

# Inhoud in vogelvlucht

<b>Inleiding</b> .....	1
<b>Deel 1: Aan de slag met Python</b> .....	5
HOOFDSTUK 1: Met je computer praten .....	7
HOOFDSTUK 2: Python downloaden en installeren .....	21
HOOFDSTUK 3: Met Python communiceren .....	39
HOOFDSTUK 4: Je eerste app schrijven .....	57
<b>Deel 2: Geen woorden maar daden</b> .....	81
HOOFDSTUK 5: Gegevens opslaan en aanpassen .....	83
HOOFDSTUK 6: Gegevens beheren .....	95
HOOFDSTUK 7: Beslissingen nemen .....	119
HOOFDSTUK 8: Herhaalde taken uitvoeren .....	137
HOOFDSTUK 9: Fouten verwerken .....	153
<b>Deel 3: Algemene taken verrichten</b> .....	187
HOOFDSTUK 10: Met modules communiceren .....	189
HOOFDSTUK 11: Met strings werken .....	209
HOOFDSTUK 12: Lijsten beheren .....	229
HOOFDSTUK 13: Allerlei soorten gegevens verzamelen .....	249
HOOFDSTUK 14: Klassen maken en gebruiken .....	271
<b>Deel 4: Geavanceerde taken verrichten</b> .....	295
HOOFDSTUK 15: Gegevens in bestanden opslaan .....	297
HOOFDSTUK 16: Een e-mailbericht versturen .....	315
<b>Deel 5: Het deel van de tientallen</b> .....	333
HOOFDSTUK 17: Tien prachtige programmeerbronnen .....	335
HOOFDSTUK 18: Tien manieren om met Python de kost te verdienen .....	345
HOOFDSTUK 19: Tien interessante tools .....	355
HOOFDSTUK 20: Tien bibliotheken die je moet kennen .....	365
<b>Index</b> .....	375

# Inleiding

---

**S**nel! Met welke programmeertaal kun je apps voor elk populair besturingssysteem schrijven? Weet je het? Inderdaad, dat is Python. Het ongelofelijke aan Python is dat je echt een app op één besturingssysteem kunt programmeren (bijvoorbeeld Windows) en op een ander besturingssysteem kunt gebruiken (zoals Mac OS X of Linux). In tegenstelling tot andere programmeertalen die platformafhankelijkheid beloven, is dit met Python echt mogelijk. In dit geval is de belofte net zo veel waard als het resultaat.

Python biedt betrouwbare code en een bondige syntaxis waarmee je apps kunt maken met minder regels code dan met andere programmeertalen. Bovendien wordt Python voor allerlei onderwerpen gebruikt door mensen zonder programmeerervaring. Sommige mensen zien Python als een scripttaal, maar in werkelijkheid is het veel meer. Blader bijvoorbeeld eens naar hoofdstuk 18 om een indruk te krijgen van de verscheidenheid aan beroepen waarbij Python wordt ingezet om de gewenste resultaten te behalen.

## Over dit boek

---

De bedoeling van *Programmeren met Python voor Dummies* is om je direct aan het werk te zetten met Python. Je wilt deze programmeertaal snel leren kennen, zodat je er productief mee kunt zijn in je werk, wat van alles kan zijn. In tegenstelling tot andere boeken over dit onderwerp laat dit boek direct zien wat Python van andere talen onderscheidt en hoe je er nuttige dingen mee kunt doen in ander werk dan programmeren. Vanaf het begin ontdek je wat je moet doen aan de hand van praktische voorbeelden en door het uitvoeren van nuttige taken. Je krijgt zelfs hulp bij het installeren van Python bij jouw besturingssysteem.

Nadat je Python op jouw specifieke platform hebt geïnstalleerd, lees je eerst over de beginselen en breid je je kennis vervolgens gestaag uit. Nadat je de voorbeelden in dit boek hebt doorgenomen, ben je in staat om eenvoudige apps te schrijven, bijvoorbeeld voor het versturen van een e-mailbericht met Python. Je bent dan nog geen expert, maar wel in staat om Python te gebruiken voor specifieke doeleinden in een professionele

omgeving. Om het lezen te vergemakkelijken, worden in dit boek de volgende conventies gebruikt:

- » Tekst die je precies zo moet invoeren als in het boek staat wordt in deze letter weergegeven. De uitzonderingen hierop zijn de genummerde stappen: aangezien elke stap vet is, is de tekst die je moet typen niet in deze letter.
- » Zie je een *cursief* woord tussen de tekst die je moet invoeren, dan moet je deze vervangen met je eigen invoer. Zie je bijvoorbeeld 'Typ *je naam* en druk op Enter', dan is het de bedoeling dat je *je naam* met je eigen naam vervangt.
- » Webadressen hebben geen afwijkende zetwijze. Lees je een digitale versie van dit boek op een apparaat met een internetverbinding, dan kun je op zo'n webadres klikken om de bijbehorende website te bezoeken, bijvoorbeeld op [www.dummies.nl](http://www.dummies.nl).
- » Bij menuopties die je moet kiezen, zie je speciale pijlen, zoals bij: File ⇄ New File. In dit geval open je eerst het menu File en kies je dan de menuoptie New File. Er wordt in dit geval een nieuw bestand gemaakt in de editor IDLE, die Engelstalig is.

## Wie ben jij?

Misschien vind je het een beetje gek dat ik allerlei aannamen over jou heb gedaan — we hebben elkaar immers nooit ontmoet! Toegegeven, de meeste aannamen zijn een beetje vreemd, maar ik heb ze als uitgangspunt voor dit boek gebruikt.

Het is belangrijk dat je iets afweet van het besturingssysteem dat je wilt gebruiken, want hierover vind je geen hulp in dit boek. (In hoofdstuk 2 vind je wel informatie over de installatie van Python op verschillende platforms.) Om zo veel mogelijk informatie over Python te kunnen bieden, worden in dit boek geen platformspecifieke onderwerpen behandeld. Je moet dus weten hoe je apps installeert en gebruikt en hoe je werkt met het besturingssysteem dat op je computer is geïnstalleerd als je dit boek optimaal wilt gebruiken.

In dit boek ga ik er ook van uit dat je weet hoe je dingen op internet vindt. Er staan in dit boek veel verwijzingen naar online materiaal dat je helpt om Python beter te begrijpen. Maar je hebt er alleen wat aan als je deze bronnen kunt bereiken.

# De pictogrammen in dit boek

Tijdens het lezen van dit boek zie je regelmatig pictogrammen in de kantlijn staan die interessant materiaal markeren (of je op potentiële rampen wijzen). In deze paragraaf vind je een korte uitleg over elk pictogram.



TIP

Tips zijn handig, want je bespaart er tijd mee of ontdekt hoe je een taak uitvoert zonder veel overbodig werk te doen. De tips in dit boek zijn tijdsparingen of verwijzingen naar bronnen die de moeite waard zijn om uit te proberen.



PAS OP

Ik wil niet als een boze ouder of een of andere fanatiekeling klinken, maar zorg ervoor dat je de dingen die met dit pictogram zijn gemarkeerd niet doet. Doe je dit wel, dan maak je apps die gebruikers verwarren, waarna ze het vertikken om ermee te werken.



TECHNISCHE  
INFO

Bij dit pictogram moet je denken aan een geavanceerde tip of techniek. Misschien vind je deze alinea's met nuttige informatie dodelijk saai, maar mogelijk vind je er de oplossing om een haperende app weer aan de praat te krijgen. Sla deze stukken over als je ze niet wilt lezen.



BELANGRIJK

Steek je niets op van een specifieke alinea of een heel hoofdstuk, lees dan het materiaal bij dit pictogram nog eens. Hier vind je meestal informatie over een essentieel proces of over iets wat je moet weten om succesvolle Python-apps te schrijven.

## De code van de voorbeelden

Wie wil er nu echt alle code die in dit boek staat met de hand invoeren? De meeste lezers geven er de voorkeur aan om met de codevoorbeelden te werken, in plaats van onbekende code te typen. Je boft! De broncode kun je gewoon downloaden, dus je hoeft alleen maar dit boek te lezen om programmeertechnieken te leren kennen. Bij elk van de voorbeelden staat duidelijk aangegeven welk project je moet gebruiken. Je vindt de bestanden op <http://www.dummies.nl/downloads>.

# Wat nu?

---

Nu is het tijd om te beginnen aan je programmeeravontuur met Python! Is programmeren volledig nieuw voor je, begin dan met hoofdstuk 1 en lees het boek in een tempo dat je in staat stelt om zo veel mogelijk van het materiaal in je op te nemen.

Ben je een beginner die zo snel mogelijk met Python aan de slag moet, dan zou je met hoofdstuk 2 kunnen beginnen, al kunnen sommige onderwerpen dan in de toekomst verwarrend zijn. Met hoofdstuk 3 beginnen is mogelijk als je Python al geïnstalleerd hebt, maar blader hoofdstuk 2 voor de zekerheid wel even door om erachter te komen welke aannamen zijn gemaakt tijdens het schrijven van dit boek.

Lezers die al iets van Python weten, kunnen tijd besparen door met hoofdstuk 5 te beginnen. Je kunt dan altijd nog naar eerdere hoofdstukken terugbladeren als je met vragen zit. Het is wel belangrijk dat je elk codevoorbeeld begrijpt voordat je met het volgende doorgaat. Elk voorbeeld omvat een belangrijke les, dus de kans bestaat dat je essentiële inhoud mist als je te veel informatie overslaat.

# 1

## **Aan de slag met Python**

## IN DIT DEEL . . .

Ontdek je wat programmeren is en waarom je hiervoor Python nodig hebt.

---

Download je Python en installeer je het op je computer.

---

Werk je in de interactieve omgeving van Python.

---

Maak je je eerste app met Python.

---

Leer je de voordelen van commentaar in een app kennen.

**Met je computer praten****Apps maken die met je computer praten****De werking van een app en de reden om er een te maken leren kennen****Redenen om Python als programmeertaal te gebruiken**

## Hoofdstuk 1

# Met je computer praten

**M**isschien klinkt het voeren van een conversatie met je computer als het script van een sciencefictionfilm. De bemanning van de *Enterprise* in *Star Trek* sprak immers regelmatig met haar computer. En die computer kon gewoon terugpraten. Maar met de komst van Apple's Siri (<http://www.apple.com/ios/siri/>) en andere interactieve software is het praten met computers niet langer toekomstmuziek.



BELANGRIJK

De computer om informatie vragen is één ding, maar hem opdrachten geven is iets compleet anders. In dit hoofdstuk lees je over de redenen om je computer instructies te geven en over de voordelen die dit biedt. Je ontdekt ook waarom je hiervoor een speciale taal nodig hebt en de voordelen die Python je hierbij biedt. Maar het belangrijkste dat je van dit hoofdstuk opsteekt, is dat programmeren eenvoudigweg een vorm van communicatie met je computer is die lijkt op andere communicatiemethoden.



# Waarom zou je met je computer willen praten?

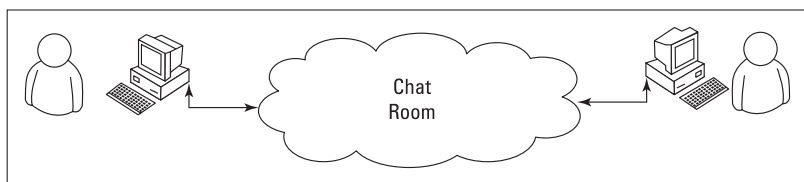
Met een apparaat praten lijkt in het begin misschien een beetje raar, maar je ontkomt er niet aan, want een computer kan geen gedachten lezen — in ieder geval nu nog niet. Maar ook als de computer je gedachten wel kon lezen, zou hij met je communiceren. Er kan niets gebeuren zonder een uitwisseling van informatie tussen het apparaat en jou. De volgende activiteiten zijn allemaal voorbeelden van communicatie tussen jou en je computer:

- » je e-mail lezen;
- » over je vakantie schrijven;
- » het mooiste cadeau ter wereld zoeken.

Het feit dat de computer ook nog met andere apparaten en mensen communiceert om jouw verzoeken door te geven, toont aan dat je zonder communicatie geen resultaten bereikt.

In de meeste gevallen vindt de communicatie op een bijna onzichtbare manier plaats, tenzij je erover nadenkt. Bezoek je bijvoorbeeld een online chatroom, dan denk je misschien dat je met een andere persoon communiceert. In werkelijkheid communiceer jij met je computer, communiceert je computer met de computer van de andere persoon via de chatroom (ongeacht waar deze uit bestaat) en communiceert deze computer op zijn beurt met de andere persoon. In figuur 1.1 zie je hoe dit in zijn werk gaat.

**FIGUUR 1.1:**  
Communicatie met je computer kan onzichtbaar lijken tot je erover nadenkt.



Zie je de wolk in het midden van figuur 1.1? In deze wolk kan van alles staan, maar je weet dat er in ieder geval andere computers aanwezig zijn die andere apps of programma's uitvoeren. Deze computers maken het mogelijk dat jij en je vriend met elkaar communiceren. Bedenk eens hoe eenvoudig dit hele proces lijkt als je een chatprogramma gebruikt. Aangezien al deze dingen zich achter de schermen afspelen, lijkt het erop dat jij eenvoudigweg met je vriend kletst; het proces zelf is onzichtbaar.

# Een programma of app is een vorm van communicatie

Computercommunicatie vindt plaats met behulp van programma's, tegenwoordig beter bekend als apps. Je gebruikt een programma om je e-mail te beantwoorden, een ander programma om dingen te kopen en weer een ander programma om een presentatie te maken. Een programma of *app* (kort voor *applicatie* of toepassing) biedt de mogelijkheid om menselijke ideeën aan een computer voor te leggen op een manier die de computer begrijpt. Zo'n app beschikt over hulpmiddelen om de benodigde gegevens voor de communicatie vorm te geven. Gegevens om de inhoud van een presentatie te tonen, verschillen van de gegevens om een cadeautje voor je moeder te kopen. De manier waarop je gegevens bekijkt, gebruikt en begrijpt verschilt van taak tot taak. Je hebt dus verschillende apps nodig om met verschillende soorten gegevens te communiceren, zodat jij en de computer begrijpen wat er gebeurt.

Er zijn tegenwoordig apps om zo'n beetje elke algemene taak die je kunt verzinnen, uit te voeren. Je hebt waarschijnlijk zelfs toegang tot apps waarvan je het doel nog niet eens verzonnen hebt. Programmeurs zijn al jaren bezig met het maken van miljoenen apps voor een verscheidenheid aan taken. Het kan dus lastig zijn om een nieuwe methode te bedenken om via een app met je computer te communiceren. Hierbij moet je goed nadenken over de gegevens waar je mee wilt communiceren. Sommige gegevens zijn eenvoudigweg niet algemeen genoeg om de aandacht van een programmeur te hebben getrokken. Het kan ook zijn dat je moet werken met gegevens in een indeling die niet door de huidige apps wordt ondersteund, waardoor je de computer er alleen maar op kunt wijzen door zelf een app te maken.

In de volgende paragrafen lees je over apps die op een speciale manier met unieke gegevens werken. Misschien heb je een videobibliotheek, maar geen manier om er op een handige manier toegang toe te krijgen. Aangezien de gegevens uniek zijn en je er op een speciale manier toegang toe zoekt, kan het handig zijn om zelf een app te maken waarin zowel de gegevens als je wensen verwerkt worden.

# Procedures die je dagelijks gebruikt

Een *procedure* is niet meer dan een reeks stappen die je moet uitvoeren om een taak te voltooien. Bij het maken van toast volg je bijvoorbeeld de volgende procedure:

1. Pak een paar sneetjes brood uit de broodtrommel.
2. Haal boter uit de koelkast.
3. Verwijder het deksel van de broodrooster.
4. Plaats de sneetjes brood in de broodrooster.
5. Duw de schuif op de broodrooster omlaag om het brood te roosteren.
6. Wacht tot het brood volledig is geroosterd.
7. Haal de toast uit de broodrooster.
8. Leg de toast op een bord.
9. Smeer boter op de toast.

Misschien wijkt jouw procedure wat af van deze stappen, maar de kans is klein dat je eerst boter op het brood smeert voordat je het in de broodrooster plaatst. Je moet de sneetjes brood natuurlijk uit de zak halen voordat je ze roostert (want het plastic van de zak geeft een ongewenste bij-smaak aan de toast). De meeste mensen denken niet aan de procedure voor het maken van toast als ze brood gaan roosteren. Maar ook als je er niet over nadenkt, gebruik je een dergelijke procedure.



BELANGRIJK

Computers zijn niet in staat om taken uit te voeren zonder een procedure te volgen. Je moet de computer laten weten welke stappen hij moet volgen, in welke volgorde dit moet gebeuren en welke uitzonderingen op de regels tot fouten kunnen leiden. Al deze informatie (en meer) maakt deel uit van een app. In feite is een app niet meer dan een geschreven procedure waarmee je een computer vertelt wat hij moet doen, wanneer hij het moet doen en hoe hij dit moet doen. Aangezien jij al je hele leven procedures gebruikt, hoef je alleen maar je bestaande kennis over een specifieke taak op de computer toe te passen.

## Procedures op papier zetten

Op de basisschool moesten we een keer opschrijven hoe je toast maakt. Nadat we het opstel hadden ingeleverd, pakte de juf een broodrooster en een gesneden brood. Elk opstel werd voorgelezen en gedemonstreerd. Geen van de procedures werkte als verwacht, maar ze gaven allemaal

grappige resultaten. In mijn geval was ik vergeten het brood uit het plastic te halen, dus propte de juf de sneetjes brood met plastic en al in de broodrooster. Deze les is me altijd bijgebleven. Het noteren van procedures kan heel lastig zijn, want ook als we precies weten hoe iets in zijn werk gaat, laten we vaak stappen weg. We gaan ervan uit dat de ander ook precies weet wat we willen doen.

In het dagelijks leven maken we veel gebruik van procedures. Denk maar eens aan de checklist die piloten nalopen voordat ze vertrekken. Zonder goede procedure zou het vliegtuig kunnen crashen. Het schrijven van een goede procedure kost tijd, maar het is te doen. Na enkele pogingen lukt het wel, al moet je misschien verschillende pogingen ondernemen voordat je procedure volledig klopt. Maar het opschrijven van procedures is niet genoeg — je moet de procedure ook laten testen door iemand die de beschreven taak niet kent. Bij het werken met computers is de computer een perfecte testkandidaat.

## Apps als een procedure beschouwen

Een computer gedraagt zich als de juf van de basisschool in het voorbeeld met de broodrooster eerder in dit hoofdstuk. Tijdens het maken van een app schrijf je een procedure die een reeks stappen definieert die de computer moet volgen om de door jou bedachte taak uit te voeren. Laat je een stap weg, dan krijg je onvoorspelbare resultaten. De computer weet dan niet wat je bedoelt of heeft niet door dat je verwacht dat sommige taken automatisch worden uitgevoerd. Het enige wat de computer weet, is dat je een specifieke procedure hebt geboden en dat hij deze procedure moet volgen.

## Computers nemen alles letterlijk

Mensen wennen uiteindelijk aan de procedures die je maakt. Ze passen automatisch foutjes aan of noteren ontbrekende onderdelen. Mensen lossen de problemen van de procedures die je schrijft gewoon zelf op.



BELANGRIJK

Als je met het schrijven van computerprogramma's begint, is de kans groot dat je gefrustreerd raakt door het feit dat computers je instructies letterlijk nemen en taken exact uitvoeren. Laat je de computer bijvoorbeeld weten dat een specifieke waarde 5 moet zijn, dan zoekt de computer naar een waarde die exact 5 is. Een mens denkt waarschijnlijk dat de waarde 4,99 goed genoeg is, maar een computer is het hier niet mee eens. De computer ziet de waarde 4,99 en besluit dat deze niet precies overeenkomt met 5. Computers zijn niet flexibel, hebben geen intuïtie en ook geen fantasie. Een procedure wordt door een computer iedere keer op dezelfde manier letterlijk gevolgd; hij zal hem nooit aanpassen en nooit besluiten dat je iets anders bedoelde.