

# 1. Hoe gebruik je dit boek

## Wat leer ik?

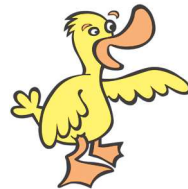
In dit boek leer je de programmeertaal **Scratch** gebruiken: je maakt spellen, doolhoven, tekenprogramma's, oefent met tafels en nog veel meer!

Je kunt straks spannende projecten bouwen en je krijgt in dit boek heel veel hulp om je te ondersteunen bij het leren **programmeren**.

## Hoe gebruik je de lessen

Leren programmeren doe je vooral door te oefenen en te experimenteren. Volg nauwkeurig alle lessen en zorg dat je de instructies goed leest. Bekijk ook goed de aanwijzingen in de plaatjes. Het lijkt dus best een beetje op hoe **Technic LEGO** werkt. Daar moet je immers ook goed de instructies stap voor stap volgen. Als je veel hebt nagebouwd, kun je zelf gaan ontwerpen. Het belangrijkste is om ook een beetje te spelen met de programmeercode als je eenmaal een les hebt afgerond. Probeer eens om stukjes code te veranderen of toe te voegen. Wees niet bang om je code te verknoeien.

De beste **programmeurs** ter wereld maken allemaal weleens een fout, maar het herstellen van die fouten, ook wel **debuggen** genoemd, is de beste manier om het te leren.

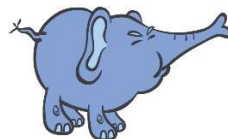


**Programmeur:**  
Iemand die programmeercode ontwerpt en schrijft



**Debuggen:**  
Code testen om fouten te vinden en te herstellen

Er is zelfs een beroemd citaat over programmeren en fouten maken van Mark Zuckerberg – de man die Facebook heeft bedacht:



“Beweeg snel en maak dingen stuk. Als je geen dingen stuk maakt, beweeg je niet snel genoeg.”

# Wat is programmeren?

## Wat zijn computers?

Overal om ons heen zien we computers. Waarschijnlijk zijn er bij jou thuis meerdere computers; je hebt misschien geen idee hoeveel het er zijn! Heeft er iemand in jouw familie een laptop? Natuurlijk is dat een computer. Maar wat dacht je van een mobiele telefoon? Wist je dat dat ook een computer is? Er zit zelfs een computer in de televisie, de magnetron en de auto.

Soms gebruiken we computers die honderden of zelfs duizenden kilometers verderop staan. Wanneer je zit te internetten, kan de computer waarop de webpagina is opgeslagen die je aan het bekijken bent (ook wel server genoemd), soms in een heel ander land staan.

Computers zijn machines die de antwoorden kunnen uitrekenen van hele simpele rekensommen. Zo kunnen computers bijvoorbeeld getallen optellen of aftrekken, of zij kunnen zien wanneer twee getallen hetzelfde zijn of niet. Zij kunnen heel snel rekenen. Als je tot een miljoen moest tellen, zou je er waarschijnlijk meer dan een week over doen. Een computer kan in minder dan één seconde tot een miljoen tellen.

Snel rekenen is heel indrukwekkend, maar wat computers echt heel uitzonderlijk maakt, is dat zij instructies kunnen opvolgen. Deze instructies worden **programma's** genoemd. Computers kunnen deze instructies uitvoeren en heel slimme dingen doen, zoals monsters over het scherm laten bewegen in een computerspel of jou in een echt vliegtuig meenemen op vakantie.

Zonder programma's is een computer volkomen nutteloos. Computers zijn juist zo bijzonder door de programma's die voor ze geschreven zijn.

## Wat is coderen?

Bij het schrijven van een computerprogramma kan een programmeur geen gewone mensentaal gebruiken, zoals de taal die we dagelijks met elkaar spreken. Je kunt bijvoorbeeld niet zomaar tegen een computer zeggen “tel tot duizend”.

In plaats daarvan moet je een speciale taal gebruiken: een programmeertaal. Er bestaan honderden verschillende programmeertalen. Als je een computer tot duizend wilt laten tellen in Scratch, de programmeertaal die je met dit boek leert, zouden we dit schrijven:



In Python, een andere programmeertaal, zouden we iets heel anders schrijven:

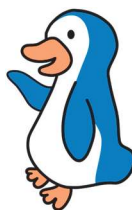
```
for i in range(1001):  
    print(i)
```

In weer een andere taal die JavaScript heet, schrijf je:

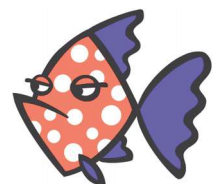
```
for(i=0; i<1001; i++){  
    console.log(i)  
};
```

De woorden en symbolen die in programmeertalen worden gebruikt, worden samen **code** genoemd. We noemen het schrijven van computerprogramma's dan ook vaak **coderen**.

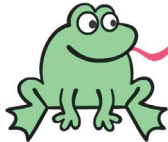
**Coderen:**  
Code schrijven voor een computer



**Code:**  
Instructies die in een programmeertaal zijn geschreven, zodat een computer ze kan opvolgen



Als je klaar bent met het schrijven van een computerprogramma, moet het nog vertaald worden in enen (1) en nullen (0). Dit wordt **compileren** of **interpreteren** genoemd en zorgt ervoor dat de computer de instructies kan lezen. Het maakt niet uit welke programmeertaal je gebruikt, uiteindelijk wordt die taal door de computer altijd omgezet in enen en nullen.



**Compileren:**

Een programma omzetten in een taal die door de computer kan worden begrepen en uitgevoerd



**Interpreteren:**

Dit gebeurt als de computer een programma regel voor regel uitvoert

## Wat is syntax?

**Syntax** is de verzamelnaam voor alle regels van een taal. Bekijk deze twee zinnen eens:

“Ik ben een fantastische codeerder.”

“Codeerder ben een ik fantastische.”

De eerste zin is te begrijpen, omdat die de syntax (of regels) van de Nederlandse taal volgt. De tweede zin is niet volgens die regels opgebouwd en daar snappen we dus niet zoveel van. Omdat jij slim bent, heb je waarschijnlijk wel een idee van wat er ongeveer staat, ook al staan de woorden in de verkeerde volgorde. Maar computers zijn niet op dezelfde manier intelligent als mensen. Zij snappen niet wat je bedoelt als de code die je schrijft niet helemaal goed in elkaar zit.



**Syntaxfout:**

Een fout in een computerprogramma die wordt veroorzaakt door het niet volgen van de regels van de programmeertaal

Als je je niet houdt aan de regels van de programmeertaal, gaat een computer niet proberen de instructies te compileren of interpreteren; je krijgt gewoon een melding dat je iets fout hebt gedaan. Dit wordt een **syntaxfout** genoemd.

## Waarom is coderen belangrijk?

Overal zijn er computers en die voeren elke dag enorm veel belangrijke taken uit. Wij gebruiken computers om te leren, te werken, te spelen, te praten en te reizen... En elke keer dat wij een computer gebruiken, werkt het apparaat doordat het codes uitvoert die geschreven zijn door een programmeur. Zonder programmeurs zouden computers helemaal niet eens werken en zou de wereld er heel anders uitzien!

## Klaar om te en

Scratch is een programmeertaal die speciaal gemaakt is om je te leren coderen. Je zult het snel door hebben en dan kun je de taal gebruiken om spelletjes, interactieve verhalen, tekeningen en animaties te maken.

Voordat je Scratch begint te gebruiken, moet je een account aanmaken. Als je jonger bent dan 13 jaar, moet je een volwassene vragen of je zijn of haar e-mailadres mag gebruiken.

## Scratch-account aanmaken

1. Open een webbrowser op je computer.
2. Ga naar de website [www.scratch.mit.edu](http://www.scratch.mit.edu)
3. Daar zie je een knop . Klik hier op.

## Namen op het internet

Internet is een geweldige plek om mensen te ontmoeten en met ze te praten, maar je weet niet altijd met wie je nu precies aan het praten bent en of je die persoon kunt vertrouwen. Gebruik daarom altijd een gebruikersnaam die niet op je echte naam lijkt. Kies een favoriet figuur uit een spel of een stripverhaal, of de naam van een huisdier.

## Wachtwoorden

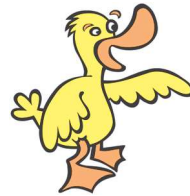
Een wachtwoord zorgt ervoor dat niemand zomaar jouw account kan gebruiken en net kan doen of jij het bent. Kies een wachtwoord dat niemand kan raden.

Gebruik bijvoorbeeld een wachtwoordzin,  
zoals deze:

**IkEetGraagHoeden**

of

**MijnVisSlaaptNooit**



**Wachtwoordzin:**

Een groep woorden die voor jou gemakkelijk te onthouden is, maar moeilijk door iemand anders geraden kan worden

4. Nu moet je een gebruikersnaam en een wachtwoord kiezen. Gebruik niet je echte

naam. Klik op **Volgende** als je klaar bent.

5. Nu moet je je geboortedatum, geslacht en land invullen en daarna klik je op

**Volgende**.

6. Voor het volgende deel heb je de hulp van een volwassene nodig. Vraag hem of haar om het e-mailadres in te vullen. Daarna kan diegene de e-mail openen om dit adres te

bevestigen. Daarna klik je op **Volgende**.

7. Als je een venster ziet zoals de volgende, betekent het dat je bent aangemeld bij de Scratch-website en klaar bent om te beginnen met coderen.

**Welkom bij Scratch,**

.Je bent nu ingelogd! Je kunt starten met ontdekken en maken van projecten.

Als je commentaar wilt plaatsen of je projecten wilt delen, klik dan op de link in de e-mail die we je gestuurd hebben op

Onjuist e-mailadres? Verander je e-mailadres in je [accountinstellingen](#).

Having problems? [Please give us feedback](#)



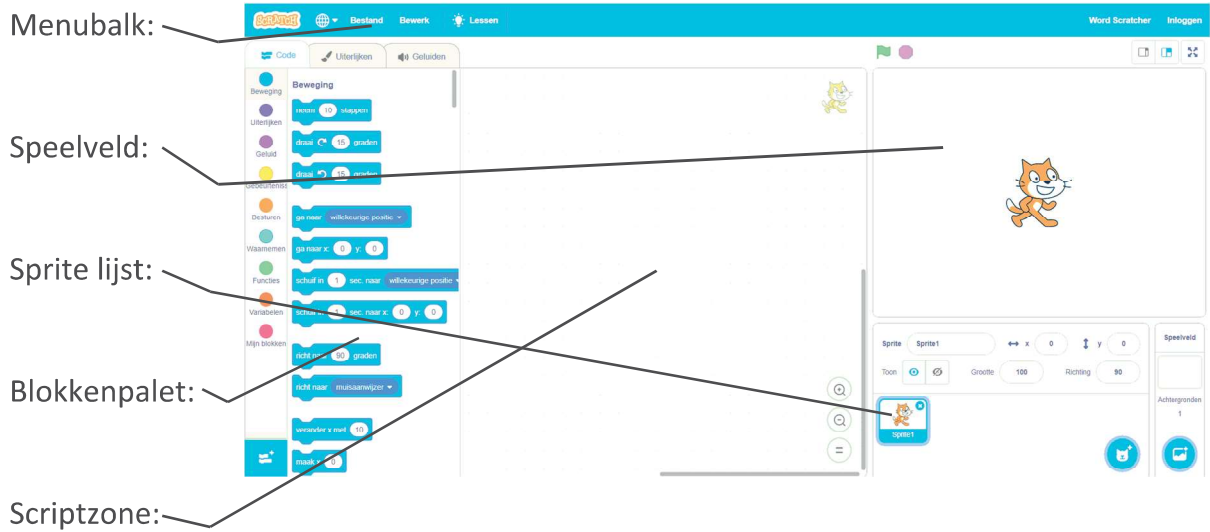
8. Klik op **Oké, laten we beginnen!** en begin met het verkennen van Scratch.

## Je weg vinden in Scratch

1. Om de projecten in dit boek goed te kunnen begrijpen, moet je iets weten over de Scratch **Graphical User Interface** (GUI), ofwel het startscherm van Scratch.

2. Klik op de hoofdpagina op **Maak**.

3. Nu zie je een scherm dat er net zo uit ziet als het plaatje hierna.

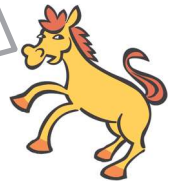


**Graphical User Interface:**

Op het (GUI) scherm kun je op knoppen klikken of items uit een menu kiezen, meestal met je muis, trackpad of touchscreen

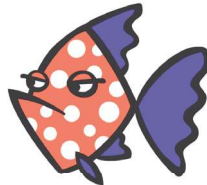
**Spritelijst:**

Hierin bewaar je alle figuurtjes die je in jouw project gebruikt



**Scriptzone:**

Hier zet je alle code die in jouw project wordt gebruikt



**Menubalk:**

Hier kun je kiezen uit opties om bijvoorbeeld je project op te slaan of een nieuw project te maken

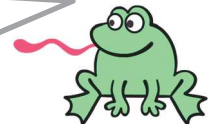


**Blokkenpalet:**

Hier vind je alle programmeer-codes die je nodig hebt om je project te bouwen. Afhankelijk van wat ieder blok doet, heeft deze zijn eigen kleur

**Speelveld:**

Hier kun je je project uitvoeren



Ieder hoofdstuk van dit boek begint met: Wat leer je in dit hoofdstuk? Daarin wordt kort verteld wat je zoal gaat leren, altijd fijn. Een hoofdstuk eindigt steeds met: Ga je de uitdaging aan? Dit deel is voor gebruikers die al veel ervaring hebben met Scratch. In dit deel wordt je verteld wát je moet doen, niet hÓe je het moet doen. Zo, nu weet je iets meer over Scratch en is het tijd om je eerste project te gaan coderen.