

 **Projecten**

Beginner Scratch Sushi kaarten

Maak een leuk spel in je eerste Scratch-programma!

Scratch



Stap 1 Inleiding

Deze Sushi-kaarten helpen je bij het leren maken van computerprogramma's in Scratch!

Wat je gaat maken

Je maakt een spel waarbij je een haai laat bewegen om vissen te vangen.

Hier is een voorbeeld van hoe je voltooide game eruit zou kunnen zien:

Druk op de groene vlag om het spel te starten. Gebruik de pijltjestoetsen om de haai te verplaatsen en de vis te vangen.

Wat ga je leren

- Toevoegen en verwijderen van codeblokken
- Basisbeweging: stap voor stap bewegen en draaien per graden
- Gebruik van toetsenbord knoppen
- Gebruik maken van (x, y) coördinaten om afbeeldingen te plaatsen
- Veranderen van het uiterlijk van het Speelveld en sprites met kostuums en het weergeven/verbergen commando
- Meerdere sprites op het scherm
- Gebruik maken van de **herhaal** lussen
- Opmerken wanneer sprites elkaar raken

Wat je nodig hebt

Je hebt **Scratch** nodig, dus doe één van de volgende dingen:

- Download Scratch en installeer het op je computer door naar **dojo.soy/downloadscratch** (<http://dojo.soy/downloadscratch>) te gaan
- Gebruik Scratch in je webbrowser door naar **dojo.soy/usescratch** (<http://dojo.soy/usescratch>) te gaan

Als je dit gedaan hebt, kun je door naar de volgende stap!

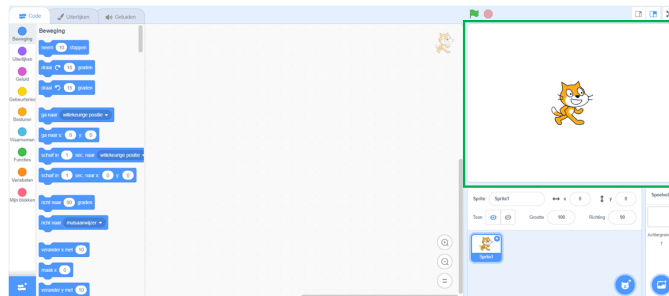
Stap 2 **Beginnen**

Zodra je Scratch opent, zie je een venster zoals hieronder.

Zoek naar de volgende gebieden en onthoud waar ze zijn.



Het speelveld

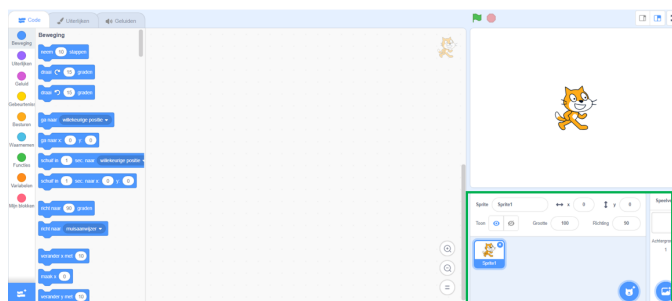


i Wat is het Speelveld?

Dit is waar je Scratch programma's worden uitgevoerd. Het heeft:

- Eén of meer **achtergronden** (afbeeldingen op de achtergrond van het scherm)
- Alle **codeblokken** die daarbij horen (hier komen we zo op)

De sprite lijst



i Wat is een sprite?

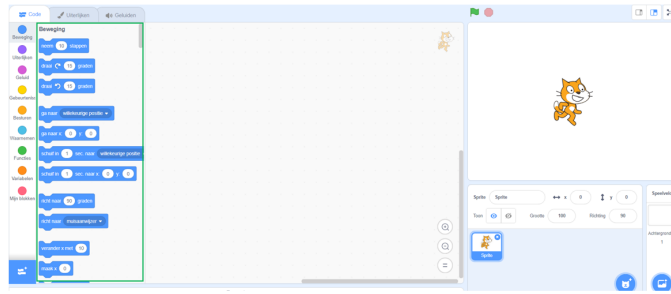
Elk item dat je in het Speelveld kunt toevoegen, is een **sprite**.

Een sprite bevat:

- De **afbeelding** op het Speelveld
- Mogelijke extra **kostuums** (uiterlijken) dat het heeft
- Alle **geluiden** die bij de sprite horen
- Alle **codeblokken** die bij de sprite horen



Het codeblokken palet

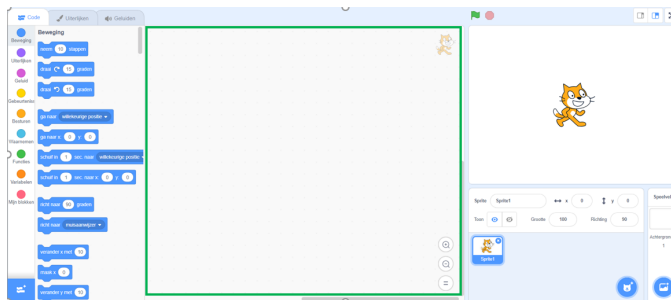


i Codeblokken

Code in Scratch komt in de vorm van blokken die je met elkaar verbindt om programma's te maken. Je kiest blokken uit het palet met **codeblokken**, sleept ze naar het **huidige sprite paneel** en verbindt ze dan met elkaar.

Er zijn tien categorieën van blokken. Ze hebben een kleurcode en je kunt kiezen tussen de blokken door op de items in de lijst bovenaan het palet met **codeblokken** te klikken.

Het huidige sprite paneel



i Wat is de huidige sprite?

De **huidige sprite** is degene die is geselecteerd in de **sprite lijst**.

Het **huidige sprite paneel** is waar je de code, kostuums en geluiden voor de geselecteerde sprite kunt zien.

Tijd om te gaan programmeren!

Klik op de Scratch kat in de sprite lijst. De kat is nu de huidige sprite. ✓

Kies de **Gebeurtenissen** categorie in het code blokken palet, klik op het **wanneer op groene vlag wordt geklikt :: events hat** blok, en sleep het naar het huidige sprite paneel.

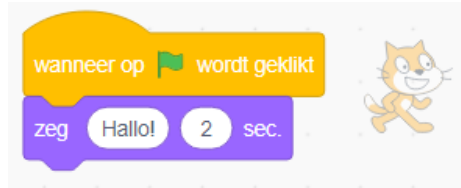
wanneer op de groene vlag wordt geklikt

Ga nu naar **Uiterlijk** in het code blokken palet en zoek dit blok: ✓



Klik erop, houd de muisknop ingedrukt en sleep hem naar het huidige sprite paneel en laat de knop los.

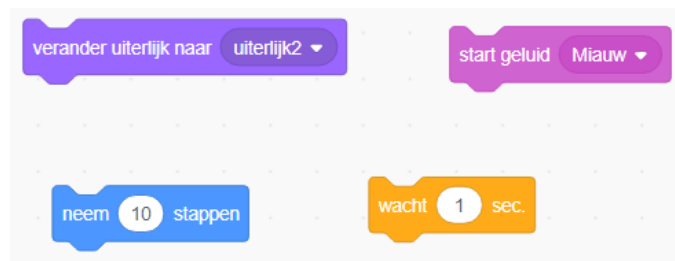
Nu het tweede blok ook in het huidige sprite paneel staat, verbindt je het aan de onderkant van het eerste blok door erop te klikken en het onder het eerste blok te slepen zodat ze aan elkaar vastklikken, zoals dit:



Klik nu op de **Ga** knop en kijk wat er gebeurt!

Stap 3 Toevoegen en verwijderen van codeblokken

Geweldig! Je hebt je allereerste Scratch programma geschreven. Tijd om wat meer te leren over hoe je code in en uit Scratch kunt zetten! Scratch code bestaat uit **blokken** zoals deze:



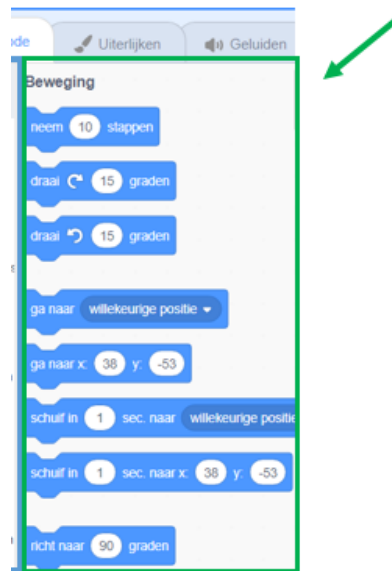
Je vindt alle blokken in het palet met **codeblokken**, onderverdeeld in verschillende categorieën op basis van wat ze doen.

i Gebruik blokken uit de verschillende categorieën

Klik op een categorienaam om de blokken in die categorie te bekijken. Hier is de **Beweging** categorie geselecteerd:



Alle blokken in de categorie waarop je hebt geklikt, worden weergegeven in de lijst:



Je kunt op het gewenste blok klikken en deze vervolgens naar het huidige sprite paneel slepen en loslaten. Zodra het in het paneel zit, kun je het verplaatsen en verbinden met andere blokken.

Als je wilt zien wat een blok doet, kun je erop dubbelklikken om het uit te laten voeren!

Dubbelklik op de blokken om te zien wat ze doen.



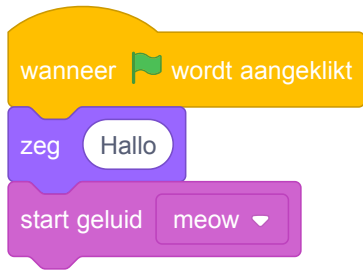
De code uitvoeren

Meestal wil je dat je code automatisch wordt uitgevoerd wanneer er iets specifiek gebeurt. Daarom beginnen veel van je programma's met een blok uit de **gebeurtenissen** categorie, meestal deze:



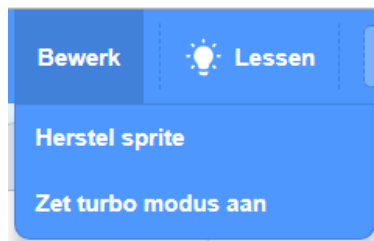
De codeblokken die zijn aangesloten op dit blok wordt uitgevoerd nadat op de **groene vlag** is geklikt.

Codeblokken lopen van boven naar beneden, dus de volgorde waarin je de blokken samenvoegt, is van belang. In dit voorbeeld zal de sprite **Hallo!** zeggen voordat het het **meow** geluid afspeelt.



Het verwijderen van codeblokken die je niet wilt gebruiken in je programma is eenvoudig! Sleep ze gewoon terug naar het palet met codeblokken.

Pas op: door je blok naar het codeblok paneel te slepen, worden alle blokken die aan dat blok hangen ook verwijderd. Zorg er dus voor dat de blokken die je wilt houden losgekoppeld zijn van het blok dat je wilt verwijderen. Als je per ongeluk blokken hebt verwijderd en ze terug wilt halen, klik dan met je rechtermuisknop en vervolgens op de **Ongedaan maken** optie om alles weer terug te krijgen.



Probeer eens codeblokken toe te voegen, te verwijderen en weer terug te halen!

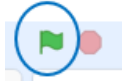
Alles bij elkaar brengen

Nu je weet hoe je code kunt verplaatsen en dingen kunt laten gebeuren, is het tijd om een programma te maken waarmee de Scratch Cat in een cirkel kan lopen!

Zorg dat je de kat sprite geselecteerd hebt in de spritelijst, en sleep dan de volgende blokken in het sprite paneel en verbindt ze met elkaar. Je vindt ze in de **Gebeurtenissen** en **Beweging** categorieën.



Klik nu op de groene vlag boven het Speelveld.



Je zou de kat in een rechte lijn moeten zien lopen... niet precies wat je wilt, toch?

Opmerking: als je te vaak op de vlag klikt en de kat loopt uit beeld, dan kun je hem terug slepen!

Zet het draai blok aan het eind zodat de kat in een rondje kan lopen. Het zit ook in de **Beweging** categorie. ✓



Hoe werkt draaien?

Dit blok zorgt ervoor dat de sprite 15 graden van de volledige 360 graden draait die een cirkel vormen. Je kunt dat aantal en het aantal stappen wijzigen door op het nummer te klikken en een nieuwe waarde in te voeren.



Sla nu je project op! ✓

Stap 4 **Het instellen van de scène**

Scratch heeft een bibliotheek met achtergronden en sprites die je kunt gebruiken om je project er geweldig uit te laten zien.

Selecteer het Speelveld. ✓



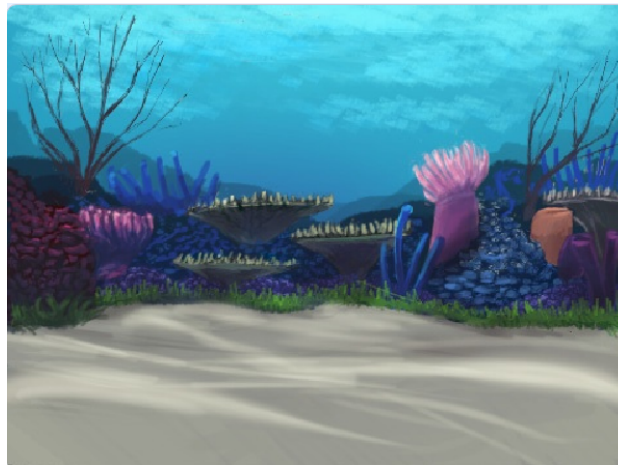
Klik op **Achtergronden**.



Klik op het **Kies een achtergrond** icoon.

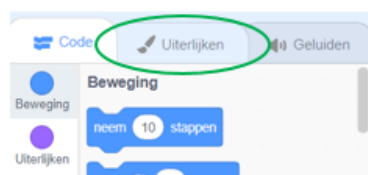


Kies dan je favoriete onderwaterachtergrond!

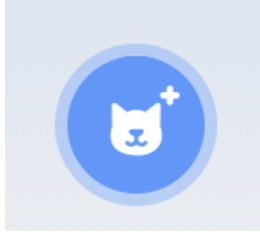


Geweldig! Maar nu heb je natuurlijk een kat onder water, en katten zijn daar meestal geen grote fans van. Maar dat kun je oplossen, want je kunt de kat veranderen in een haai!

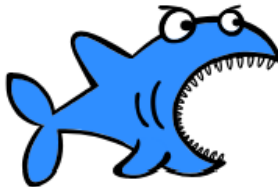
Selecteer eerst de kat sprite en klik op de **Uiterlijken** tab.



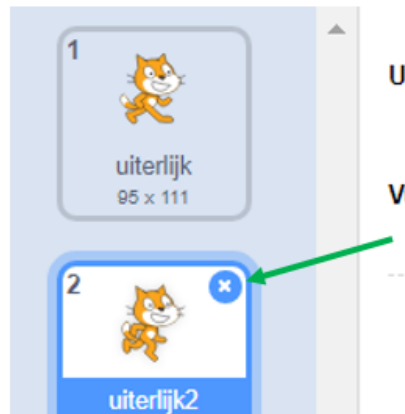
Klik dan op **Kies een uiterlijk**



Selecteer deze haai afbeelding en klik op **OK**.



Verwijder nu de kat uiterlijken door ze stuk voor stuk te selecteren en op **x** te klikken.



Nu heb je een haai - uitstekend!

Stap 5 Dingen verplaatsen

Op dit moment beweegt je haai in een rondje, en het zou veel gaver zijn om hem aan te sturen met de pijltjestoetsen. Met deze kaart leer je hoe dat moet!

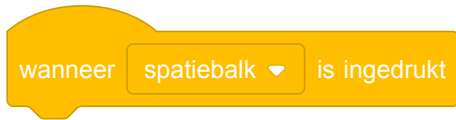
Begin met het verwijderen van alle code die je hebt voor de haai.



Zoals je vast al geraden hebt, heb je de **Gebeurtenissen** en **Beweging** blokken weer nodig!

Zoek deze keer dit blok op en sleep het naar het huidige sprite paneel:

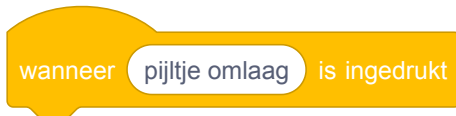
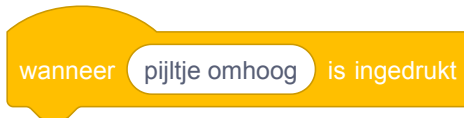




Klik op de kleine pijl (▼) naast **spatie**. Je ziet nu een lijst met alle toetsen waaruit je kunt kiezen.

Je hebt vier van de **wanneer toets is ingedrukt** blokken nodig - één voor elk van je pijltjestoetsen.

Om je haai te laten bewegen, verbindt je deze blokken op de volgende manier aan de **beweging** blokken: ✓



Let op: -10 betekent 'ga 10 stappen terug'.

Klik nu op de groene vlag om je code te testen. ✓

Nu beweegt je haai naar voren en naar achteren, wat heel gaaf is, maar hij gaat nog niet omhoog of naar beneden. Als je kijkt bij de **Beweging** blokken, zul je zien dat er geen blokken zijn voor 'omhoog' of 'omlaag'. Er zijn wel veel blokken die te maken hebben met **x** en **y** coördinaten - laten we die proberen!

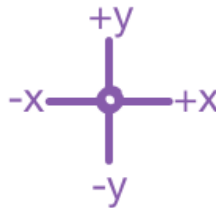
Neem twee **verander y met** blokken en werk je code als volgt bij: ✓



Als je nu op de pijltjestoetsen klikt, beweegt je haai over het hele speelveld!

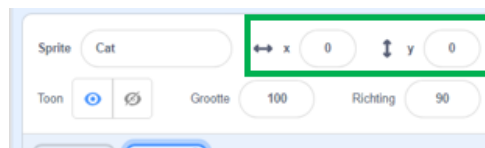
Hoe werken x- en y-coördinaten?

Als we het hebben over de positie van dingen, zoals sprites, dan gebruiken we vaak x- en y-coördinaten. De **x-as** van het Speelveld coördinaten systeem loopt van **links naar rechts**, en de **y-as** van **beneden naar boven**.



Een sprite kan gevonden worden op de coördinaten van zijn middelpunt, bijvoorbeeld (15, -27), waarbij 15 de positie op de x-as is, en -27 de positie op de y-as.

- Om een indruk te krijgen van hoe dit werkt, selecteer je een sprite en gebruik je de **x** en **y** besturingselementen om het over het speelveld te laten bewegen, hierbij vul je verschillende waarden in voor de coördinaten.



- Probeer verschillende waarden uit om te zien waar de sprite heen gaat! In Scratch loopt de x-as van -240 tot 240, en de y-as van -180 tot 180.


Het spel herstarten

De haai beweegt nu over je hele scherm, maar stel je voor dat dit een spel is: hoe herstart je het dan, en wat gebeurt er aan het begin van elk spel?

De haai moet weer op zijn beginpositie komen als een speler het spel start. Ze beginnen het spel door op de groene vlag te klikken, dus je moet de x- en y-coördinaten van de haai veranderen als dat gebeurt.

Dat is best wel simpel! Het midden van het speelveld is (0, 0) in (x, y) coördinaten.

Je hebt alleen een **Gebeurtenissen** blok nodig voor de groene vlag, en het **ga naar** blok uit **Beweging**.

Sleep een **wanneer groene vlag is ingedrukt**, **Gebeurtenis** blok naar het huidige sprite paneel. 


wanneer op de groene vlag wordt geklikt

Zoek dan het **ga naar** beweging blok en zet het vast aan je vlag **Gebeurtenissen** blok.

wanneer groene vlag is ingedrukt


ga naar x: 0 y: 0

Stel zowel de x als y coördinaten in op 0 in het **ga naar** blok als ze nog niet op 0 staan.

Klik nu op de groene vlag: je zou nu moeten zien dat de haai weer in het midden van het speelveld staat! 

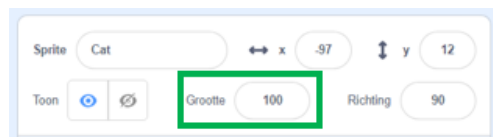
Stap 6 Alle sprites

Nu heb je een haai die je kunt laten bewegen met de pijltjestoetsen. Gaaf! Hoog tijd om wat vissen toe te voegen die hij kan vangen.

Klik op de **Kies een sprite** knop en kies een vis uit het scherm dat geopend wordt. 



Als je vis een beetje groot is vergeleken met je haai, dan kun je Grootte gebruiken om beide sprites het juiste formaat te geven!



Verander het getal achter Grootte om de sprite groter of kleiner te maken.

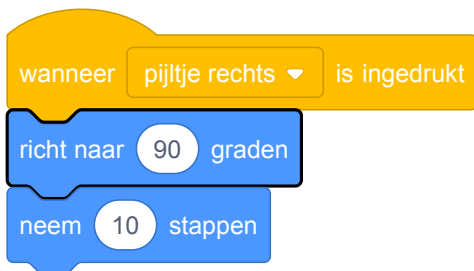
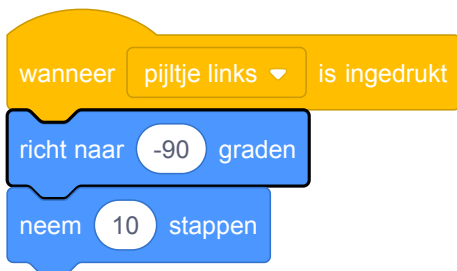
Geweldig! Later zul je code toevoegen om de vis zelfstandig te laten bewegen, zonder dat de speler helpt. Je speler zal de haai aansturen om te proberen de vis te vangen.

i Hoe zit het met de achterstevoren haai?

Het ziet er een beetje gek uit om de haai achterstevoren te laten zwemmen. Net zoals jij je liever omdraait in plaats van achteruit te lopen, wil de haai zich ook omdraaien in plaats van achterstevoren te zwemmen. Gelukkig heeft Scratch hier een blok voor!

Het **richt naar graden** blok laat je de richting bepalen waarnaar je sprite wijst. Je vindt het in de **Beweging** blokken categorie. Je kunt zelf het aantal graden bepalen om de sprite in de juiste richting te laten wijzen.

Gebruik een paar **richt naar graden** blokken uit de **Beweging** lijst en voeg ze als volgt aan de haai-code toe:



Verander het aantal stappen in de **beweging** blokken van **-10** tot **10**.


Als je de haai beweegt nu je de **ga naar graden** blokken hebt toegevoegd, zul je iets raars zien. De haai draait niet helemaal goed!



Waarom draait de haai ondersteboven?

Het probleem is dat de haai begon, zoals alle sprites doen, met de 'helemaal rond' **draaistijl**, en wat je nodig hebt is de 'links-rechts' stijl.

Zoals altijd is daar een blok voor in **Beweging**!

Zoek in de **Beweging** categorie naar het blok **maak draaistijl**. 

Voeg dit blok toe aan de herstartcode van de haai, en maak als volgt de draaistijl **links-rechts**:

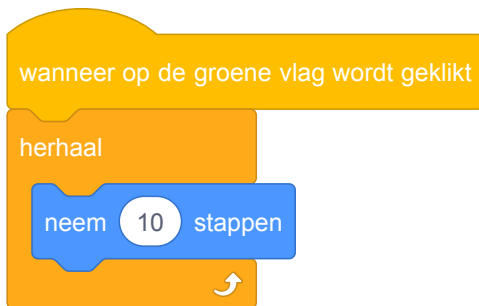


Stap 7 **Vis op afstand bestuurbaar**

Oké, hoog tijd om de vis zelfstandig te laten zwemmen. Hiervoor heb je een nieuw soort blok nodig: een **Besturen** blok.

Selecteer je vis sprite. 

Sleep een **wanneer op de groene vlag wordt geklikt** **Gebeurtenissen** blok, een **herhaal** **Besturen** blok en een **neem 10 stappen** **Beweging** blok naar het **sprite paneel**:



Wat doet dit nieuwe blok?

Besturen blokken zorgen ervoor dat je programma dingen een bepaald aantal keren doet, of onder bepaalde omstandigheden.

Hier herhaalt de vis alles wat in het **herhaal** blok staat voor eeuwig. Dus als het het laatste ding (blok) binnen het **herhaal** heeft gedaan, begint hij bovenaan weer helemaal op nieuw, en blijft dat herhalen.

Klik nu op de groene vlag en kijk wat er gebeurt!



Tjonge, die vis botste tegen de zijkant van het Speelveld en ging véél te snel voor je haai.

Eerst moet je de vis langzamer laten gaan. Dat is vrij simpel, je moet een pauze inlassen als de vis 10 stappen genomen heeft. Er is een **Besturen** blok dat je hierbij helpt:



Voeg het **wacht** blok toe aan je code binnen het **herhaal** blok, en verander het getal naar 0.5:



i Aanpassingen maken

Het getal dat je ingevoerd hebt in het **wacht** blok verteld hoeveel **seconden** jij wilt dat de vis moet wachten. 0.5 is een halve seconde.

Je kunt verschillende getallen invoeren om uit te zoeken wat het beste bij het spel past. En onthoud dat je het aantal stappen binnen het **neem stappen** blok ook kunt veranderen!

De vis beweegt nu, maar je wilt ook dat hij omdraait aan de rand van het Speelveld. Ook hier is een **Beweging** blok voor!

Zoek het **keer om aan de rand** blok en zet het onder het **wacht** blok.



i Wat doet dit nieuwe blok?

Het **keer om aan de rand** blok controleert of de sprite de rand van het Speelveld raakt, en zo ja, zorgt dat de sprite al naar gelang naar links, rechts, boven of beneden draait.

Natuurlijk betekent dit dat je vis ondersteboven gaat zwemmen, dus heb je weer een **maak draaistijl** nodig.

Werk je code bij door de draaistijl van de vis **links-rechts** aan het begin van de sprite code toe te voegen:



```

    wanneer op de groene vlag wordt geklikt
      maak draaistijl links-rechts
      herhaal
        neem 10 stappen
        wacht 0.5 sec.
        keer om aan de rand
  
```

De vis beweegt nu vooruit en achteruit, maar alleen in een rechte lijn - de speler kan nu met de haai wel heel makkelijk de vis pakken! Je moet de vis minder voorspelbaar maken.

Je weet van een vorige stap al hoe je een sprite kunt laten draaien, dus daar begin je mee.

Voeg een draai toe aan de zweminstructies van de vis, en klik op de groene vlag.



```

    wanneer op de groene vlag wordt geklikt
      maak draaistijl links-rechts
      herhaal
        neem 10 stappen
        draai 10 graden
        wacht 0.5 sec.
        keer om aan de rand
  
```

Al beter, maar nog steeds te voorspelbaar. Het moet willekeuriger worden. Gelukkig kan Scratch willekeurig voor je uitvoeren! Je hebt een nieuw blok nodig, een **functie** blok.



Wat is een functie?

Functies nemen één of meer waarden (zoals getallen, tekst, of **waar/Niet waar** waarden) en geven daar een enkele waarde voor terug. Je kunt zien wat voor waarde het terug zal geven door de vorm van het blok: afgeronde blokken geven getallen of tekst, puntige blokken geven **Waar/Niet waar**.



Zoek het **willekeurig getal** tussen **Beweging** blok door erop te klikken en het in het getalvakje van de graden te slepen.



Let op: je kan het laagste en hoogste getal veranderen dat gekozen moet worden, maar de standaard waarden (1 en 10) zijn zeer geschikt voor dit spel, dus je mag ze ook laten staan.

Klik op de groene vlag om de code uit te voeren!

Dus wat doet het herhaalblok nu?

Het herhaalblok laat de vis vier dingen in deze volgorde doen:

1. Beweeg vooruit
2. Draai een beetje
3. Wacht even
4. Controleer of het aan de rand van het Speelveld is

Zodra de sprite gecontroleerd heeft, zal het, zolang je het Scratch programma uitvoert, weer teruggaan naar het begin van de loop en bewegen, draaien, wachten, controleren.

Cool! Volgende stap: pak die vis!

Stap 8 Vissen!

De haai beweegt, de vis zwemt, maar ze reageren niet op elkaar: als de vis in de bek van de haai zwemt, gebeurt er niets. Hoog tijd om dat te veranderen!

Eerst moet je weten of de vis de haai aanraakt. Hiervoor heb je een **Besturen** blok en een **Waarnemen** blok nodig.

Voeg het **als...dan Besturen** blok toe aan de **herhaal** lus van de vis sprite, onder het **keer om aan de rand** blok. ✔

Sleep het **raak ik** blok boven in het vlakje van het **als...dan** blok, en klik op het kleine driehoekje om de naam van de haai sprite te selecteren. Als je die naam niet verandert hebt, zal dat 'Sprite1' zijn.



i Hoe werkt het?

Het **als...dan Besturen** blok heeft een **waar/Niet waar** waarde nodig.

Waarnemen blokken verzamelen informatie, zoals waar de sprite is, wat hij aanraakt, etc. Je gebruikt dit blok:

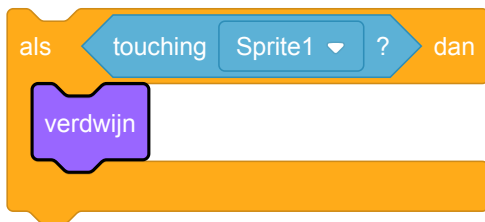


Door de puntige hoeken van dit blok, weet je dat het je een **Waar/Niet waar** waarde zal geven dat het **als...dan** blok nodig heeft.

Uiteraard heb je net een **als...dan** blok toegevoegd zonder iets voor het 'dan' gedeelte te zetten. Dus nu controleert je code of de vis de haai aanraakt, maar gebeurt er verder helemaal niets.

Je kunt de vis laten verdwijnen, alsof de haai hem heeft opgegeten, door het **verdwijn** blok te gebruiken.

Zoek het **verdwijn** blok in **Uiterlijken** en zet het in het **als...dan** blok: ✓



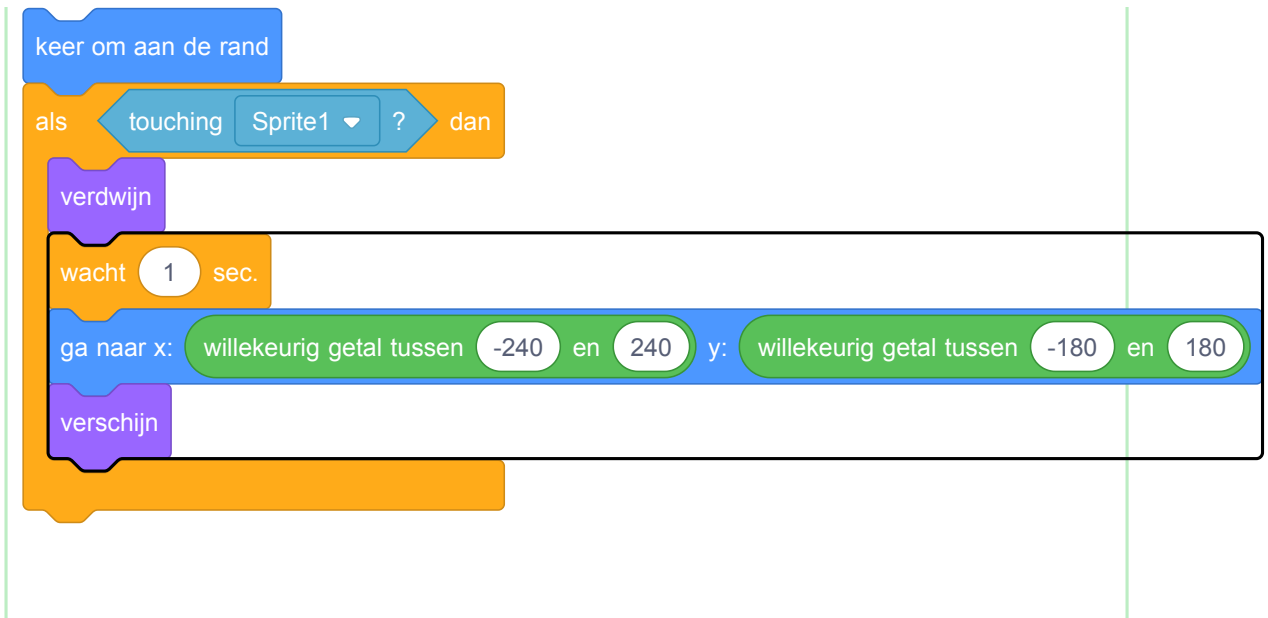
Als de haai nu de vis pakt, verdwijnt de vis definitief. Dat is niet zo mooi.

Zet het **verschijn** blok uit **Uiterlijken** helemaal aan het begin van de vis code, zodat je het spel kunt hervatten. ✓



Dat is al beter, maar je wil niet dat de speler het spel moet herstarten zodra er een vis is gevangen!

Werk je code bij in het **als...dan** blok om het er zo uit te laten zien: ✓



Hoe werkt dit?

Jij bent slim: als de vis verdwenen is, dan wacht hij, beweegt, en komt weer te voorschijn.

Het lijkt alsof er heel veel vissen verschijnen, maar het is slechts één sprite die beweegt!

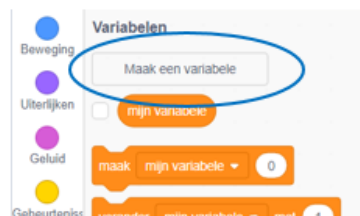
Dat is pas een spell Maar je hebt nog geen manier om de score bij te houden, of te winnen. Ook dat kun je oplossen - op de volgende kaart!

Stap 9 Score bijhouden

Om bij te houden hoeveel vissen de speler vangt, moet je iets hebben om de score in op te slaan, een manier om erbij op te tellen, en een manier om opnieuw te beginnen als het spel herstart wordt.

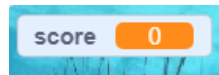
Ten eerste: de score opslaan!

Ga naar de **Variabelen** blokken categorie en klik op **Maak een variabele**. 



Type **score** als de naam.


Bekijk je nieuwe variabele!



Wat zijn variabelen?

Als je informatie wilt opslaan in een programma, dan gebruik je iets dat **variabele** heet. Zie het als een doos met een label erop: je kunt er iets instoppen, controleren wat er in zit, en veranderen wat er in zit. Je vindt de variabelen in de **Variabelen** categorie, maar je moet ze eerst zelf maken zodat ze te zien zijn!

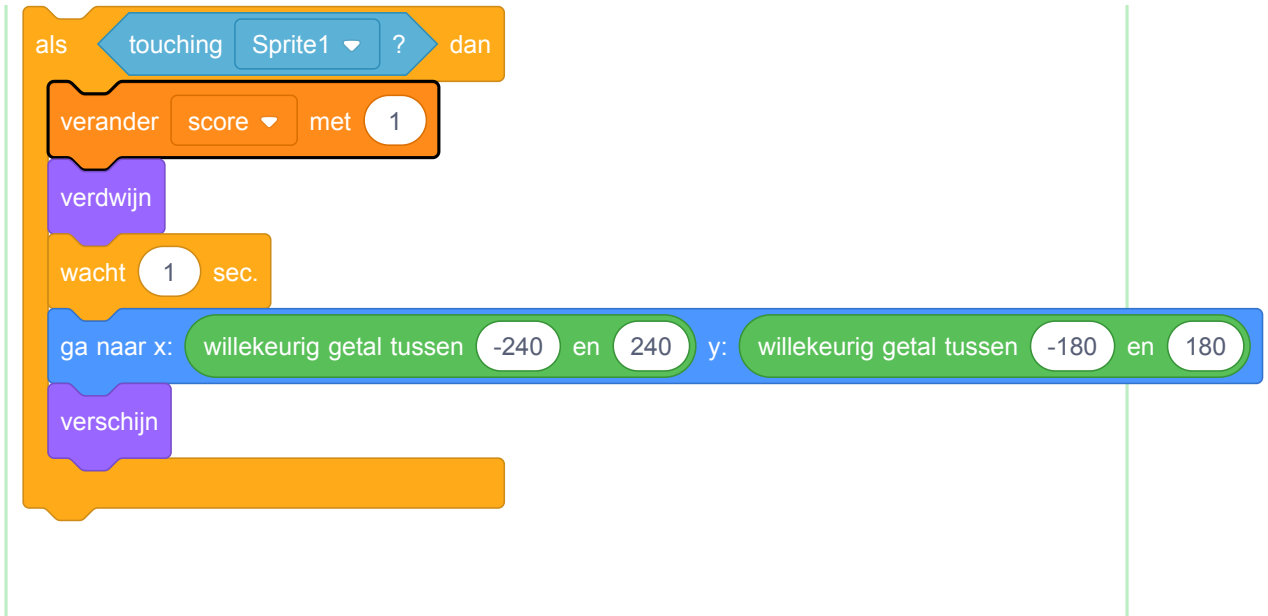
Nu moet de variabele elke keer bijgewerkt worden als de haai een vis pakt, en opnieuw beginnen als het spel wordt herstart. Beide zijn vrij simpel om te doen:

Pak uit de **Variabelen** categorie de **maak [mijn variabele v] [0]** en **verander [mijn variabele v] met [1]** blokken. Klik op de kleine pijltjes in de blokken, kies **score** uit de lijst, en zet de blokken in je programma: 

Code voor de haai



Code voor de vis



Gaaf! Nu heb je zelfs een score.

Uitdaging!

Uitdaging: het spel winnen

Kies een score waarbij de speler wint, en laat iets leuks gebeuren als ze die score halen. Laat de haai de speler feliciteren, laat een 'Jij wint!' sprite te voorschijn komen, of speel muziek af!

Door de community geleverde vertaling

Dit project werd vertaald door **Sanneke van der Meet** en gecontroleerd door **Lieve Cox**.

Onze geweldige vertalers helpen ons om kinderen over de hele wereld de kans te geven te leren coderen. Jij kunt ons helpen nog meer kinderen te bereiken door onze projecten te vertalen - lees meer op **rpf.io/translators** (<https://rpf.io/translators>).

Gepubliceerd door Raspberry Pi Foundation (<https://www.raspberrypi.org>) onder een Creative Commons license (<https://creativecommons.org/licenses/by-sa/4.0/>).

Bekijk project en licentie op GitHub (<https://github.com/RaspberryPiLearning/cd-beginner-scratch-sushi>)