

# Introduzione: iniziamo da qui

Benvenuto!

Grazie per aver scelto questo libro. Spero che non lo troverai noioso e che ti divertirai a leggerlo. Se non ti è mai capitato di scrivere un programma prima d'ora, non preoccuparti. Partiremo dall'inizio e faremo un passo alla volta. Non serve che tu abbia già esperienza.

Minecraft piace a tutti. Uno dei motivi del suo successo è che permette a chiunque di partecipare alla realizzazione del gioco. Ti dà infatti la possibilità di costruire e fabbricare. Che sia un rifugio d'emergenza in modalità Sopravvivenza, una simulazione di Pietrarossa o il tuo castello, potrai creare qualcosa da solo.

A volte, però, le funzioni che il gioco ti mette a disposizione non sono sufficienti. Potresti volere delle mucche che volano mentre vanno a fuoco, per esempio, o avere la possibilità di rinchiodare un nemico in una gabbia di roccia. Per conquistare queste e molte altre capacità, dovrai aggiungerle tu al gioco.

Le applicazioni sul tuo computer o sul tuo smartphone sono scritte in un testo speciale chiamato *linguaggio di programmazione*. Un linguaggio di programmazione non è complesso da imparare come le lingue che si parlano nel mondo (come il cinese o il tedesco) ma è diverso da quello che usi tutti i giorni per scrivere e comunicare.

I linguaggi di programmazione sono tantissimi. Alcuni sono famosi ma poco potenti. Altri vengono usati solo da poche persone molto preparate, ma sono incredibilmente solidi e molto più difficili da capire.

Minecraft è scritto nel linguaggio di programmazione chiamato Java, che è piuttosto potente ma che in alcune sue parti è abbastanza complicato e può creare confusione. Qui ti spiegheremo i suoi elementi di base, i più facili, e tralascieremo quelli più difficili.

In questo libro ti insegneremo in poco tempo a programmare in Java. Alla fine, ne saprai abbastanza da poter svolgere le operazioni più comuni di questo linguaggio e creare da solo i tuoi plug-in di Minecraft. (Se non sai cos'è un plug-in, te lo spieghiamo tra poco.)

Vedremo anche come impostare il tuo server di Minecraft, creando sul cloud delle copie di riserva del codice che scrivi, mentre verso la fine daremo un'occhiata ad alcune tecniche di programmazione avanzate.

## Questo libro fa per te?

Questo libro è fatto per i lettori che non hanno alcuna esperienza di programmazione ma che vogliono intervenire in Minecraft da veri esperti. Se non hai familiarità con il gioco, puoi trovare un aiuto guardando i numerosi video in Internet o leggendo alcuni libri che ne parlano.

### NOTA

Se non sei ancora molto esperto di Minecraft, se hai dei dubbi o se vuoi avere una panoramica completa sul gioco e sui suoi elementi, puoi visitare il wiki in lingua italiana all'indirizzo [http://minecraft-it.gamepedia.com/Minecraft\\_Wiki](http://minecraft-it.gamepedia.com/Minecraft_Wiki).

Se però sei qui, suppongo che tu sia già un fan entusiasta di Minecraft, e che tu voglia imparare qualcosa sulla sua programmazione.

Se hai meno di 8-9 anni o se hai qualche difficoltà a capire come si programma in Java, puoi fare una piccola “deviazione” e provare prima a lavorare con un linguaggio più semplice.

Per esempio, Scratch e la sua versione più recente, Snap!, sono due piccoli linguaggi che mostrano attraverso le immagini come si combinano i vari elementi della programmazione, e che possono aiutarti a imparare i concetti di base (vai agli indirizzi <http://scratch.mit.edu> e <http://snap.berkeley.edu>).

### NOTA

In questo libro troverai dei riferimenti a siti o libri che purtroppo si trovano solo in inglese. Se non conosci questa lingua, dovrai farti aiutare a interpretarli.

Una volta che avrai acquisito un po' di familiarità con l'argomento, un linguaggio basato su testo come Java ti sarà più comprensibile.

Se invece sai già come fare, per partire ti basterà avere un computer con installato Windows, OS X o Linux.

## Cosa serve per iniziare

Minecraft è un'applicazione *client-server*, cioè divisa in due parti.

La prima parte è il *client*, cioè l'applicazione che gira sul tuo computer, portatile o desktop. Il client mostra le immagini del mondo di Minecraft e accetta i tuoi comandi quando ti sposti e agisci nel gioco.

La seconda parte è il *server*, che tiene traccia di tutto quello che avviene nel gioco, compresi i giocatori che sono connessi, i loro inventari, quello che hanno costruito, dove si trovano e così via. La maggior parte del tempo il server lavora su una macchina dall'altra parte del paese, ma potrebbe girare anche sul tuo computer.

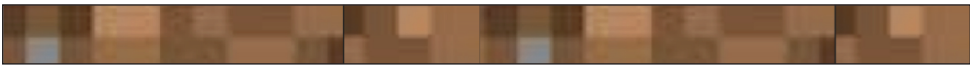
Il client e il server parlano tra loro attraverso Internet, esattamente come fai tu quando usi un browser web per collegarti e giocare online o vedere un po' di foto di gatti.

Per aggiungere o modificare una funzionalità in Minecraft, devi aggiungere o modificare il programma Java sul server. Ecco quello che imparerai in questo libro: programmare in Java scrivendo istruzioni (che chiameremo “codice sorgente”, solo “codice” oppure “il programma”) per creare dei plug-in per il server di Minecraft. Un *plug-in* non è altro che un pezzo di codice che aggiungi a un programma esistente. (Alcuni scrivono delle modifiche di Minecraft, chiamate *mod*, all'interno del client grafico, ma qui non ce ne occuperemo.)

Prima di iniziare a lavorare con i plug-in, devi impostare un server Minecraft locale, che ti servirà per fare i test, e installare il linguaggio di programmazione Java e un paio di altre applicazioni. Ci occuperemo di tutto questo nei primi capitoli. Installare tutti questi elementi non è un gran divertimento, anzi, è proprio noioso, ma cercheremo di sbrigarcela in fretta.

Per aiutarti a tenere una traccia dei tuoi progressi, alla fine di ogni capitolo trovi un paragrafo chiamato “La tua toolbox” che riporta una barra con la percentuale di nuove conoscenze che hai acquisito.

Partirai da zero...



... e finirai con un bel 100%:



In alcuni capitoli andremo più rapidamente che in altri, e in alcuni troverai più informazioni che in altri, ma farai sempre e comunque dei progressi.

## Niente panico

Visto che avremo tanto di cui parlare ma poco spazio per farlo, prima ti mostrerò le cose e come usarle e poi te le illustrerò nel dettaglio. A volte ti sentirai come se ti avessero buttato in una piscina senza fondo: sarai disorientato, farai fatica a trovare la direzione, ma non preoccuparti. Da lì a poco ti spiegherò tutto. Potrà capitare che tu riesca a usare qualche caratteristica con successo anche se in realtà non hai capito come funziona.

Questo capita anche nella vita di tutti i giorni: puoi accendere una lampada e utilizzarla senza sapere come viene generata l'elettricità. Volendo, potresti anche costruirne una senza avere la minima idea di com'è fatto un generatore. Quello che ti serve sapere è solo come i vari pezzi stanno insieme.

Ed è proprio su come assemblare le varie parti che ci concentreremo.

## Se ti serve aiuto

Su Internet trovi uno spazio dedicato a questo libro con domande, risposte, aggiornamenti e suggerimenti. Vai alla pagina <https://pragprog.com/book/ahmine2> e fai clic sulla scheda *Discuss*.

## Scarica i file degli esempi

Dalla stessa pagina, fai clic su *Source code* per scaricare tutto il codice sorgente del libro; il link esatto è <https://media.pragprog.com/titles/ahmine2/code/ahmine2-code.zip>. Scaricalo sul Desktop del tuo computer (utilizzerai il Desktop per la maggior parte del lavoro, ma ne parleremo meglio in seguito).

Quello che stai scaricando è un file archivio compresso, quindi dovrai decomprimerlo. Se lavori in OS X o Linux puoi farlo dalla riga di comando; se invece lavori con Windows puoi usare WinZip o 7-Zip (disponibili agli indirizzi <http://www.winzip.com/it/downwz.html> o <http://www.7-zip.org>).

Fatto? Bene!

Man mano che andremo avanti, conoscerai nuovi strumenti o imparerai nuovi modi per impiegarli. Terremo traccia di tutto questo nella “toolbox” – la tua cassetta degli attrezzi – alla fine di ogni capitolo. Arrivati in fondo al libro, la tua toolbox sarà abbastanza ricca da permetterti di progettare i tuoi plug-in e il relativo codice da zero!

## Alcune regole

Il codice o i comandi che ti mostro come esempio appaiono così:

```
$ Ho digitato questa riga come esempio per te.
```

Il codice o i comandi che invece dovrai digitare ti appaiono in grassetto:

**\$ questa parte devi digitarla tu (ma senza il segno del dollaro)**

Le parole in *corsivo* sono dei segnaposto, quindi non devi digitarle direttamente. Per esempio, in una riga come:

```
me.chat(string msg);
```

devi sostituire solo la parte in corsivo, così:

```
me.chat("Stanno arrivando dei Creeper");
```

Pronto? E allora iniziamo!