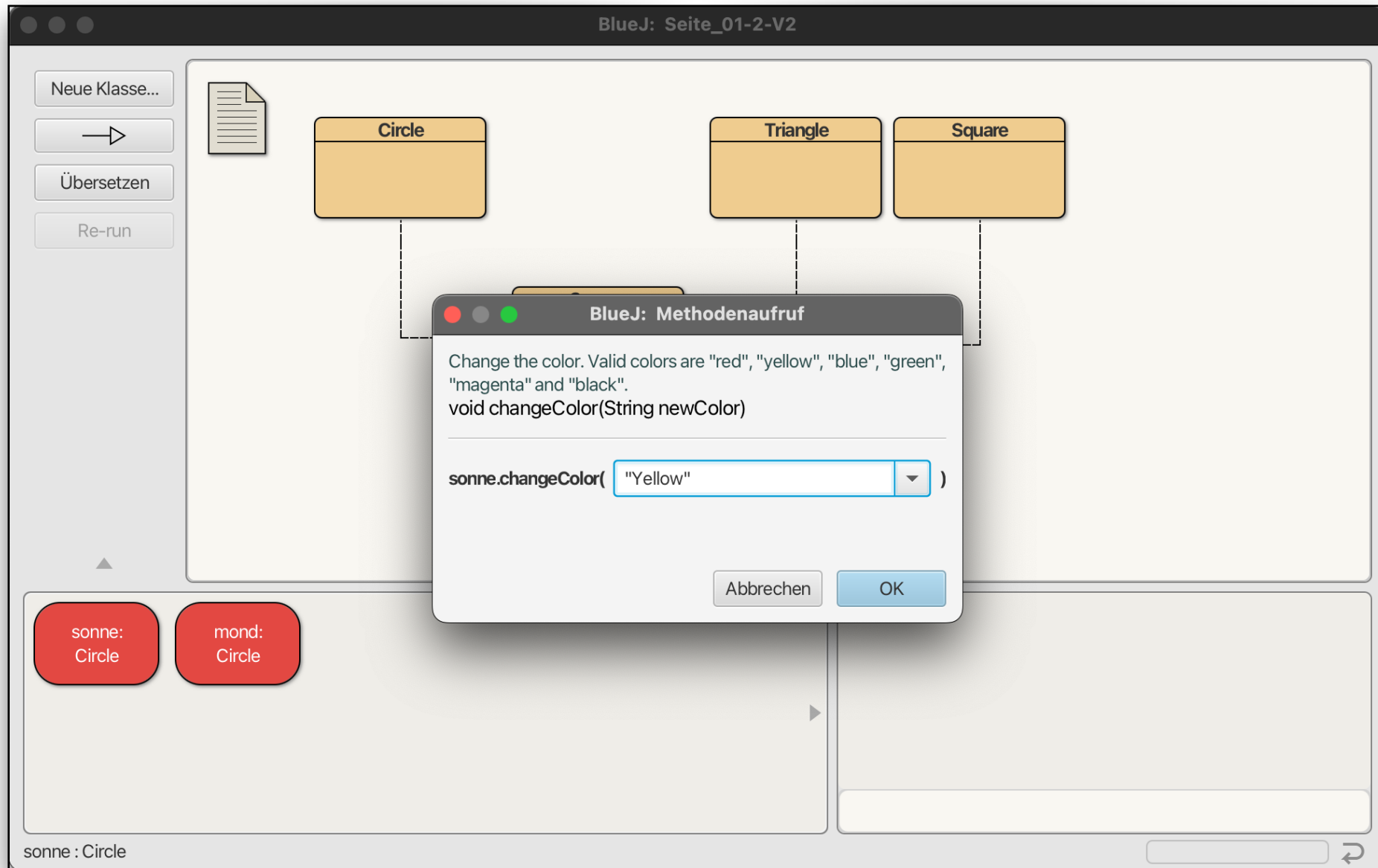
Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



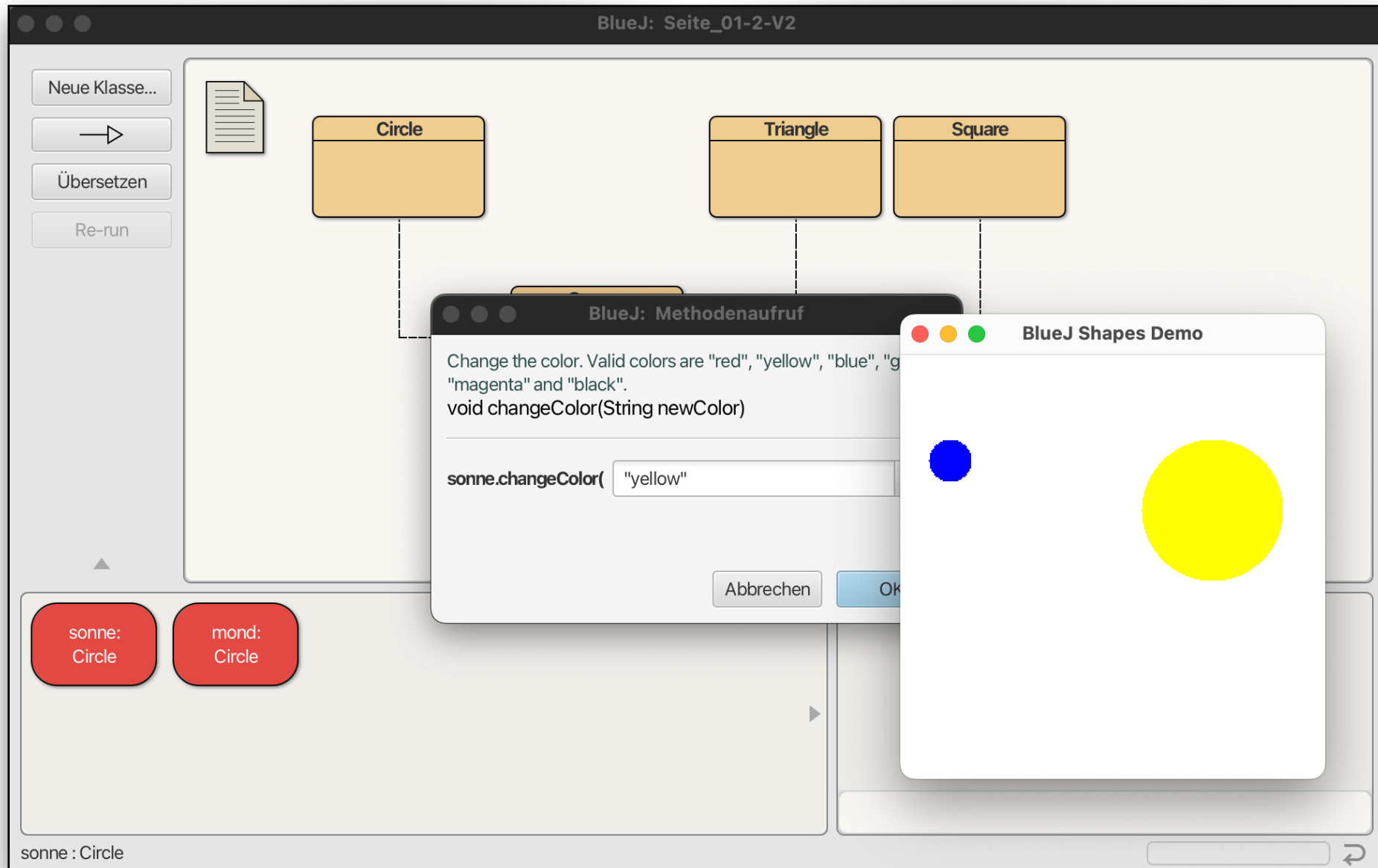
Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden



Einführung in die Objektorientierte Programmierung (OOP)

1.2.2 Arbeiten mit Methoden

