

Übungen zur Vorlesung Software-Entwicklung

Erstellen und Ausführen eines Java-Programms.

In dieser Übungsstunde sollen Programme in der Programmiersprache Java erstellt, für den Ablauf vorbereitet und schließlich ausgeführt werden. Es werden zwei unterschiedliche Programme erstellt: das erste wird direkt auf dem Rechner ausgeführt und das zweite wird als Teil einer Internet-Seite durch einen Internet-Browser ausgeführt.

Das Ziel dieser Übung ist das Verstehen der technischen Abläufe, die zur Erstellung eines Programms notwendig sind. Es wird nicht erwartet, dass Sie anschließend Java programmieren können.

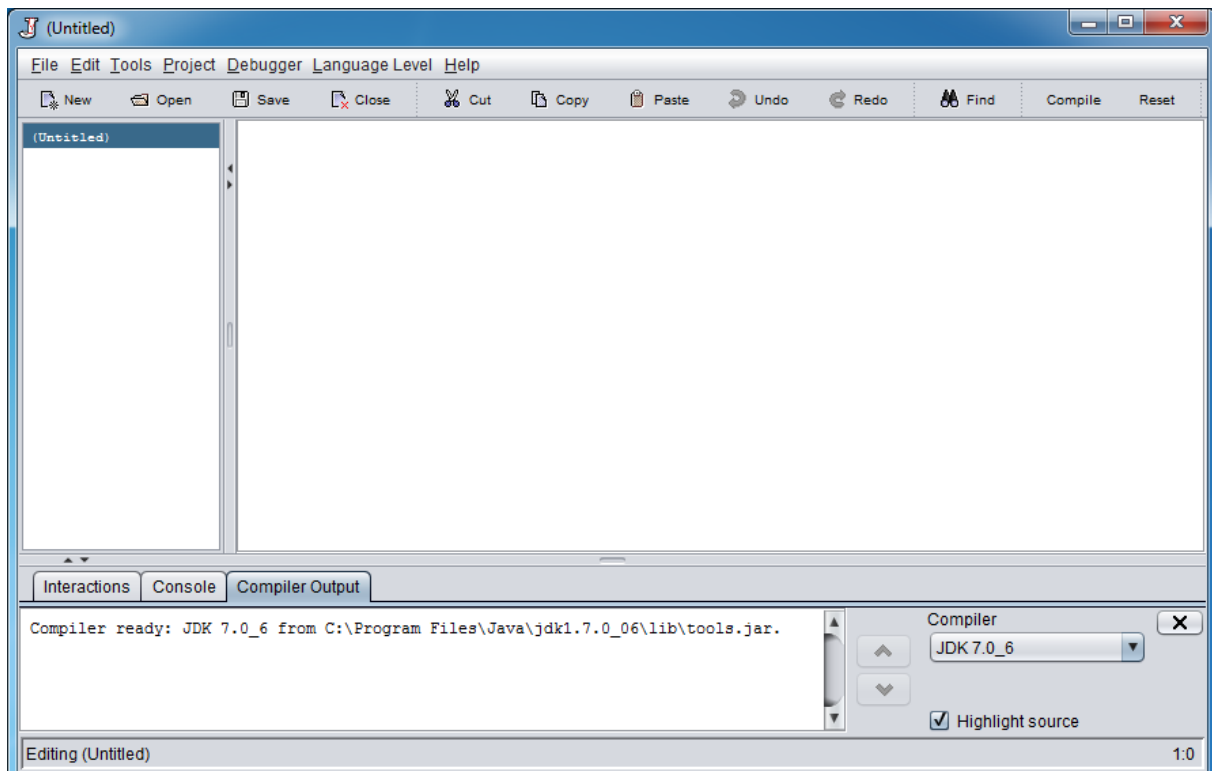
1. Voraussetzungen

- Der Rechner muss angeschaltet und hochgefahren sein.
- Man sollte als Benutzer auf dem Rechner eingeloggt sein.

2. Das Programm erstellen und abspeichern

Zur Programmerstellung wird ein einfacher Programmeditor verwendet. Editoren sind interaktive Systeme zum Erstellen, Modifizieren und Abspeichern von Inhalten unterschiedlichen Typs. Programme sind besondere Texte. Um sie zu erstellen und zu bearbeiten würde sogar ein einfacher Texteditor genügen. Wir verwenden hier „DrJava“, einen einfachen Programmeditor für die Programmiersprache Java.

Starten Sie den Programmeditor durch Doppelclick auf das DrJava-Piktogramm:

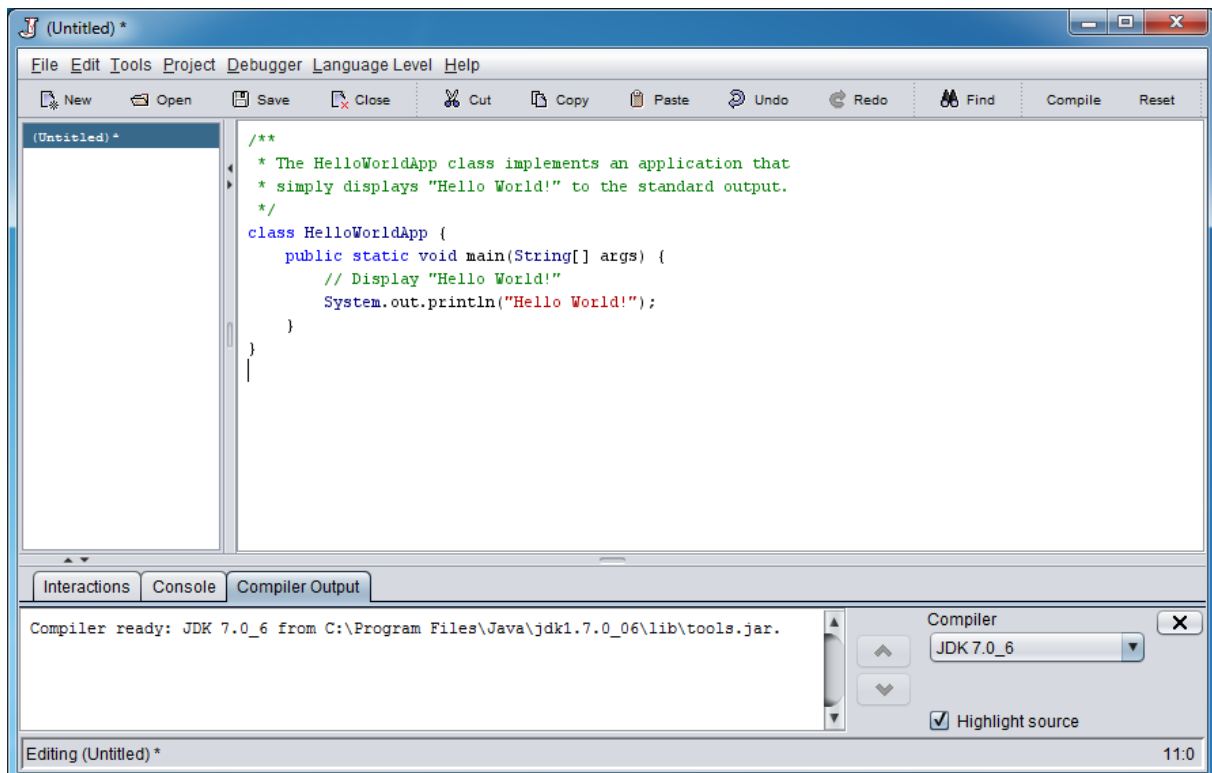


In den leeren Hauptbereich geben Sie den folgenden Text ein. Am besten, Sie tippen die wenigen Zeilen einfach ab.¹ Achten Sie darauf, dass Groß- und Kleinschreibung erhalten bleiben. Die Anzahl von Leerzeichen zur Einrückung ist dagegen nicht relevant.

```
/**
 * The HelloWorldApp class implements an application that
 * simply displays "Hello World!" to the standard output.
 */
class HelloWorldApp {
    public static void main(String[] args) {
        // Display "Hello World!"
        System.out.println("Hello World!");
    }
}
```

Dieser Programmtext ist ein Programm in der Programmiersprache Java. Er besteht aus fest vorgegebenen Symbolen, die in der Sprache Java eine festgelegte Bedeutung haben (z.B. die Symbole „class“, „{“, „}“, „public“ und „void“) und aus freien, vom Programmierer gewählten Wörtern (z.B. „HelloWorldApp“, „args“, „Hello“ und „World“). Die „Anordnung“ der Symbole erfolgt nach den Vorschriften (der Syntax) von Java und muss penibel und genau eingehalten werden. Kleinste Abweichungen (Tippfehler, Umstellungen, etc.) führen dazu, dass das Programm fehlerhaft wird. Obwohl die Anzahl der Leerzeichen und Zeilenwechsel zwischen den Worten hierbei keine Rolle spielt, sollten Sie sie trotzdem möglichst genau übertragen.

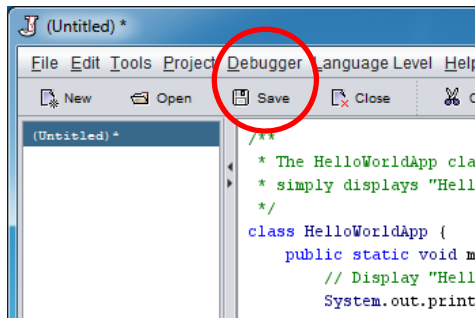
Nach dem Eingeben des Textes sollte das DrJava-Fenster wie folgt aussehen:



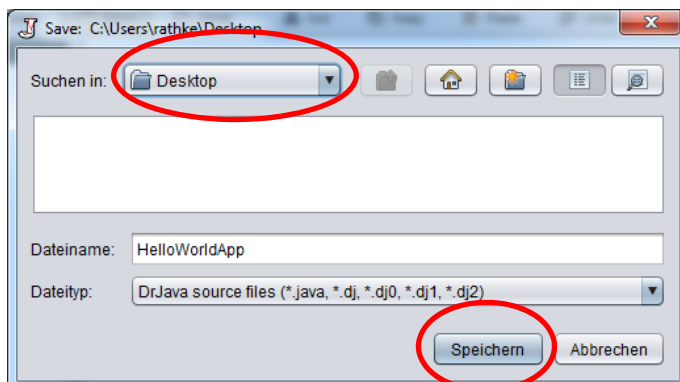
¹ Wenn Sie den Text mit „Kopieren-und-Einfügen“ übertragen, besteht die Gefahr, dass Formatierungselemente mitkopiert werden, die dazu führen, dass das Programm fehlerhaft wird.

Das eingegebene Programm ist das in etwa kleinste, das noch etwas Sichtbares tut: wenn es später ausgeführt wird, gibt es den Text „Hello World!“ auf dem Bildschirm aus. Wir werden das noch sehen. Man verwendet dieses Programm häufig um zu zeigen, wie einfach oder kompliziert eine solch triviale Aufgabe in einer Programmiersprache bewerkstelligt werden kann. Uns dient es als Beispiel, an Hand dessen wir die beim Programmieren erforderlichen Schritte darstellen können.

Als nächstes sollten Sie Ihr Java-Programm abspeichern. Dies geschieht entweder über das File-Menü links oben oder über das Save-Symbol:



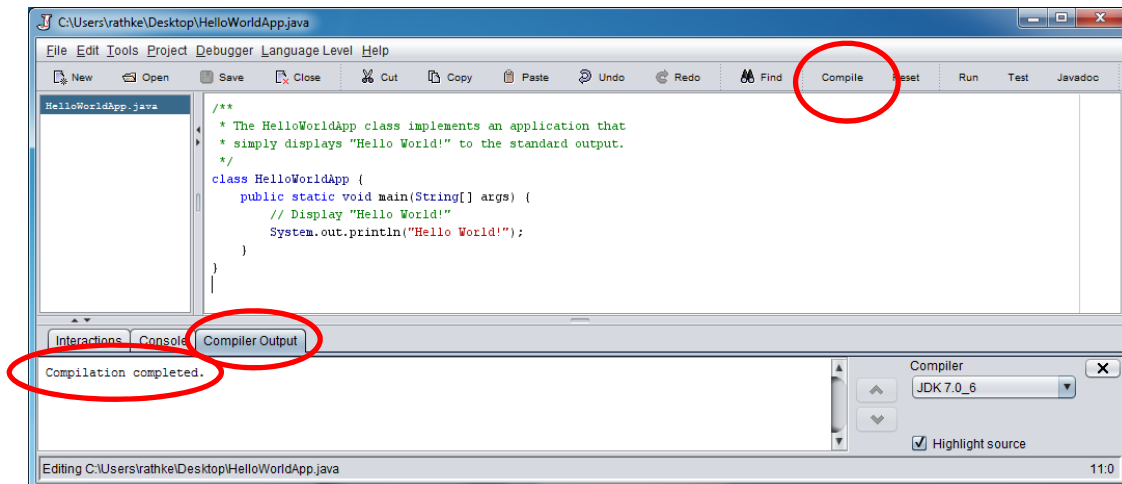
An dieser Stelle werden Sie aufgefordert, das Programm in eine Datei namens „HelloWorldApp“ abzuspeichern. Diesen Namen hat der Programmeditor aus dem eingegebenen Java-Programm abgeleitet und sollte nicht verändert werden. Am besten, Sie navigieren über die DropDown-Liste zum „Desktop“ und bestätigen den Namen durch Betätigen der Schaltfläche „Speichern“.



3. Das Programm in seine ausführbare Form übersetzen

Damit das abgespeicherte Programm auf dem Computer ausgeführt werden kann, muss es in eine andere sprachliche Form *übersetzt* (= *compiliert*) werden. Diese Aufgabe übernimmt ein Programm aus dem Java-Entwickler-Werkzeugkasten, der *Java-Compiler*.

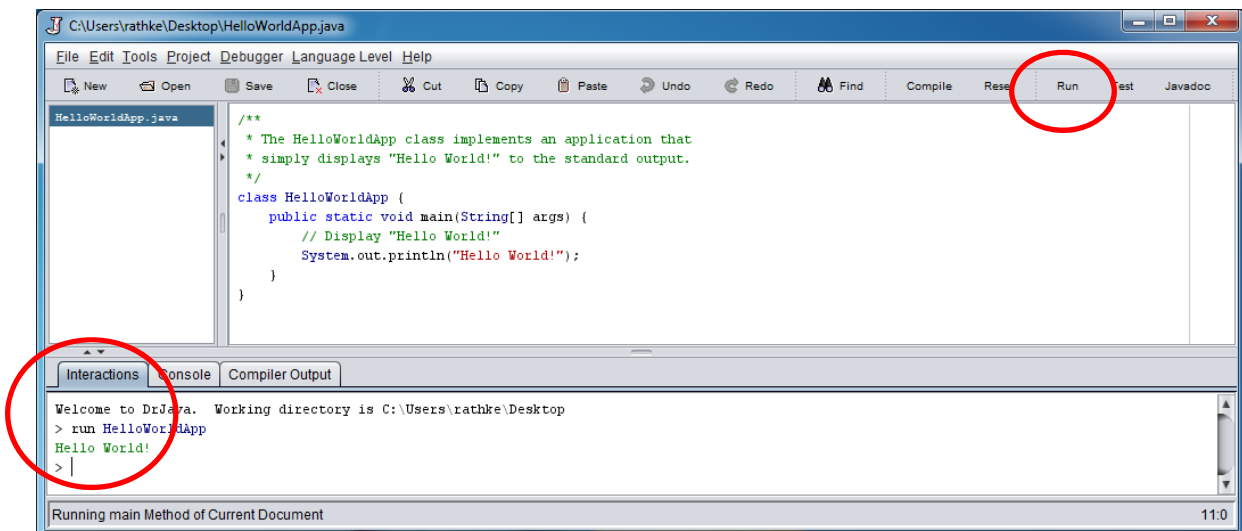
Aktiviert wird der Java-Compiler durch Betätigung der Schaltfläche „Compile“. Danach erscheint im unteren Bereich unter dem Reiter „Compiler Output“ die Meldung „Compilation completed.“



Die übersetzte Form (im Fall von Java-Programmen heißt diese Form "Bytecode") wird vom Java-Compiler in einer Datei abgespeichert. Dies geschieht für uns ohne besondere Rückmeldung und erfolgt automatisch als Ergebnis der Übersetzung. Sie können diese Datei aber auf dem Desktop sehen, wenn Sie das Java-Programm vorher dort abgespeichert hatten.

4. Das Programm ausführen

Das übersetzte Programm wird mit Hilfe des sog *Java-Interpreters* ausgeführt. Betätigen Sie hierzu die Schaltfläche „Run“ im Programmeditor:



Daraufhin sollte nach kurzer Zeit in der „Interactions“-Anzeige von DrJava der Text „Hello World!“ ausgegeben werden. Herzlichen Glückwunsch: Sie haben Ihr (erstes?) Java-Programm erfolgreich eingegeben, übersetzt und ausgeführt!

Rückblick: Typisches Vorgehen beim Entwickeln von (Java-) Programmen

1. Den Programmeditor starten.
2. Das Java-Programm (hier: HelloWorldApp) im Editor laden/eingeben/modifizieren.
3. Das Programm in eine Datei (hier: HelloWorldApp.java) abspeichern!

4. Das Programm compilieren. Hierbei entsteht eine Datei, die den Bytecode des Programms enthält (hier: HelloWorldApp.class)
5. Beim Auftreten von Fehlern zum Überarbeiten des Programms zurück zu Punkt 2.
6. Das Programm ausführen.
7. Beim Auftreten von Fehlern zum Überarbeiten des Programms zurück zu Punkt 2.
8. Das Programm macht seine Ausgaben (hier: Hello World!)

Übungen

1. Modifizieren Sie das Java-Programm, so dass es anstatt "Hello World!" Ihren Namen ausgibt. Versuchen Sie dazu im Programmtext zu raten, an welcher Stelle dieser Text ersetzt werden muss. Lassen Sie das Programm anschließend übersetzen und ausführen.
2. Bauen Sie einen Fehler in das Programm ein, z.B. ersetzen Sie das Wort „public“ durch „oeffentlich“. Was passiert, wenn Sie es übersetzen lassen?
 - Wo erscheint die Fehlermeldung?
 - Was besagt sie? Können Sie mit ihrer Hilfe erkennen, wo der Fehler auftritt und was die Ursache ist?

5. Erstellen und Ausführen eines Java-Applets

Neben auf dem Rechner direkt ausführbaren Programmen können mit der Programmiersprache Java sog. *Applets* erstellt werden. Applets sind Java-Programme, die als Teil einer Internet-Seite durch einen Internet-Browser ausgeführt werden. Die Internetseite wird dadurch zum Träger des Java-Programms.

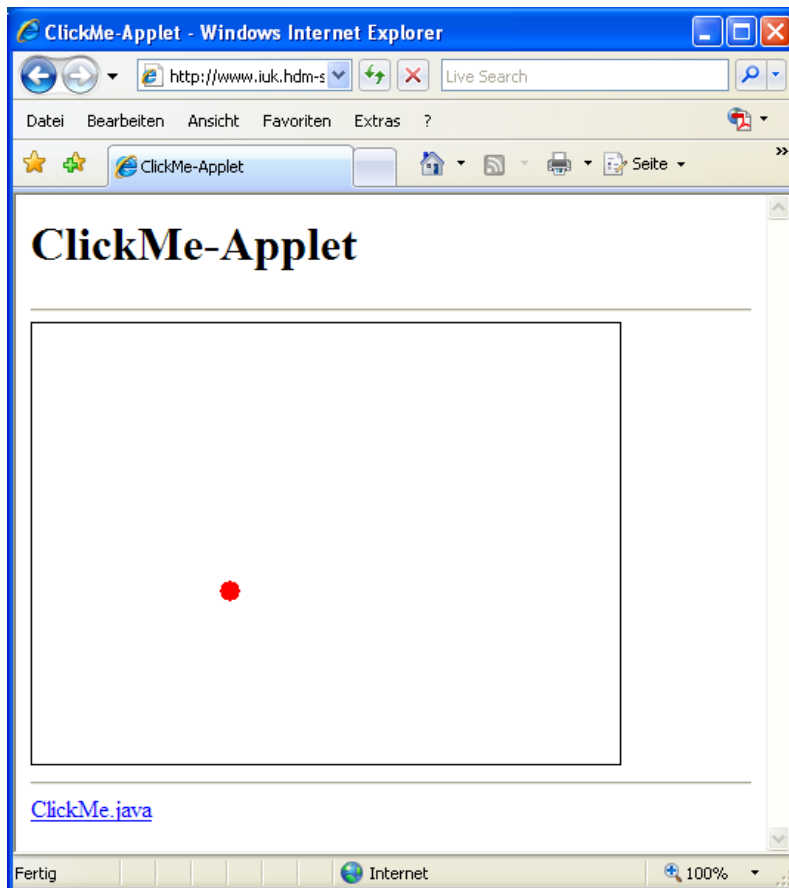
Damit wir uns besser vorstellen können, worauf wir hinaus wollen, schauen wir uns das Ergebnis einmal vorab im Browser an.

Starten Sie dazu einen Internet-Browser (MS Internet Explorer oder Firefox) und geben Sie die folgende URL in das Adressfeld des Browsers ein:²

<http://www.hdm-stuttgart.de/~rathke/ClickMe.html>

Bitte bestätigen Sie dabei alle Abfragen, die Ihr Browser bzgl. der Vertrauenswürdigkeit der Webseite stellt und lassen Sie das Applet ausführen.

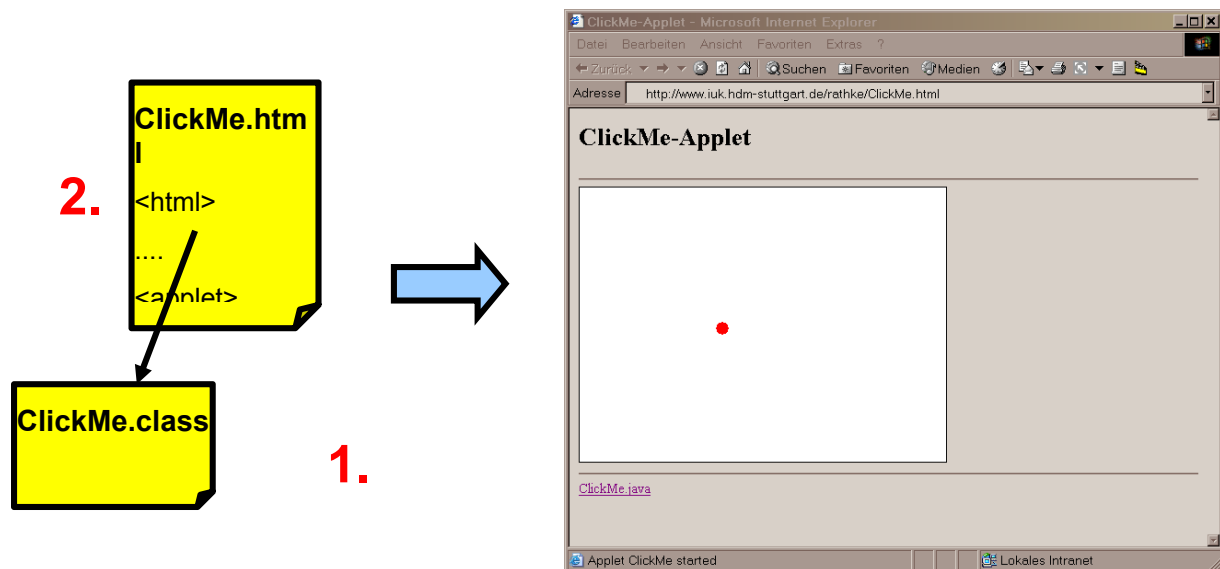
² Wenn diese URL „nicht funktioniert“, haben Sie sie wahrscheinlich nur den Link angeklickt oder mit Copy-Paste aus dem Dokument in das Adressfeld übertragen. Probieren Sie es noch einmal, indem Sie die URL abtippen.



Wenn Sie im Bereich zwischen den Linien mit der Maus klicken, erscheint an dieser Stelle ein roter Punkt.

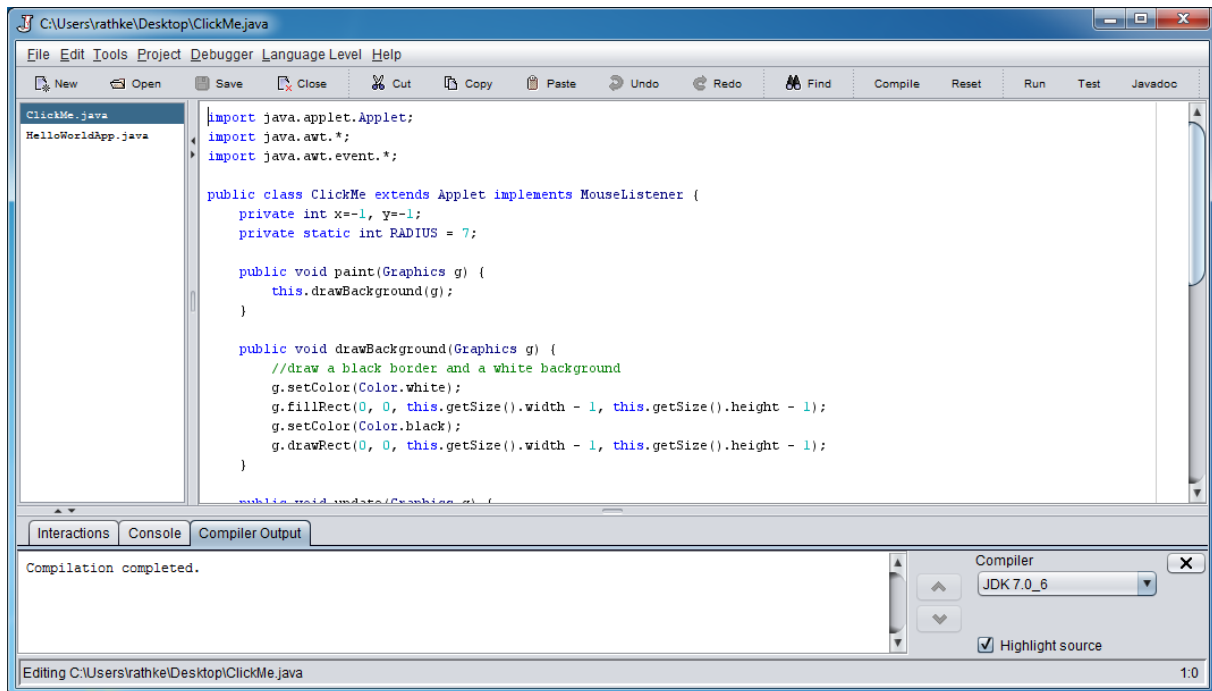
Um ein Java-Programm im Browser ausführen zu lassen, ist neben dem Java-Programm auch die Internet-Seite zu erstellen, in der das Programm „lebt“. Wir müssen beides tun:

1. das Java-Programm schreiben und übersetzen und
2. die Internet-Seite schreiben und darin das Java-Programm „einbetten“.



Wir könnten so vorgehen wie zu Beginn der Übung, d.h. DrJava wird gestartet und der Programmtext wird eingetippt. Zur Vereinfachung schreiben wir das Programm allerdings nicht neu, sondern kopieren uns den Quellcode aus einer vorhandenen Datei im Intranet.

Wechseln Sie hierzu nochmals zum Browserfenster und klicken Sie mit der rechten Maustaste auf den Link „ClickMe.java“ unter dem Applet und speichern Sie diese Datei am besten auf dem Desktop ab. Öffnen Sie dann diese Datei im Programmeditor DrJava und schauen Sie sich den Quellcode einmal an:



Der Quellcode eines ähnlichen Java-Programms, das nicht im Browser sondern direkt auf dem PC abläuft, wird Gegenstand des nächsten Übungstermins sein. Wir werden dann diesen Code versuchen soweit zu verstehen, dass wir selbst Modifikationen darin vornehmen können, um z.B. die Farbe oder die Größe des Punktes zu verändern, der bei einem Mausklick erscheint.

Wie zuvor wird der Quellcode dieses Programms manuell über die Schaltfläche „Compile“ übersetzt. Wie das letzte Mal hat auch jetzt der Compiler eine Datei erzeugt, deren Inhalt als Programm ausgeführt werden kann. Da Applets allerdings nicht ohne Browser ausgeführt werden können, werden Sie dies nicht einfach über die Schaltfläche „Run“ bewerkstelligen können.

Es muss noch eine HTML-Seite erstellt werden, in die das Applet eingebettet wird. Erstellen Sie hierzu eine neue HTML-Datei, in dem Sie in DrJava „New“ auswählen und anschließend den folgenden Text eingeben:

```
<HTML>
  <Head>
    <Title>ClickMe-Applet</Title>
  </Head>
  <Body>
    <H1>ClickMe-Applet</H1>
    <Applet Code="ClickMe.class" Width=400 Height=300>
    </Applet>
  </Body>
</HTML>
```


Speichern Sie diese Datei unter dem Namen „ClickMe.html“ ab. Achten Sie darauf, dass sie sich im selben Verzeichnis wie das zuvor abgespeicherte Java-Programm befindet. Öffnen Sie anschließend die Datei „ClickMe.html“ im Browser. Diese Mal wird das von Ihnen abgespeicherte und übersetzte ClickMe.java verwendet.

Exkurs: Funktionsweise eines Browsers beim Anzeigen von HTML-Dateien

Zum Verständnis dessen, was passiert, sind ein paar Bemerkungen zur Funktionsweise von Browsern im Allgemeinen und zu HTML im Besonderen notwendig. Wenn Sie sich mit HTML schon auskennen, können Sie diesen Exkurs überspringen.

Seiten, die von einem Internet-Browser (kurz: Browser) angezeigt werden, sind normalerweise Inhalt einer Datei auf einem Rechner, der als „Web-Server“ fungiert. Ein Web-Server erhält von einem Browser die Aufforderung, eine Internet-Seite an den Browser zu übermitteln, damit der Browser die Seite anzeigen kann. Als Reaktion sendet der Server den Inhalt der Datei an den Browser zurück und die Seite wird angezeigt.

Neben Text enthält eine Seite Hinweise, wie der Text formatiert werden soll und ob zusätzlich Bilder, Schaltflächen, dynamische Effekte, Formulare, usw. in der Seite enthalten sind. Diese zusätzlichen Hinweise werden in Form sog. Markierungen kenntlich gemacht. Man erkennt diese Markierungen an den spitzen Klammern „<“ und „>“. HTML steht nämlich für „Hypertext Markup Language“ – eine Sprache, die mit in ihrer Bedeutung festgelegten Markierungen arbeitet.

In unserem Beispiel der Datei ClickMe.html gibt es mehr Markierungen als Text. In der Reihenfolge von oben nach unten sind die Markierungen: <HTML>, <Head>, <Title>, </Title>, </Head>, <Body>, <H1>, </H1>, <HR>, <Applet Code=“Clickme.class“ Width=400 Height=300>, </Applet>, <HR>, </Body>, </HTML>.

Es fällt auf, dass es von nahezu allen Markierungen (Ausnahme ist die Markierung <HR>) eine Form mit „/“ und eine Form ohne „/“ gibt. Markierungen sind wie Klammerpaare, die einen Teil des Textes auf der Seite umschließen. Die sich „öffnende Klammer“ enthält keinen Schrägstrich und die sich „schließende Klammer“ beginnt mit einem Schrägstrich. In der dritten Zeile der Datei ClickMe.html sieht man das besonders deutlich. Der Text „ClickMe-Applet“ ist zwischen den Markierungen <Title> und </Title> eingeschlossen. Diese Markierung legt übrigens fest, welcher Text in der Randleiste des Browser-Fensters erscheinen soll, wenn die Seite angezeigt wird.

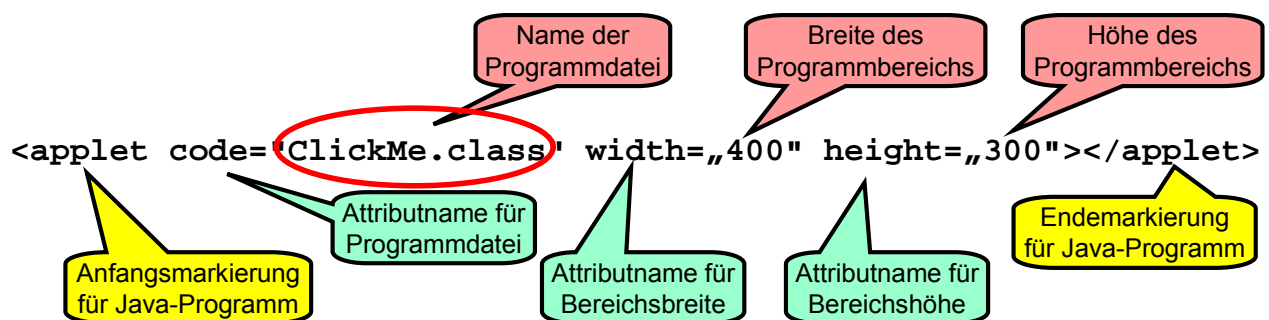
Aber auch die anderen Markierungen haben jeweils einen Anfang und ein Ende:

Markierung	Bedeutung
<html> Text </html>	Text besteht aus Zeichen und HTML-Markierungen.
<head> Text </head>	Text ist der Seitenkopf der HTML-Seite. Im Kopf werden allgemeine Angaben über den Hauptteil der Seite, den Seitenkörper, gemacht.
<title> Text </title>	Text erscheint im Randbereich des Browsers

<code><body> Text </body></code>	Text ist der Inhalt der Seite. Er wird durch den Browser angezeigt.
<code><hr></code>	„Horizontal Rule“ produziert einen waagrechten Strich.
<code><applet> Text </applet></code>	Text sind zusätzliche Angaben, die einem Java-Applet bei Ausführung zur Verfügung stehen.

Die für uns interessante Markierung ist offensichtlich die `<applet>`-Markierung.

Die Struktur und der Aufbau der Applet-Markierung erlaubt es dem Browser, unser Java-Programm in einem Bereich der anzuzeigenden HTML-Seite ablaufen zu lassen. Dazu enthält die Applet-Markierung verschiedenen Informationen über das Java-Programm:



Zunächst kann man feststellen, dass die Anfangsmarkierung von der öffnenden bis zur schließenden spitzen Klammer 3 sog. Attribute enthält. Diese haben die folgende Bedeutung:

Attribut	Bedeutung
<code>code="ClickMe.class"</code>	Die Programmdatei, die ausführbaren Byte-Code enthält, wie er vom Java-Compiler erzeugt wurde.
<code>width="400"</code>	Die Breite des Bereichs, in der das Programm auf der angezeigten Seite Ausgaben machen kann.
<code>height="300"</code>	Die Höhe des Bereichs, in der das Programm auf der angezeigten Seite Ausgaben machen kann.

Der Inhalt der Applet-Markierung, d.h. das was zwischen der Anfangs- und der Endmarkierung steht, ist leer. Alle notwendigen Informationen befinden sich „innerhalb“ der Anfangsmarkierung als Markierungsattribute.

Wenn nun ein Browser die HTML-Seite anzeigt und auf die Applet-Markierung trifft, fordert er vom Web-Server die im Code-Attribut angegebene Datei an und führt das vom Server übermittelte Programm aus. Das Java-Programm wird also direkt vom Browser ausgeführt! Dazu muss natürlich der Browser Java „können“. Dies bedeutet der Ausdruck „Java enabled“, wenn Hersteller von den Eigenschaften ihres Browsers sprechen.

Übungen

Speichern Sie die Datei WavingDuke.html aus dem Moodle Kurs dieser Lehrveranstaltung auf Ihrem Desktop und aktivieren Sie die kopierte Datei durch Doppelklick.

Bitte bestätigen Sie dabei alle Abfragen, die Ihr Browser bzgl. der Vertrauenswürdigkeit der Webseite stellt und lassen Sie das Applet ausführen.

- a. Öffnen Sie die HTML-Datei in DrJava und ergänzen Sie eigene Texte vor und nach der Applet-Markierung. Überprüfen Sie Ihre Änderungen nach Abspeichern durch Öffnen im Browser.³
- b. Experimentieren Sie mit den Parameterwerten, d.h. verändern Sie die Angaben für die Pausen und beobachten Sie die Veränderungen.
- c. Wenn Sie schon eigene HTML-Seiten erstellt haben, integrieren Sie das Applet in eine ihrer Seiten!

³ Evtl. müssen Sie die Seite explizit neu laden, damit der Browser nicht die gemerkte aber veraltete Version zeigt.