

Denken zoals een computer?

Vaak doen we dingen waarbij we niet meer nadenken zoals een boterham smeren of schoenen aandoen. Vandaag gaan we leren denken als een robot. Misschien weten jullie het niet maar een robot die gemaakt is dom. De robot doet en zegt dingen die hij kent. Vandaag gaan we leren, hoe we handelingen aan een robot kunnen leren.

1.1 Instructie geven

Opdracht 1 bekijk het filmpje en beantwoord de vragen.

1) Waar gaat de video over?

2) Was de opdracht makkelijk voor Lucas ?

3) Wat deed de vader van Lucas na?

Wel we zien dat het niet zo gemakkelijk is om instructie te geven. Om te bewijzen dat het niet makkelijk is gaan we een oefening doen.

Opdracht 2

We gaan het spel spelen dat we juist in de video hebben gezien. Jullie gaan elkaar niet leren hoe je een boterham moet smeren. Leerling A leert leerling B hoe hij een fles moet opendoen. Leerling B leert leerling A hoe hij zijn boek van Mens & samenleving op tafel kan leggen.

1.2 Stappenplan

Om zeker van te zijn dat je ook in de toekomst op een correcte manier een instructie kunt geven hoe je een boterham kunt smeren kun je een stappenplan gebruiken. De verschillende stappen kun je weergeven in een algoritme. Een algoritme is een reeks logische instructies die er stap voor stap voor zorgen dat er bepaalde handelingen uitgevoerd worden.

Om een algoritme op te stellen, kun je drie controlestructuren gebruiken.

- Wanneer we een stappenplan maken over hoe je een boterham kun smeren gebruiken we vaak verschillende stappen in een vast volgorde. Indien we de eerste stap uitvoeren, volgt automatisch de volgende stap. Eerste neem ik brood daarna neem ik mijn mes vast.... Indien we opeenvolging stappen gebruiken noemen we het een sequentie.
- Wanneer we een stappenplan maken over hoe je moet oversteken gebruiken we een bepaalde voorwaarde. Indien de voorwaarde voldaan is er een actie anders niet (bijvoorbeeld als het licht groen is dan steek ik over anders blijf ik staan). Wanneer we een stappenplan hebben waarbij we een voorwaarde gebruiken noemen we het een selectie.
- Wanneer we een stappenplan maken over hoe je telkens je agenda moet bekijken, gaan we vaak herhalingen gebruiken. We gaan dus opeenvolgende stappen meermaals moeten herhalen (bijvoorbeeld agenda bekijken op maandag, dinsdag...). Indien we opeenvolgende stappen meermaals moet herhalen noemen we het een herhaling.

Opdracht 3

1) Geef drie voorbeelden waarbij je een sequentie nodig hebt om een algoritme op te stellen?

2) Geef drie voorbeelden waarbij je herhalingen nodig hebt om een algoritme op te stellen?

3) Geef drie voorbeelden waarbij je een selectie nodig hebt om een algoritme op te stellen?

1.3 Algoritme opstellen

Opdracht 4:

Vlaams minister van economie had ons gevraagd of we een app konden maken. Hij zou graag hebben dat zijn app kon zeggen of je genoeg budget had of een product of dienst te kopen.

Stap 1: Probleemdefinitie

Hoeveel budget heeft de minister?

Hoeveel kost het product dat de minister wil?

Het resultaat berekenen we door het budget af te trekken van de kost van het product.

Vervolgens moeten we werken met een bepaalde voorwaarde, als het resultaat $<$ is dan 0 dan kan de minister zijn product niet kopen anders kan hij het wel kopen.

Stap 2: Probleemanalyse

Bij de probleemanalyse is het de bedoeling dat we ons probleem ontleden in detail. Welke informatie heb je nodig om tot de oplossing te komen. Op welke manier kom je tot de oplossing.

Invoer:

- Budget en kost

Verwerking:

- Het resultaat bepalen door het budget af te trekken van de kost.
- Als het resultaat kleiner is dan nul kan ik het product niet kopen;
- Indien het resultaat groter is dan nul kan ik het product kopen.

Resultaat

- Toon het resultaat

Stap 3: Probleem omzetten in een computertaal

Om de probleemanalyse in een app te zetten, is het belangrijk dat we onze probleemanalyse omzetten naar een begrijpbare, logische taal voor de computer. Zoiets noemen we een pseudocode. Een pseudocode is bedoeld om het programmeren in een uiteindelijke programmeertaal te vergemakkelijken.

Pseudocode	Algoritme
Invoer Lees Budget Lees Kosten	Invoer De budget en kosten invoeren.
Verwerking Het resultaat = budget – kosten Als resultaat < 0 dan zeg “je kan het niet kopen”. Anders zegt “je kan het kopen”.	Verwerking Het resultaat bepalen we door budget – kosten te doen. Indien het resultaat < is dan nul, zegt de kat “je kan het niet kopen. Als het resultaat nu groter is dan nul zegt de kat je kan het kopen.
Resultaat Print resultaat	Resultaat Het resultaat wordt geprint

Stap 4 controle

Om zeker te zijn dat ons programma gaat werken gaan we controle moeten uitvoeren. Zo kunnen we eventuele fouten tijdig opsporen en verbeteren.

Invoer: welke gegevens kan de minister invoeren

Verwerking: Welke formule en voorwaarde heb ik nodig?

Opdracht 5

Vul de Synthese aan.

Om de app te maken voor de minister, heb je een reeks logische instructie gemaakt om te zorgen dat je de app kon maken. Dat noem je _____ een reeks van logische instructie die er stap voor stap voor zorgen dat bepaalde handelingen uitgevoerd worden.

De algoritme is dan de basis zijn om de app te kunnen maken in een programmeertaal.

Om een algoritme op te stellen, kun je drie controlestructuren gebruiken.

- Opeenvolging of _____ : wanneer er verschillende stappen moeten worden uitgevoerd in een bepaalde volgorde. Zodra de eerste stap is uitgevoerd, volgt automatisch de volgende stap.
- Keuze of _____ : wanneer aan een bepaalde voorwaarde wordt voldaan, volgt keuze 1; wanneer niet aan die voorwaarde wordt voldaan, volgt keuze 2.
- Herhaling of _____ : wanneer verschillende opeenvolgende stappen meermaals voorkomen en dus herhaald worden.

Om het algoritme om te zetten in een begrijpbare _____ taal, gebruik je de _____ Die is bedoeld om het _____ te vergemakkelijken.

Bij het opstellen van een algoritme gebruik je altijd een stappenplan volgens bepaalde structuur.

Stap 1: **probleemdefinitie**: Welke probleem moet er opgelost worden?

Stap 2: **probleemanalyse**: ontleed je het probleem in detail. Welke informatie heb je nodig om op welke manier tot welke oplossing te komen? Beschrijf de analyse in: invoer, verwerking en resultaat.

Stap 3: **algoritme in pseudocode**: Hoe zet ik mijn algoritme in een logische programmeertaal.

Stap 4: **controle**: Om zeker te zijn dat je algoritme goed is, moet je controle uitvoeren. Zo kunnen we eventuele fouten tijdig opsporen en verbeteren

1.4 Programmeren

Je hebt een algoritme geschreven, het is de bedoeling om het algoritme te programmeren. Om te programmeren gaan we werken met Scratch.

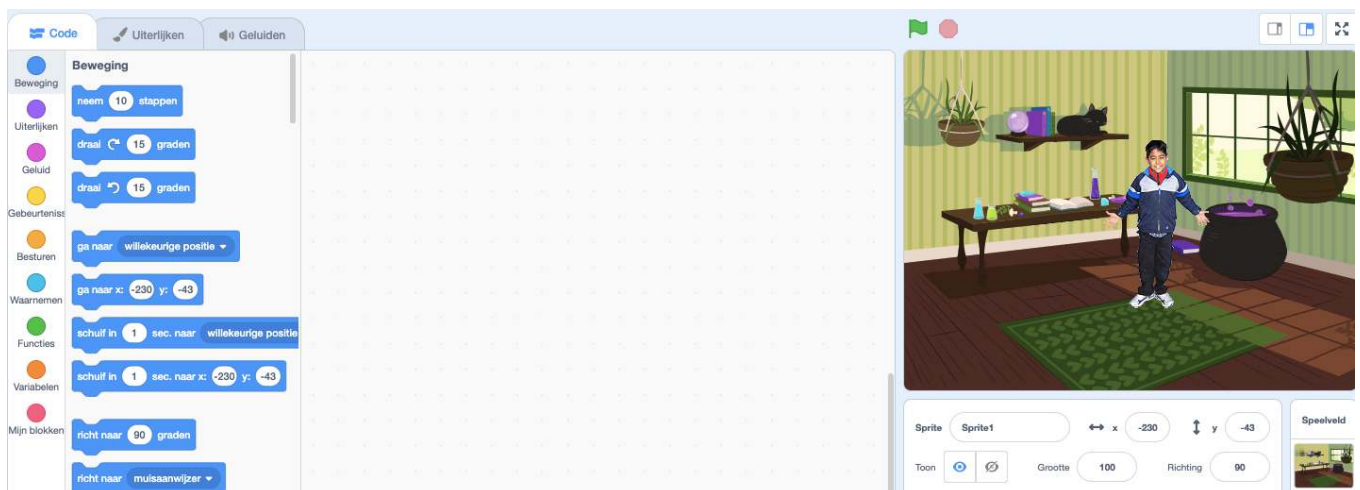
STAP 1

We moeten eerste zorgen dat het programma kan starten. Hiervoor moeten zoeken naar de logische programmeertaal.

We willen dat er iets **gebeurt**, in welke code van Scratch moeten we zoeken.

Welke blok van de code moeten we kiezen.

Sleep het gekozen blok naar het midden.



Code Uiterlijken Geluiden

Gebeurtenissen

- Beweging
- Uiterlijken
- Geluid
- Gebeurtenissen**
- Besturen
- Waarnemen
- Functies
- Variabelen
- Vlijn blokken

wanneer op  wordt geklikt

wanneer spatiebalk  is ingedrukt

wanneer op deze sprite wordt geklikt

wanneer achtergrond verandert naar  achtergr

wanneer volume  > 10

wanneer ik signaal  bericht1 ontvang


zend signaal  bericht1


zend signaal  bericht1 en wacht

Code Uiterlijken Geluiden


Gebeurtenissen


- Beweging
- Uiterlijken
- Geluid
- Gebeurtenissen**
- Besturen
- Waarnemen
- Functies
- Variabelen
- Vlijn blokken


wanneer op  wordt geklikt


wanneer spatiebalk  is ingedrukt


wanneer op deze sprite wordt geklikt

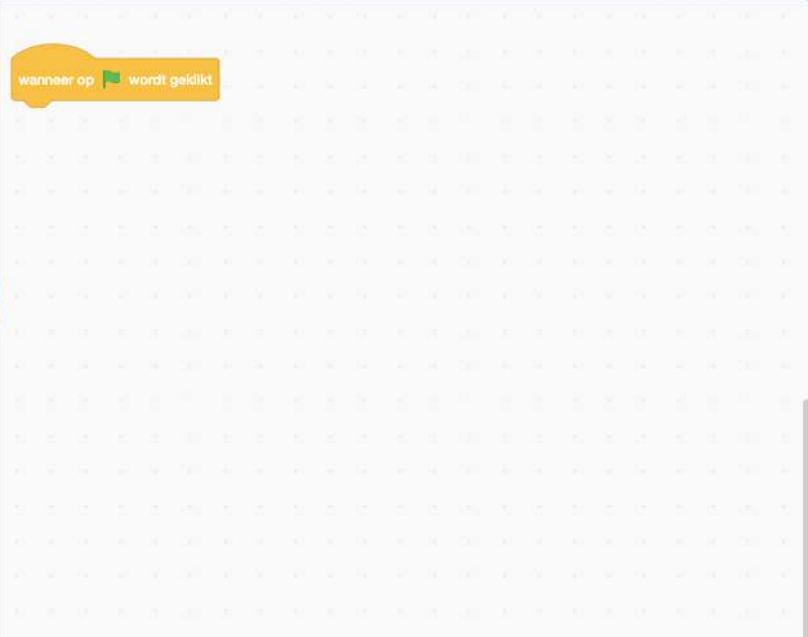
wanneer achtergrond verandert naar  achtergr

wanneer volume  > 10

wanneer ik signaal  bericht1 ontvang

zend signaal  bericht1

zend signaal  bericht1 en wacht



STAP 2

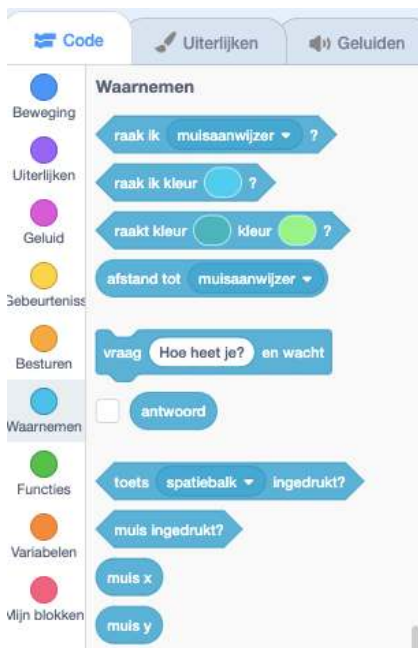
Het is de bedoeling dat de figuur ons vraagt: "Wat is jouw budget?"

Je wilt **zien** dat de figuur vraagt: Wat is jouw budget, in welke code van Scratch moeten we zoeken.

Welke blok van de code moeten we kiezen.

Sleep het gekozen blok naar het midden.

Klik op "Hoe heet je" en verander het naar: "Wat is jouw budget?"

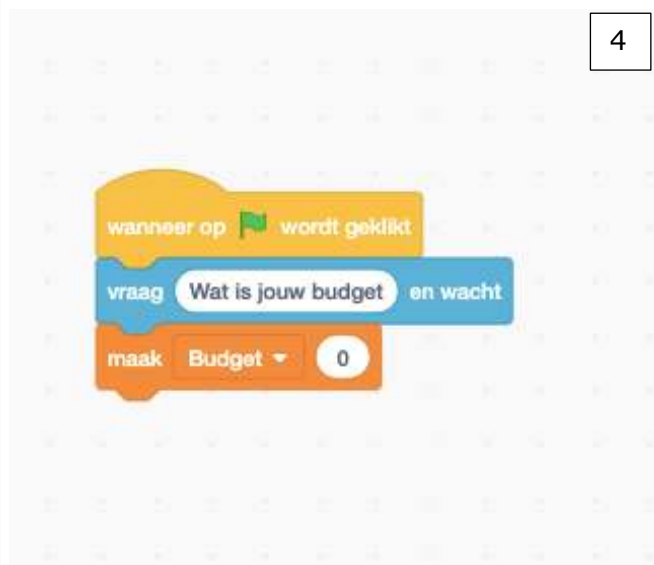
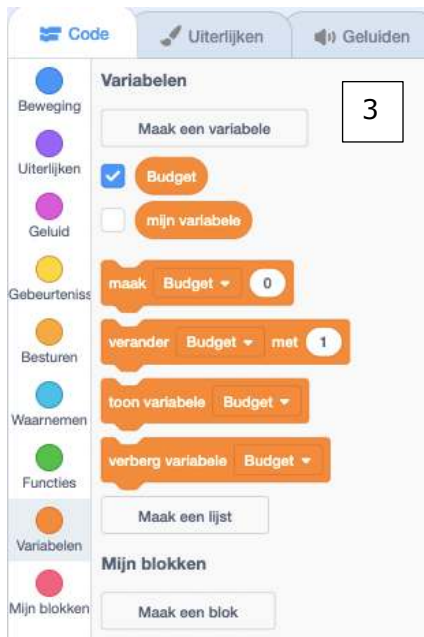
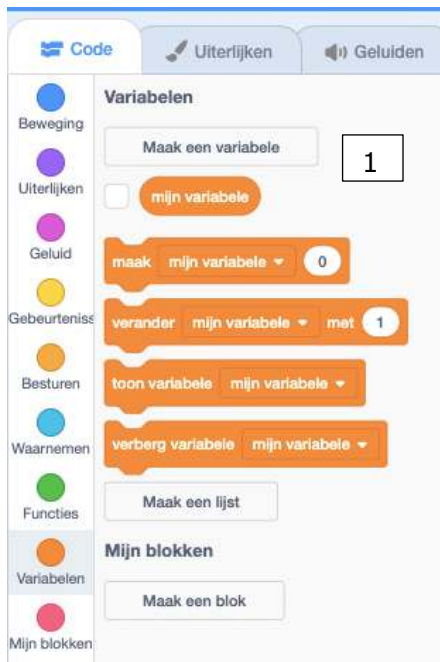




STAP 3

Je moet nu een box maken waar je het antwoord kunt opslaan van "wat is jouw budget". Hiervoor moet je naar Variabelen gaan. In variabelen gaan we een box maken waar we een antwoord in kunnen opslaan.

- 1 Ga naar Variabelen
- 2 Klik op "maak een variabele"
- 3 Geef de naam Budget
- 4 Verander maak mijn variabele naar budget en sleep het naar het midden

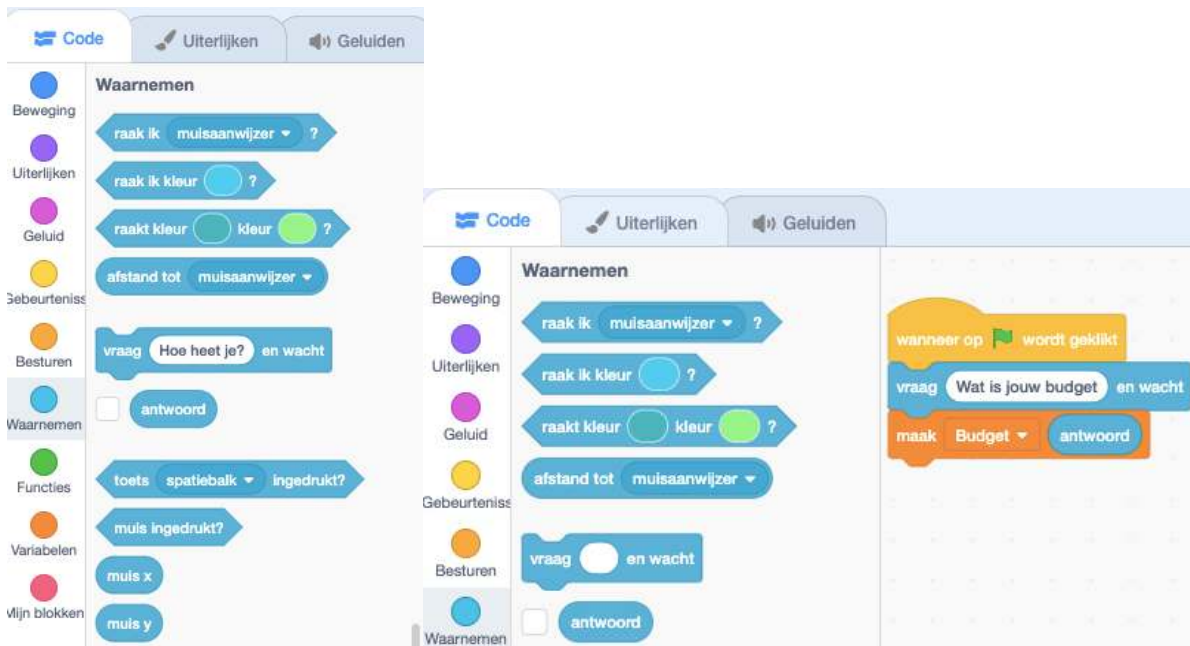


STAP 4

Je moet ervoor zorgen dat we het antwoord van "Wat is jouw budget" koppelen aan de variabele.

Ga naar waarnemen zoek naar "antwoord"

Sleep "antwoord" naar waar het getal 0 staat.



STAP 5

Het is de bedoeling dat de figuur ons vraagt: "Hoeveel kost het product?"

Je wilt **zien** dat de figuur vraagt: Wat is jouw budget, in welke code van Scratch moeten zoeken.

Welke blok van de code moeten we kiezen.

Sleep het gekozen blok naar het midden.

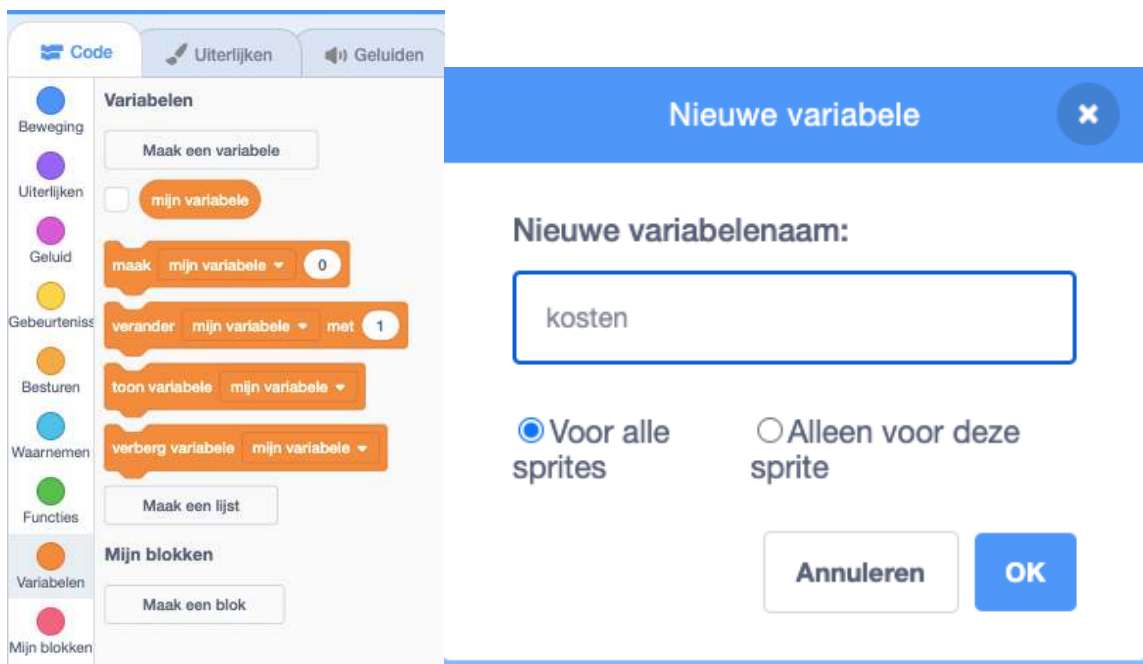
Klik op "Hoe heet je" en verander het naar: "Hoeveel kost het?"



STAP 6

Je moet nu een box maken waar je het antwoord kunt opslaan van "Hoeveel kost het product". Hiervoor moet je naar Variabelen gaan. In variabelen gaan we een box maken waar we een antwoord in kunnen opslaan.

- 1 Ga naar Variabelen
- 2 Klik op "maak een variabele"
- 3 Geef de naam Kosten
- 4 Verander "maak mijn variabele" naar "Kosten" en sleep het naar midden

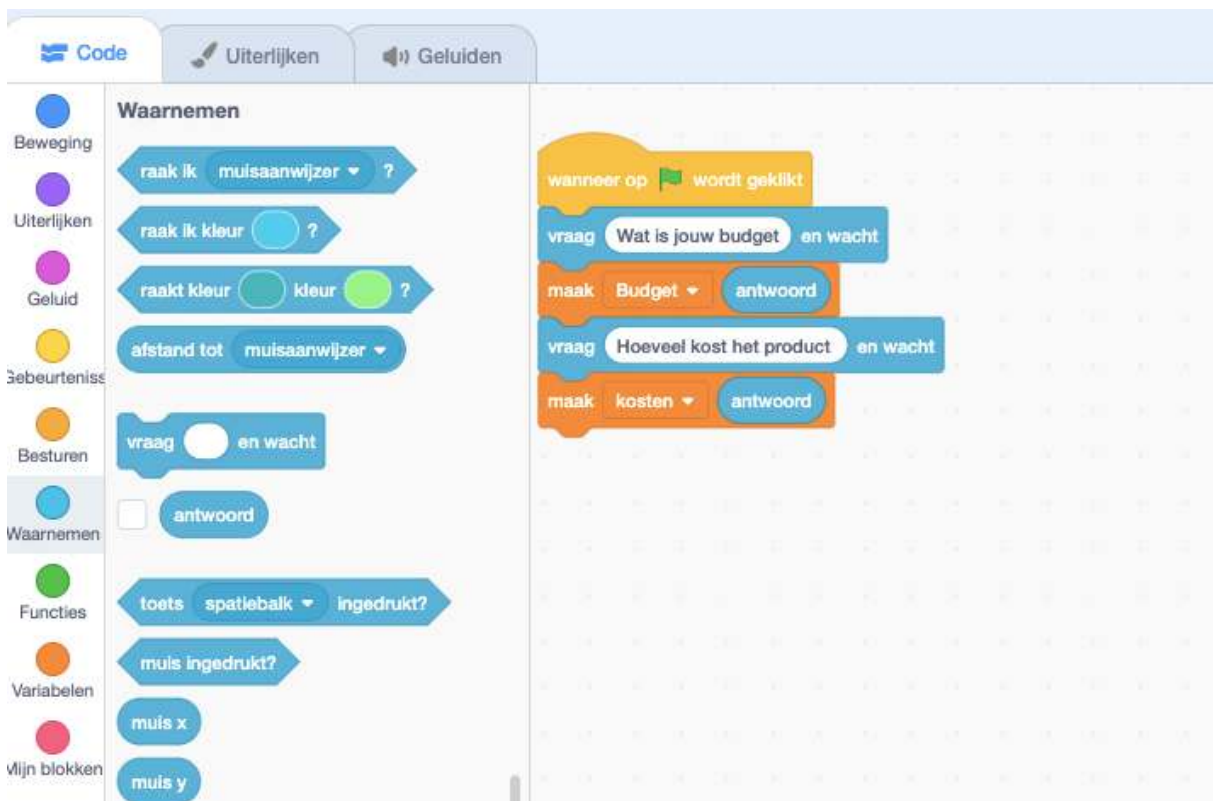
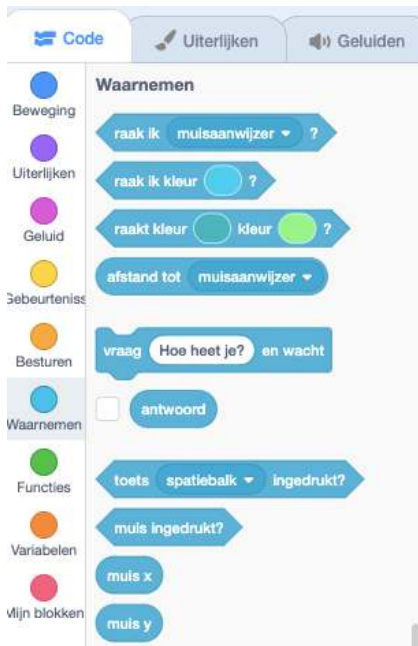


STAP 7

Je moet ervoor zorgen dat we het antwoord van "Hoeveel kost het product" koppelen aan de variabele.

Ga naar waarnemen zoek naar antwoord

Sleep antwoord naar waar het getal 0 staat.



STAP 8

Je moet nu een box maken waar we Het resultaat kunnen opslaan. Om het resultaat te berekenen gingen we budget – kosten doen. Hiervoor moet je naar Variabelen gaan. In variabelen ga je een box maken waar je het antwoord in kunt opslaan.

1 Ga naar Variabelen

2 Klik op "maak een variabele"

3 Geef het de naam "Het resultaat"

4 Verander maak mijn variabele naar budget en sleep het naar midden achter (maak kosten antwoord)

The image consists of four numbered screenshots showing the process of creating a variable in Scratch:

- 1**: The 'Variabelen' (Variables) menu is open, showing the 'Maak een variabele' (Make a variable) button.
- 2**: The 'Nieuwe variabele' (New variable) dialog box is shown. The variable name is 'Het resultaat' (The result).
- 3**: The 'Variabelen' menu is shown with 'Budget', 'Het resultaat', and 'kosten' (costs) checked, indicating they are active variables.
- 4**: A Scratch script is shown with the following blocks: 'wanneer op vlag wordt geklikt' (when green flag clicked), 'vraag Wat is jouw budget en wacht' (ask 'What is your budget' and wait), 'maak Budget antwoord' (make 'Budget' answer), 'vraag Hoeveel kost het product en wacht' (ask 'How much does the product cost' and wait), 'maak kosten antwoord' (make 'kosten' answer), and 'maak Het resultaat 0' (make 'Het resultaat' 0).

STAP 9

De bedoeling is om budget – kosten te koppelen aan het resultaat.

Je wilt **een functie gebruiken**, in welke code van Scratch moeten we zoeken.

Welke blok van de code moeten we kiezen.

Sleep het gekozen blok naar de nul (maak het resultaat 0).



The screenshot shows the Scratch Code editor interface. The top navigation bar includes 'Code', 'Uiterlijken', and 'Geluiden'. The left sidebar shows various block categories: Beweging, Uiterlijken, Geluid, Gebeurtenissen, Besturen, Waarnemen, Functies, and Variabelen. The 'Functies' category is selected, displaying several arithmetic blocks: addition (+), subtraction (-), multiplication (*), division (/), and a 'willekeurig getal tussen 1 en 10' block. The main workspace shows a script starting with 'wanneer op vlag wordt geklikt', followed by 'vraag Wat is jouw budget en wacht', 'maak Budget antwoord', 'vraag Hoeveel kost het product en wacht', 'maak kosten antwoord', and 'maak Het resultaat' with a comparison block set to '< 0'.

STAP 10

Je moet je formule afmaken want nu leest scratch het resultaat is niks – niks. We moeten naar variabelen om de formule af te maken.

Sleep budget naar het eerste vak

Sleep erna kosten naar het tweede vak.

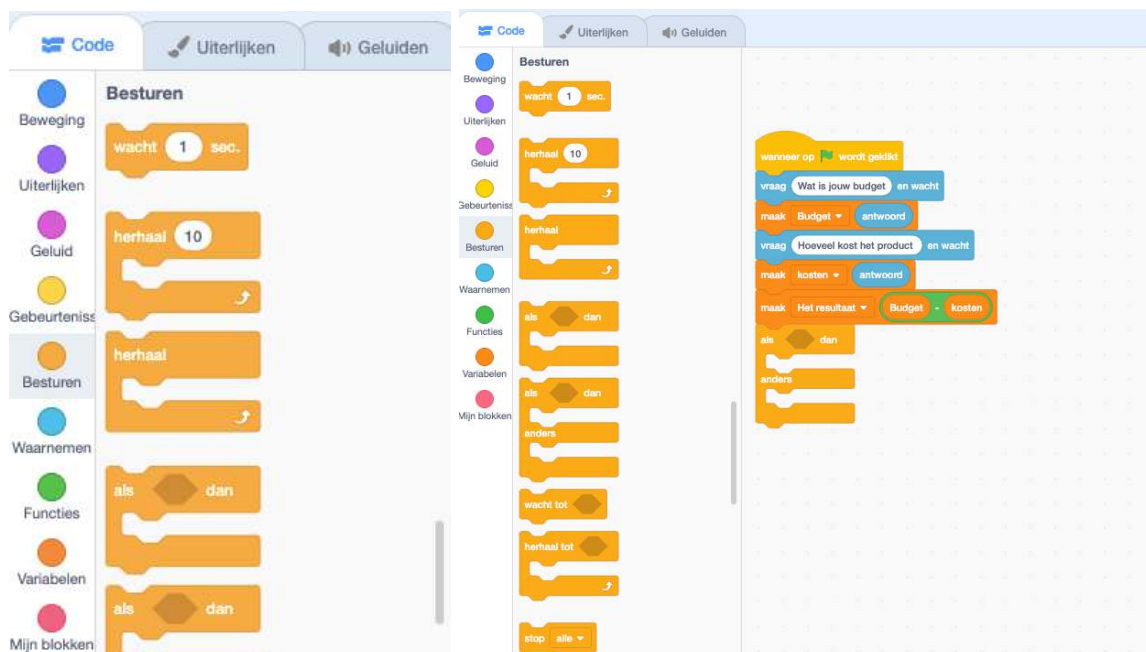


STAP 11

Je bent bijna klaar we moeten nu nog een voorwaarde toevoegen aan het resultaat.

Daarvoor zal je moeten kijken naar de blokken onder besturen. De structuur die we gaan gebruiken is de als en anders.

Sleep als en anders naar het midden. (Na maak het resultaat budget – kosten)



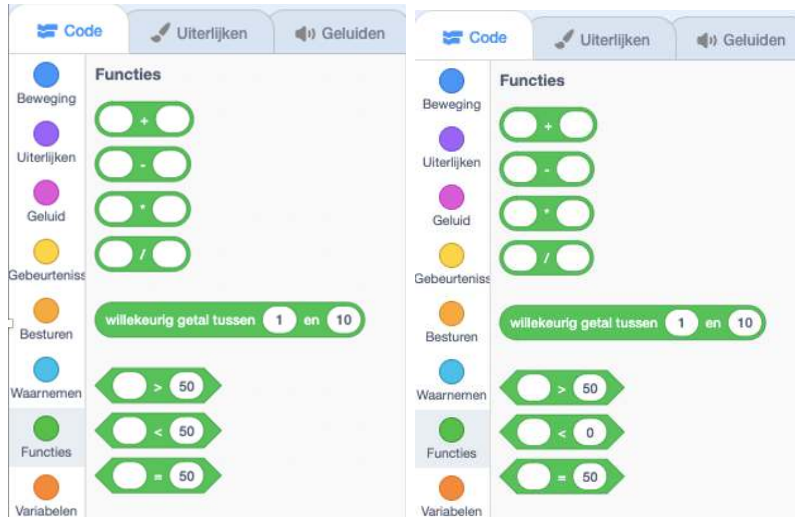
STAP 12

Je gaat een functie toevoegen aan de voorwaarde

Kies kleiner dan 50

Verander 50 naar 0

Sleep " < 0 " in de voorwaarde bij de "als en dan"



STAP 13

Je moet nu iets vergelijken met 0. Iets bestaat niet, je hebt het resultaat berekend. Zet de variabele "het resultaat" in het leeg vak. Zo leest het programma "als resultaat < is dan nul".

Ga naar Variabelen

Sleep de Het resultaat naar het leeg vak.



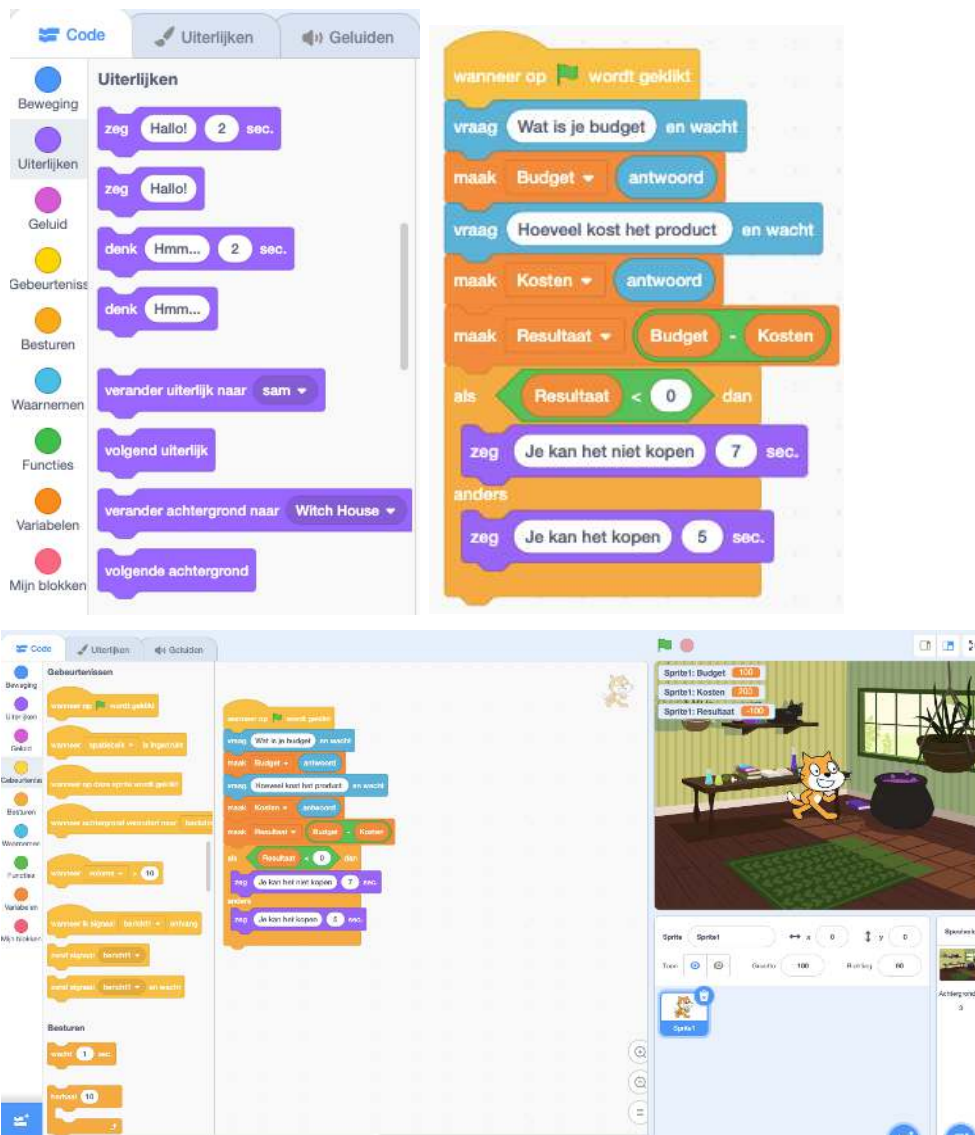
STAP 14

Je hebt de functie toegevoegd aan de voorwaarde. Nu gaan we de actie toevoegen. Als het resultaat kleiner is dan nul zegt het programma "Je kan het niet kopen", anders zegt het programma "Je kan het kopen".

We willen dat de **figuur** zegt: "Je kan het niet kopen of Je kan het kopen", tussen welke blokken van Scratch moeten we zoeken.

Welke blok van de code moeten we kiezen.

Sleep het gekozen blok naar het midden (tussen als en anders zet je: "Je kan het niet kopen", en tussen anders en niks zet je "Je kan het kopen")



1.5 Opdracht

Je hebt geleerd hoe je een algoritme kunt opstellen en daarna programmeren. De bedoeling is dat je zelf onderzoek gaat naar een probleem. Je lost het probleem op a.d.h.v. algoritmen, daarna ga je er een programma over schrijven in Scratch. Inspiratie kan je vinden op deze site. Het algoritme wordt weergegeven in een Word-document.

<https://sites.google.com/psdiepenbeek.be/techniek/de-lessen/scratch>

Totaal /15	
Een algoritme juist schrijven	-4
Het programma werkt	-4
Je document opslaan als achternaam_voornaam	-2
Deadline gehaald	-5