

Kucken, was Ozobot Evo kann

Lektion 1 (Lehrer)
Ozobot Evo

Zusammenfassung der Lektion

Die Schüler lernen Ozobot Evo kennen und lernen Bewegungen, Lichteffekte und Geräusche anzuwenden. Anhand eines kurzen Szenarios lernen sie, einen Code zu erstellen. Dafür wird den Schülern ein Hilfsmittel von gedruckten Wörtern gegeben. In späteren Lektionen lernt der Schüler dies nach und nach selbstständig zu tun.

Was ist nötig?

- Ozobot
- Tablet, Laptop oder Computer

Zeit:

40-50 Minuten

Achtung: Für diese Lektion ist es wichtig, dass Ozobot Evo eine Firmware von v1.4 oder höher besitzt. Im Lehrerleitfaden finden Sie ein separates Blatt, das erklärt, wie Sie dies überprüfen können und wie Sie Ozobot aktualisieren können.

Lernziele

- 1 Problem umformulieren.
- 1.3 (Um)formulieren eines Problems, so dass ein Computer es lösen kann.
- 2 Daten sammeln.
- 2.1 Sammeln von brauchbaren Daten aus einer Datensammlung.
- 4 Daten visualisieren.
- 4.1 Aufbereiten von verschiedenen Arten von Daten für die Verwendung mit dem Computer (z.B. binäre Codes).
- 7 Algorithmen und Methoden.
 - 7.1 Ausführen einer Aufgabe durch schrittweises Ausführen einer Reihe von Handlungen.
 - 7.10 Kreieren von einfachen Reihen von Anweisungen, die zu einem bestimmten Ziel führen.
 - 7.12 Sammeln von praktischer Erfahrung mit einer Programmierumgebung (Software).

Antworten

Die Antwort kann von Schüler zu Schüler unterschiedlich sein. Kontrollieren Sie, ob der Schüler die folgenden Elemente seinem Code in der richtigen Reihenfolge hinzugefügt hat:

- Fahren
- Anhalten
- Rotes Licht
- Warten (lang)
- Evo ungeduldig/betreten
- Grünes Licht
- Wegrennen (schnell)
- Polizei Licht
- Evo erschrickt
- Stillstehen
- Evo fröhlich
- Ruhig fahren

Die richtige Antwort muss dem Code, der im Folgenden wiedergegeben ist, einigermaßen ähneln.

Es ist wichtig, dass der Schüler in jedem Fall die folgenden Blöcke verwendet hat:

- "Movement" (gelb)
- "Light effect" (rosa)



- "Wait" (blau)
- "Sounds" (violett)



Kucken, was Ozobot Evo kann

Lektion 1
Ozobot Evo

Was lernst du?

Wir werden Ozobot Evos neue Funktionen ausprobieren. Evo hat Ton, zusätzliche Lämpchen und Sensoren!

Was werden wir machen?

Du wirst anhand einer Geschichte einen eigenen Code erstellen. Die fett gedruckten Wörter in der Geschichte musst du in deinem Code verwenden.

Was brauchst du?

- Ozobot Evo
- Tablet, Laptop oder Computer

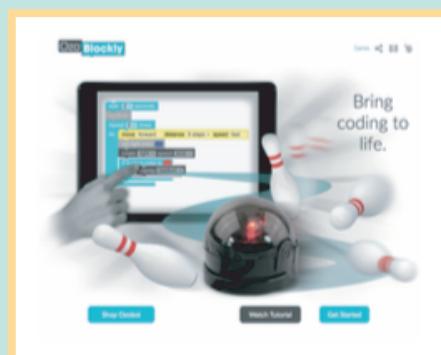
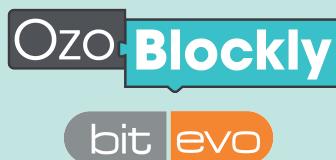
Zeit:

20 Minuten

Los geht's!

Gehe auf www.ozoblockly.com und klicke auf 'Get Started'

- 1 Schließe den Bildschirm, der nun aufgeht, indem du auf das 'x' klickst.
- 2 Klicke links oben auf 'Evo' unter dem Wort Ozoblockly.



- 3 Wir fangen immer mit einem leeren Bildschirm an. Das tust du, indem du rechts unten auf das Papierkorb-Icon und danach auf 'yes' (ja) klickst.
- 4 Wähle Level 1 aus. Das machst du, indem du die Zahl „1“ oben im Menü auf der linken Seite anklickst.

Der Auftrag

Lasst uns mal sehen, was Ozobot Evo so alles kann. Erstelle anhand der folgenden Geschichte einen Code für Evo.

Stell' dir vor, dass Evo auf der Straße fährt. Er muss wegen einer roten Ampel anhalten und einen Moment warten. Weil es so lange dauert wird Evo ungeduldig und ein bisschen niedergeschlagen. Die Ampel springt endlich auf grün und Evo fährt los. Er fährt so schnell, dass die Polizei kommt! Ozobot erschrickt und muss am Straßenrand stehenbleiben. Zum Glück muss er kein Bußgeld zahlen! Er kann glücklich und entspannt weiterfahren.



- 5 Erstelle anhand der oben stehenden Geschichte einen Code für Evo. Schreibe eventuelle zuerst für dich selbst auf, welche Dinge aus der Geschichte du in deinem Code verwenden kannst.

- 6 Wenn du fertig bist, ist dein Code bereit, um geladen zu werden. Du kannst Ozobot Evo genauso programmieren wie Ozobot Bit 2.0. Hast du vergessen, wie das geht? Dann schaue dir Folgendes an!
- 7 Klicke auf das Icon unter Flashing.  FLASHING
- 8 Schalte Ozobot Evo an.
- 9 Halte ihn gegen die weiße Stelle auf dem Bildschirm (Kalibriere Ozobot Evo zuerst an diese Stelle!).
- 8 Klicke auf 'Load bit'.  Load Bit
- 8 Spiele deinen Code jetzt ab, indem du zweimal auf Evos An-Aus-Knopf drückst. Lies währenddessen die Geschichte und schaue dabei, ob Evo die Geschichte abbildet. Hat es geklappt? Super!!

Sollte etwas nicht funktionieren, dann kannst du deinen Code natürlich immer anpassen, bis dein Code richtig ist.
Viel Erfolg!

Hindernissen aus dem Weg gehen

Lektion 2 (Lehrer)
Ozobot Evo

Achtung: Für diese Lektion ist es wichtig, dass Ozobot Evo eine Firmware von v1.4 oder höher besitzt. Im Lehrerleitfaden finden Sie ein separates Blatt, das erklärt, wie Sie dies überprüfen können und wie Sie Ozobot aktualisieren können.

Zusammenfassung der Lektion

Die Schüler erhalten einen Code, in dem Fehler sind. Die Fehler müssen Sie herausnehmen. Durch 'error and trial' (Fehler machen und immer wieder neu probieren) filtern sie die Fehler heraus. Wenn sie den Code verbessert haben, können sie Wettbewerbe gegeneinander austragen. Dadurch, dass sie Ozobot Evo jetzt richtig programmiert haben, können sie Evo mit ihren Händen um die Fahnen und zum schwarzen Block führen. Der Evo, der zuerst am schwarzen Block ist und 'black' (schwarz) sagt, hat gewonnen!

Was ist nötig?

- Ozobot
- Tablet, Laptop oder Computer
- Arbeitskarte 2 (Vor- und Rückseite)

Zeit:

30 Minuten.

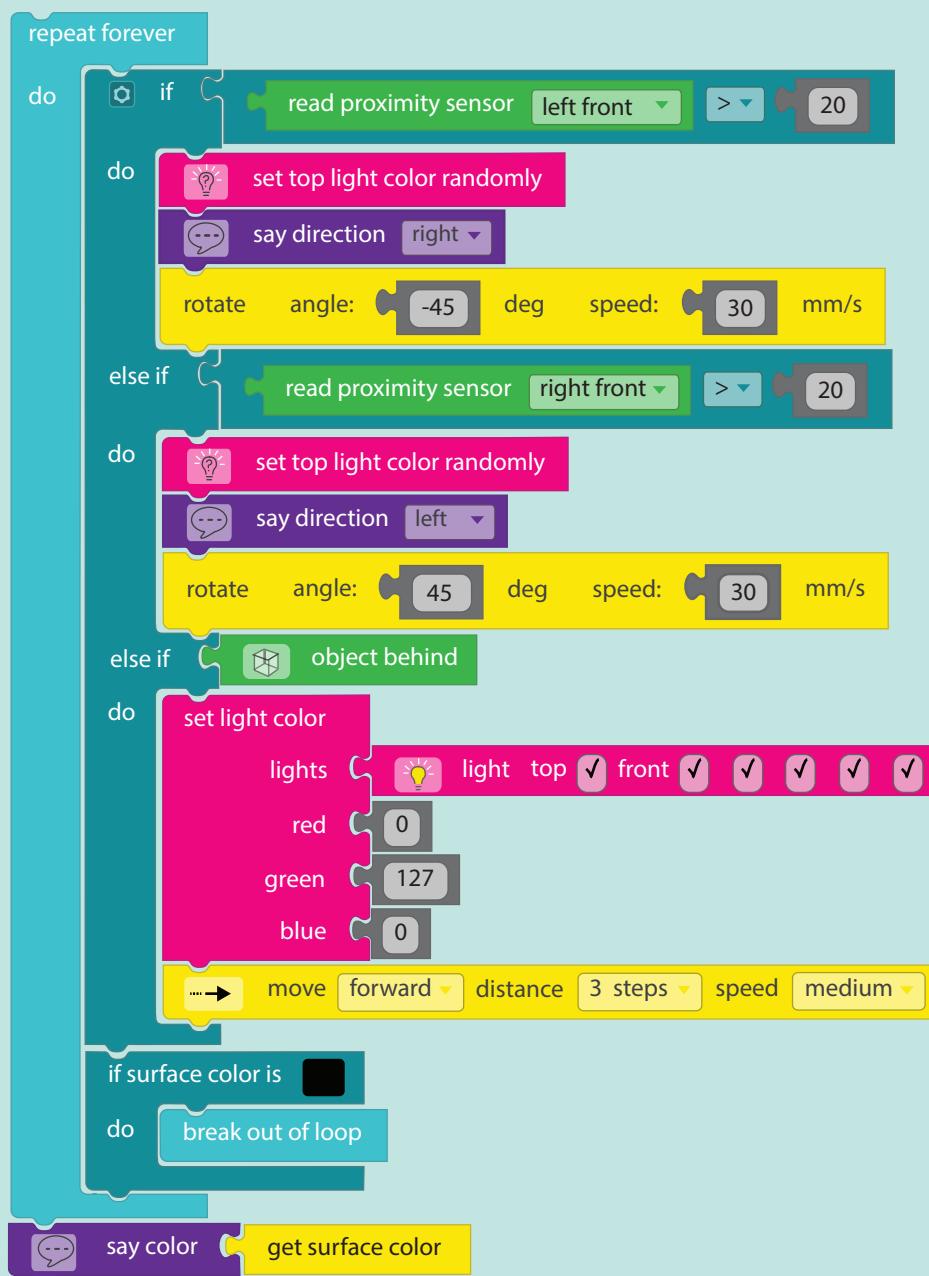
Lernziele

- 3 Daten analysieren.
- 3.2 Fortsetzen und Erstellen von Mustern in konkreten Situationen.
- 3.5 Entdecken von zu vereinfachenden Mustern.
- 5 Problemzerlegung.
- 5.2 (Teil-) Aufträge in eine logische Reihenfolge bringen.
- 5.6 Überprüfen, ob kein wichtiger Teil bei der Ausführung von Teilaufträgen ausgelassen oder vergessen wurde.
- 6 Automatisierung.
- 6.2 Erkennen, dass ein Computer eine Aufgabe endlos wiederholen kann (z. B. in Übungen).
- 7 Algorithmen und Methoden.

- 7.11 Fehlererkennung in Algorithmen durch logisches Schlussfolgern und Verbessern davon (Debugging).
- 7.16 Verwendung einer Wenn-Dann-Konstruktion beim Beschreiben von Schritten in einem Prozess.
- 7.17 Vorhersagen von Verhalten bei der Arbeitsweise einfacher (Computer-) Programme durch logisches Schlussfolgern.

Antworten

Die Antwort kann von Schüler zu Schüler leicht variieren. Wir empfehlen Ihnen in der Praxis zu schauen, ob der Code funktioniert. Die richtige Antwort sollte dem folgenden Code etwas ähneln.



Ihr Schüler kann Sie um Ihr Antwortblatt bitten, um seinen/ihren Code nachzuschauen, da dies auf seiner/ihrer Arbeitskarte angegeben ist.

Hindernissen aus dem Weg gehen

Lektion 2
Ozobot Evo

Was lernst du?

In dieser Lektion lernst du, die Sensoren von Ozobot Evo zu verwenden und Ozobot Evo mit deinen Händen zu steuern, ohne dass du ihn anfasst.

Was werden wir machen?

Wir werden Ozobot Evo so programmieren, dass er Hindernissen ausweicht. Im ersten Auftrag nimmst du die Fehler aus dem Code, der auf deiner Arbeitskarte steht.

Was brauchst du?

- Ozobot Evo
- Tablet, Laptop oder Computer
- Arbeitskarte 2 (Vorder- und Rückseite)

Zeit:

30 Minuten

Der Auftrag

- 1 Verwende Arbeitskarte 2. Auf der Rückseite der Arbeitskarte findest du den Code für Ozobot. Der Code sorgt dafür, dass du Ozobot Evo mit den Händen durch den Parcours führen kannst (den kannst du auf der Vorderseite der Arbeitskarte 2 finden). Wenn du Evo um die Fähnchen fahren lassen und zum schwarzen Block gebracht hast, dann muss Ozobot anhalten und die Farbe Schwarz sagen. Aber der Code, so wie er auf der Arbeitskarte steht, ist nicht ganz richtig... Es ist deine Aufgabe, die Fehler aus dem Code zu nehmen.
- 2 Bilde den Code von deiner Arbeitskarte auf Ozoblockly.com nach. Achtung: Es befinden sich Fehler im Code der Arbeitskarte. Diese musst du herausnehmen. Du arbeitest in Level 3 & 4 (links oben auf dem Bildschirm kannst du die Level finden). Sorge dafür, dass du alle Blöcke gelöscht hast, bevor du mit dem Kodieren beginnst.
- 3 Wenn du nicht weißt, was du tun musst, dann probiere zuerst aus, den Code, der auf der Arbeitskarte steht, auf Ozobot Evo zu laden und ihn zu steuern, indem du deine Hände hinter und neben Ozobot hältst (ohne ihn anzufassen!). So kannst du dahinter kommen, was das Problem ist.

Der Code ist richtig, wenn:

- Ozobot Evo von deinen Händen wegfährt. Also: Evo muss nach vorne fahren, wenn du deine Hand hinter ihm hältst. Er muss nach rechts fahren, wenn du deine Hand links von ihm hältst und nach links, wenn du deine Hand rechts von ihm hältst.
- Wenn er so programmiert ist, musst du ihn, wenn es gut läuft, um die Fähnchen herumführen können.
- Ozobot Evo muss die Schleife verlassen, wenn er auf den schwarzen Block kommt;
- Evo muss die Farbe Schwarz sagen, wenn er auf den schwarzen Block kommt.

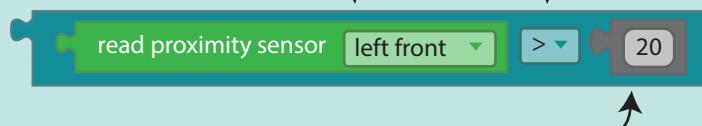
Tipps & Hilfe

Was bedeuten die Codes?

Bedeutet: Die Empfindlichkeit des Sensors muss sein....

"left front" bedeutet „links vorne“. Gebe hier den Sensor ein, den du kodieren willst.

Bedeutet: „Größer als“. Eine andere Option ist zum Beispiel „kleiner als“ oder „gleich“.



Bedeutet: Je höher die Zahl, desto näher muss ein Objekt vor Ozobot Evo sein, um es zu erkennen. 20 steht für eine hohe Empfindlichkeit von Evos Sensor. Oder auch: Ozobot Evo sieht ein Objekt sehr schnell, auch wenn er weiter weg ist. Du kannst in dieses Feld Zahlen von 0 bis 127 eingeben.

Bedeutet: Der Drehwinkel. Mit dieser Option kannst du genau einstellen, um wie viel Grad sich Ozobot drehen muss. Das Wort "deg" nach dem grauen Feld bedeutet degree, oder auch ‚Grad‘.

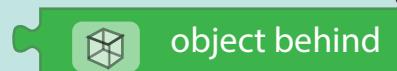
Hier gibst du an, wie schnell Ozobot sich drehen muss. Ozobot kann zwischen 15 und 85 Millimeter pro Sekunde fahren.

Tipp: Wenn Ozobot beim Drehen rutscht oder umfällt, dann kannst du die Geschwindigkeit anpassen, um dies zu verhindern.



In dieses Feld trägst du die Gradzahl ein, um die sich Ozobot Evo drehen muss. Du kannst Zahlen zwischen -128 und 127 eintragen. Willst du, dass sich Ozobot noch weiter dreht, dann füge einfach noch einmal den gleichen Block hinzu. Wenn du eine negative Zahl (zum Beispiel -20) einträgst, dann wird sich Ozobot nach rechts drehen. Wenn du eine positive Zahl einträgst (zum Beispiel 40), dann wird sich Ozobot nach links drehen.

Bedeutet: 'Objekt oder Gegenstand hinter Ozobot Evo' in diesem Fall haben wir Ozobot Evo dann so kodiert, dass er nach vorne fährt, um nicht gegen den Gegenstand zu fahren.



Mit diesem Block kannst du den LED Leuchten von Ozobot Evo eine Farbe geben. Die Lämpchen, die du angeklickt hast, leuchten dann in der Farbe, die du eingegeben hast.



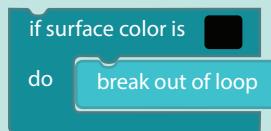
Hier gibst du an, in welcher Farbe du die LED Lampen sehen willst. Du kannst hier einen Wert zwischen 0 und 127 eingeben.

Abhängig von der Menge, die du in jedes Feld eingetragen hast, wird Ozobot jede Farbe, die du dir überlegt hast, anzeigen können.

Ein bisschen rot mit ein bisschen blau ergibt natürlich violett, ein bisschen rot mit ein bisschen grün ergibt orange. Probier's mal aus! In diesem Auftrag ist dieser Block nicht so wichtig.

Bedeutet: "Höre auf mit dem Folgen der Schleife".

Das kennst du vielleicht bereits. Bedeutet: "Wenn die Untergrund-Farbe von Ozobot Evo ist, dann tue...."



Ein violetter Block bedeutet immer, dass es etwas mit einem Geräusch zu tun hat. In diesem Fall muss er eine Farbe sagen ('say color'). Der gelbe Block steht für den Untergrund ('surface') von Evo. Ozobot Evo soll die Farbe sagen, auf der er steht.



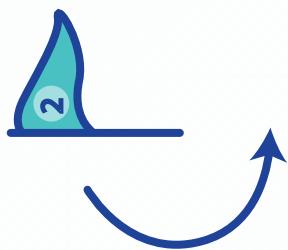
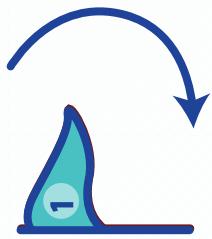
TIPP: Willst du mehr über die Codes wissen? Du kannst auf Ozoblockly nachschauen, was ein Code bedeutet. Klicke auf das Buch im rechten Menü (die zweite Schaltfläche von oben). Benutze die Suchfunktion, um einen Code zu suchen oder klicke auf einen Code, den du schon auf deinen Bildschirm gezogen hast.

Spielregeln

- Beginne auf dem Startpunkt.
- Führe Ozobot Evo mit deinen Händen die richtige Seite hoch, aber du darfst Evo nicht anfassen.
Um ihn nach vorne zu bewegen, halte deine Hand hinter Evo. Um Ozobot nach rechts zu bewegen, halte deine Hand links von Ozobot (sodass er von der Hand wegfährt)
- Wenn Ozobot Evo ein Fähnchen berührt, beginnst du von vorn, indem du Evo wieder auf den Startpunkt setzt.

Arbeitskarte 2 Vorderseite

Lektion 2



Arbeitskarte 2 Rückseite

Lektion 2

```
repeat forever
  do
    if [read proximity sensor > 20]
      do
        set top light color randomly
        say direction right ▶
        rotate angle: 45 deg speed: 30 mm/s
    else if [read proximity sensor > 20]
      do
        set top light color randomly
        say direction left ▶
        rotate angle: -45 deg speed: 30 mm/s
    else if [object behind]
      do
        set light color
          lights light top front ✓ ✓ ✓ ✓
          red 0
          green 127
          blue 0
        move backward distance 3 steps ▶ speed medium ▶
    if surface color is blue
      break out of loop
    do
      say color
      get surface color
```

Bis 10 zählen

Lektion 3 (Lehrer)
Ozobot Evo

Achtung: Diese Lektion besteht aus 2 Teilen. Es wird empfohlen, den ersten Teil der Lektion in der Klasse durchzunehmen. Da Ozobot Evo umfangreichere Optionen und Codes enthält, ist es wichtig, die Wortschatzliste und die verschiedenen Code-Blöcke zu erklären, bevor Ihre Schüler selbstständig arbeiten.

Zusammenfassung der Lektion

Die Schüler lernen eine Anzahl von neuen Anwendungen für Evo. Sie erhalten den Auftrag, Ozobot Evo bis 10 zählen zu lassen, zusammen mit einigen Spezifikationen, die ihr Code erfüllen muss.

Was brauchen Sie?

- Ozobot
- Tablet, Laptop oder Computer
- Arbeitskarte 3

Zeit:

Gesamtzeit Selbstständige Lektion (Teil 1, 2 & 3):
30-40 Minuten
Zeit in der Klasse erklären (Teil 1): 20 Minuten
Zeit selbstständige Lektion (Teil 2 & 3): 20 Minuten

Lernziele

- 1 Problem umformulieren.
 - 1.1 Das Erkunden der Möglichkeiten, Probleme mit einem Computer zu lösen.
 - 1.3 Umformulieren eines Problems, sodass ein Computer es lösen kann.
- 2 Daten sammeln.
 - 2.1 Sammeln von brauchbaren Daten aus der Datensammlung.
- 3 Daten analysieren.
 - 3.1 Realisieren, was ein Muster ist.
 - 3.2 Fortführen und Erstellen von Mustern in konkreten Situationen.
 - 3.5 Entdecken von zu vereinfachenden Mustern.

- 4 Daten visualisieren.
- 4.1 Erstellen verschiedener Datentypen für die Verwendung mit dem Computer.
- 5 Problemzerlegung.
- 5.1 Aufteilen einer einfachen Aufgabe in Teilaufgaben.
- 5.6 Kontrollieren, ob bei der Ausführung von Teilaufträgen kein wichtiger Teil ausgelassen oder vergessen wurde.
- 6 Automatisierung.
- 6.1 Erkennen von Wiederholungen von Aufgaben in verschiedenen Situationen.
- 7 Algorithmen und Methoden.
- 7.6 Erfahren, dass ein Algorithmus eine Liste von Anweisungen ist, die zu einem bestimmten Ergebnis führt.
- 7.8 Erstellen eines einfachen Algorithmus in einer konkreten Situation mit einem festen Satz vordefinierter Anweisungen.
- 7.13 Beschreiben einer Wiederholungsschleife mit einer festen Anzahl von Wiederholungen.

Erläutern

Es ist möglich für den Schüler dieser Lektion selbstständig zu folgen. Trotzdem empfehlen wir, Teil 1 dieser Lektion in der Klasse durchzunehmen und die Wortliste zu besprechen. Um Teil 1 in der Klasse zu besprechen, können Sie folgenden Punkten folgen:

- 1 Um Ihre Schüler mit den Ton-Blöcken (und möglichen anderen Blöcken) vertraut zu machen, können Sie den Schülern Zeit geben, die verschiedenen Ton-Blöcke anzuschauen. Die Ton-Blöcke heißen in Ozoblockly 'Sounds' und haben die Farbe Violett.

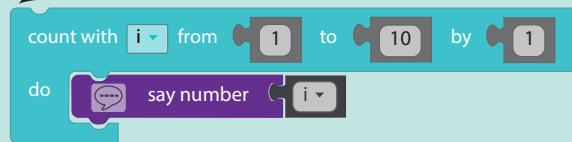


- Ermutigen Sie Ihre Schüler, verschiedene Ton-Blöcke aus verschiedenen Leveln (Level 1 bis 4) in einer Schleife zu verwenden, um sie in Ozobot Evo zu laden und zu beobachten, was passiert.
- Erklären Sie, dass zum Beispiel der "Emotions-Ton-Block" nur in Level 2 gefunden werden kann, aber mit Blöcken aus anderen Leveln verwendet werden kann.
- Wenn die Schüler nicht genau wissen oder verstehen, was einige Blöcke tun können, machen Sie deutlich, dass sie die Erklärung aller Blöcke in Ozoblockly finden können. Um zur Erklärung der Blöcke zu gelangen, klicken sie auf die zweite Schaltfläche des Menüs auf der rechten Seite. Wenn sie nun auf einen Block in Ozoblockly klicken, erscheint die Erklärung auf der rechten Seite.

2 Besprechen Sie nun die Wortliste mit Ihrer Klasse.

- Variable: Eine Variable ist der allgemeine Name für eine Einheit. Diese Einheit kann immer einen anderen Wert haben. In der Code-Sprache verwendest du eine Variable, um Berechnungen durchzuführen. Manchmal brauchst du zum Beispiel einen bestimmten Wert mehr als einmal in deinem Code. In diesem Fall kannst du den Wert in einer Variablen speichern und dann einfach den Namen der Variablen verwenden. In einer komplexen Berechnung kannst du Zwischenschritte mit Hilfe von Variablen aufteilen, von denen du jedes Zwischenergebnis in einer separaten Variablen speichern kannst. Eigentlich ist eine Variable eine Art Gedächtnis, in dem Angaben gespeichert werden können.
- Iterator: Erklären Sie, dass dies eine Variable ist. Der Iterator macht es möglich zu speichern, wie oft eine Schleife ausgeführt wurde. In diesem Auftrag ist dies der 'count with i' Block, den Ihre Schüler gleich sehen sollen. Das 'i' darin ist die Variable, die die Anzahl der Male speichert, die der Code in der Schleife ausgeführt wurde.
- Counter: Im Deutschen „der Zähler“. Diese Variable speichert die Anzahl der Elemente, die eine bestimmte Anforderung erfüllen. Wenn du zum Beispiel eine Liste von Ziffern hast und wissen willst, wie oft die Ziffer 9 in der Liste steht, dann kannst du eine Counter Variable anlegen, die speichert, wie oft die 9 enthalten ist.

3 Stellen Sie den Auftrag dann Ihrer Klasse vor. Fragen Sie Ihre Schüler, wie sie Ozobot kodieren können, so dass er laut von 1 bis 10 zählt. Es gibt verschiedene richtige Antworten, aber machen Sie deutlich, dass die richtige Antwort nur 2 Blöcke enthält. Wenn Ihre Schüler nicht darauf kommen, dann zeigen Sie ihnen den folgenden Code:

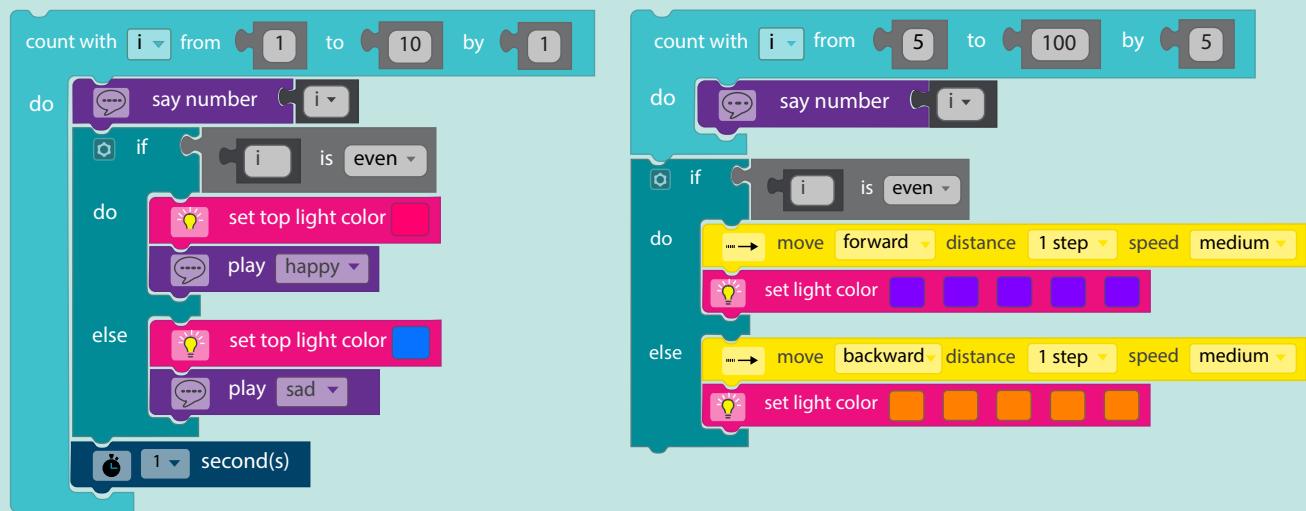


4 Sprechen Sie in der Klasse darüber, was der Code macht, Schritt für Schritt. Erklären Sie, dass der Block die Variable 'i' verwendet, um von 1 bis 10 zu zählen, wodurch Ozobot immer die richtige Zahl sagt. Erklären Sie, wo jede Ziffer im Code steht. Die ersten beiden Felder repräsentieren die Zahlen, von und bis zu der Ozobot zählt. In diesem Fall also von 1 bis 10. Das letzte Feld gibt an, mit welchem Schritt Ozobot von 1 bis 10 zählt. In diesem Fall 1. Wenn es 2 sein soll, dann soll Ozobot wie folgt zählen: 1, 3, 5, ...

5 Lassen Sie Ihre Schüler nun die Aufgabe auf ihrem Arbeitsblatt machen. Machen Sie dabei deutlich, dass sie die 'count with i' Schleife, wie sie sie gerade gelernt haben, in ihrem Code verwenden müssen.

Antworten

Es sind mehrere Antworten möglich, im Folgenden finden Sie für jeden Auftrag eine der möglichen Lösungen:



Bis 10 zählen

Lektion 3
Ozobot Evo

Was lernst du?

In dieser Lektion lernst du, was eine Variable, ein Iterator und ein Counter ist. Wir lernen, wie wir Ton, Ziffern und Lämpchen zusammen arbeiten lassen können, sodass Evo von 1 bis 10 zählt.

Was werden wir machen?

Wir werden Evo so programmieren, dass er von 1 bis 10 zählt, wobei er jedes Mal ein anderes Lämpchen leuchten und eine andere Emotion hören lässt.

Was brauchst du?

- Ozobot Evo
- Tablet, Laptop oder Computer
- Arbeitskarte 3

Zeit:

30-40 Minuten

TEIL I

Wortschatz:

Variable: Vielleicht weißt du schon, was das bedeutet. Eine Variable ist der allgemeine Name für eine Einheit. Diese Einheit kann immer einen anderen Wert haben. In der Code-Sprache verwendest du eine Variable, um Berechnungen durchzuführen. Eigentlich ist eine Variable eine Art Gedächtnis, in dem Angaben gespeichert werden können. Das klingt sehr kompliziert, aber es ist ganz einfach. Schau mal weiter und du kommst dahinter!

Iterator: Ein Iterator ist eine Art von Variable. Diese Variable macht es möglich zu speichern, wie oft eine Schleife durchlaufen wurde. In diesem Auftrag ist das zum Beispiel der 'count with i' Block. Das 'i' darin ist die Variable, die die Anzahl der Male speichert, die der Code in der Schleife ausgeführt wurde.

Counter: Auf Deutsch bedeutet das 'der Zähler'. Diese Variable speichert die Anzahl der Elemente, die eine bestimmte Anforderung erfüllen. Wenn du zum Beispiel eine Liste von Ziffern hast und wissen willst, wie oft die Ziffer 9 in der Liste steht, dann kannst du eine Counter Variable anlegen, die speichert, wie oft die 9 enthalten ist.

Der Auftrag Teil 1:

Kodiere Ozobot Evo so, dass er laut von 1 bis 10 zählt.

- Tipp: Der richtige Code enthält nur zwei Blöcke.
- Noch ein Tipp: Der Geräusch-Block, der verwendet werden muss, befindet sich in Level 2 von Ozoblockly Evo.
- Hat es geklappt? Oder kommst du nicht drauf? Bitte deinen Lehrer oder deine Lehrerin um Hilfe, wenn du nicht drauf kommst.

Teil 2

Der Auftrag Teil 2:

Kodiere Ozobot Evo so, dass er laut von 1 bis 10 zählt.

- Verwende die 'count with i' Schleife von 1 bis 10 bei 1. Verstehst du nicht, was jede Ziffer im Code-Block bedeutet? Schaue dir dann nochmal genau die Antwort von Teil 1 auf deinem Arbeitsblatt an!
- Lass' Ozobot laut die Ziffer (i) sagen.
- Wenn die Ziffer (i) gerade ist, kodiere Ozobot so, dass er ein fröhliches Geräusch abgibt und ein rotes Lämpchen angeht
- Wenn die Ziffer (i) ungerade ist, kodiere Ozobot so, dass er ein betrübtes Geräusch abgibt und ein blaues Lämpchen angeht.
- Programmiere Ozobot so, dass er eine Sekunde wartet, bevor er die folgende Ziffer ansagt.

TIPP: Wenn du nicht genau weißt oder verstehst, was manche Blöcke können, dann kannst du das in Ozoblockly nachschauen. Um zur Erklärung der Code-Blöcke zu gelangen, klicke auf die zweite Schaltfläche des Menüs auf der rechten Seite. Wenn du jetzt auf einen Code-Block in Ozoblockly klickst, erscheint die Erklärung auf der rechten Seite. Achtung: Der erklärende Text ist auf Englisch. Du kannst einen Übersetzer verwenden, wenn du nicht zurechtkommst.

Har det fungerat? Ladda upp ditt program och se om Ozobot följer uppdraget! Tror du att du har rätt kod? Fråga din lärare eller lärare för svaret! Eller, om du säkert vet: Fortsätt till del 3!

Teil 3

Der Auftrag Teil 3:

Kodiere Ozobot Evo so, dass er laut von 5 bis 100 in 5er-Schritten zählt. Also: 5, 10, 15...

- Verwende die gleiche 'count with i' Schleife, die wir im vorherigen Auftrag verwendet haben.
- Lasse Ozobot laut die Ziffer (i) sagen.
- Wenn die Ziffer (i) gerade ist, dann lasse Ozobot nach vorne fahren, mit violetten Lämpchen an der Vorderseite
- Wenn die Ziffer (i) ungerade ist, dann lasse Ozobot rückwärtsfahren, mit orangen Lämpchen an der Vorderseite.

Hat es geklappt? Lade das Programm hoch und schaue, ob Ozobot dem Auftrag folgt. Jetzt weißt du, wie du eine Variable in Ozobots Codes verwenden kannst!

Arbeitskarte

Lektion 3

