O	O	Р

Ein erstes "Hello world!" Programm

Henrik Horstmann

14. September 2014

Inhaltsverzeichnis

1 Bedeutung der Symbole	1
2 Die Benutzer Oberfläche von HOOPLU	2
2.1 Projekte öffnen und speichern	2
2.2 Die Klasse Program und die Methode main	2
3 Das Programm "Hello world!"	3
3.1 Der Programmcode	3
3.2 Einfügen einer Variablen	4
3.3 Ein Objekt erzeugen	6
3.4 Einfügen eines Kommentars	6
3.5 Eine Methode aufrufen	7
4 Das Programm ausführen	10

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.

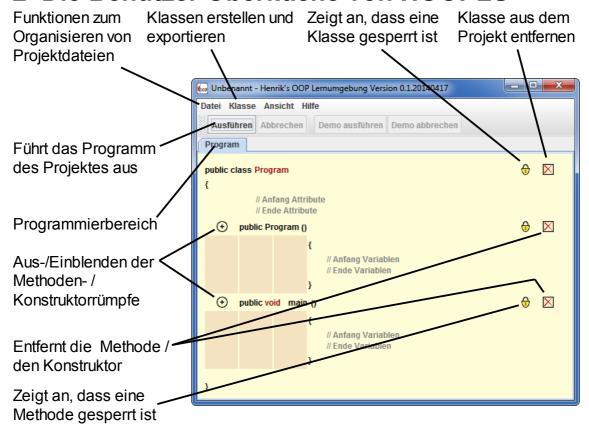


Hier erfahren Sie, wie es gemacht wird.



Es folgt eine Aufgabe die zu lösen ist.

2 Die Benutzer Oberfläche von HOOPLU



2.1 Projekte öffnen und speichern

Im Menü *Datei* finden sich alle Menüpunkte zum Verwalten von Projekten.



Hinweis:

Beim Programmieren sollte in regelmäßigen Abständen das Projekt gespeichert werden um einem möglichen Datenverlust vorzubeugen.



2.2 Die Klasse Program und die Methode main

Jedes Projekt in HOOPLU enthält die Klasse *Program* mit der Methode *main*. Weder die Klasse, noch die Methode können entfernt werden.

Wird ein Programm über den *Ausführen* Button gestartet, so erzeugt HOOPLU ein Objekt der Klasse *Program* und ruft als erstes die Methode *main* auf.

3 Das Programm "Hello world!"

Das Programm soll auf der *Konsole* Text "Hello world!" ausgeben. Um Text auf der *Konsole* auszugeben wird ein Objekt der Klasse *Computer* benötigt. Die Klasse *Computer* ist in HOOPLU standardmäßig enthalten (so wie die Klasse *Program*).

3.1 Der Programmcode

```
public class Program
   // Anfang Attribute
   // Ende Attribute
  public Program ()
      // Anfang Variablen
      // Ende Variablen
   }
   public void main ()
      // Anfang Variablen
      Computer comp;
      // Ende Variablen
      // 1. Ein Objekt fuer die Variable
      //
          comp erzeugen.
      comp = new Computer();
      // 2. Methode print des Objektes in comp
      //
            aufrufen, um den Text "Hello world!"
            auszugeben.
      comp.print("Hello world!");
```



Hinweis:

Bevor ein Objekt verwendet werden kann muss es mit **new** erzeugt werden. Folgende Codezeilen würden zu einem Fehler führen:

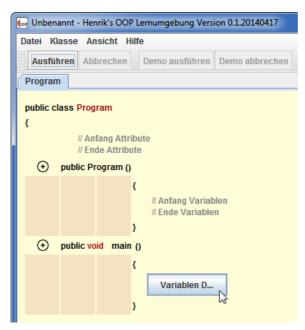
```
comp.print("Hello world!");
comp = new Computer();
```

3.2 Einfügen einer Variablen

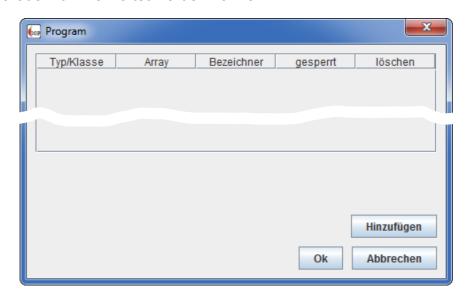
Nach der Kopfzeile von Konstruktoren und Methoden ist im Rumpf an oberster Stelle der ein Kommentar eingefügt:

```
// Anfang Variablen
// Ende Variablen
```

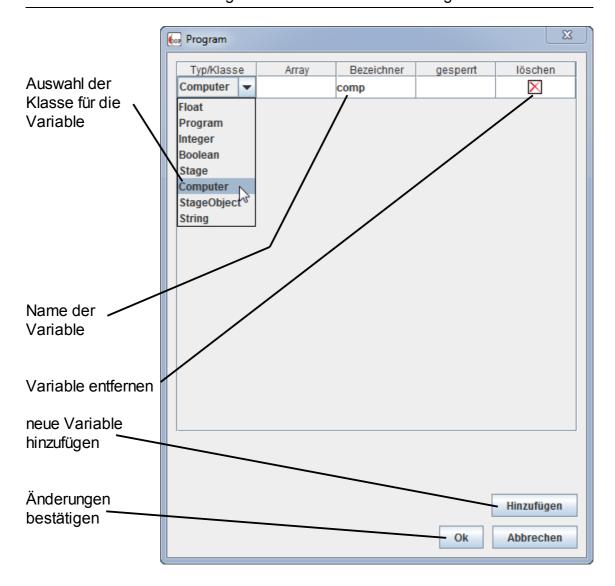
Fährt der Mauszeiger auf diesen Bereich im Programmierbereich, so erscheint ein Button.



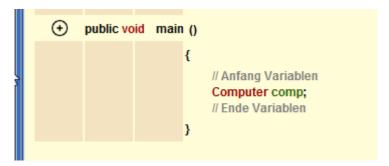
Ein Mausklick auf diesen Button öffnet ein Dialogfenster in dem die Variablen der Methode *main* verwaltet werden können.



Mit einem Mausklick auf den Button Hinzufügen wird eine neue Variable eingefügt:

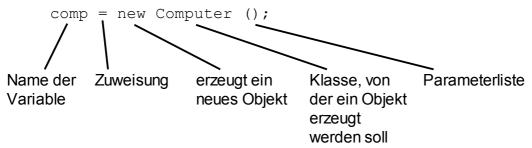


Mit einem Mausklick auf den Button *OK* wird der Dialog wieder geschlossen Im Programmbereich sind die Änderungen anschließend sichtbar:



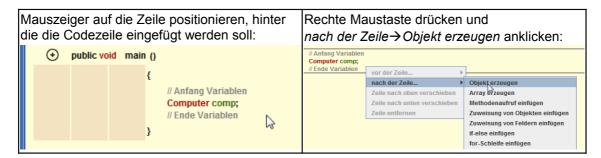
3.3 Ein Objekt erzeugen

Bevor ein Objekt verwendet werden kann muss es "zum Leben erweckt" werden. Dies geschieht mit dem **new** Operator. Das erzeugte Objekt wird z.B. einer Variablen mit dem '=' zugewiesen.



Einfügen der Programmzeile in HOOPLU:

Den Mauszeiger auf die Programmzeile positionieren, auf der die neue Codezeile folgen soll und die rechte Maustaste drücken. ein Kontextmenü öffnet sich. Den Eintrag nach der Zeile → Objekt erzeugen anklicken:



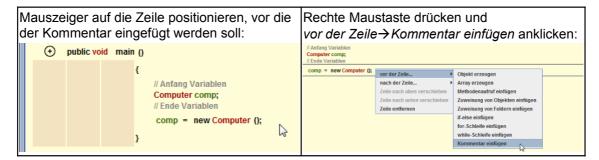
Anschließend ist die Codezeile im Methodenrumpf eingefügt:

3.4 Einfügen eines Kommentars

Damit der Programmcode verständlich bleibt, sollte er mit Kommentaren versehen werden. Kommentare sind Codezeilen, die bei der Ausführung des Programms ignoriert werden. Kommentare beginnen immer mit einem // und enden mit dem Zeilenende:

```
// 1. Ein Objekt fuer die Variable
// comp erzeugen.
```

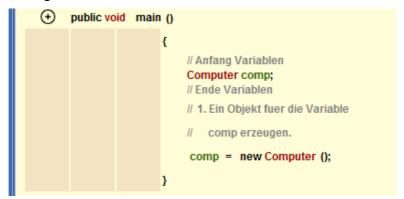
Einfügen einer Kommentarzeile in HOOPLU:



Anschließend ist eine leere Kommentarzeile in den Programmcode eingefügt. Wird der Mauszeiger auf die Kommentarzeile positioniert, so erscheint ein Textfeld, in dem der Kommentar eingetragen wird:

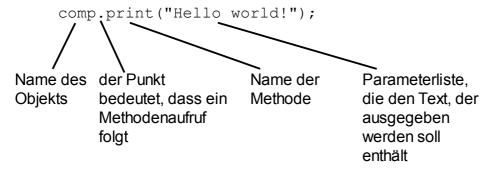
```
// Anfang Variablen
Computer comp;
// Ende Variablen
// _____
comp = new Computer ();
```

Diese Schritte werden für die zweite Kommentarzeile wiederholt, so dass das Programm wie folgt aussieht:

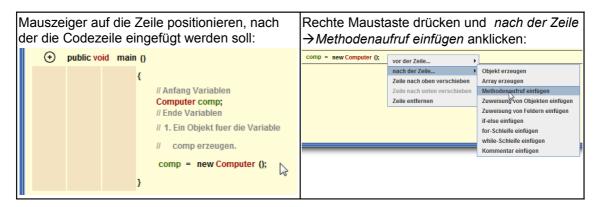


3.5 Eine Methode aufrufen

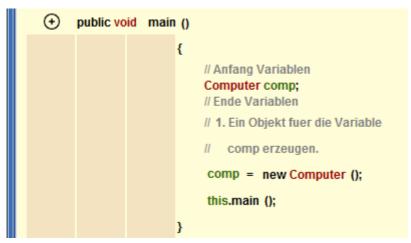
Im nächsten Schritt soll die Methode *print* des Objekts *comp* zur Ausgabe des Textes "Hello world!" aufgerufen werden. Der Programmcode dazu sieht wie folgt aus:



Einfügen der Programmzeile in HOOPLU:



Anschließend ist ein Methodenaufruf eingefügt. Allerdings ist es nicht der gewünschte Methodenaufruf:



Die Variable *this* muss durch *comp* ersetzt werden. Dazu wird der Mauszeiger auf den Variablen Namen *this* positioniert. Eine Liste der zur Auswahl stehenden Variablen wird eingeblendet. Mit einem Mausklick auf *comp* ist die Variable ausgewählt:



Fehlt nur noch der Parameter, mit dem der Text an die Methode *print* übergeben wird. Dazu den Mauszeiger auf die Parameterliste positionieren. Ein Button erscheint:



Nach einem Mausklick auf den Button öffnet sich folgendes Dialogfenster:



Mit einem Mausklick auf den Auswahlbutton (siehe Abbildung), erscheint ein Textfeld, in dem der Text direkt eingeben werden kann:



Mit einem Mausklick auf Ok die Eingabe bestätigen.



Wie in Abschnitt 3.4 beschrieben wird die Programmzeile dokumentiert:



4 Das Programm ausführen

Um das Programm auszuführen muss lediglich mit der Maus auf den Button *Ausführen* geklickt werden:



Ein Dialog zum Speichern des Projektes öffnet sich:



Es ist ein Verzeichnis auszuwählen und ein Name für das Projekt zu bestimmen. Die Angaben werden mit einem Mausklick auf den Button Speichern bestätigt.

Danach startet das Programm und auf der Konsole ist die Ausgabe zu sehen:



ENDE