# Was ist *VEXcode VR*?

Bei uns im dEIn-Labor an der TU Berlin kannst du lernen, Roboter zu bauen und zu programmieren. Vielleicht kennst du *Lego-mindstorms*-Roboter, die baut man hauptsächlich aus Legosteinen. Dazu kommen Sensoren, damit die Roboter sich orientieren können, und ein programmierbarer Baustein, auf dem das Programm läuft.

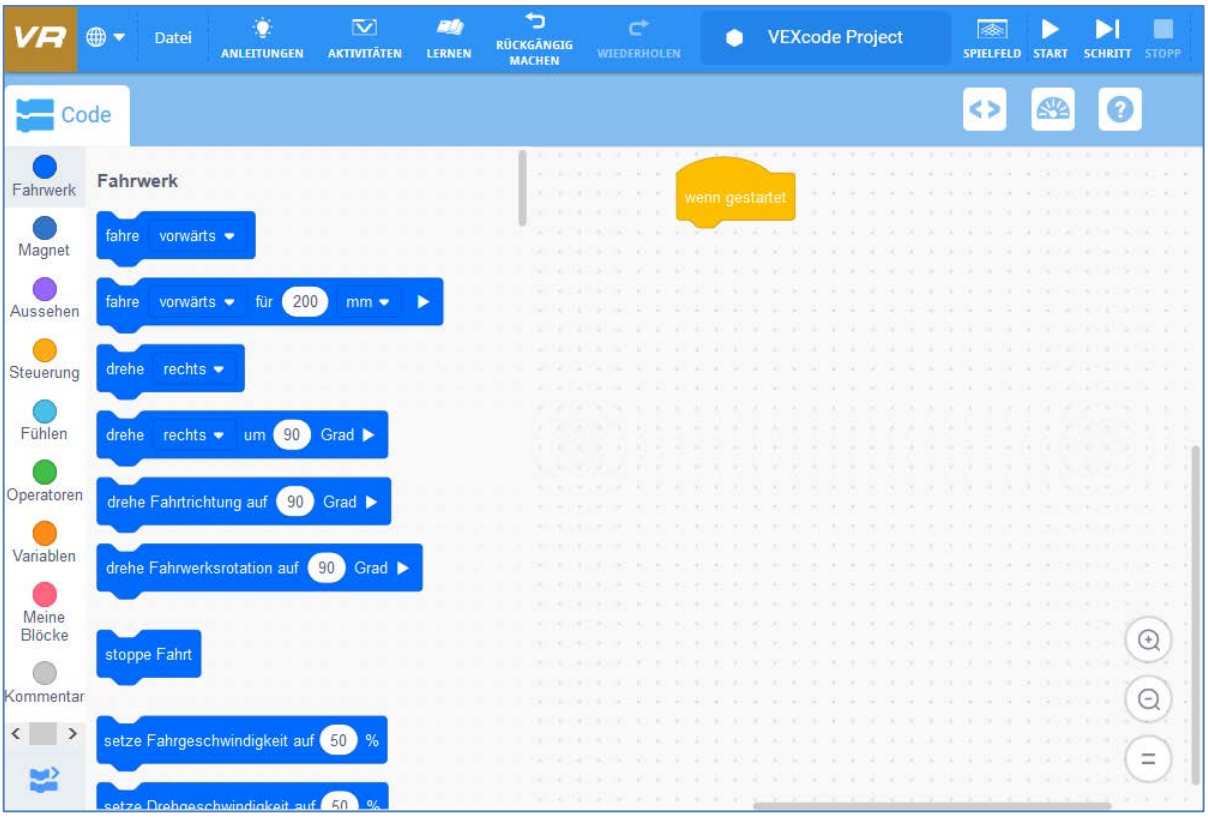
Die Roboter, die wir im dEIn-Labor haben, sind so ähnlich wie die Legoroboter. Aber sie sind von einer anderen Firma namens VEX. Man programmiert sie in der Programmiersprache *VEXcode*.

Nun hat aber nicht jeder einen Roboter zu Hause, die sind ja auch ziemlich teuer. Darum haben sich schlaue Programmierer einfallen lassen, einen *virtuellen Roboter* zu programmieren. Den gibt es nicht wirklich, sondern nur virtuell, also vom Computer so echt wie möglich am Bildschirm nachgeahmt. Das ist auch wegen Corona praktisch.

Der virtuelle Roboter hat vier Räder und ein paar Sensoren. Du kannst ihn also nicht selber bauen, aber du kannst immerhin am Bildschirm sehen, wie er sich bewegt und dein Programm ausführt. Das hilft natürlich sehr, wenn du wissen willst, ob dein Programm korrekt ist und wenn du Fehler in deinem Programm aufspüren möchtest.

## 1. Programmieren im Web-Browser

Die Umgebung zum Programmieren von virtuellen VEX-Robotern heißt *VEX code VR* und läuft im Browser. Die Webadresse ist https://vr.vex.com/ .



Du siehst links die Blöcke für den Programmcode. Im Bild sieht man gerade die blauen Blöcke aus der Abteilung *Fahrwerk*. Das sind alles Blöcke, die für das Fahren des Roboters benutzt werden können. Wenn du Scratch kennst, kommt dir diese Art Blöcke sicher bekannt vor.

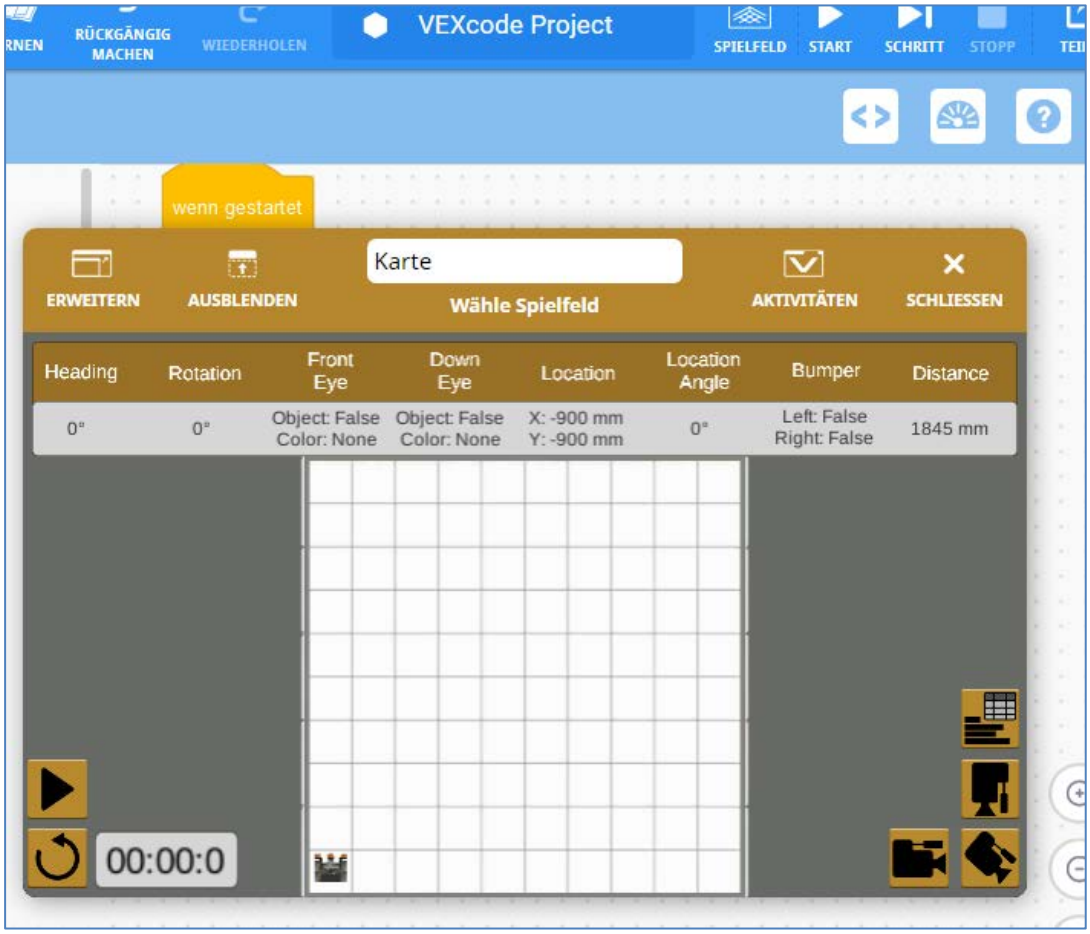
Rechts ist die leere Fläche, auf der dein Programm entsteht. Nein, sie ist gar nicht ganz leer: der gelbe *wenn gestartet*-Block ist schon da. An diesen werden alle Programmblöcke unten drangepuzzelt, die ablaufen sollen, wenn du dein Programm startest.

Ganz links sind verschiedene Code-Abteilungen. Wenn du eine Abteilung durch Klicken auf den entsprechenden Kreis öffnest, dann erscheinen daneben die Blöcke dieser Abteilung. Außer den Fahr-Blöcken gibt es Abteilungen mit Blöcken für das Benutzen von Sensoren (*Fühlen*), für Berechnungen (*Operatoren*), zum Steuern, wie oft etwas geschehen soll (*Steuerung*) und zum Aufheben von Sachen mithilfe eines Magneten. Du wirst sie alle beim Programmieren nach und nach kennen lernen.

Zum Programmieren ziehst du mit der Maus Blöcke aus einer Code-Abteilung auf die rechte Fläche und klickst sie in der gewünschten Reihenfolge aneinander.

## 2. Der virtuelle Roboter und sein Spielfeld

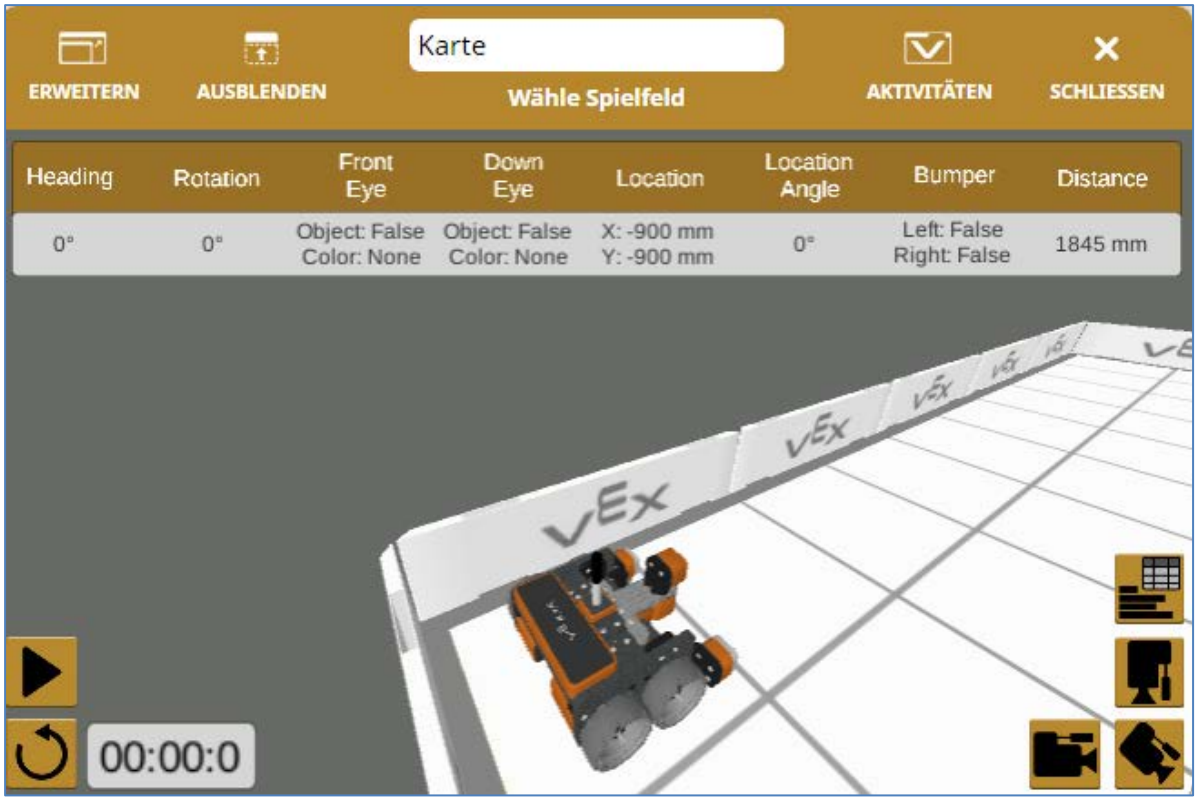
Aber wo ist denn nun der virtuelle Roboter zu sehen? Klicke oben in der blauen Kopfleiste auf *Spielfeld*. Ein neues, braunes Fenster schiebt sich in den Vordergrund:



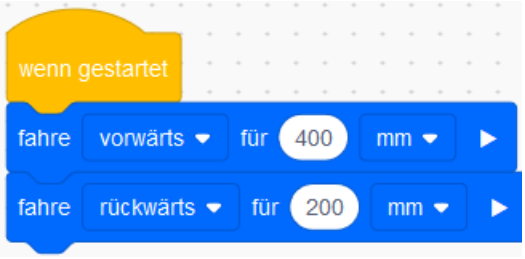
Im weißen Textfeld oben in der Mitte über *Wähle Spielfeld* steht „Karte“. Das ist der Name der aktuellen Spielfläche, also in diesem Fall ein Gitterfeld aus weißen Quadraten. Hier könntest du auch ein anderes Spielfeld wählen. Es gibt ein paar Felder, wo schon Spielelemente aufgebaut sind, mit denen du dann Aufgaben lösen sollst.

Im Quadrat ganz unten links steht der Roboter. Er ist nur schwer zu sehen, weil die Kamera ihn von ganz weit weg von oben zeigt, so dass du auch das komplette Spielfeld sehen kannst. Aber die Kameraperspektive kannst du ändern, wenn du auf die schräg gedrehte Videokamera ganz rechts unten klickst. Dann siehst du den Roboter größer und in 3D. Mit der Maus kannst du das Spielfeld dann auch drehen, so dass du den Roboter von allen Seiten betrachten kannst.

Oben links kannst du mit den Buttons *ERWEITERN / VERKLEINERN* und *AUSBLENDEN / ANZEIGEN* das Spielfeld mit dem Roboter anpassen.




Lass deinen Roboter mal zur Probe zwei Kästchen vorwärts und dann wieder ein Kästchen rückwärts fahren. Ein Kästchen ist quadratisch und 200 mm lang. Du musst zwei *fahre*-Blöcke an den *wenn-gestartet*-Block anhängen, erst einen zum Vorwärts-Fahren, dann einen zum Rückwärts-Fahren.




Mit einem Klick auf den Play-Button (das schwarze Dreieck links auf dem Spielfeld) schickst du dein Programm an den Roboter, und der führt es aus. Du siehst, wie dein virtueller Roboter hin und her fährt.

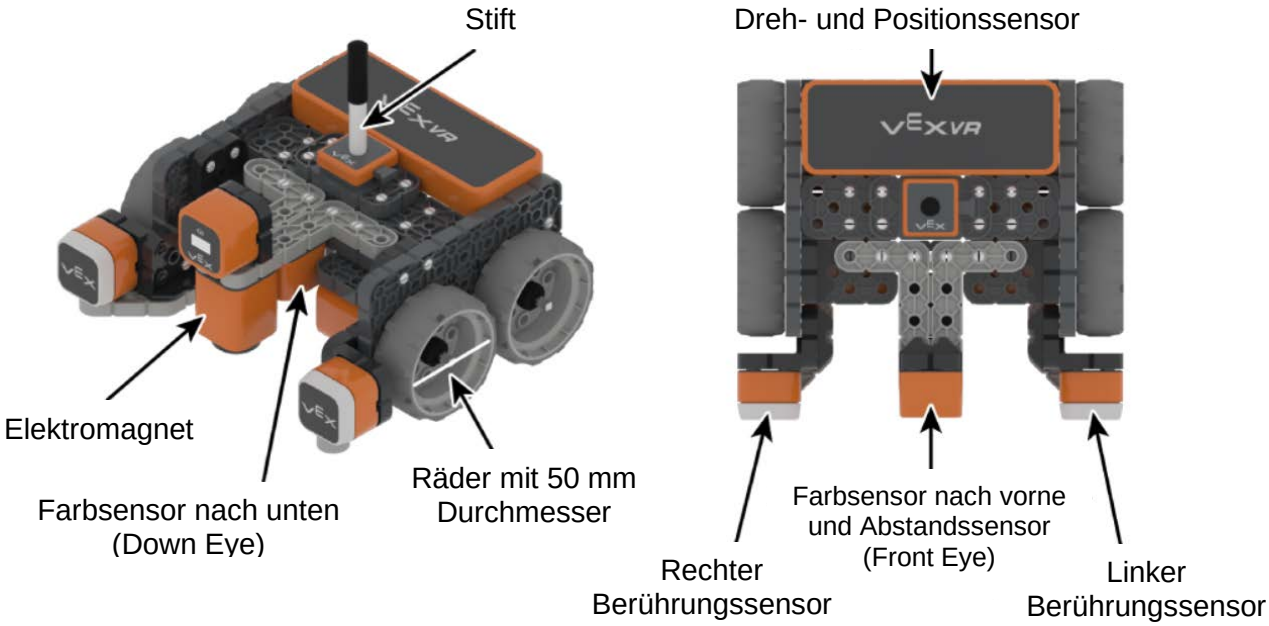
Ein Klick auf den Reset-Button (den runden Pfeil) stellt die Ausgangssituation wieder her.

# 3. Sensoren und Aktoren

Die Zeile unter *Wähle Spielfeld* zeigt die Werte über die aktuelle Robotersituation an, die von den Sensoren gemessen werden. In welche Richtung fährt er (*Heading*), um wieviel Grad dreht er sich gerade (*Rotation*), welche Farbe sehen die Farbsensoren, die nach vorne und nach unten blicken (*Front Eye* und *Down Eye*), wo befindet sich der Roboter und wie steht er (*Location*, *Location Angle*), berührt er einen Gegenstand (*Left Bumper*, *Right Bumper*), und in welcher Entfernung (*Distance*) ist das nächste Hindernis?

Außer den *Sensoren* zum Messen (zwei Farbsensoren, ein Abstandssensor, zwei Berührungssensoren (Bumper) und ein Drehsensor (Gyro) hat der Roboter natürlich seine Räder zum Fahren, einen Elektromagneten, den man an- und abschalten kann und einen Stift zum Malen auf der Spielfläche, den du heben und senken kannst. Wenn er unten ist, berührt er den Boden und hinterlässt beim Fahren eine Spur. So kannst du malen. Wenn du nicht malen willst, lass den Stift oben. Räder, Stiftmotor und Magnet heißen *Aktoren*.

So sind die Sensoren und Aktoren am Roboter verbaut:




Nun bist du bereit, dein erstes Spiel anzugehen. Die Anleitungen erklären dir jeweils ein Spielfeld, das Ziel des Spiels und helfen dir mit Programmiertipps. Keine Angst, einiges bleibt für dich auch noch zu tun.

Viel Spaß!