

Scratch 1

Introductie en start

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

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.

Voorbereiding: Open dit project in Scratch:

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

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

Je zou dit startscherm moeten zien:

image.png
image not found or type unknown

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

Beschrijf in één zin in je eigen woorden wat je denkt dat je gaat maken.

Inleveren

Typ je zin in het tekstvak hieronder. 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/lGtsy24vbiE?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 om acties te blijven herhalen.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/OFzSUt23Lho?si=r63G16qrEFOA2YY>

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

Opdracht

Leg 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? Je leert nu het *als-dan-anders*-blok 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. In deze stap leer je hoe je zijn positie reset.

Bekijk de uitlegvideo

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

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

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.

Bekijk de uitlegvideo

<https://www.youtube.com/embed/UoqZhd4ggn0?si=j8a5GLqdgiNfATHc>

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](#)

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-dacTo0dg

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

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/cHWgQNulJlU?si=5iQpiZdUwrxSB8UU>

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

Opdracht

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

Inleveren

Lever jouw eigen unieke versie van het spel in.

Stap 11 – Wat heb je geleerd?

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

Opdracht

- **Vraag 1:** Waarvoor heb je in dit spel een lus gebruikt?
- **Vraag 2:** Waarvoor heb je een als-dan-anders-blok gebruikt?
- **Vraag 3:** Waarvoor heb je een variabele gebruikt?

Inleveren

Schrijf je drie antwoorden op en lever ze in.

Revision #21

Created 19 April 2025 16:13:47 by Max

Updated 14 May 2025 09:49:45 by Max