OOP

Mit Zahlen rechnen

Henrik Horstmann

15. September 2014

Inhaltsverzeichnis

1 Bedeutung der Symbole....1
2 Ein Taschenrechner zum Addieren....2
3 Die Benutzereingaben....3
4 Strings in Fließkommazahlen umwandeln....3
5 Mit Floats rechnen....4
6 Fließkommazahlen in Strings umwandeln....4
7 Der große Auftritt....5
7.1 Bühne und Taschenrechner....5
7.2 Das Ergebnis im Display anzeigen....6
8 Übungen....7
8.1 Flächeninhalt eines Dreiecks....7
8.2 Mittelwert berechnen....8
8.3 Sag mir wie spät es ist....9
8.4 Quadratische Gleichung....12
9 Lösungen zu den Aufgaben....13
9.1 Lösung zu Aufgabe 1....13
9.2 Lösung zu Aufgabe 2....14
9.3 Lösung zu Aufgabe 3....15
9.4 Lösung zu Aufgabe 4....16
9.5 Lösung zu Aufgabe 5....17
9.6 Lösung zu Aufgabe 6....18
9.7 Lösung zu Aufgabe 7....19
9.8 Lösung zu Aufgabe 10....20
9.9 Hilfe zur Umrechnung der Stunden....21
9.10 Hilfe zur Umrechnung der Minuten....22
9.11 Lösung zu Aufgabe 11....23
9.12 Lösung zu Aufgabe 12....24
9.13 Lösung zu Aufgabe 13....25
9.14 Lösung zu Aufgabe 14....26

1 Bedeutung der Symbole

Symbol	Beschreibung
	Dieses Symbol kennzeichnet einen Tipp oder Hinweis.
	Aufgepasst, hier passieren leicht Fehler oder es ist mit Schwierigkeiten zu rechnen.
	Hier erfahren Sie, wie es gemacht wird.
	Es folgt eine Aufgabe die zu lösen ist.

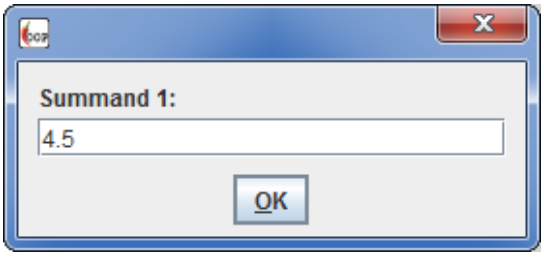
© 2014 Henrik Horstmann

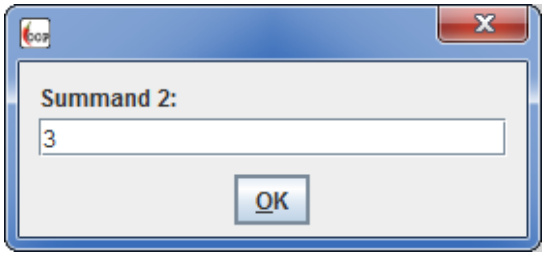
2 Ein Taschenrechner zum Addieren

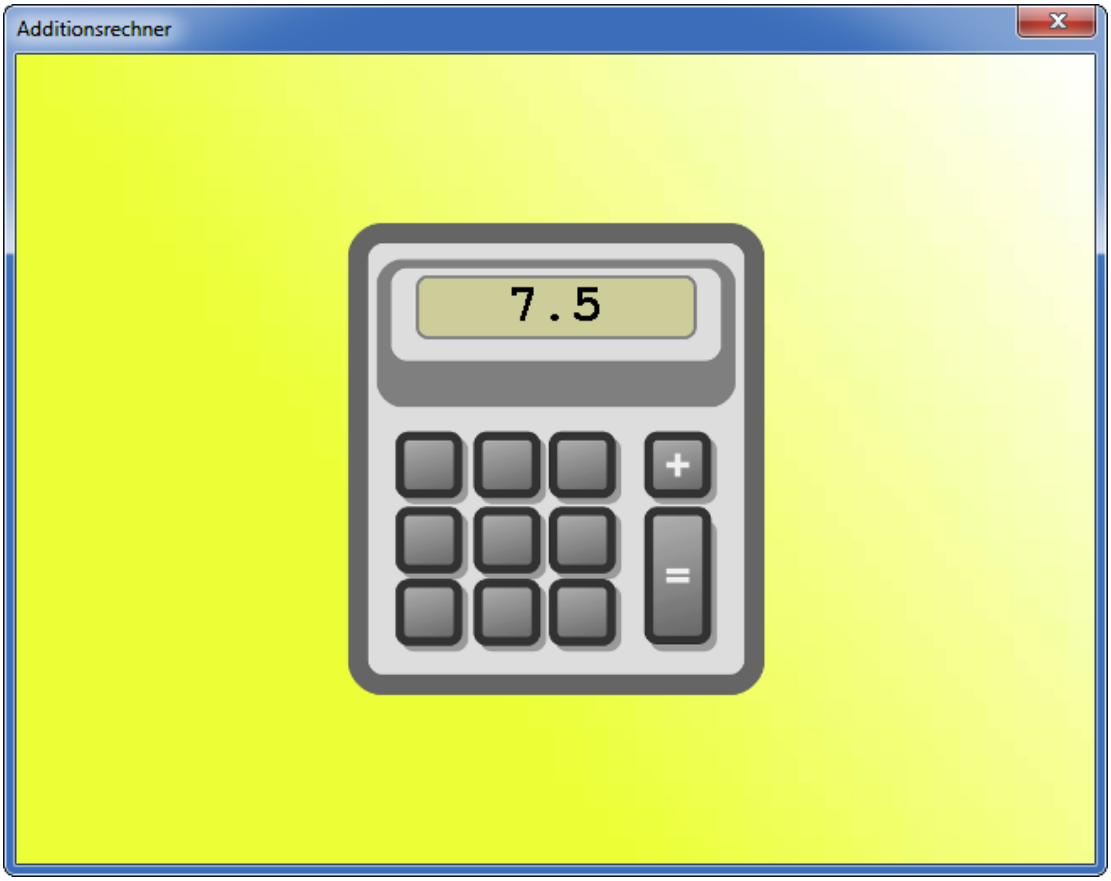
In diesem Tutorial kommt eine neue Klasse ins Spiel: *Float*. Objekte der Klasse *Float* repräsentieren Fließkommazahlen.

Die Funktionsweise der Klasse *Float* wird an einem einfachen Beispiel, einem Additionsrechner gezeigt.

Das Programm in diesem Tutorial erwartet vom Benutzer die Eingabe von zwei Summanden und gibt die Summe wieder aus.







3 Die Benutzereingaben

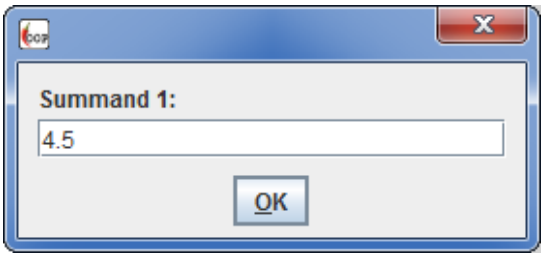
Aufgabe 1:

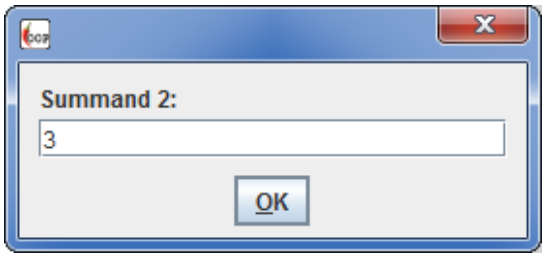
Öffnen Sie ein neues OOP-Projekt und speichern Sie es unter dem Namen *MitZahlenRechnen.oop*.

Legen Sie in der Methode *main* zwei Variablen *input1* und *input2* der Klasse *String* an. Programmieren Sie zwei Benutzerabfragen für die zwei Summanden und speichern Sie die Eingaben in den Variablen *input1* und *input2*.

Hier geht es zur Lösung auf Seite 13.

Wenn Sie Ihr Programm starten, sollten sich nacheinander zwei Dialogfenster öffnen und nach den Summanden fragen. Mehr passiert noch nicht.





4 Strings in Fließkommazahlen umwandeln

Bevor die beiden eingegebenen Zahlen addiert werden können, müssen sie in Fließkommazahlen umgewandelt werden.

Aufgabe 2:

1. Legen Sie zwei weitere Variablen *summand1* und *summand2* der Klasse *Float* an.
2. Erzeugen Sie für jede der Variablen *summand1* und *summand2* ein neues Objekt. Beim Erzeugen übergeben Sie einmal die Variable *input1* und einmal die Variable *input2* als Parameter: Beispiel: `summand1 = new Float(input1);`

Hier geht es zur Lösung auf Seite 14.

Hinweis:

Enthält *input1* oder *input2* Zeichen, die nicht zu einer Zahl gehören, so erzeugt `new Float(input1)` die Zahl 0. Beispiel:
`new Float("abc")` ergibt die Zahl 0.

Statt einem Komma muss ein Punkt eingegeben werden:

`new Float("1,5")` ergibt 0, da ',' ein unzulässiges Zeichen ist.

`new Float("1.5")` ergibt 1,5.

5 Mit Floats rechnen

Mit Objekten der Klasse *Float* kann gerechnet werden.

Beispiel:

```
...
Float f1;
Float f2;
f1 = new Float(1.5);
f2 = new Float(4.0);
// Addiere f1 und f2 und speichere das Ergebnis in f1
f1 = f2.add(f1);
```

Folgende Rechenoperationen stehen zur Verfügung:

Rechenoperation	Methode
Addition	`Float add(Float f)`
Subtraktion	`Float sub(Float f)`
Multiplikation	`Float mul(Float f)`
Division	`Float div(Float f)`
Wurzelziehen	`Float sqrt()`

Aufgabe 3:

1. Legen Sie eine weitere Variable *summe* der Klasse *Float* an.
2. Addieren Sie die beiden Werte aus *summand1* und *summand2* und weisen Sie das Ergebnis der Variablen *summe* zu.

Hier geht es zur Lösung auf Seite 15.

6 Fließkommazahlen in Strings umwandeln

Um die Summe anzeigen zu können, muss diese wieder in ein Objekt der Klasse String umgewandelt werden.

Beispiel:

```
...
Float f;
String s;
f = new Float(1.5);
s = f.toString();
```

Aufgabe 4:

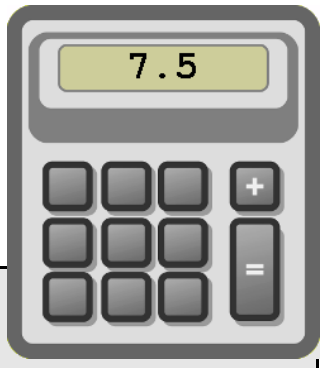
1. Lege eine Variable *output* der Klasse *String* an.
2. Rufe vom Objekt in der Variable *summe* die Methode *toString()* auf und weise das Ergebnis der Variablen *output* zu.
3. Geben Sie den String *output* auf der Konsole aus.

Testen Sie Ihr Programm.

Hier geht es zur Lösung auf Seite 16.

7 Der große Auftritt

Statt das Ergebnis auf der Konsole auszugeben, soll in einem Programmfenster ein Taschenrechner angezeigt werden, in dessen Display das Ergebnis erscheint.



7.1 Bühne und Taschenrechner

Aufgabe 5:

1. Legen Sie eine Variable *stage* der Klasse *Stage* an.
2. Erzeugen Sie ein neues Objekt der Klasse *Stage* mit dem Titel "Additionsrechner", der Breite von 640 Pixel und einer Höhe von 480 Pixel. Weisen Sie das neue Objekt der Variable *stage* zu.
3. Laden Sie in den Hintergrund der Bühne die Grafik *bg_yellow.jpg*.
4. Öffnen Sie das Fenster von *stage* mit der Methode *start()*.

Hier geht es zur Lösung auf Seite 17.

Aufgabe 6:

1. Legen Sie eine Variable *calculator* der Klasse *StageObject* an.
2. Erzeugen Sie ein neues Objekt der Klasse *StageObject* mit folgenden Parametern:

   name = "Additionsrechner"
   x = 320
   y = 240
   resName = "Calculator.png"
   scaleFactor = 100

   Weisen Sie das neue Objekt der Variable *calculator* zu.
3. Platzieren Sie den Additionsrechner mit der Methode *addStageObject(StageObject obj)* auf der Bühne *stage*.

Hier geht es zur Lösung auf Seite 18.

© 2014 Henrik Horstmann

7.2 Das Ergebnis im Display anzeigen

Um Das Ergebnis im Display anzuzeigen wählen wir ein transparentes Objekt, beschriften es mit dem Ergebnis der Addition und platzieren es über dem Taschenrechner, so dass es aussieht, als wenn in der Anzeige das Rechenergebnis stehen würde.

Aufgabe 7:

1. Lege eine Variable *display* der Klasse *StageObject* an.
2. Erzeugen Sie ein neues Objekt der Klasse *StageObject* mit folgenden Parametern:

   name = "Anzeige"
   x = 320
   y = 137
   resName = "Transparenz.png"
   scaleFactor = 100

   Weisen Sie das neue Objekt der Variable *dislay* zu.
3. Beschriften Sie das Display in der Variable *display* mit den String aus *output*.
4. Platzieren Sie das Display mit der Methode *addStageObject(StageObject obj)* auf der Bühne *stage*.

Testen Sie Ihr Programm.

Hier geht es zur Lösung auf Seite 19.

Damit ist der Additionsrechner fertig.

Auf der nächsten Seite geht es weiter mit Übungen.

8 Übungen

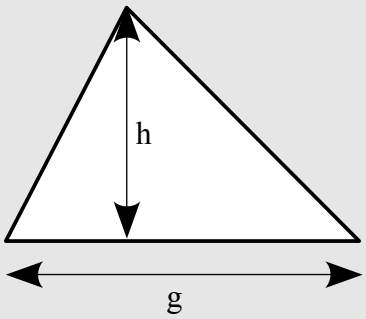
8.1 Flächeninhalt eines Dreiecks

Das Programm *Dreieck.oop* soll so vervollständigt werden, dass es den Flächeninhalt eines Dreiecks berechnet. Dazu werden vom Benutzer die Länge der Grundseite und Höhe des Dreiecks erfragt.

Tipp:

Der Flächeninhalt A eines Dreiecks berechnet sich wie folgt:

$$A=\frac{h\cdot g}{2}$$



Aufgabe 8:

1. Öffnen Sie die Projektdatei *Dreieck.oop*.
2. Fügen Sie zwischen den Kommentarzeilen

```
// Beginn der Verechnung des Flaecheninhalts
// Ende der Verechnung des Flaecheninhalts
```

   die Berechnung des Flächeninhalts ein. Die Werte g und h befinden sich in den Variablen *g* und *h*. Das Ergebnis soll in der Variablen *a* gespeichert werden.
3. Testen Sie Ihr Programm mit folgenden Werten:

g	h	A
3.0	4.0	6.0
5.0	3.0	7.5
13.0	7.0	45.5

8.2 Mittelwert berechnen

Das Programm *Mittelwert.oop* soll so vervollständigt werden, dass es den Mittelwert von drei Zahlen berechnet.

Tipp:

Der Mittelwert m von drei Werten f_1, f_2 und f_3 berechnet sich wie folgt:

$$m=\frac{f_1+f_2+f_3}{3}$$

Aufgabe 9:

1. Öffnen Sie die Projektdatei *Mittelwert.oop*.
2. Fügen Sie zwischen den Kommentarzeilen

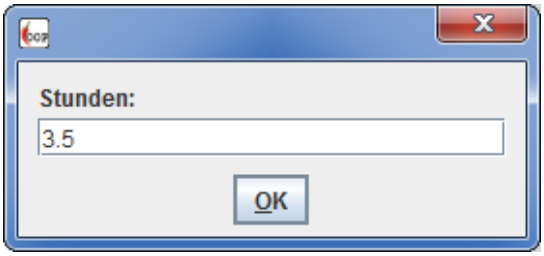
```
// Beginn der Mittelwertberechnung
// Ende der Mittelwertberechnung
```

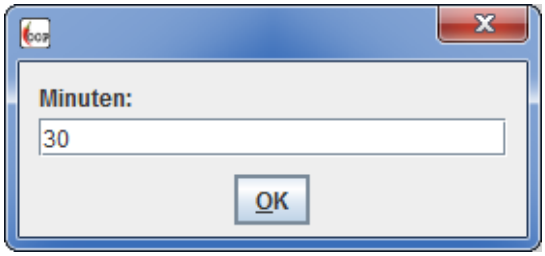
   die Berechnung des Mittelwerts ein. Die Werte f_1, f_2 und f_3 befinden sich in den Variablen *f1*, *f2* und *f3*. Das Ergebnis soll in der Variablen *m* gespeichert werden.
3. Testen Sie Ihr Programm mit folgenden Werten:

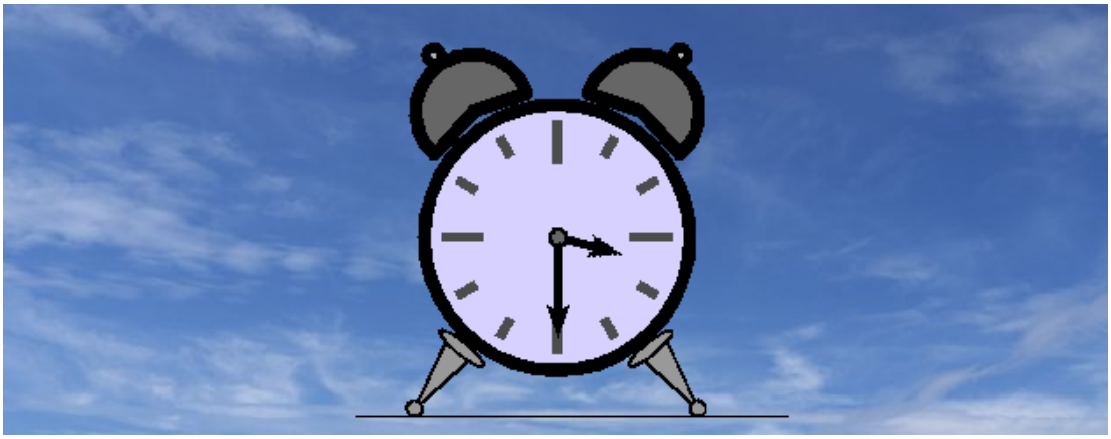
f_1	f_2	f_3	m
2	3	4	3.0
1.2	1.6	1.7	1.5
-2	1	2.5	0.5

8.3 Sag mir wie spät es ist

Im folgenden sollen Sie ein Programm *Wecker.oop* erstellen, das vom Benutzer eine Uhrzeit in Stunden und Minuten abfragt und diese dann mit einer analogen Uhr anzeigt.







Aufgabe 10:

1. Erstellen Sie ein neues Projekt *Wecker.oop*.
2. Fügen Sie der *main* Methode zwei Variablen *stunden* und *minuten* der Klasse *Float* hinzu. Als erstes soll die Methode zwei Werte abfragen, die in den Variablen *stunden* und *minuten* gespeichert werden.

Hier geht es zur Lösung auf Seite 20.

Aufgabe 11:

1. Rechnen Sie die Stundenangabe und die Minutenangabe in Grad um. Versuchen Sie die Umrechnung selbstständig zu lösen. Benötigen Sie trotzdem Hilfe, so finden Sie auf Seite 21 Hilfe zur Umrechnung der Stunden und auf Seite 22 Hilfe zur Umrechnung der Minuten.

Hier geht es zur Lösung auf Seite 23.

Aufgabe 12:

1. Fügen Sie der *main* Methode eine Variable *stage* der Klasse *Stage* zu.
2. Erzeugen Sie ein neues *Stage* Objekt mit 640×480 Pixel und weisen Sie es der Variablen *stage* zu. Laden Sie als Hintergrundbild das Image *sky.jpg*.
3. Öffnen Sie das "Bühnenfenster" mit *start()*.

Hier geht es zur Lösung auf Seite 24.

Aufgabe 13:

1. Fügen Sie der *main* Methode drei Variablen *wecker*, *stundenzeiger* und *minutenzeiger* der Klasse *StageObjekt* zu.
2. Erzeugen Sie ein neues Objekt der Klasse *StageObject* mit folgenden Parametern:

   name = "Wecker"
   x = 320
   y = 240
   resName = "Wecker.png"
   scaleFactor = 50

   und weisen Sie es der Variable *wecker* zu. Fügen Sie der *stage* mit *addStageObject(obj)* das StageObjekt *wecker* zu.

4. Erzeugen Sie ein neues Objekt der Klasse *StageObject* mit folgenden Parametern:

   name = "Minutenzeiger"
   x = 320
   y = 240
   resName = "Minutenzeiger.png"
   scaleFactor = 50

   und weisen Sie es der Variable *minutenzeiger* zu. Fügen Sie der *stage* mit *addStageObject(obj)* das StageObjekt *minutenzeiger* zu.

5. Erzeugen Sie ein neues Objekt der Klasse *StageObject* mit folgenden Parametern:

   name = "Stundenzeiger"
   x = 320
   y = 240
   resName = "Stundenzeiger.png"
   scaleFactor = 50

© 2014 Henrik Horstmann

und weisen Sie es der Variable stnden*zeiger* zu. Fügen Sie der *stage* mit *addStageObject(obj)* das StageObjekt *stundenzeiger* zu.

Hier geht es zur Lösung auf Seite 25.

Tipp:

Ein Objekt der Klasse *StageObject* kann mit folgender Methode gedreht werden:

```
void setRotation(Float angle)
```

Dabei ist *angle* die Gradzahl, um die das Objekt gedreht werden soll.

Aufgabe 14:

1. Drehen Sie Stunden- und Minutenzeiger entsprechend der Werte in den Variablen *stunden* und *minuten*. Verwenden Sie dazu die Methode *setRotation* für Objekte der Klasse *StageObjekt*.
2. Testen Sie Ihr Programm.

Hier geht es zur Lösung auf Seite 26.

8.4 Quadratische Gleichung

Das Programm *Loesungsformel.oop* soll so vervollständigt werden, dass es Gleichungen der Form $a x^2 + b x + c = 0$ lösen kann. De Koeffizienten a, b und c werden vom Benutzer abgefragt.

Tipp:

Die Lösungen einer Gleichung $a x^2 + b x + c = 0$ berechnet sich wie folgt:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Tipp:

$$b^2 = b \cdot b$$

Aufgabe 15:

1. Öffnen Sie die Projektdatei *Loesungsformel.oop*.
2. Fügen Sie zwischen den Kommentarzeilen

```
// Beginn der Berechnung von x1 und x2
// Ende der Berechnung von x1 und x2
```

   die Berechnung der Lösungen der Gleichung ein. Die Werte a, b und c befinden sich in den Variablen *a*, *b* und *c*. Das Ergebnis soll in den Variablen *x1* und *x2* gespeichert werden.
3. Testen Sie Ihr Programm mit folgenden Werten:

a	b	c	x_1	x_2
2	0	-8	2	-2
1	-3	0	3	0
3	9	-12	1	-4

Tipp:

Nutzen Sie zusätzliche Variablen um Zwischenergebnisse zu speichern.

ENDE

9 Lösungen zu den Aufgaben

9.1 Lösung zu Aufgabe 1

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String input2;
      String input1;
      Computer comp;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
   }
}
```

Zurück zur Aufgabe auf Seite 3.

9.2 Lösung zu Aufgabe 2

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
   }
}
```

Zurück zur Aufgabe auf Seite 3.

9.3 Lösung zu Aufgabe 3

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      Float summe;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
      // Summe berechnen
      summe = summand1.add(summand2);
   }
}
```

Zurück zur Aufgabe auf Seite 4.

9.4 Lösung zu Aufgabe 4

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String output;
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      Float summe;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
      // Summe berechnen
      summe = summand1.add(summand2);
      // Ergebnis für die Ausgabe in einen String umwandeln
      output = summe.toString();
      // Ergebnis auf der Konsole ausgeben
      comp.print(output);
   }
}
```

Zurück zur Aufgabe auf Seite 5.

9.5 Lösung zu Aufgabe 5

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String output;
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      Stage stage;
      Float summe;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
      // Summe berechnen
      summe = summand1.add(summand2);
      // Ergebnis für die Ausgabe in einen String umwandeln
      output = summe.toString();
      // Ergebnis auf der Konsole ausgeben
      comp.print(output);
      // Stage Objekt erzeugen und gelben Hintegrund laden
      stage = new Stage("Additionsrechner",640,480);
      stage.loadBackground("bg_yellow.jpg");
      stage.start();
   }
}
```

Zurück zur Aufgabe auf Seite 5.

9.6 Lösung zu Aufgabe 6

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      String output;
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      Stage stage;
      StageObject calculator;
      Float summe;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
      // Summe berechnen
      summe = summand1.add(summand2);
      // Ergebnis für die Ausgabe in einen String umwandeln
      output = summe.toString();
      // Ergebnis auf der Konsole ausgeben
      comp.print(output);
      // Stage Objekt erzeugen und gelben Hintegrund laden
      stage = new Stage("Additionsrechner",640,480);
      stage.loadBackground("bg_yellow.jpg");
      stage.start();
      // Taschenrechnergrafik laden und auf der Stage positionieren
      calculator = new StageObject("Additionsrechner",320,240,
                                   "Calculator.png",100);
      stage.addStageObject(calculator);
   }
}
```

Zurück zur Aufgabe auf Seite 5.

9.7 Lösung zu Aufgabe 7

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute

   public Program ()
   {
   }

   public void main ()
   {
      // Anfang Variablen
      String output;
      String input2;
      String input1;
      Computer comp;
      Float summand2;
      Float summand1;
      Stage stage;
      StageObject display;
      StageObject calculator;
      Float summe;
      // Ende Variablen

      // Objekt Computer erzeugen
      comp = new Computer();
      // Einlesen der Summanden
      input1 = comp.read("Summand 1:");
      input2 = comp.read("Summand 2:");
      // Eingaben in Float Werte umwandeln
      summand1 = new Float(input1);
      summand2 = new Float(input2);
      // Summe berechnen
      summe = summand1.add(summand2);
      // Ergebnis für die Ausgabe in einen String umwandeln
      output = summe.toString();
      // Ergebnis auf der Konsole ausgeben
      comp.print(output);
      // Stage Objekt erzeugen und gelben Hintegrund laden
      stage = new Stage("Additionsrechner",640,480);
      stage.loadBackground("bg_yellow.jpg");
      stage.start();
      // Taschenrechnergrafik laden und auf der Stage positionieren
      calculator = new StageObject("Additionsrechner",320,240,
                                   "Calculator.png",100);
      stage.addStageObject(calculator);
      // Das Rechenergebnis im Display des Tascherchners anzeigen.
      display = new StageObject("Anzeige",320,137,
                                "Transparenz.png",100);
      display.setText(output,32);
      stage.addStageObject(display);
   }
}
```

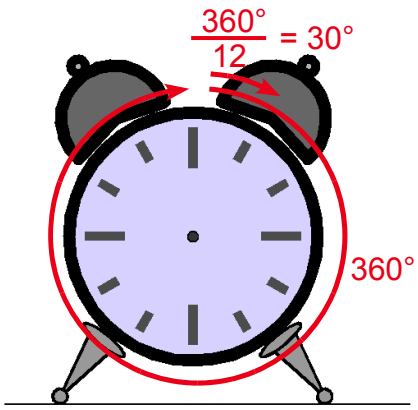
Zurück zur Aufgabe auf Seite 6.

9.8 Lösung zu Aufgabe 10

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      Float stunden;
      Computer computer;
      Float minuten;
      String str;
      // Ende Variablen

      // Benutzereingaben für die Zeit in Stunden und Minuten.
      computer = new Computer();
      str = computer.read("Stunden:");
      stunden = new Float(str);
      str = computer.read("Minuten:");
      minuten = new Float(str);
   }
}
```

Zurück zur Aufgabe auf Seite 9.

9.9 Hilfe zur Umrechnung der Stunden

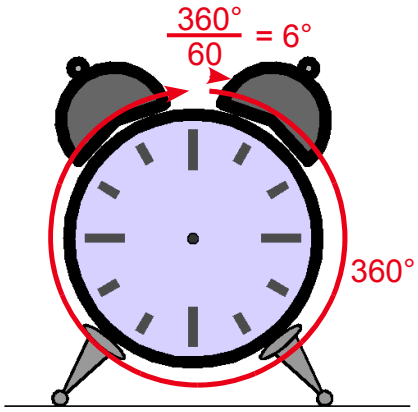


Für die Stunden werden Wert zwischen 0 und 11 eingegeben. Das bedeutet das der Winkel des Stundenzeigers sich durch folgende Formel ergibt:

$$winkel = stunden \cdot 30$$

Zurück zur Aufgabe auf Seite 9.

9.10 Hilfe zur Umrechnung der Minuten



Um den Minutenzeiger um eine Minute weiter zu bewegen, muss er um 6° gedreht werden.

Für die Minuten werden Wert zwischen 0 und 59 eingegeben. Das bedeutet das der Winkel des Minutenzeigers sich durch folgende Formel ergibt:

$$winkel = minuten \cdot 6$$

Zurück zur Aufgabe auf Seite 9.

9.11 Lösung zu Aufgabe 11

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      Float stunden;
      Computer computer;
      Float minuten;
      String str;
      // Ende Variablen

      // Benutzereingaben für die Zeit in Stunden und Minuten.
      computer = new Computer();
      str = computer.read("Stunden:");
      stunden = new Float(str);
      str = computer.read("Minuten:");
      minuten = new Float(str);
      // Drehung der Uhrzeiger in Grad berechnen:
      minuten = minuten.mul(6.0);
      stunden = stunden.mul(30.0);
   }
}
```

Zurück zur Aufgabe auf Seite 9.

9.12 Lösung zu Aufgabe 12

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      Float stunden;
      Stage stage;
      Computer computer;
      Float minuten;
      String str;
      // Ende Variablen

      // Benutzereingaben für die Zeit in Stunden und Minuten.
      computer = new Computer();
      str = computer.read("Stunden:");
      stunden = new Float(str);
      str = computer.read("Minuten:");
      minuten = new Float(str);
      // Drehung der Uhrzeiger in Grad berechnen:
      minuten = minuten.mul(6.0);
      stunden = stunden.mul(30.0);
      // Die Stage aufbauen.
      stage = new Stage("Wecker",640,480);
      stage.loadBackground("sky.jpg");
      stage.start();
   }
}
```

Zurück zur Aufgabe auf Seite 10.

9.13 Lösung zu Aufgabe 13

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      Float stunden;
      Stage stage;
      Computer computer;
      Float minuten;
      String str;
      // Ende Variablen

      // Benutzereingaben für die Zeit in Stunden und Minuten.
      computer = new Computer();
      str = computer.read("Stunden:");
      stunden = new Float(str);
      str = computer.read("Minuten:");
      minuten = new Float(str);
      // Drehung der Uhrzeiger in Grad berechnen:
      minuten = minuten.mul(6.0);
      stunden = stunden.mul(30.0);
      // Die Stage aufbauen.
      stage = new Stage("Wecker",640,480);
      stage.loadBackground("sky.jpg");
      stage.start();
      wecker = new StageObject("Wecker",320,240,"Wecker.png",50);
      stage.addStageObject(wecker);
      minutenzeiger = new StageObject("Minutenzeiger",320,240,
                                      "Minutenzeiger.png",50);
      stage.addStageObject(minutenzeiger);
      stundenzeiger = new StageObject("Stundenzeiger",320,240,
                                      "Stundenzeiger.png",50);
      stage.addStageObject(stundenzeiger);
   }
}
```

Zurück zur Aufgabe auf Seite 10.

9.14 Lösung zu Aufgabe 14

```
public class Program
{
   // Anfang Attribute
   // Ende Attribute


   public Program ()
   {
      // Anfang Variablen
      // Ende Variablen

   }

   public void main ()
   {
      // Anfang Variablen
      Float stunden;
      Stage stage;
      Computer computer;
      Float minuten;
      String str;
      // Ende Variablen

      // Benutzereingaben für die Zeit in Stunden und Minuten.
      computer = new Computer();
      str = computer.read("Stunden:");
      stunden = new Float(str);
      str = computer.read("Minuten:");
      minuten = new Float(str);
      // Drehung der Uhrzeiger in Grad berechnen:
      minuten = minuten.mul(6.0);
      stunden = stunden.mul(30.0);
      // Die Stage aufbauen.
      stage = new Stage("Wecker",640,480);
      stage.loadBackground("sky.jpg");
      stage.start();
      wecker = new StageObject("Wecker",320,240,"Wecker.png",50);
      stage.addStageObject(wecker);
      minutenzeiger = new StageObject("Minutenzeiger",320,240,
                                      "Minutenzeiger.png",50);
      stage.addStageObject(minutenzeiger);
      stundenzeiger = new StageObject("Stundenzeiger",320,240,
                                      "Stundenzeiger.png",50);
      stage.addStageObject(stundenzeiger);
      // Zeiger drehen:
      minutenzeiger.setRotation(minuten);
      stundenzeiger.setRotation(stunden);
   }
}
```

Zurück zur Aufgabe auf Seite 11.