



mitp

Daniel Braun

3. Auflage



LET'S PLAY

Programmieren lernen

mit Python und Minecraft

Plugins erstellen
ohne Vorkenntnisse

Für
Bukkit &
Spigot
unter Windows, Linux und macOS

KEIN OFFIZIELLES MINECRAFTPRODUKT.
NICHT VON MOJANG GENEHMIGT ODER MIT MOJANG VERBUNDEN.



Hinweis des Verlages zum Urheberrecht und Digitalen Rechtemanagement (DRM)

Der Verlag räumt Ihnen mit dem Kauf des ebooks 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 ebooks vor Missbrauch des Urheberrechts durch ein digitales Rechtemanagement. Bei Kauf im Webshop des Verlages werden die ebooks mit einem nicht sichtbaren digitalen Wasserzeichen individuell pro Nutzer signiert.

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

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

Kapitel 1: Minecraft-Server

- 1.1 Java installieren
- 1.2 Installation
 - 1.2.1 CraftBukkit
 - 1.2.2 Spigot
- 1.3 Konfiguration
- 1.4 Befehle
- 1.5 Verbinden
- 1.6 Updates

Kapitel 2: Python

- 2.1 Programmiersprachen
- 2.2 Besonderheiten von Python
- 2.3 Einrichtung
 - 2.3.1 Jython
 - 2.3.2 PPLoader
- 2.4 Editor

Kapitel 3: Das erste Plugin

- 3.1 Ordner anlegen
- 3.2 plugin.py

- 3.3 plugin.yml
- 3.4 Testen
- 3.5 Fehler finden
- 3.6 Entdecken

Kapitel 4: Chat-Kommandos

- 4.1 Eigene Befehle definieren
- 4.2 Chat-Nachrichten versenden

Kapitel 5: Variablen

- 5.1 Namen
- 5.2 Werte
 - 5.2.1 Operatoren
 - 5.2.2 Umwandlung
 - 5.2.3 Runden
- 5.3 +1-Plugin
- 5.4 Listen und Arrays
- 5.5 Konstanten

Kapitel 6: Schleifen

- 6.1 Kürbis-Plugin
 - 6.1.1 Positionierung
 - 6.1.2 Blöcke platzieren
- 6.2 Die verschiedenen Schleifen
 - 6.2.1 for-Schleife
 - 6.2.2 while-Schleife
 - 6.2.3 Verschachtelte Schleifen

Kapitel 7: Verzweigungen

7.1 if

7.2 else

7.3 elif

Kapitel 8: Funktionen

8.1 Deklaration von Funktionen

8.2 Rückgabewerte

8.3 Parameter

8.4 Anwendungsbeispiel

Kapitel 9: Bauen

9.1 Notunterkunft

9.1.1 Decke und Wände

9.1.2 Tür

9.1.3 Bett

9.1.4 Fackel

9.2 Runde Objekte

9.2.1 Kreise

9.2.2 Kugeln

Kapitel 10: Schilder

10.1 Hängende Schilder

10.2 Stehende Schilder

10.3 Text festlegen

10.3.1 Farbe

10.3.2 Formatierung

10.4 Schilder-Plugin

10.4.1 Wiederholung: Listen

10.4.2 Das Plugin

Kapitel 11: Listener

- 11.1 Grundgerüst
- 11.2 Spieler-Events
- 11.3 Kreaturen-Events
- 11.4 Block-Events
- 11.5 Inventar-Events
- 11.6 Server-Events
- 11.7 Fahrzeug-Events
- 11.8 Wetter-Events
- 11.9 Welt-Events
- 11.10 Mehrere Listener in einem Plugin

Kapitel 12: Klassen und Objekte

- 12.1 Die ganze Welt ist ein Objekt
- 12.2 Funktionen in Klassen
- 12.3 Zugriffskontrolle
- 12.4 Vererbung
- 12.5 Mehrfachvererbung und mehrstufige Vererbung
- 12.6 Bau-Plugin

Kapitel 13: Crafting-Rezepte

- 13.1 Rezepte festlegen
- 13.2 Eigene Rezepte entwerfen
- 13.3 Feuerschwert
- 13.4 Enderbogen

Kapitel 14: Informationen dauerhaft speichern

- 14.1 Konfigurationsdateien
 - 14.1.1 Lesen
 - 14.1.2 Schreiben
- 14.2 Objekte in Dateien speichern

Kapitel 15: Eigene Spielmodi entwickeln

- 15.1 Schneeballschlacht
 - 15.1.1 Schneebälle verteilen
 - 15.1.2 Schneebälle auffüllen
 - 15.1.3 Punkte zählen
 - 15.1.4 Punkte dauerhaft speichern
 - 15.1.5 Highscore-Liste anzeigen
 - 15.1.6 Vollständiger Quellcode
- 15.2 Sammelspiel
 - 15.2.1 Aufbau des Plugins
 - 15.2.2 Plugin starten
 - 15.2.3 Spieler betritt den Server
 - 15.2.4 Gegenstände zählen
 - 15.2.5 Auftrag anzeigen
 - 15.2.6 Vollständiger Quellcode

Kapitel 16: Eigenständige Python-Programme

- 16.1 Python einrichten
- 16.2 Grundgerüst
- 16.3 Ein- und Ausgabe
- 16.4 Quiz programmieren

Anhang A: Befehlsreferenz

Anhang B: Materialien

Daniel Braun

Let's Play: Programmieren lernen mit Python und Minecraft

Plugins erstellen ohne Vorkenntnisse



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 978-3-7475-0507-6
3. 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

KEIN OFFIZIELLES MINECRAFT-PRODUKT.
NICHT VON MOJANG GENEHMIGT ODER MIT MOJANG
VERBUNDEN.

Minecraft and its graphics are a trademark of Mojang Synergies AB.

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: Sabine Schulz, Janina Bahlmann

Sprachkorrektorat: Petra Kleinwegen

Coverbild: Daniel Braun

electronic **publication**: Ill-satz, Husby, www.drei-satz.de

Dieses Ebook 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 ebooks 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 ebooks vor Missbrauch des Urheberrechts durch ein digitales Rechtemanagement. Bei Kauf im Webshop des Verlages werden die ebooks mit einem nicht sichtbaren digitalen Wasserzeichen individuell pro Nutzer signiert.

Bei Kauf in anderen ebook-Webshops erfolgt die Signatur durch die Shopbetreiber. Angaben zu diesem DRM finden Sie

auf den Seiten der jeweiligen Anbieter.

Einleitung

Liebe Leserin, lieber Leser,

die Welt von Minecraft steckt voller Dinge, die es zu entdecken gilt. Verschiedene Landschaften, Hunderte verschiedene Gegenstände und allerlei Tiere und Monster sind nur einige der Dinge, die dich erwarten.

Irgendwann ist aber selbst diese Vielzahl an Möglichkeiten erschöpft und man hat das Gefühl, alles schon einmal gesehen oder gemacht zu haben. Wenn es dir so geht, dann ist dieses Buch genau richtig für dich. Denn im Verlaufe dieses Buches lernst du, wie man mithilfe von Python und dem Bukkit- oder Spigot-Server eigene Erweiterungen für die Java-Edition von Minecraft programmiert, sogenannte Plugins, die du dann zusammen mit deinen Freunden auf deinem eigenen Minecraft-Server ausprobieren kannst.

Egal ob du neue Crafting-Rezepte entwerfen, ganze Häuser mit einem einfachen Chat-Befehl bauen oder sogar einen eigenen Spielmodus programmieren möchtest, mit eigenen Plugins steckt die Welt von Minecraft wieder voller Herausforderungen und Dinge, die entdeckt werden wollen. Und ganz nebenbei lernst du auch noch zu programmieren – und wer weiß, vielleicht kommt das nächste Minecraft eines Tages von dir!

Bevor es soweit ist, liegt allerdings noch ein ordentliches Stück Weg vor dir. Die ersten beiden Kapitel dieses Buches beschäftigen sich deshalb zunächst einmal damit, wie du deinen Computer für das Programmieren und Testen eigener Plugins vorbereitest. Dazu wird dir erklärt, wie du den Bukkit- oder Spigot-Server installierst, der in diesem Buch verwendet wird, ihn nach deinen Wünschen konfigurierst

und wie du deinen Computer so einrichtest, dass du Python-Plugins schreiben kannst.

Direkt im Anschluss geht es im [dritten Kapitel](#) ohne Umschweife direkt los mit dem Programmieren deines ersten eigenen Plugins. Die ersten Schritte werden dir vielleicht noch etwas unspektakulär vorkommen, aber mit jedem der folgenden Kapitel wirst du immer mehr Möglichkeiten besitzen, um immer ausgeklügeltere Plugins zu programmieren. Schon im [vierten Kapitel](#) wirst du zum Beispiel lernen, wie du eigene Chat-Befehle programmieren und verwenden kannst.

Die [Kapitel 5 bis 8](#) beschäftigen sich mit grundlegenden Konzepten des Programmierens im Allgemeinen und der Programmiersprache Python im Besonderen. Was du hier liest, wird dir nicht nur beim Programmieren von Minecraft-Plugins helfen, sondern beim Programmieren jedes Programms in jeder Programmiersprache. Trotzdem entstehen dabei natürlich auch einige praktische kleine Plugins, wie zum Beispiel das Mauer-Plugin, das es dir erlaubt, mit einem einfachen Chat-Befehl auf die Schnelle eine Mauer zu bauen, wenn du möchtest, sogar aus purem Gold.

Das [neunte Kapitel](#) widmet sich dann ganz der Baukunst. Häuser, Kreise und Kugeln – hier wird kein Block auf dem anderen gelassen. Und wenn du schon einmal versucht hast, eine Kugel in Minecraft von Hand zu bauen, dann wirst du ganz besonders die Dienste des Kugel-Plugins zu schätzen wissen, das dir auf Knopfdruck eine nahezu perfekte Kugel zaubern kann. Weiter geht es danach mit dem Bau von Schildern, denen das gesamte [zehnte Kapitel](#) gewidmet ist.

Wenn dir selbst ein Knopfdruck zum Bauen noch zu viel ist, dann wird dir das [elfte Kapitel](#) besonders gefallen. Dort geht es nämlich um Plugins, die vollautomatisch auf Geschehnisse in der Spielwelt reagieren. Egal ob ein Creeper über die Karte schleicht, ein Spieler etwas isst oder ein Baum wächst, hier lernst du wie deinem Plugin nichts mehr von dem entgeht, was auf deinem Server passiert und natürlich auch, wie du darauf reagieren kannst.

Ein sehr grundlegendes und wichtiges Konzept moderner Programmiersprachen, die objektorientierte Programmierung, lernst du in [Kapitel 12](#) kennen, hier dreht sich alles um Objekte und Klassen.

Falls du dich um die umherschleichenden Creeper aber doch lieber ganz manuell kümmern möchtest, kannst du die Informationen aus [Kapitel 13](#) nutzen, um ganz eigene Waffen zu kreieren. In diesem Kapitel geht es nämlich um das Erstellen eigener Crafting-Rezepte und ein Beispiel, das dir dort begegnen wird, ist ein Rezept für ein Flammenschwert, das alles in Brand setzt, worauf es trifft.

Das [vierzehnte Kapitel](#) ist dann wieder etwas technischer, aber nicht weniger nützlich. Hier lernst du nämlich, wie du Informationen dauerhaft speichern kannst, die auch dann erhalten bleiben, wenn der Server zwischenzeitlich ausgeschaltet wird. Das ist zum Beispiel praktisch, wenn du wie in [Kapitel 15](#) eigene Spielmodi kreieren willst, also sozusagen ein Spiel im Spiel. Wie wäre es zum Beispiel mit einem Schneeballschlacht-Mod mit eigener Highscore-Liste, die die Treffer zählt? Oder lieber ein lustiges Suchspiel, bei dem der Gewinner mit Erfahrungspunkten oder wertvollen Gegenständen belohnt wird? Ganz wie du möchtest: Deiner Kreativität sind keine Grenzen gesetzt!

Im [letzten Kapitel](#) bekommst du dann noch einen kurzen Ausblick darauf, was du mit deinen neu gewonnenen Programmierfähigkeiten noch anstellen kannst, außer Minecraft-Plugins zu programmieren. Denn wenn du am Ende des Buches angelangt bist, hört der Spaß noch lange nicht auf: Nun hast du alle Werkzeuge und alles Wissen, das du benötigst, um ganz eigene Plugins ganz nach deinen Vorstellungen zu entwerfen. Dabei helfen dir einige Listen im [Anhang](#) des Buches, in denen du Befehle und besonders häufig benötigte Dinge schnell nachschlagen kannst. Denn egal wie erfahren man als Programmierer ist, alles kann und muss man nicht auswendig können, man muss nur wissen, wo man es nachschlagen kann – und genau dazu dient der [Anhang](#) dieses Buches.

Falls du Fragen, Kritik oder Anregungen zum Buch oder generell zu Minecraft-Plugins hast, kannst du mich gerne jederzeit kontaktieren. Du erreichst mich per Mail an info@daniel-braun.com oder über meine Website www.daniel-braun.com.

Downloads zum Buch

Unter der Webadresse buch.daniel-braun.com findest du:

- Links zu allen Downloads, die du benötigst
- Alle Plugins, die du im Rahmen des Buches programmieren wirst, falls du den Code nicht aus dem Buch abtippen möchtest

Mein besonderer Dank gilt Karl-Heinz Barzen, der den Entstehungsprozess dieses Buches unermüdlich mit zahlreichen hilfreichen Kommentaren und Anmerkungen begleitet und damit einen wichtigen Beitrag dazu geleistet hat, dass dieses Buch möglichst verständlich und einsteigerfreundlich wird.

Nun wünsche ich dir aber vor allem viel Spaß beim Lesen, Programmieren und Entdecken!

Daniel Braun

Kapitel 1

Minecraft-Server

Alleine Minecraft zu spielen, kann schon jede Menge Spaß machen, noch lustiger wird es aber, wenn du dich mit anderen Spielern zusammentust, um mit ihnen oder auch gegen sie zu spielen. Dazu kannst du dir entweder einen der hunderten öffentlichen Server aussuchen, die du überall im Internet findest, oder du kannst deinen eigenen Server nutzen – dann hast du die volle Kontrolle über alle Einstellungen. Noch mehr Spaß wird dir dein eigener Server machen, wenn du im Laufe des Buches lernst, immer ausgefeiltere Plugins für ihn zu programmieren, mit denen du Minecraft nach deinen Vorstellungen erweitern kannst.

Um deinen eigenen Server zu betreiben, benötigst du neben dem normalen Minecraft-Spiel, das auch **Client** genannt wird, noch ein weiteres Programm, nämlich den Minecraft-**Server**. Den »normalen« Minecraft-Server, manchmal auch »Vanilla-Server« genannt, kannst du auf der offiziellen Minecraft Webseite www.minecraft.net herunterladen. Neben dieser Version gibt es aber auch noch zahlreiche sogenannte Mods, also Modifikationen des Original-Servers. Als Mods oder Modifikationen bezeichnet man im Zusammenhang mit Spielen Versionen eines Spiels, die in irgendeiner Form verändert, also modifiziert wurden. Diese meist von Fans entwickelten Mods bieten häufig viele zusätzliche Funktionen und Annehmlichkeiten, über die der Vanilla-Server nicht verfügt, wie zum Beispiel auch die Möglichkeit eigene Plugins zu programmieren.

Merke

Das normale Minecraft-Spiel, das du auch startest, wenn du alleine spielst, wird **Client** genannt. Das Programm, das wir in diesem Kapitel installieren werden, das du benötigst, um mit Freunden zusammen spielen zu können, heißt hingegen **Server**.

Dieses Buch ist für gleich zwei der beliebtesten Server ausgelegt. Du kannst dich entscheiden zwischen dem **CraftBukkit**-Server, häufig auch einfach nur Bukkit genannt, und dem **Spigot**-Server. Da der Spigot- auf dem Bukkit-Server aufbaut, funktionieren alle Plugins, die wir im Rahmen dieses Buches programmieren werden, auf beiden Servern. Der einzige Unterschied liegt in der Administration der Server, hier bietet Spigot mehr Möglichkeiten, ist dafür in der Bedienung aber auch etwas komplexer. Außerdem ist der Spigot-Server etwas effizienter, was bedeutet, dass er insbesondere etwas weniger Arbeitsspeicher (RAM) benötigt. Für Anfänger, die zum ersten Mal einen eigenen Server betreiben, ist es daher ratsam zunächst auf Bukkit zu setzen; wer schon Erfahrung mit der Verwaltung eines Minecraft-Servers hat, kann sich auch an Spigot herantrauen. Ein Wechsel ist ohnehin jederzeit möglich.

1.1 Java installieren

Egal für welchen der beiden Server du dich entscheidest: Um sie später starten zu können, muss auf deinem Computer eine aktuelle Java-Version installiert sein, denn die Server sind, wie auch Minecraft selbst, in Java programmiert. Um das zu testen, kannst du unter Windows die Eingabeaufforderung öffnen, indem du den Namen einfach in das Suchfeld im Startmenü eingibst, beziehungsweise unter GNU/Linux und macOS ein Terminal

öffnest. Dort gibst du dann den Befehl `java -version` ein und bestätigst deine Eingabe mit der `Enter`-Taste. Siehst die darauf folgende Ausgabe aus wie in [Abbildung 1.1](#) gezeigt, so ist Java bereits korrekt auf deinem Computer installiert und du kannst zum nächsten Abschnitt springen.

```
C:\>java -version
openjdk version "17.0.1" 2021-10-19
OpenJDK Runtime Environment (build 17.0.1+12-39)
OpenJDK 64-Bit Server VM (build 17.0.1+12-39, mixed mode, sharing)
```

Abb. 1.1: Ausgabe nach `java -version`

Bekommst du stattdessen eine Meldung angezeigt wie »Der Befehl "java" ist entweder falsch geschrieben oder konnte nicht gefunden werden.«, so ist Java noch nicht auf deinem Computer installiert. In diesem Fall kannst du unter buch.daniel-braun.com einen Link zum Download der aktuellsten Java-Version finden.

Unter GNU/Linux kannst du Java direkt über den Paketmanager deiner Wahl installieren. Unter macOS und Windows lädst du zunächst ein gepacktes Verzeichnis herunter, das nach dem Download entpackt werden muss. Dieses Verzeichnis, das je nach Version zum Beispiel den Namen `jdk-17.0.1` trägt, kopierst du unter macOS in den Ordner `/library/Java/JavaVirtualMachines/` und unter Windows in ein beliebiges Verzeichnis deiner Wahl, zum Beispiel direkt in `c:\`. Unter Windows musst du dieses Verzeichnis dann noch zur sogenannten *PATH-Variable* hinzufügen. Dazu öffnest du zunächst die erweiterten Systemeinstellungen deines Computers.

Windows XP, Vista und 7: Unter Windows XP, Vista und 7 öffnest du dafür zunächst das Startmenü und dann die **SYSTEMSTEUERUNG**. Dort wählst du aus der Kategorie **SYSTEM**

UND SICHERHEIT den Eintrag **SYSTEM** aus und im sich danach öffnenden Fenster den Eintrag **ERWEITERTE SYSTEMEINSTELLUNGEN**.

Windows 8, 10 und 11: Unter Windows 8, 10 und 11 kannst du die erweiterten Systemeinstellungen öffnen, indem du den Begriff einfach direkt in die Suche eingibst. Alternativ kannst du auch zunächst mit der rechten Maustaste auf das Windows-Logo in der unteren linken Ecke klicken und dort dann auf **SYSTEM** und in dem sich öffnenden Fenster wieder auf **ERWEITERTE SYSTEMEINSTELLUNGEN**.

Nun solltest du, unabhängig von deiner verwendeten Windows-Version, das in [Abbildung 1.2](#) gezeigte Fenster sehen.

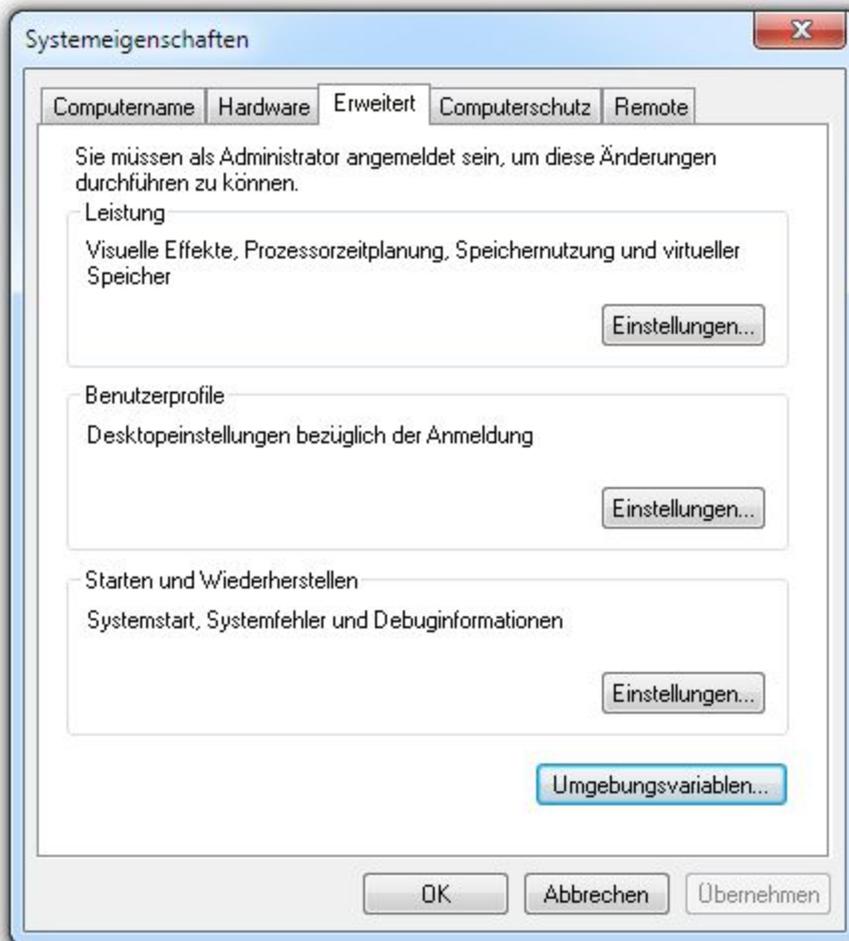


Abb. 1.2: Erweiterte Systemeinstellungen

Dort findest du in der rechten unteren Ecke einen Button mit der Beschriftung **UMGEBUNGSVARIABLEN**. Bei einem Klick darauf öffnet sich das in [Abbildung 1.3](#) gezeigte Fenster.

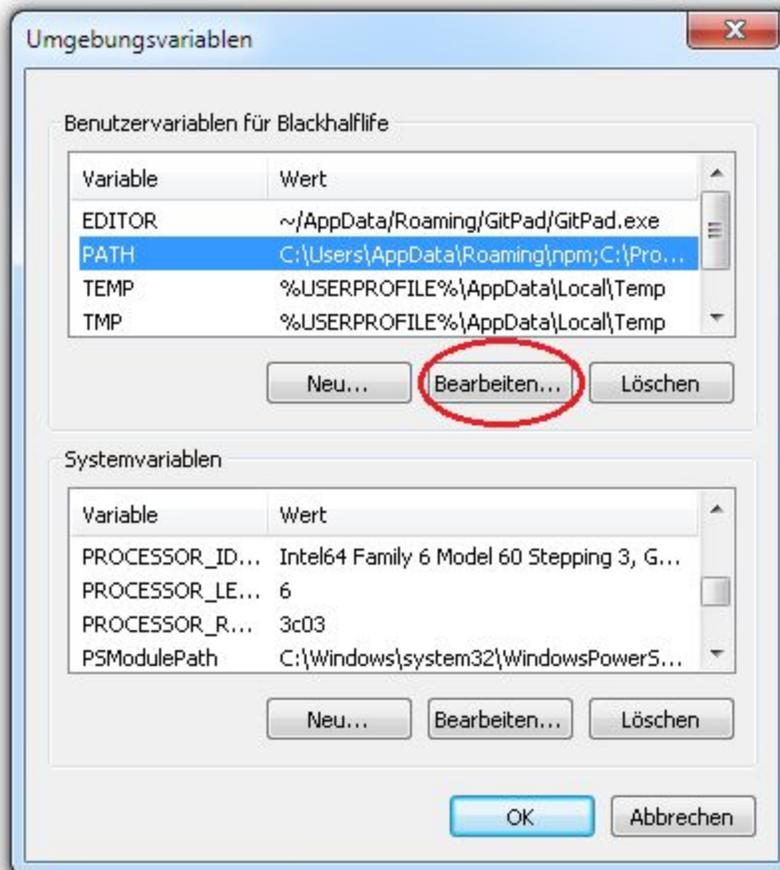


Abb. 1.3: Umgebungsvariablen

Dann wählst du, wie in [Abbildung 1.3](#) gezeigt, den Eintrag **PATH** aus und klickst anschließend auf **BEARBEITEN**. Sollte der Eintrag nicht vorhanden sein, so kannst du direkt zum nächsten Absatz springen. Danach öffnet sich ein langes Textfeld, in dem es schon zahlreiche Einträge gibt, die auf keinen Fall geändert werden dürfen. Stattdessen solltest du am Ende, abgetrennt durch ein Semikolon, den Pfad angeben, an den du zuvor das entpackte Verzeichnis kopiert hast, gefolgt von `\bin\`. Also zum Beispiel:

```
C:\jdk-17.0.1\bin\;
```

Je nachdem, welche Java-Version du installiert hast, kann der Pfad aber, insbesondere bei der Versionsnummer, leicht abweichen. Daher solltest du unbedingt darauf achten, den tatsächlichen Installationspfad zu nutzen. Danach musst du die Änderungen nur noch mit **OK** und **ÜBERNEHMEN** bestätigen.

Sollte es bei dir noch keinen Eintrag mit dem Namen **PATH** geben, so kannst du diesen ganz einfach selbst anlegen. Dazu klickst du statt auf **BEARBEITEN** einfach auf **NEU**. Im Fenster, das sich daraufhin öffnet, gibst du als **NAME DER VARIABLEN** das Wort **PATH** ein und als **WERT DER VARIABLEN** den Pfad zur Installation, beendet durch ein Semikolon, und bestätigst deine Eingabe mit **OK**. Anschließend sollte der Befehl `java -version` dann in der Eingabeaufforderung funktionieren.

Hinweis

Sollte beim späteren starten des Servers die in [Abbildung 1.4](#) gezeigte Fehlermeldung erscheinen, so ist die auf deinem Computer installierte Java-Version veraltet. Bitte führe die oben beschriebenen Schritte dann mit der neusten Java-Version durch.

```
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main" java.lang.UnsupportedClassVersionError: org/bukkit/craftbukkit/
bootstrap/Main has been compiled by a more recent version of the Java Runtime (class file
version 60.0), this version of the Java Runtime only recognizes class file versions up to
52.0

    at java.lang.ClassLoader.defineClass1(Native Method)
    at java.lang.ClassLoader.defineClass(Unknown Source)
    at java.security.SecureClassLoader.defineClass(Unknown Source)
    at java.net.URLClassLoader.defineClass(Unknown Source)
    at java.net.URLClassLoader.access$100(Unknown Source)
    at java.net.URLClassLoader$1.run(Unknown Source)
    at java.net.URLClassLoader$1.run(Unknown Source)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at sun.misc.Launcher$AppClassLoader.loadClass(Unknown Source)
    at java.lang.ClassLoader.loadClass(Unknown Source)
    at sun.launcher.LauncherHelper.checkAndLoadMain(Unknown Source)
```

Abb. 1.4: Fehlermeldung nach dem Starten des Servers bei veralteter Java-Version

1.2 Installation

An dieser Stelle musst du dich nun entscheiden, welchen Server du zum Testen deiner Plugins verwenden möchtest. Wenn du dich für den Bukkit-Server entscheidest, kannst du in [Abschnitt 1.2.1](#) weiterlesen, möchtest du lieber den Spigot-Server verwenden, dann kannst du direkt zu [Abschnitt 1.2.2](#) springen.

1.2.1 CraftBukkit

Einen Link zum Download der neusten Version des Bukkit-Servers findest auf der Website zum Buch unter buch.daniel-braun.com. Dabei handelt es sich um eine einzelne sogenannte Jar-Datei, die, je nach Version, zum Beispiel den Namen `craftbukkit-1.18.jar` trägt. Zunächst solltest du einen leeren Ordner anlegen, in den du diese Datei kopierst. Prinzipiell kannst du diesen Ordner nennen, wie du möchtest, im Verlaufe des Buches werden wir davon ausgehen, dass der Ordner den Namen `server` trägt und in

C:\server unter Windows, /home/Benutzername/server unter GNU/Linux beziehungsweise /Users/Benutzername/server unter OS X, abgelegt ist.

Um den Server nun zum ersten Mal zu starten, musst du zunächst wieder die Eingabeaufforderung beziehungsweise ein Terminal öffnen, und in den Server-Ordner wechseln. Das kannst du mithilfe des Befehls `cd`. Die englische Abkürzung steht für »change directory«, also »Ordner wechseln«, und genau das, also zwischen verschiedenen Ordnern hin- und herwechseln, kann man mit diesem Befehl auch tun. Unter Windows gibst du also zum Beispiel `cd C:\server` ein und unter GNU/Linux `cd /home/Benutzername/server`. Bist du erst einmal im richtigen Ordner, so kannst du den Server mit dem Befehl `java -jar craftbukkit-1.18.jar` starten. Beim ersten Starten wirst du aber zunächst einmal nur die in [Abbildung 1.5](#) gezeigten Warnhinweise sehen.

```
C:\Users\Blackhalflife>cd C:\server
C:\server>java -jar craftbukkit-1.9.jar
Loading libraries, please wait...
[15:43:15 INFO]: Starting minecraft server version 1.9
[15:43:15 WARN]: To start the server with more ram, launch it as "java -Xmx1024M
-Xms1024M -jar minecraft_server.jar"
[15:43:15 INFO]: Loading properties
[15:43:15 WARN]: server.properties does not exist
[15:43:15 INFO]: Generating new properties file
[15:43:15 WARN]: Failed to load eula.txt
[15:43:15 INFO]: You need to agree to the EULA in order to run the server. Go to
eula.txt for more info.
[15:43:15 INFO]: Stopping server
>
C:\server>
```

Abb. 1.5: Ausgabe nach dem ersten Starten des Servers

Merke

Der Server wird mit dem Befehl `java -jar craftbukkit-1.18.jar` gestartet. Achte darauf, die Versionsnummer im

Befehl an die von dir verwendete Server-Version anzupassen.

Dort steht im Wesentlichen, dass du zunächst den Nutzungsbedingungen zustimmen musst, bevor du den Server verwenden kannst. Wenn du jetzt einen Blick in deinen Server-Ordner wirfst, dann wird dir auffallen, dass es dort, wie in [Abbildung 1.6](#), nun zwei weitere Dateien und einen Ordner gibt.

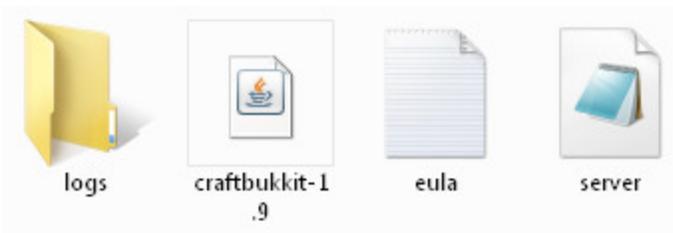


Abb. 1.6: Inhalt des Server-Ordners nach dem ersten Start

Tipp

Sollte beim Starten des Servers folgender Hinweis angezeigt werden

```
*** Error, this build is outdated ***  
*** Please download a new build as per instructions from  
https://www.spigotmc.org/go/outdated-spigot ***  
*** Server will start in 20 seconds ***
```

dann bedeutet das, dass du nicht die aktuellste Version des Servers benutzt. Du kannst den Server trotzdem weiterhin wie gewohnt nutzen oder eine neuere Version aus dem Internet herunterladen.

Um den Nutzungsbedingungen zuzustimmen, musst du die dort nun vorhandene Datei `eula.txt` öffnen. In dieser Datei findest du auch einen Link, unter dem du die Bedingungen lesen kannst. Wenn du diesen Link öffnest, wirst du auf die offizielle Seite des Minecraft-Herstellers Mojang geleitet, wo du die Nutzungsbedingungen glücklicherweise auch auf Deutsch vorfindest. Dort wird geregelt, was du mit dem Spiel und dem Server machen darfst – und was nicht. Außerdem steht dort auch explizit, dass du, solltest du unter 18 sein, die Zustimmung eines gesetzlichen Vertreters einholen musst, also zum Beispiel eines Elternteils. Auf jeden Fall solltest du die Bedingungen sorgfältig lesen.

Den Inhalt der `eula.txt` findest du auch in [Listing 1.1](#). Bist du mit den Bedingungen einverstanden, so kannst du dies kenntlich machen, indem du die letzte Zeile der Datei von `eula=false` zu `eula=true` änderst. Nur wenn du das tust, kannst du den Server benutzen. Genau das wird in der ersten Zeile der Datei auf Englisch erklärt.

```
#By changing the setting below to TRUE you are indicating
your agreement to our EULA
(https://account.mojang.com/documents/minecraft_eula).
#Mon Apr 01 13:37:00 BST 2020
eula=false
```

Listing 1.1: Inhalt der Datei `eula.txt`

Wenn du die Änderungen gespeichert hast, kannst du wieder versuchen, den Server mit dem Befehl `java -jar craftbukkit-1.18.jar` zu starten. Der Startvorgang wird dieses Mal wahrscheinlich eine Weile dauern und es werden sehr viele Zeilen relativ schnell über den Bildschirm laufen. Wichtig ist besonders die letzte Zeile. Steht dort so etwas

Wie Done (13,370s)! For help, type "help" or "?", dann bedeutet das, dass dein Server nun problemlos läuft. Ein erneuter Blick in den Server-Ordner wird dir zeigen, dass es dort nun, wie in [Abbildung 1.7](#) zu sehen, noch einmal deutlich mehr Dateien gibt.

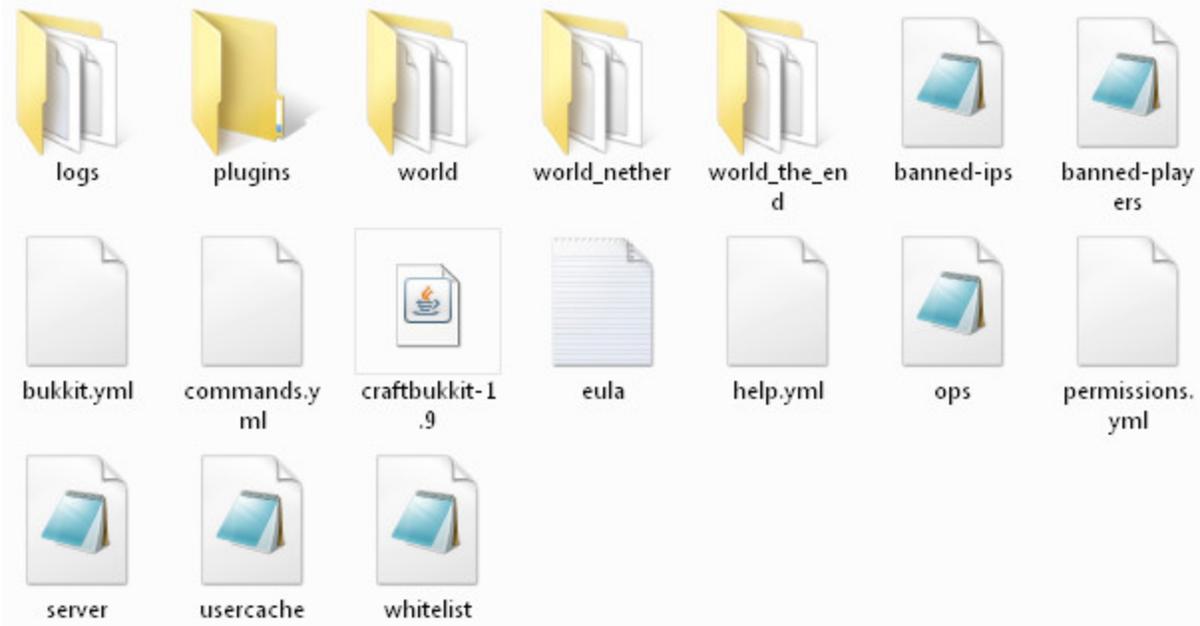


Abb. 1.7: Inhalt des Server-Ordners nach erfolgreichem Starten des Servers

Hinweis

Der Server läuft nur, solange das entsprechende Fenster der Eingabeaufforderung beziehungsweise des Terminals geöffnet bleibt. Schließt du das Fenster, so wird auch der Server geschlossen.

1.2.2 Spigot