

HANS-GEORG SCHUMANN

JAVA

FÜR KIDS

8. AUFLAGE

**DER EINFACHE EINSTIEG
OHNE VORKENNTNISSE**



Hinweis des Verlages zum Urheberrecht und Digitalen Rechtemanagement (DRM)

Liebe Leserinnen und Leser,

dieses E-Book, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Mit dem Kauf räumen wir Ihnen das Recht ein, die Inhalte im Rahmen des geltenden Urheberrechts zu nutzen. Jede Verwertung außerhalb dieser Grenzen ist ohne unsere Zustimmung unzulässig und strafbar. Das gilt besonders für Vervielfältigungen, Übersetzungen sowie Einspeicherung und Verarbeitung in elektronischen Systemen.

Je nachdem wo Sie Ihr E-Book gekauft haben, kann dieser Shop das E-Book vor Missbrauch durch ein digitales Rechtemanagement schützen. Häufig erfolgt dies in Form eines nicht sichtbaren digitalen Wasserzeichens, das dann individuell pro Nutzer signiert ist. Angaben zu diesem DRM finden Sie auf den Seiten der jeweiligen Anbieter.

Beim Kauf des E-Books in unserem Verlagsshop ist Ihr E-Book DRM-frei.

Viele Grüße und viel Spaß beim Lesen,

Ihr mitp-Verlagsteam



Neuerscheinungen, Praxistipps, Gratiskapitel,
Einblicke in den Verlagsalltag –
gibt es alles bei uns auf Instagram und Facebook



[instagram.com/mitp_verlag](https://www.instagram.com/mitp_verlag)



[facebook.com/mitp.verlag](https://www.facebook.com/mitp.verlag)

Inhaltsverzeichnis

Impressum

Einleitung

Was heisst eigentlich Programmieren?

Was ist eine Entwicklungsumgebung?

Warum gerade Java?

Eclipse, die Entwicklungsumgebung zum Buch

Und was bietet dieses Buch?

Wie arbeitest du mit diesem Buch?

Arbeitsschritte

Aufgaben

Notfälle

Achtung

Spezialwissen

Spickzettel

Was brauchst du für dieses Buch?

Betriebssystem

Speichermedien

Hinweise für Lehrer

Übungsmedien

Regelmäßig sichern

Kapitel 1: Aller Anfang ist schwer: Das erste Projekt

Eclipse starten

Willkommen in Java

Ein erstes Hallo
Objekte, Klassen und Pakete
Eclipse beenden
Zusammenfassung
Ein paar Fragen ...
... aber noch keine Aufgabe

Kapitel 2: Wenn, Dann, Sonst: Kontrolle und Auswahl

Eingabe und Ausgabe
Dialog mit »Schwung«
Gut oder schlecht?
Die if-Struktur
Strings oder Zahlen?
Plus oder minus, mal oder durch
Die if-else-Struktur
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 3: Zensuren und Zahlenraten: Bedingungen

Von int zu float
Die Sache mit try und catch
Von 1 bis 6
Von Fall zu Fall
Punkt für Punkt
Und und Oder, oder?

Ein kleines Spielchen
Die while-Struktur
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 4: Geld-Spielereien: Schleifen und Felder

Dein PC zählt mit
Abbruch bei Unlust?
Auf dem Weg zum Millionär
Schleifenvariationen
Zählen mit for
Variablenfelder
Lottoziehung
Feldsortierung
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 5: Do it yourself: Funktionen und Klassen

Java ist lernfähig
Funktionen fürs Ratespiel
Lokal oder global?
Parameter und Rückgabe
Ein neues Baby?
Laufen lernen ...

Kapselung und Vererbung
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 6: Nicht nur was fürs Auge: Einstieg in Swing

Erst mal ein Fenster
Hallo, wie geht es?
Es passiert etwas
Gut oder schlecht?
Es gibt was zu erben
super, this und andere Erbstücke
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 7: Alles unter Kontrolle: Komponentensammlung

Kleine Knopfparade
Feldoptimierung
Listenwahl
Von Pünktchen ...
... und Häkchen
Körper, Geist und Seele
Der letzte Schliff
Zusammenfassung
Ein paar Fragen ...

... und ein paar Aufgaben

Kapitel 8: Wer weiss was? Quiz-Projekt Teil 1

Erst der Plan, dann der Bau

Frage und Antworten

Datensammlung

Datentransfer

Aufsammeln und einordnen

Zusammenfassung

Ein paar Fragen ...

... aber keine Aufgabe

Kapitel 9: Spielen und Lernen: Quiz-Projekt Teil 2

Du hast die Wahl

Aufgabenkontrolle

Antwort als Optionen

Vokabeln lernen?

Mehrfachauswahl

Zusammenfassung

Ein paar Fragen ...

... und ein paar Aufgaben

Kapitel 10: Jetzt wird's bunt: Grafik in Java

Von Punkten und Koordinaten

Das erste Bild

Jetzt wird's bunt

Eckig und rund

Mit Text geht auch
Farbtupfer
Von Graphics zu Graphics2D
Zusammenfassung
Ein paar Fragen ...
... und ein paar Aufgaben

Kapitel 11: Bilder lernen laufen: Animationen

Erst mal ein Kreis
Und es bewegt sich was
Endlich ein (richtiges) Bild
Bildersammlung
Da läuft was
Drehen, verschwinden, auftauchen
Movie komplett
Zusammenfassung
Keine Fragen ...
... doch ein paar Aufgaben

Kapitel 12: Buntes Trio: Kleine Spielesammlung

Wie viele Augen?
Anzeigen und auswerten
Schere - Stein - Papier
Qual der Wahl
Das Nim-Spiel
Streichholz-Parade

Nur einer kann gewinnen
Zusammenfassung
Keine Fragen ...
... aber zwei Aufgaben

Kapitel 13: Hangman: Raten oder »braten«

Anzeige und Eingabe
Auf ein Wort
Angst vor dem Galgen?
Wortschatz
Komplettierung
Zusammenfassung
Keine Fragen ...
... und keine Aufgaben

Kapitel 14: Kleiner Krabbelkurs: Tasten- und Maussteuerung

Ein Käfer bewegt sich
Der KeyListener
Richtungswechsel
Kein Spiel ohne Grenzen
Maussteuerung
Ohne Mathe geht es nicht
Winkelberechnung
Zusammenfassung
Keine Fragen ...
... und keine Aufgaben

Kapitel 15: Insektenjagd: Opfer und Jäger

Freilauf mit Kehrtwende

Runnable und run()

Klick und Platt

Neuerscheinung

Insektarium?

MyGame-Klasse komplett

Zusammenfassung (und Schluss)

Keine Fragen ...

... und keine Aufgaben

Anhang A: Anhang A

Java installieren

Eclipse installieren

Einsatz der Buch-Dateien

Anhang B: Anhang B

Kleine Checkliste

Dem Fehler auf der Spur

 Debug-Modus

 Haltepunkte setzen

 Ein Programm in Einzelschritten

*Für
Janne, Julia, Daniel und Katrin*

Hans-Georg Schumann

Java für Kids

**Programmieren lernen ohne
Vorkenntnisse**



mitp

Impressum

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-7475-0522-9

8. Auflage 2022

www.mitp.de

E-Mail: mitp-verlag@sigloch.de

Telefon: +49 7953 / 7189 - 079

Telefax: +49 7953 / 7189 - 082

© 2022 mitp Verlags GmbH & Co. KG

Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Lektorat: Katja Völpel
Sprachkorrektorat: Petra Heubach-Erdmann
Covergestaltung: Christian Kalkert
electronic **pub**lication: Ill-satz, Flensburg, www.drei-satz.de

Dieses E-Book verwendet das ePub-Format und ist optimiert für die Nutzung mit dem iBooks-reader auf dem iPad von Apple. Bei der Verwendung anderer Reader kann es zu Darstellungsproblemen kommen.

Der Verlag räumt Ihnen mit dem Kauf des E-Books das Recht ein, die Inhalte im Rahmen des geltenden Urheberrechts zu nutzen. Dieses Werk, einschließlich aller seiner Teile, ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlages unzulässig und strafbar. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und Einspeicherung und Verarbeitung in elektronischen Systemen.

Der Verlag schützt seine E-Books vor Missbrauch des Urheberrechts durch ein digitales Rechtemanagement. Bei Kauf im Webshop des Verlages werden die E-Books mit einem nicht sichtbaren digitalen Wasserzeichen individuell pro Nutzer signiert.

Bei Kauf in anderen E-Book-Webshops erfolgt die Signatur durch die Shopbetreiber. Angaben zu diesem DRM finden Sie auf den Seiten der jeweiligen Anbieter.

Einleitung

Java – ist das nun eine Insel mit vorwiegend warmem Klima? Oder eine Programmiersprache, die auch etwas mit dem Internet zu tun haben soll? Beides ist richtig, aber mit einem Reisebericht auf die besagte Insel kann ich hier nicht dienen, obwohl die gleichnamige Sprache durchaus etwas damit zu tun hat.

Entstanden ist diese Sprache, indem sich ihre Erfinder zuerst mal kräftig bei einer anderen Programmiersprache bedient haben, nämlich bei C++. Über die gäbe es eine Menge zu schreiben, u.a. muss man zugeben, dass sie die schwierigere Sprache von beiden ist. (Über C++ habe ich ein anderes Kids-Buch geschrieben.)

Ursprünglich wurde die Programmiersprache, um die es in diesem Buch geht, Oak genannt (zu Deutsch so viel wie »Eiche«). Kein besonders attraktiver Name, fand man schließlich, da passte dann der einer Südseeinsel schon besser.

Java zu lernen ist nicht zu schwierig, und dennoch hat diese Sprache die Fähigkeiten berühmter großer Konkurrenten wie z.B. C++, C# und Delphi. Java wird von der Firma Oracle verwaltet und kann kostenlos aus dem Internet heruntergeladen werden.

Was heisst eigentlich Programmieren?

Wenn du aufschreibst, was ein Computer tun soll, nennt man das Programmieren. Das Tolle daran ist, dass du selbst bestimmen kannst, was getan werden soll. Lässt du dein Programm laufen, macht der Computer die Sachen, die du ausgeheckt hast. Natürlich wird er dann dein Zimmer nicht aufräumen und dir auch keine Tasse Kakao ans Bett bringen. Aber kannst du erst mal programmieren, kannst du den Computer sozusagen nach deiner Pfeife tanzen lassen.

Allerdings passiert es gerade beim Programmieren, dass der Computer nicht so will, wie du es gerne hättest. Meistens ist das ein Fehler im Programm. Das Problem kann aber auch irgendwo anders im Computer oder im Betriebssystem liegen. Das Dumme bei Fehlern ist, dass sie sich gern so gut verstecken, dass die Suche danach schon manchen Programmierer zur Verzweiflung gebracht hat.

Vielleicht hast du nun trotzdem Lust bekommen, das Programmieren zu erlernen. Dann brauchst du ja nur noch eine passende Entwicklungsumgebung, und schon kann's losgehen.

Was ist eine Entwicklungsumgebung?

Um ein Programm zu erstellen, musst du erst mal etwas eintippen. Das ist wie bei einem Brief oder einer Geschichte, die man schreibt. Das Textprogramm dafür kann sehr einfach sein, weil es ja nicht auf eine besondere Schrift oder Darstellung ankommt wie bei einem Brief oder einem Referat. So etwas wird Editor genannt.

Ist das Programm eingetippt, kann es der Computer nicht einfach so lesen und ausführen. Jetzt muss es so übersetzt werden, dass der PC versteht, was du von ihm willst. Weil er

aber eine ganz andere Sprache spricht als du, muss ein Dolmetscher her.

Du programmierst in einer Sprache, die du verstehst, und der Dolmetscher übersetzt es so, dass es dem Computer verständlich wird. So etwas heißt dann Compiler. In Java klingt dieser Dolmetscher noch ein bisschen technischer: Die Java Virtual Machine (kurz JVM) ist eine Art »Zwischencomputer«.

Das heißt: Eigentlich wird ein Java-Programm an die JVM weitergereicht, die es dann für den jeweiligen Computer passend zubereitet: Das kann dann ein Windows-PC oder ein Linux-PC sein, ein Macintosh oder irgendein anderes Computersystem. Ein und dasselbe Java-Programm funktioniert so auf jedem beliebigen Computer, der über eine JVM verfügt. Viele Apps für das Android-System, das auf den meisten Smartphones läuft, wurden in Java programmiert.

Schließlich müssen Programme getestet, überarbeitet, verbessert, wieder getestet und weiterentwickelt werden. Dazu gibt es noch einige zusätzliche Hilfen. Daraus wird dann ein ganzes System, die Entwicklungsumgebung.

Warum gerade Java?

Leider kannst du nicht so programmieren, wie dir der Mund gewachsen ist. Eine Programmiersprache muss so aufgebaut sein, dass möglichst viele Menschen in möglichst vielen Ländern einheitlich damit umgehen können.

Weil in der ganzen Welt Leute zu finden sind, die wenigstens ein paar Brocken Englisch können, besteht auch fast jede

Programmiersprache aus englischen Wörtern. Es gab auch immer mal Versuche, z.B. in Deutsch zu programmieren, aber meistens klingen die Wörter dort so künstlich, dass man lieber wieder aufs Englische zurückgreift.

Eigentlich ist es egal, welche Programmiersprache du benutzt. Am besten eine, die möglichst leicht zu erlernen ist.

In diesem Buch hast du es mit der Programmiersprache Java zu tun. Sie ist inzwischen eine der Sprachen, die am meisten verbreitet sind. Sie ist nicht einfach, aber auch für Anfänger geeignet, die mit Java ihre erste Programmiersprache lernen wollen. (Willst du mal in andere Sprachen hineinschnuppern, dann empfehle ich dir z.B. eines der Kids-Bücher über C++, C#, JavaScript oder Python.)

Der Weg zum guten Programmierer kann ganz schön steinig sein. Nicht selten kommt es vor, dass man die Lust verliert, weil einfach gar nichts klappen will. Das Programm tut etwas ganz anderes, man kann den Fehler nicht finden und man fragt sich: Wozu soll ich eigentlich programmieren lernen, wo es doch schon genug Programme gibt? Und dann noch ausgerechnet in Java.

Zurzeit werden gute Programmierer dringend gesucht, und dieser Bedarf wird weiter steigen. In vielen Stellenanzeigen steht unter anderem oft »Programmierkenntnisse in Java erwünscht«. Wirklich gute Programmierer werden auch wirklich gut bezahlt. Es ist also nicht nur einen Versuch wert, es kann sich durchaus lohnen, das Programmieren in Java zu erlernen.

Eclipse, die Entwicklungsumgebung zum Buch

Um den Kauf einer Entwicklungsumgebung für Java musst du dich nicht weiter kümmern, denn die bekommst du kostenlos aus dem Internet. Mit der freien Software **Eclipse** hast du eine weitverbreitete Entwicklungsumgebung und kannst damit unter allen Versionen von Windows programmieren.

Das komplette Paket besteht aus zwei Teilen, die du von diesen Seiten herunterladen kannst:

<https://www.java.com/de/>

<https://www.eclipse.org/>

Und was bietet dieses Buch?

Über eine ganze Reihe von Kapiteln verteilt lernst du

- die Grundlagen von Java kennen
- mit Eclipse unter Windows umzugehen
- einiges über die objektorientierte Programmierung (OOP)
- mit Komponenten zu arbeiten (das sind Bausteine, mit denen du dir viel Programmierarbeit sparen kannst)
- die grafischen Möglichkeiten von Java kennen
- eine Reihe von Spielen selber zu programmieren

Im **Anhang** gibt es dann noch einiges an Informationen und Hilfen, u.a. über Installationen und den Umgang mit Fehlern.

Wie arbeitest du mit diesem Buch?

Grundsätzlich besteht dieses Buch aus einer Menge Text mit vielen Abbildungen dazwischen. Natürlich habe ich mich bemüht, alles so zuzubereiten, dass daraus lauter gut verdauliche Happen werden. Damit das Ganze noch genießbarer wird, gibt es zusätzlich noch einige Symbole, die ich dir hier gern erklären möchte:

Arbeitsschritte

- Wenn du dieses Zeichen siehst, heißt das: Es gibt etwas zu tun. Damit kommen wir beim Programmieren Schritt für Schritt einem neuen Ziel immer näher.

Grundsätzlich lernt man besser, wenn man einen Programmtext selbst eintippt oder ändert. Aber nicht immer hat man große Lust dazu. Deshalb gibt es alle Projekte im Buch auch als Download:

<https://www.mitp.de/0520>

Und hinter einem Programmierschritt findest du zusammen mit diesem Pfeil → auch den jeweiligen Namen des Projekts oder einer Datei (z.B. → Projekt1, → Grafik1, → Game1). Wenn du also das Projekt nicht selbst erstellen willst, kannst du stattdessen diese Datei laden (zu finden im Ordner Projekte).

Aufgaben

Am Ende eines Kapitels findest du jeweils eine Reihe von Fragen und Aufgaben. Diese Übungen sind nicht immer ganz einfach, aber sie helfen dir, noch besser zu programmieren. Lösungen zu den Aufgaben findest du in verschiedenen Formaten ebenfalls bei den Download-Dateien. Du kannst sie dir alle im Editor von Windows oder auch in deinem Textverarbeitungsprogramm anschauen. Oder du lässt sie dir ausdrucken und hast sie dann schwarz auf weiß, um sie neben deinen PC zu legen. (Auch die Programme zu den Aufgaben liegen im Ordner Projekte.)

Notfälle

Vielleicht hast du irgendetwas falsch gemacht oder etwas vergessen. Oder es wird gerade knifflig. Dann fragst du dich, was du nun tun sollst. Bei diesem Symbol findest du eine Lösungsmöglichkeit. Es kann nicht schaden, auch mal ganz hinten im Anhang B nachzuschauen, wo ein paar Hinweise zur Pannenhilfe aufgeführt sind.



Achtung

Hin und wieder findest du ein solch dickes Ausrufezeichen im Buch. Dann ist das eine Stelle, an der etwas besonders Wichtiges steht.



Spezialwissen

Wenn du ein solches »Wow« siehst, geht es um ausführlichere Informationen zu einem Thema.



Spickzettel

Unter den Downloaddateien findest du zudem Spickzettel zum Ausdrucken. Sie helfen dir beim Java-Programmieren nicht den Überblick zu verlieren.

Was brauchst du für dieses Buch?

Installiert wird Java mit dem Programm setup in ein Verzeichnis deiner Wahl, z.B. c:\Programme\Java. Auch für Eclipse gibt es ein Installationsprogramm, dafür solltest du dann einen Extra-Ordner benutzen, in dem du später auch deine Java-Projekte unterbringst.

Die Beispielprogramme in diesem Buch gibt es alle als Download von der Homepage des Verlages, falls du mal keine Lust zum Abtippen hast:

<https://www.mitp.de/0520>

Und auch die Lösungen zu den Fragen und Aufgaben sind dort untergebracht (alles im Ordner Projekte).

Betriebssystem

Die meisten Computer arbeiten heute mit dem Betriebssystem Windows. Davon brauchst du eine der Versionen 7 bis 10. (Java gibt es bei Oracle u.a. auch für Linux.)

Speichermedien

Auch wenn du deine Programme auf der Festplatte unterbringst, kann es nicht schaden, sie zusätzlich z.B. auf einem USB-Stick als Backup zu speichern.

Gegebenenfalls bitte deine Eltern oder Lehrer um Hilfe.

Hinweise für Lehrer

Dieses Buch versteht sich auch als Lernwerk für den Informatik-Unterricht in der Schule. Dort setzt natürlich jeder Lehrer seine eigenen Schwerpunkte. Benutzen Sie an Ihrer Schule bereits ein Werk aus einem Schulbuchverlag, so lässt sich dieses Buch auch als Materialienband einsetzen – in Ergänzung zu dem vorhandenen Schulbuch. Weil dieses Buch sozusagen »von null« anfängt, ist ein direkter Einstieg in Java möglich – ohne irgendwelche anderen Programmierkenntnisse.

Ein wichtiger Schwerpunkt in diesem Buch ist die objektorientierte Programmierung (OOP). Auf die wichtigsten Eigenheiten (Kapselung, Vererbung und Polymorphie) wird ausführlich eingegangen.

In den Projekten werden alle wesentlichen Elemente des Java-Wortschatzes wie auch die wichtigsten Grafik-Komponenten eingesetzt. Außerdem erfährt man hier, wie man eine eigene Komponente erstellen kann. Ein besonderer Schwerpunkt liegt auf der Spieleprogrammierung.

In den Lösungen zu den Aufgaben finden Sie weitere Vorschläge zur Programmierung in Java.

Übungsmedien

Für den Informatik-Unterricht sollte jeder Schüler ein anderes externes Speichermedium haben, um darauf seine Programmierversuche zu sichern. So wird verhindert, dass sich auf der Festplatte des Schulcomputers mit der Zeit allerlei »Datenmüll« ansammelt. Außerdem dient der eigene Datenträger dem Datenschutz: Nur der betreffende Schüler kann seine Daten manipulieren.

Regelmäßig sichern

Es kann nicht schaden, die Programmdateien, an denen gerade gearbeitet wird, etwa alle zehn Minuten zu speichern. Denn Computer pflegen gern gerade dann »abzustürzen«, wenn man seine Arbeit längere Zeit nicht gespeichert hat.

Das ist aber nur dann nötig, wenn man ein Programm längere Zeit nicht startet. In der Regel fragt nämlich Eclipse bei jedem Programmstart nach, ob die Datei gespeichert werden soll.

Kapitel 1

Aller Anfang ist schwer:

Das erste Projekt

Du willst gleich mit dem ersten Programm loslegen? Deinen Computer kannst du schon mal anschalten und Windows starten lassen. Dann machen wir gemeinsam die ersten Schritte in Java. Es wird eine Weile dauern, bis du dann dein erstes Programm erschaffen hast, aber wie der Titel des Kapitels schon sagt: Aller Anfang ist schwer.

In diesem Kapitel lernst du

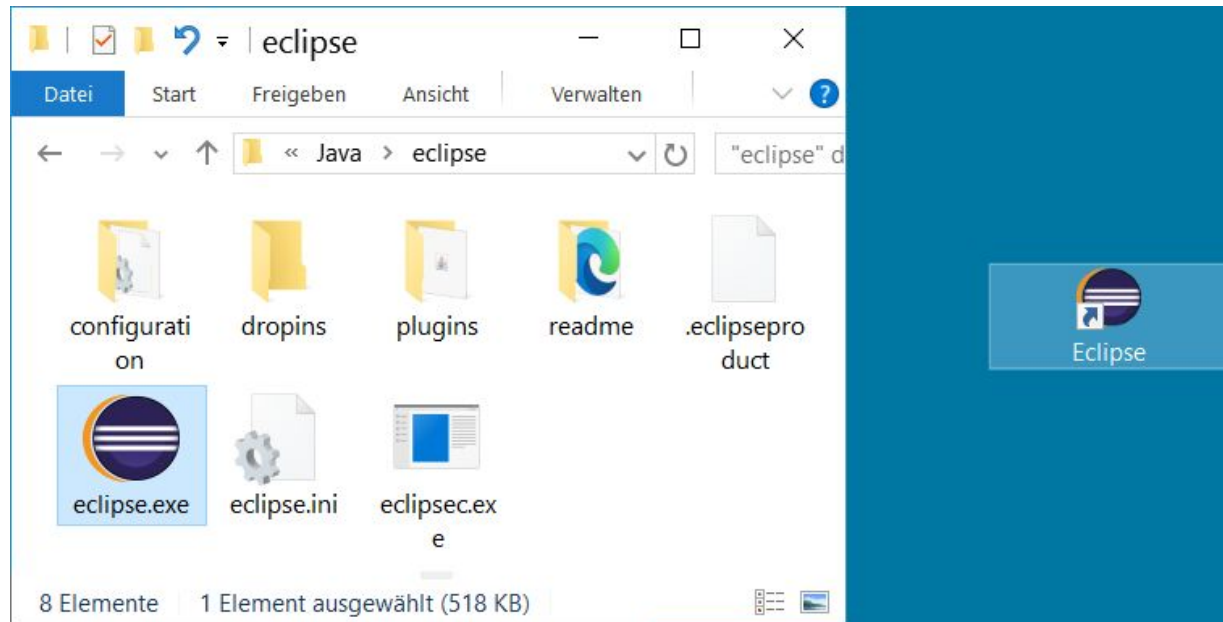
- wie man Eclipse startet
- wie man ein Programmprojekt erstellt und ausführt
- aus welchen »Schalen« ein Java-Projekt besteht
- wie man einen Text anzeigen kann
- wie man Eclipse beendet

Eclipse starten

Bevor wir mit dem Programmieren anfangen können, muss Eclipse erst installiert werden. Genauer erfährst du im Anhang A. Hier musst du dir von jemandem helfen lassen, wenn du dir das Einrichten nicht allein zutraust.

Wenn Eclipse verfügbar ist, hast du diese zwei Möglichkeiten, dieses Programm zu starten:

- Du öffnest den Ordner, in dem du Eclipse untergebracht hast, und doppelklickst mit der Maus auf das Symbol mit dem Namen **ECLIPSE.EXE**.
- Oder du doppelklickst auf das Desktop-Symbol mit dem Namen Eclipse – wenn es vorhanden ist.



Wie kommt das Eclipse-Symbol auf den Desktop? Du brauchst dazu eine **Verknüpfung**; das bedeutet, du legst ein Symbol auf dem Desktop an, das mit dem Programm **ECLIPSE.EXE** verbunden ist. So etwas nennt man Verknüpfung.

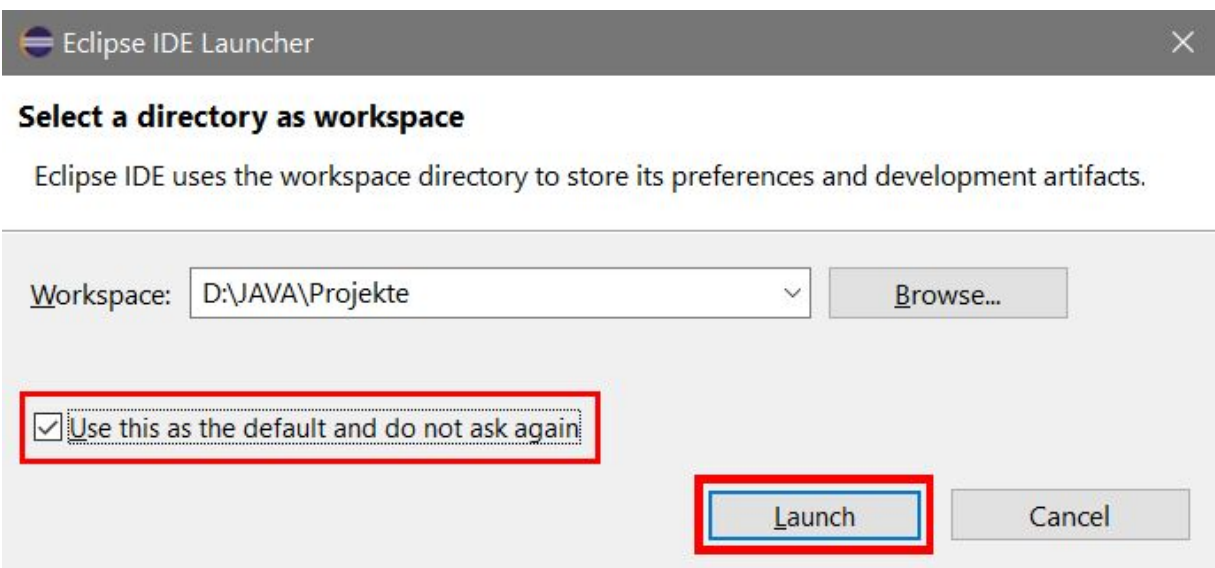


- Dazu klickst du im **ECLIPSE**-Ordner mit der rechten Maustaste auf das Symbol für **ECLIPSE.EXE**. Im Kontextmenü wählst du **KOPIEREN**.
- Dann klicke auf eine freie Stelle auf dem Desktop, ebenfalls mit der rechten Maustaste. Im Kontextmenü wählst du **VERKNÜPFUNG EINFÜGEN**.

- Es ist sinnvoll, für das neue Symbol auf dem Desktop den Text **ECLIPSE.EXE - VERKNÜPFUNG** durch **ECLIPSE** zu ersetzen.

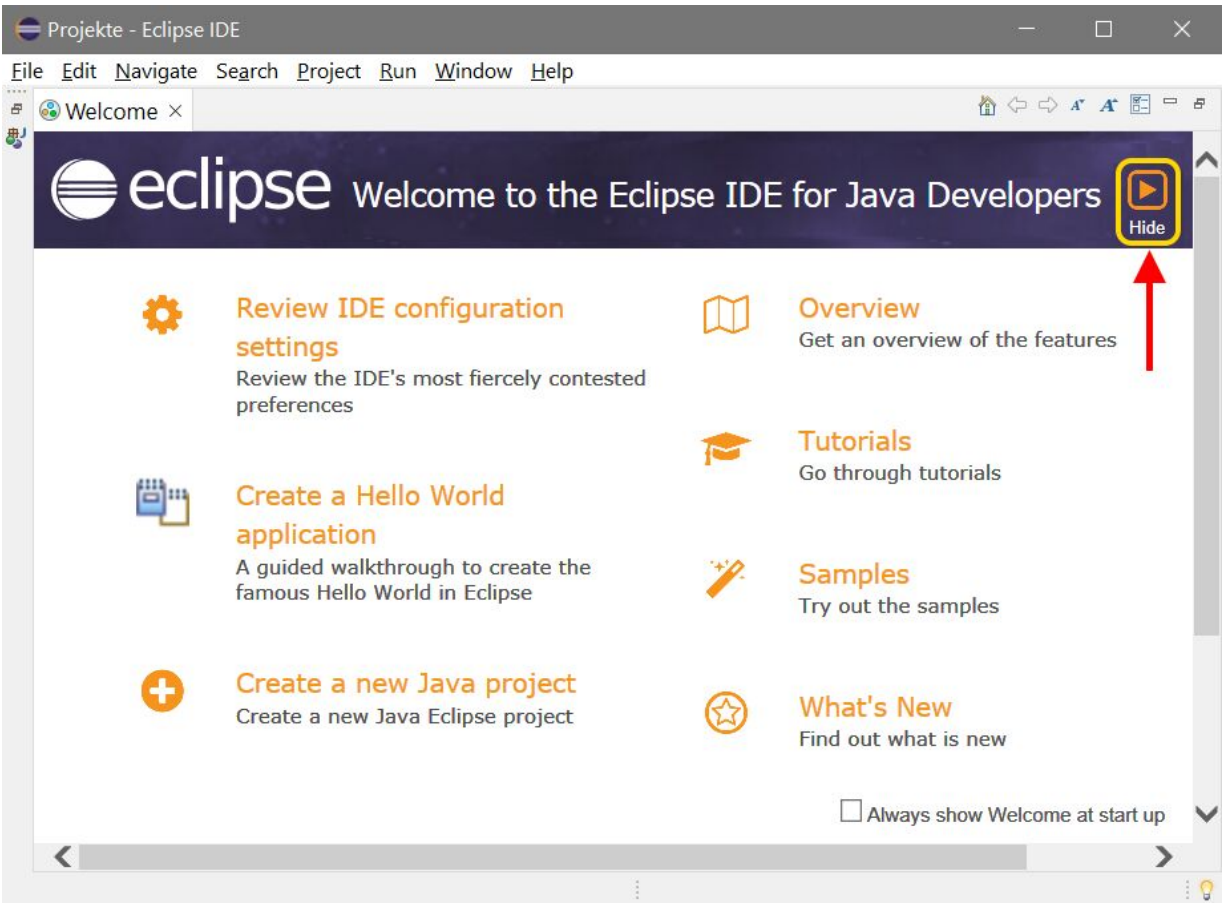
Von nun an kannst du auf das neue Symbol **doppelklicken** und damit Eclipse starten.

Je nach Computer kann es eine Weile dauern, bis Eclipse geladen ist. Zwischendurch fordert ein Dialogfeld einen Ordner an, in dem deine Projekte untergebracht werden – Workspace genannt.



- Du kannst das Verzeichnis so lassen oder ein eigenes angeben. Sorge auf jeden Fall dafür, dass vor dem Eintrag **USE THIS AS DEFAULT AND DO NOT ASK AGAIN** ein Häkchen steht (sonst erscheint dieses Dialogfeld bei jedem deiner nächsten Starts von Eclipse immer wieder). Dann klicke auf **OK**.

Einige Zeit später landest du in einem Willkommens-Fenster.



- Hier klickst du oben rechts auf das Symbol, unter dem **HIDE** steht.

Damit schließt sich das Willkommens-Fenster, und was dich erwartet, könnte etwa so aussehen: