

Scratch 1

Deze pagina biedt een **stap-voor-stap handleiding** om te leren programmeren in Scratch door het bouwen van een **doolhofspel**. Je volgt de lessen door middel van **uitlegvideo's**, maakt **opdrachten direct in Scratch** en leert daarbij belangrijke programmeerconcepten zoals beweging, loops, if-then-else statements, variabelen en het resetten van de speltoestand. De cursus begeleidt je van de introductie en het opzetten van het startproject tot het afronden van je eigen unieke versie van het spel en het reflecteren op wat je hebt geleerd.

Vind je Scratch te makkelijk en heb je hier al ervaring mee? Dan mag je ook de [Scratch Plus opdracht](#) maken waarmee je een geheel eigen spel maakt.

Cursus: [28567](#)

Stap 0 - Introductie en start

[datasource](#)

Welkom! We gaan je stap voor stap helpen om Scratch te begrijpen, zodat je zelf leert programmeren. Maar wat gaan we eigenlijk doen?

? Leerdoelen

- Je ontdekt de Scratch **interface**: waar je leert in blokken te programmeren .
- Je leert wat sprite is en je leert deze te laten **bewegen**.
- Je leert wat een herhaalblok (Engels: **loop**) is.
- Je leert wat een als-dan-anders blok (Engels: **if-then-else**) is.
- Je leert wat een **variabele** is en hoe je die kan in Scratch kan gebruiken

Scratch

Scratch is een grafische manier van programmeren. Het is goede eerste stap omdat je zo leert **nadenken als een programmeur**, zonder dat je meteen hoeft te worstelen met moeilijke code of foutmeldingen. In Scratch werk je met blokken die niet kunnen "crashen", waardoor je snel resultaat ziet. Pas als je begrijpt hoe logica, lussen en beslissingen werken, is het makkelijker om over te stappen naar een echte programmeertaal zoals Python.

Wat kun je verwachten?

- Je bekijkt korte uitlegvideo's van Felienne, een docent aan de TU Delft.
- Je maakt opdrachten direct in Scratch.
- Je leert stap voor stap hoe je een eigen **doolhofspel** bouwt.

Als je nog meer wilt weten van deze module dan kan je [hier](#) meer informatie vinden.

Spelregels

- Je bouwt het spel stap voor stap op. Daarna maak je je eigen variant.
- Het inleveren van andermans werk is **fraude** en zorgt er voor dat je een module wordt geblokkeerd. Er volgt een gesprek met je studiecoach. Samen met hem/haar bepaal je onder welke voorwaarden jij je studie nog kan voortzetten.
- Maar het is ook écht leuk om je eigen spel te maken! Kom je er even niet uit, dan mag je natuurlijk wel een medestudent om hulp vragen.
- Voor deze opdrachten gebruiken we (nog) **geen AI** — dat komt later.
- Bij elke opdracht lever je iets in: soms is dat je code (of een screenshot daarvan), soms een antwoord op één of meer vragen. Ook die antwoorden schrijf je in je **eigen woorden**. Door zelf na te denken over de vragen, begrijp je de stof beter en word je uiteindelijk een goede softwaredeveloper.

Vorbereiding: Open dit project in Scratch:

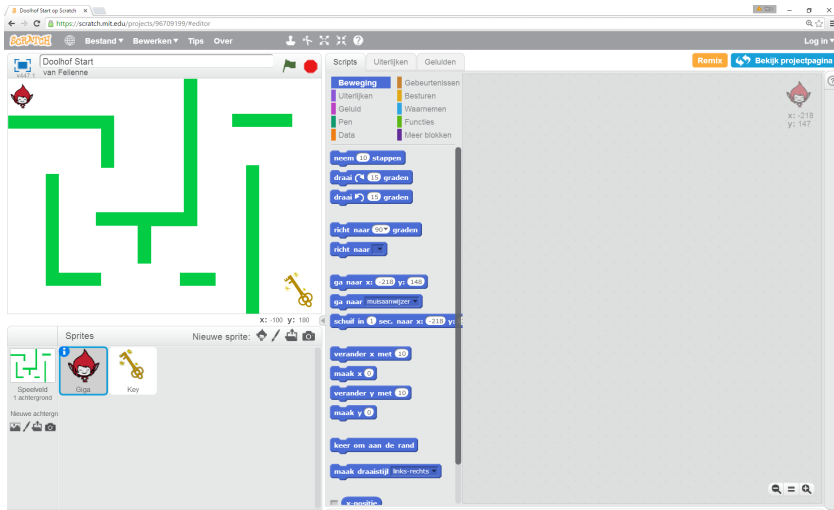
<https://scratch.mit.edu/projects/96709199/#editor>

(of download het bestand hier: [Doolhof Start.sb3](#))

Taal aanpassen

Standaard staat de taal op English, deze module is in het **Nederlands** dus als je de taal wil aanpassen dan kan dat onder **settings - language**.

Je zou dit startscherm moeten zien:



Bekijk de uitlegvideo

In deze video legt Felienne uit wat je gaat bouwen en hoe je aan de slag kunt. Let goed op de uitleg over hoe Scratch werkt.

<https://www.youtube.com/embed/JcIVNv4VZv8?si=8FT4Hu648QqKqrcY>

Alternatieve link: [Klik hier als de video hierboven niet werkt](#)

?? Opdracht

Vertel in een paar regels of je al ervaring hebt met programmeren. Wat heb je gedaan? Formuleer dit in minimaal drie zinnen.

Heb je geen ervaring? Beschrijf kort wat je met deze opdrachten hoop te leren. Wat zijn je verwachtingen? Formuleer dat in minimaal drie zinnen.

? Inleveren

Typ je antwoord in de tekstbox in.

Gebruik je eigen woorden en denk aan wat je in het spel gaat bouwen.

Stap 1 – Beweging

We beginnen met de besturing van je hoofdpersonage, Giga. In deze stap leer je hoe je Giga kunt laten bewegen met de pijltjestoetsen.

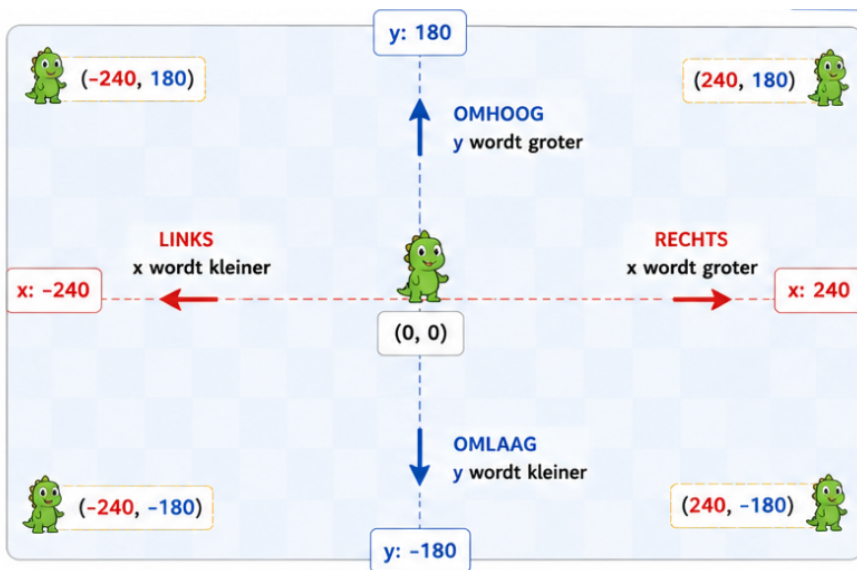
Wat leer je?

- Bewegen over de X-as (links en rechts).
- Bewegen over de Y-as (omhoog en omlaag).

Bekijk de uitlegvideo

<https://www.youtube.com/embed/IGtsy24vbiE?si=ZYnQwgoNBWrrlWw>

Alternatieve link: [Bekijk de video via HiDrive](#)



?? Opdracht

Hoeveel richtingen kan jouw poppetje bewegen aan het einde van deze stap?

? Inleveren

Vul in het tekstvak in hoeveel richtingen jouw sprite beweegt.

Stap 2 – De lus (loop)

In deze stap leren we hoe je een herhaling maakt met een lus. Hiermee kun je code telkens opnieuw uitvoeren zolang een bepaalde voorwaarde klopt.

Wat leer je?

Je gebruikt een **herhaalblok (loop)** om acties te blijven herhalen.

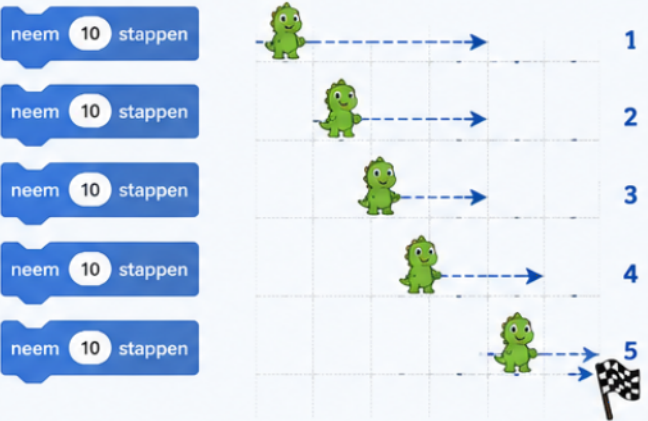
Bekijk de uitlegvideo

<https://www.youtube.com/embed/OFzSUt23Lho?si=r63G16qrEFFOA2YY>

Alternatieve link: [Bekijk de video via HiDrive](#)

ZONDER LOOP

Je moet dezelfde code 5 keer opschrijven.



neem 10 stappen 1
neem 10 stappen 2
neem 10 stappen 3
neem 10 stappen 4
neem 10 stappen 5

☹ Veel herhalen = veel blokken = onhandig en foutgevoelig!

MET LOOP

Je gebruikt een loop: korter, makkelijker en beter!



herhaal 5 keer
neem 10 stappen

😊 Kort, overzichtelijk en makkelijk aan te passen!

MAKKELIJK AAN TE PASSEN

Wil je 10 stappen in plaats van 5?



herhaal 5 keer
neem 10 stappen

Verander dit getal:

herhaal 10 keer
neem 10 stappen

BELANGRIJK OM TE ONTHOUDEN



Een loop laat de computer dezelfde opdracht meerdere keren achter elkaar uitvoeren.



**MINDRE BLOKKEN,
MEER OVERZICHT!**

?? Opdracht

Leg in 2 tot 4 zinnen in je eigen woorden uit wat een lus (loop) doet in een programma.

? Inleveren

Typ je uitleg over de lus in het tekstvak.

Stap 3 – Als-dan-anders

We voegen nu logica toe: wat moet er gebeuren als iets gebeurt?

Wat leer je?

Je leert het *als-dan-anders*-blok (if-then-else) gebruiken.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/6f9TeuFZshM?si=XWdXdtFBhWh0XuD8>

Alternatieve link: [Bekijk de video via HiDrive](#)

?? Opdracht

Wat doet een als-dan-anders (if-then-else)-blok in je code? Leg het uit in je eigen woorden.

? Inleveren

Typ je uitleg in het tekstvak.

Stap 4 – Terug naar startpositie

Als Giga een muur raakt, willen we dat hij teruggaat naar het begin.

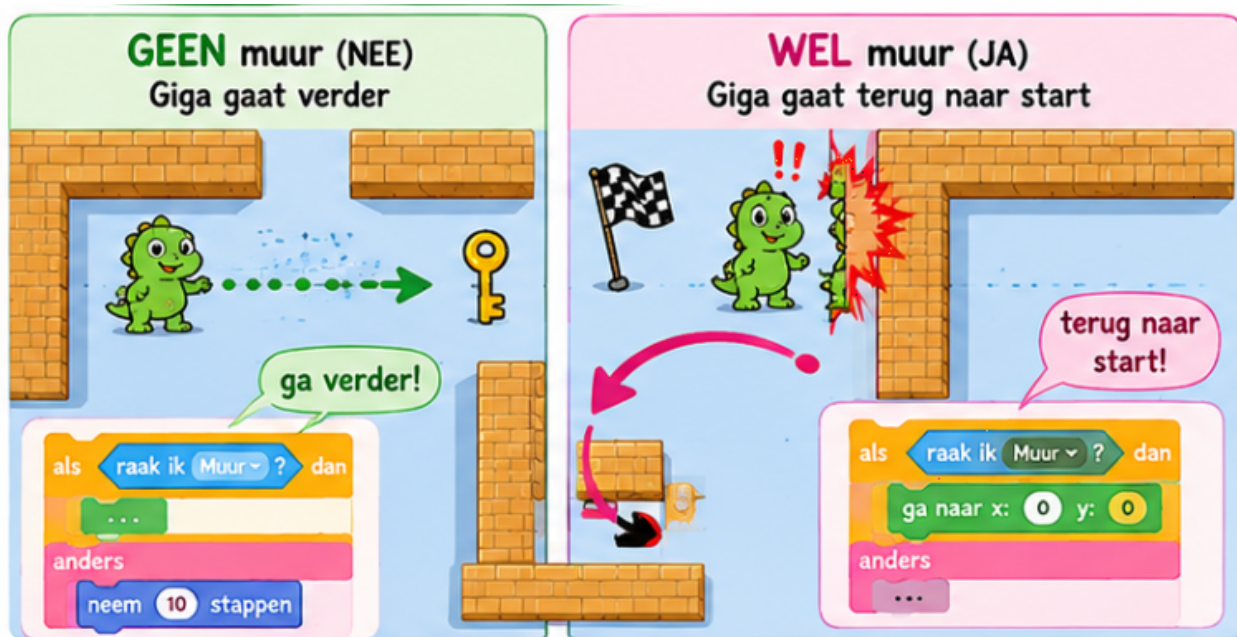
Wat leer je?

Je leert de sprite, Giga terug te zetten op zijn beginpositie.

Bekijk de uitlegvideo

https://www.youtube.com/embed/IRX_n2S0Hwk?si=YNeZtufi6l11jnJX

Alternatieve link: [Bekijk de video via HiDrive](#)



★ Kort gezegd: Als A gebeurt, doe B. Anders doe C. → Dit noemen we een **if-then-else**.

?? Opdracht

Welke coördinaten (X en Y) gebruikt jouw spel om Giga terug te zetten naar de start?

? Inleveren

Typ de X- en Y-positie in het tekstvak.

Stap 5 – De sleutel

Je spel is bijna klaar! In deze stap voeg je een sleutel toe. Als Giga deze aanraakt, is het spel gewonnen.

Wat leer je?

Je leert controleren of Giga de Sleutel aanraakt en als dat gebeurt dan wordt het spel beëindigd.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/UoqZhd4ggn0?si=j8a5GLqdgjNfATHc>

Alternatieve link: [Bekijk de video via HiDrive](#)

?? Opdracht

Maak een screenshot van je hele browser waarin je de gemaakte code laat zien.

? Inleveren

Upload de screenshot met jouw Scratch-code.

Stap 6 – Monster toevoegen

We maken het spannend! Voeg een monster toe dat je moet ontwijken. Laat het monster bewegen.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/6sYbPEse-jc?si=ugD4jr0WJ0vxC-8N>

Alternatieve link: [Bekijk de video via HiDrive](#)

?Let op

De detectie van een botsing met het spook werk niet altijd erg goed. Dt komt door het gebruik van



Dit commando wordt eerst helemaal uitgevoerd voordat de rest van het programma iets doet.

Als je dit niet helemaal begrijpt dan is dat goed en je mag een alternatieve spook beweging bedenken zonder dat je dit command gebruikt.

[Hier](#) een link naar een voorbeeld van een alternatieve spook-beweging. Het spook maakt telkens een stapje en als het de rand raakt dan draait het iets en maakt het weer stapjes tot het de rand raakt enzovoort, enzovoort.

?? Opdracht

Laat het monster bewegen. Lever een screenshot in van de code die het monster laat bewegen.

? Inleveren

Upload de screenshot van de code van het monster.

Stap 7 – Game over

Als het monster Giga raakt, is het spel afgelopen. Je leert nu hoe je het spel opnieuw kunt laten starten.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/LR5vDCgcLMM?si=Kj7SCUr9BADcQYGp>

Alternatieve link: [Bekijk de video via HiDrive](#)

?? Opdracht

Laat zien dat Giga teruggezet wordt naar het begin wanneer hij de muur of het monster raakt.

? Inleveren

Lever een screenshot in van de code die dit laat zien.

Stap 8 – Alles terugzetten

Als het spel opnieuw begint, moeten zowel het monster als Giga teruggezet worden naar hun beginpositie. Pas ook de snelheid aan als het spel te moeilijk is.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/N76K2ZfCP3o?si=Yf6DBrKI71vHcx2V>

Alternatieve link: [Bekijk de video via HiDrive](#)

?? Opdracht

Pas de snelheid van het monster aan en laat zien hoe je alles terugzet.

? Inleveren

Lever een screenshot in van de code waarin het monster wordt teruggezet en je de snelheid hebt aangepast.

Stap 9 – Punten bijhouden

We voegen nu een **score** toe met behulp van een variabele.

Een variabele is een waarde die je kunt opslaan en aanpassen tijdens het spel.

Bekijk de uitlegvideo

https://www.youtube.com/embed/1GTrDEd1ECY?si=_ALBcbZ-dacTo0dq

Alternatieve link: [Bekijk de video via HiDrive](#)

Om de score bij te houden gebruiken we een variabele.

WAT IS EEN VARIABELE?

Stel je voor: een doosje met een getal erin.
Je kan het getal lezen, veranderen en bewaren.



-  Je kunt de waarde lezen
-  Je kunt de waarde veranderen
-  De waarde blijft bewaard

VARIABELE IN SCRATCH

In Scratch werken we met blokken voor variabelen.

Voorbeeld: **score**

	zet score op 0
	tel 1 op bij score
	trek 1 af van score
	geef de waarde van score

?? Opdracht

Laat zien hoe je de score bijhoudt.

? Inleveren

Lever een screenshot in van je code waarin je de score bijhoudt met een variabele.

Stap 10 – Afronden

We maken het spel af en zorgen dat het er goed uitziet. Je kunt je spel opslaan op je laptop of online delen.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/CHWgQNulLjU?si=5iQpiZdUwrxSB8UU>

Alternatieve link: [Bekijk de video via HiDrive](#)

?? Opdracht

Maak het spel af, voeg iets unieks toe en sla het op.

★ TIP

- ✓ Gebruik duidelijke namen voor variabelen.
- ✓ Zet de score bij het begin van het spel op 0.
- ✓ Gebruik "verander score met 1" elke keer als je iets verdient!



Variabelen maken je spel levendiger en leuker!

? Inleveren

Lever jouw eigen unieke versie van het spel in.

Stap 11 – Wat heb je geleerd?



Programmeren = problemen oplossen door logisch na te denken.



Goed gedaan, programmeur! ★ YOU ROCK!

Je hebt veel geleerd over programmeren. Nu kijk je terug op wat je allemaal hebt gedaan.

Wat we hebben geleerd

Beweging



Loop



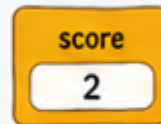
Als-dan-anders



Reset



Variabele



Deze bouwstenen gebruiken we samen om ons spel te maken!

?? Opdracht

- **Vraag 1:** Waarvoor heb je in dit spel een **lus** (Engels: **loop**) gebruikt?
- **Vraag 2:** Waarvoor heb je een **als-dan-anders-blok** (Engels: **if-then-else**) gebruikt?
- **Vraag 3:** Waarvoor heb je een **variabele** gebruikt?

? Reflectie

- Wat heb jij geleerd?
- Wat vond je leuk?
- Wat vond je lastig?
- Wat hoop je in de volgende module te leren?

? Inleveren

Beantwoord alle vragen uit de opdracht en de reflectie. Gebruik 2 to 4 zinnen per antwoord.

1. Schrijf je drie antwoorden uit de opdracht op en lever ze in.
2. Schrijf de 4 antwoorden uit de reflectie op.

--

Revision #62

Created 2025-04-19 16:13:47 UTC by Max

Updated 2026-07-05 08:41:26 UTC by Max