

Abb. 3–33 // Eclipse-IDE: Ausgabe ohne gekoppelten EV3-Brick

Nach erfolgter Kopplung (siehe Kapitel 3.4) ist die Ausgabe folgendermaßen:

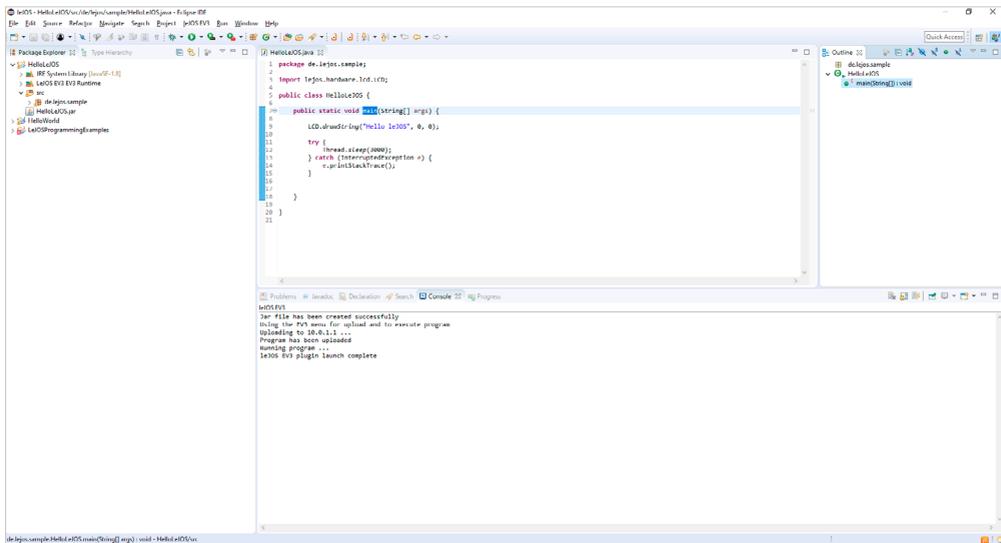


Abb. 3–34 // Eclipse-IDE: Ausgabe mit gekoppeltem EV3-Brick

Grundsätzlich lässt sich die Eclipse-IDE auch mit einem deutschen Sprachpaket ausstatten, sodass alle Menüs und Hinweistexte in Deutsch erscheinen. Dies lässt sich (wie auch das leJOS-Plug-in) über »Help« – »Install New Software« installieren. Nachdem allerdings für leJOS keine deutsche Lokalisierung existiert, wird von diesem Installationsschritt abgesehen.

3.3.4 Kurze Einführung in die Objektorientierung mit Java

In diesem Kapitel wird eine kurze Einführung zur objektorientierten Programmierung gegeben. Dabei wird auf einige wenige der wichtigsten Begriffe eingegangen. Es sollen hier nur die wichtigsten Grundlagen vermittelt werden, die für die direkte Programmierung des EV3 notwendig sind. Es ist nicht das Ziel, eine vollständige Vermittlung der Prinzipien der Objektorientierung zu versuchen, da hierfür ganze Bücher gefüllt werden.

Bei der prozeduralen Programmierung, wie sie zum Beispiel bei Small Basic verwendet wird, wird das Programm inhaltlich durch Prozeduren (sprich Unterprogrammen) strukturiert, die nacheinander gerufen werden können.

Im Gegensatz dazu geht die Objektorientierung davon aus, dass ein Gesamtsystem sich durch Objekte beschreiben lässt, die sich durch Eigenschaften und Aktivitäten/Funktionen auszeichnen. Durch die Verknüpfung von Objekten wird dabei ein Gesamtsystem aufgebaut.

Ein sehr beliebtes Beispiel ist dabei ein Auto. Ein Auto besteht aus verschiedenen Teilen, die Eigenschaften haben und durch Aktivitäten das Gesamtsystem beeinflussen. Das Auto hat dabei einen Motor und mehrere Räder. Ein Rad kann sich drehen und wird durch die Aktivität des Motors beeinflusst. Die Türen haben die Möglichkeit, sich zu öffnen, und haben als sichtbare Eigenschaften eine Farbe. Diese wenigen Beispiele sollen verdeutlichen, dass sich im Grunde alle Systeme als Verbindungen von Objekten darstellen lassen, die sich gegenseitig beeinflussen beziehungsweise sich gegenseitig bedingen.

Die Objektorientierung erfordert somit auch eine andere Art und Weise der Herangehensweise an die Erstellung eines Programms im Vergleich zu einer prozeduralen Programmiersprache.

In Java ist auch ein prozeduraler Programmierstil möglich. Bei der Umsetzung mit leJOS wird in diesem Buch sogar öfter explizit dieser Stil angewendet, um die Vergleichbarkeit der Lösungen in den drei Programmiersprachen besser zu verdeutlichen. Erst in den letzten Kapiteln wird stärker auf die Objektorientierung eingegangen, und die vorgestellten Lösungen werden auch mit entsprechenden Diagrammen erläutert.

In der Objektorientierung sind einige wichtige Kernbegriffe vorhanden, die die Grundlagen bestimmen.

■ Klasse

Eine Klasse stellt die allgemeine Definition eines Objekts dar und beinhaltet die Eigenschaftsbeschreibungen und Aktivitäten in verallgemeinerter Form, ohne dass diesen konkrete Werte zugewiesen sein müssen. Diese sind als Programmcode vorhanden. Eine Klasse kann nicht direkt ausgeführt werden.

Ein Auto stellt dabei eine Beschreibung und Definition dar, das sowohl Eigenschaften (zum Beispiel Farbe) als auch Aktivitäten (zum Beispiel Starten des Motors) haben kann.

Eine Klasse sollte in Java mit einem Großbuchstaben beginnen.