

## Over dit boek

Duizenden woorden kunnen een gezicht niet beter beschrijven dan een foto. Dit principe geldt ook bij het leren programmeren. Toen ik opleidingen volgde, was de focus gericht op theoretische uitleg. Elke les bevatte vele termen en technische woorden. Door mijn achtergrond en ervaring als docent vroeg ik me af of die saaie methode de beste manier was om te leren programmeren.

In de realiteit is het werken als programmeur anders dan wat je op de meeste traditionele opleidingen leert. In de praktijk vraagt niemand of je een bepaalde Java-term kent, maar je krijgt honderden regels code en je moet daarmee kunnen omgaan. Je wordt bijvoorbeeld gevraagd om nieuwe functionaliteiten te bouwen. Soms moet je Java-klassen schrijven of een programma ontwikkelen. Als je niet bekend bent met de code, zal de theorie weinig voor je kunnen betekenen.

Het goede nieuws is dat je zodra de code begint te begrijpen, je daar niet meer mee kunt stoppen. Veel programmeurs werken 's avonds door of zelfs tot in de nacht, omdat het erg creatief en interessant werk is. Dit boek probeert studenten voor te bereiden op wat er later van hen wordt verwacht en voorkomt dat ze technische woorden uit hun hoofd leren.

Hiermee bedoel ik niet dat theorie niet belangrijk is, maar het benadrukken van allerlei theorieën vanaf het begin leidt minder snel tot resultaat en is ineffectiever.

Als je begint met programmeren, zoek je zelf bepaalde begrippen op om problemen op te lossen. In dergelijke gevallen leer je ook veel van de theorie want die ondersteunt wat je in de praktijk doet.

Als je aan de slag gaat als programmeur kun je een aantal soorten opdrachten verwachten. Het maakt daarbij niet zoveel uit bij wat voor soort bedrijf je werkt.

De volgende programmeeropdrachten kun je verwachten:

- 1 Het begrijpen van programmacode die door andere programmeurs is geschreven.
- 2 Het bouwen van nieuwe functionaliteiten (features) voor bestaande software.
- 3 Het opsporen van bugs en problemen in een bestaand programma.
- 4 Het ontwikkelen van nieuwe programma's (vanaf scratch).

Voor het laatste punt moet je meestal een paar jaar ervaring hebben om dat te kunnen doen.

Elk hoofdstuk in dit boek begint met een beknopte uitleg over een bepaald onderwerp en bevat soms voorbeeldprogramma's. Daarna ga je werken met quizen.

- 1 Om het juiste antwoord te kunnen kiezen moet je de code begrijpen. Dat is vergelijkbaar met het eerste punt van ons lijstje. Als het je niet lukt om het juiste antwoord te vinden, kun je de stap-voor-stap-uitleg van het antwoord lezen. Dit is een zeer praktische methode om het proces te begrijpen.
- 2 Na elke quiz wordt je gevraagd een klein stukje code aan de quiz toe te voegen om een bepaald doel te bereiken. Er wordt soms ook gevraagd om iets te onderzoeken, waarbij je andere mogelijke resultaten bestudeert.
- 3 Bij sommige quizen wordt je gewaarschuwd waarom een bepaalde verandering tot een oneindige loop of een probleem kan leiden.
- 4 Vanaf het vijfde hoofdstuk krijg je opdrachten over het onderwerp van de hoofdstukken.

Programmeren is het oplossen van problemen en dat is de interessantste methode om dat te leren. Als je een probleem niet direct kunt oplossen, begin je na te denken over een oplossing. Er is ook rekening mee gehouden dat sommige Java-onderwerpen niet zo vaak voorkomen, terwijl je andere in bijna elk programma ziet.

Op [www.sarmarroof.com](http://www.sarmarroof.com) is meer informatie over dit boek te vinden. Hier vind je een discussieforum waar iedere gebruiker van het boek vragen kan stellen en opmerkingen kan plaatsen. Daar vind je ook een stap-voor-stap-uitleg hoe je de code kunt opzetten in Eclipse.

*Sar Maroof*