Goals and Standards

Implementing VEX GO STEM Labs

STEM Labs are designed to be the online teacher's manual for VEX GO. Like a printed teacher's manual, the teacher-facing content of the STEM Labs provides all of the resources, materials, and information needed to be able to plan, teach, and assess with VEX GO. The Lab Image Slideshows are the student-facing companion to this material. For more detailed information about how to implement a STEM Lab in your classroom, see the Implementing VEX GO STEM Labs article.

Goals

Students will apply

- How to create and start a VEXcode GO project that makes the Code Base move forward and backward.

Students will make meaning of

- How to solve a problem with the Code Base robot and VEXcode GO.
- How robots can do jobs that are dirty, dull or dangerous; such as unsanitary work cleaning sewers, dull work in warehouses, or dangerous work fighting fires.

Students will be skilled at

- Coding the Code Base robot to drive forward.
- Coding the Code Base robot to drive backward.
- Creating a VEXcode GO project to make the Code Base Robot move forward and backward.
- Explaining where the Drivetrain is on the Code Base robot.

Students will know

- How to create and start a project using VEXcode GO and the Code Base robot.
- How to create a VEXcode GO project that correctly orders behaviors in a sequence in order to move the Code Base robot forward and backward. This can be done both individually and collaboratively.

Objective(s)

Objective

1. Students will create and start a project that has the Code Base robot move forward.
2. Students will create and start a project that has the Code Base robot move in reverse.
3. Students will identify the position, orientation, and the location of the Code Base robot as it moves.
4. Students will identify where the drivetrain is on the Code Base robot.

Activity

1. In Play Part 1, students will create and start a project that has the Code Base robot move forward.
2. In Play Part 2, students will create and start a project that has the Code Base robot drive forward and in reverse.
3. In Play Part 1 and 2, students will be asked to place markers where the Code Base robot should end up after each project is started.
4. In the Mid-Play break the teacher will explain to students why there is a category of drivetrain blocks and where the drivetrain is on the Code Base robot.

Assessment

1. In Play Part 1, student projects will successfully drive the Code Base robot forward for a specified distance.
2. In Play Part 2, student projects will successfully drive the Code Base robot in reverse for a specified distance.
3. Students will compare their predictions versus the actual location that the Code Base robot ended up during Mid-Play Break and class discussions.
4. During the Share section, students will be able to identify where the drivetrain is on the Code Base robot using gestures.

Connections to Standards

Showcase Standards

Common Core State Standards (CCSS)

CCSS.MATH.CONTENT.K.G.A.1: Describe objects in the environment using names of shapes, and describe the relative positions of these objects using terms such as above, below, beside, in front of, behind, and next to.

How Standard is Achieved: In Play Parts 1 and 2, students predict how far the Code Base robot will move and how accurate their predictions are. As a result, they will need to describe the position of the Code Base robot relative to their prediction. Additionally, the students will be asked by the teacher how changing the orientation of the Code Base robot will affect where it ends up.

Showcase Standards

Computer Science Teachers Association (CSTA)

CSTA 1A-AP-10: Develop programs with sequences and simple loops, to express ideas or address a problem.

How Standard is Achieved: In Play Part 2, students will create and start a project where Drivetrain blocks are sequenced together to allow the Code Base robot to move forward and backward.

Showcase Standards

Computer Science Teachers Association (CSTA)

CSTA 1B-AP-11: Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

How Standard is Achieved: During the entire Lab, students will decompose the problem of how a robot should move in order to complete a job that is either dirty, dull, or dangerous. During the Play sections, students will break this problem down further by analyzing and programming their Code Base robot to drive forward and reverse a certain distance.

Summary

Materials Needed

The following is a list of all the materials that are needed to complete the VEX GO Lab. These materials include student facing materials as well as teacher facilitation materials. It is recommended that you assign two students to each VEX GO Kit.

In some Labs, links to teaching resources in a slideshow format have been included. These slides can help provide context and inspiration for your students. Teachers will be guided in how to implement the slides with suggestions throughout the lab. All slides are editable, and can be projected for students or used as a teacher resource. To edit the Google Slides, make a copy into your personal Drive and edit as needed.

Other editable documents have been included to assist in implementing the Labs in a small group format. Print the worksheets as is or copy and edit those documents to suit the needs of your classroom. Example Data Collection sheet setups have been included for certain experiments as well as the original blank copy. While they offer suggestions for setup, these documents are all editable to best suit your classroom and the needs of your students.

Materials	Purpose	Recommendation
VEX GO Kit	For students to build their Code Base 2.0.	1 per group
Code Base 2.0 Build Instructions (3D) or Code Base 2.0 Build Instructions (PDF)	For students to build the Code Base 2.0 if they have not already.	1 per group
Pre-Built Code Base 2.0	For students to start projects in Lab activities.	1 per group
VEXcode GO	For students to create and start projects on the Code Base robot.	1 per group
Robotics Roles & Routines Google Doc / .docx / .pdf	Editable Google Doc for organizing group work and best practices for using the VEX GO Kit. For students to build the Code Base if they have not already.	1 per group
Tablet or Computer	For the students to launch VEXcode GO.	1 per group
Lab 2 Image Slideshow Google Doc / .pptx / .pdf	For teachers and students to reference throughout the Lab.	1 for teacher facilitation
Pencils	For students to fill out the Robotics Roles & Routines Worksheet.	1 per group
Placement markers	For students to visually predict where the Code Base robot will end up after it has completed its movement.	At least one per group
Pin Tool	To help remove pins or pry beams apart.	1 per group
Get Ready...Get VEX...GO! PDF Book (optional)	To read with students to introduce them to VEX GO through a story and introductory build.	1 for demonstration purposes
Get Ready...Get VEX...GO! Teacher's Guide Google Doc / .pptx / .pdf	For additional prompts when introducing students to VEX GO with the PDF Book.	1 for teacher use

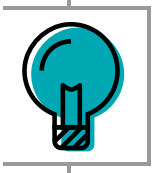
Engage

Begin the lab by engaging with the students.

Hook

Ask students to describe how to get to a certain landmark in the school building.

Note: If students are new to VEX GO, use the Get Ready...Get VEX...GO! PDF book and Teacher's Guide (Google Doc/.pptx/.pdf) to introduce them to learning and building with VEX GO. Add an additional 10-15 minutes to your lesson time to accommodate this additional activity.



Leading Question

If someone was new to school and didn't know how to get to the principal's office, what directions would we provide? Why is it important to give specific instructions? How do we give instructions to the Code Base robot?

Build

Code Base 2.0

Play

Allow students to explore the concepts introduced.

Part 1

Students will create and start a project that moves the Code Base robot forward for a specified distance. Before they start the project, they'll predict where the Code Base robot will end up using placement markers. Students will then start the project and observe the Code Base robot's movement. Students will then edit their project to change the distance to see how this affects the Code Base robot's movement.

Mid-Play Break

Discuss the Code Base robot's movement from Play Part 1. Ask the following questions, "did Code Base robot end up where you thought it was going to? How close?" Then, discuss what a drivetrain is, and where to find it on the Code Base robot.

Part 2

Students will create and start a project that moves the Code Base robot in reverse for a specified distance. Before they start the project, they'll predict where the Code Base robot will end up using placement markers. Students will then start the project and observe the Code Base robot's movement. Students will then edit their project to change the distance to see how this affects the Code Base robot's movement. Students will combine forward and backward movements.

Share

Allow students to discuss and display their learning.

Discussion Prompts

- How did you decide where the Code Base robot would be after the project was started?
- How do you change how far the Code Base robot moves?
- If you changed the direction that the Code Base robot was facing, would it change your prediction? Why?

Engage

Launch the Engage Section

ACTS is what the teacher will do and ASKS is how the teacher will facilitate.

ACTS	ASKS
1. Facilitate a discussion that introduces the concept of directions and why they are important. Ask students to describe how to get to a certain landmark in the school building. 2. As the students give directions, write them at the front of the class. 3. Write a copy of the directions next to the initial directions except mixing up a few directions. 4. Connect the importance of giving directions to the “new student” correctly to the importance of giving explicit, sequenced, and correct directions to robots. Then, show students a pre-built Code Base robot.	1. If someone was new to school and didn’t know how to get to the principal’s office, what directions would we provide? Why is it important to give specific instructions? 2. What directions could we give the student? 3. Why is it important to give specific instructions? Would the student be able to get to the location? 4. Now that we understand how to give directions to a new student, how do we give instructions to the Code Base robot?

Getting the Students Ready to Build

Let’s learn how to give our Code Base instructions to make it move!

Facilitate the Build

1

Instruct
Instruct students to join their groups and complete the Robotics Roles & Routines sheet. Use the Suggested Role Responsibilities slide in the Lab Image Slideshow as a guide for students to complete this sheet.

- Instruct students to check all their materials to prepare for the Lab challenges. They need to make sure that they have the necessary materials, and everything is charged and the Code Base is built and connected correctly. Give a thumbs up to the teacher when their group is ready to go!
- The Code Base will need to be built if it is not already. Model for students the steps in the Connect a VEX GO Brain VEX Library article for your device, to guide students through the connection process.

Note: When you first connect your Code Base to your device, the Gyro built into the Brain may calibrate, causing the Code Base to move on its own for a moment. **This is an expected behavior, do not touch the Code Base while it's calibrating.**

2

Distribute

Distribute a pre-built Code Base 2.0 and a device to launch and use VEXcode GO to each group. Or, distribute build instructions and ask students to build the Code Base if it is not yet built.

Code Base 2.0

3

Facilitate

Facilitate preparing the groups for the Play sections by walking them through the steps to check their materials.

- Is the battery charged?
- Is the Code Base built properly and not missing any pieces?
- Are all cables connected to the correct ports?
- Launch VEXcode GO on your device. Is your Code Base connected to your device?

Offer

4 Offer support to groups that need assistance in launching VEXcode GO or preparing their Code Bases.

Teacher Troubleshooting

- Ensure laptops, tablets, and VEX GO Batteries are charged before beginning the Lab.
- Remind students where the ports are for the motors. Looking at the Brain with the VEX logo oriented at the bottom, the students should plug the left motor into Port 4 and the right motor into Port 1. Ensure that the cables do not cross underneath the robot. Use the Lab 2 Image Slideshow to show where the ports are located.
- For more information about the VEX GO Brain see the Using the VEX GO Brain article in the VEX Library.

Facilitation Strategies

- Establish a consistent "start up" practice as routine before working with VEX GO. If it is consistently implemented, students will take ownership of this routine and it will foster good practices for independent robotics activities.
- Offer in the moment observation as teams work well, and invite them to share teamwork strategies with the class.
- Use the Get Ready...Get VEX...GO! PDF Book and Teacher's Guide - If students are new to VEX GO, read the PDF book and use the prompts in the Teacher's Guide (Google Doc/.pptx/.pdf) to facilitate an introduction to building and using VEX GO before beginning the Lab activities. Students can join their groups and gather their VEX GO Kits, and follow along with the building activity within the book as you read.
    - Use the Teacher's Guide to facilitate student engagement. To focus on VEX GO connections in a more concrete or tangible way, use the Share, Show, or Find prompts on each page to give students an opportunity to get to know their kits in more depth.
    - To focus on the habits of mind that support building and learning with VEX GO, like persistence, patience, and teamwork, use the Think prompts on each page to engage students in conversations about mindset and strategies to support successful group work and creative thinking.
    - To learn more about using the PDF book and accompanying Teacher's Guide as a teaching tool any time you are using VEX GO in your classroom, see this VEX Library article.

Play

Part 1 - Step by Step

1

Instruct

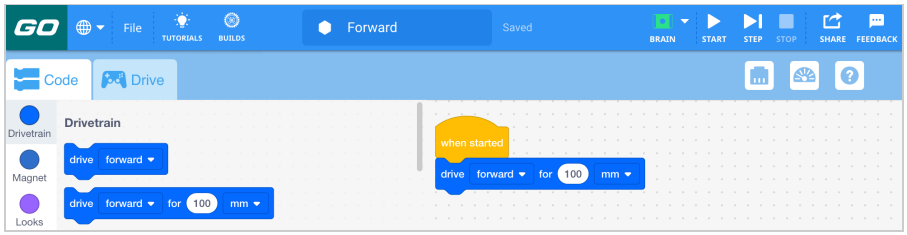
Instruct students that they will explore how to move their Code Base robot forward! Before they start the project, they'll predict where the Code Base robot will end up. Watch the animation below to see examples of the Code Base moving forward for different distances. In the animation, the Code Base starts in the lower left corner of the Tile and first drives forward 150mm and stops. It then appears back at the starting location, and drives forward 75mm and stops.

2

Model

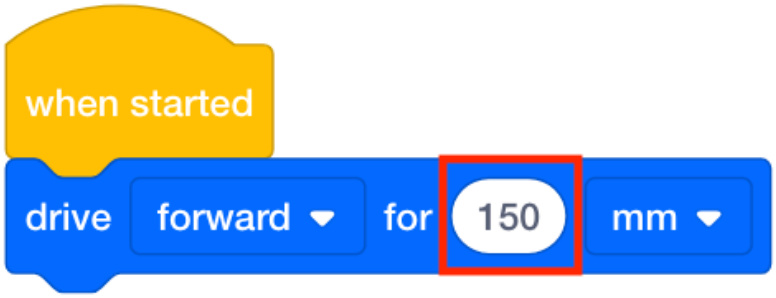
Model how to launch VEXcode GO on a device and create a project that moves the Code Base forward with the [Drive for] block.

- Model for students the steps of the Open and Save a Project VEX Library article and have them follow the steps to open and save their project.
- Instruct students to name their project *Forward*.
- Have the students then connect the Brain of their Code Base robot to their device.
- Once students have named their project and connected the Brain to their device, they need to follow the steps to configure for the Code Base robot. Model the steps from the Configure a Code Base VEX Library article and ensure students can see the Drivetrain blocks in the Toolbox.
- Show how to drag in the [Drive for] block into the Workspace and place it under the {When started} block.

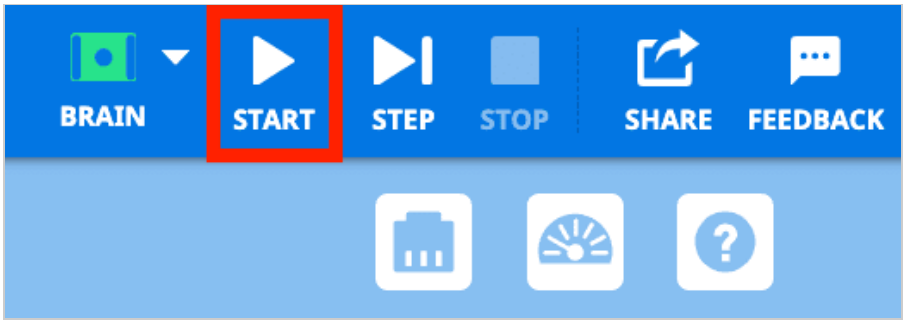


Add the [Drive for] block

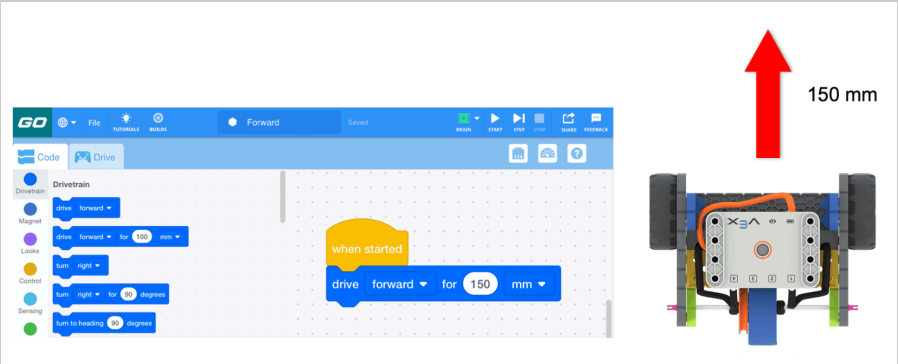
- Change the parameter of the [Drive for] block to 150mm.



Change the parameter

- Model for students how to predict how far the Code Base robot will move based on the parameters in the [Drive for] block. Have students place the Code Base in the starting position, then estimate how far the robot will move. They should place a marker where they think the Code Base will stop.
- Model for students how to select the 'Start' button in the Toolbar to start the project.
  - 

- Once the students have observed the behavior, model for students how to go back to their project, edit the parameters of the [Drive for] block from 150mm to another distance, such as 200mm or 250mm. Then, start the project again to see how the change in parameters affected the movement of the Code Base robot.



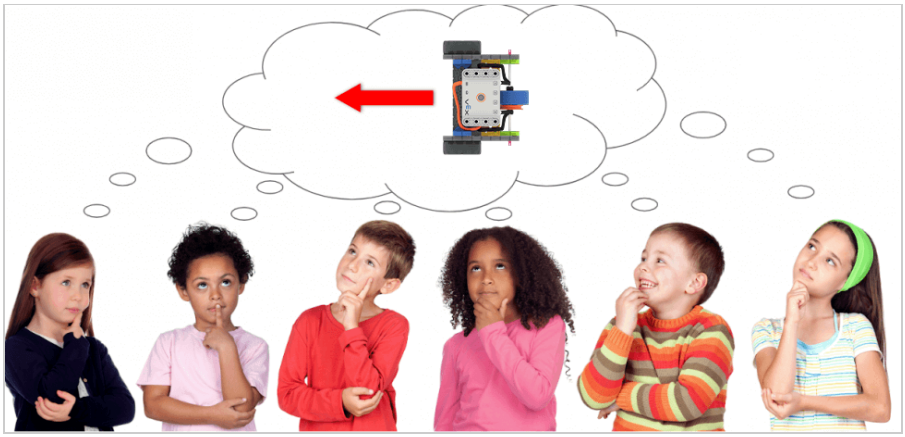
Forward 150 mm

3

Facilitate

Facilitate a discussion around student observations and the goals of the project by asking the following:

- Can you show me using your hands how far you thought the Code Base robot would move before you started the project?
- What did you change the distance parameter to and why? How far do you think the Code Base robot will travel now that the distance has been changed?
- How did the distance travelled compare to your estimation?
- What category of blocks did you use for this project?



Discuss the Code Base Robot's Movement

4

Remind

Remind students that they may have questions when they are creating and starting their project. Remind students that learning new concepts may take multiple tries and encourage them to try again if they are unsuccessful on the first try.

5

Ask

Ask students to think about how far the Code Base robot would need to move to travel across the classroom. Have students make a connection to why this type of planning is useful in everyday life. Ask students how being able to plan and give accurate directions could be useful for a job? Ask the students if they can think of any jobs where directions are needed?

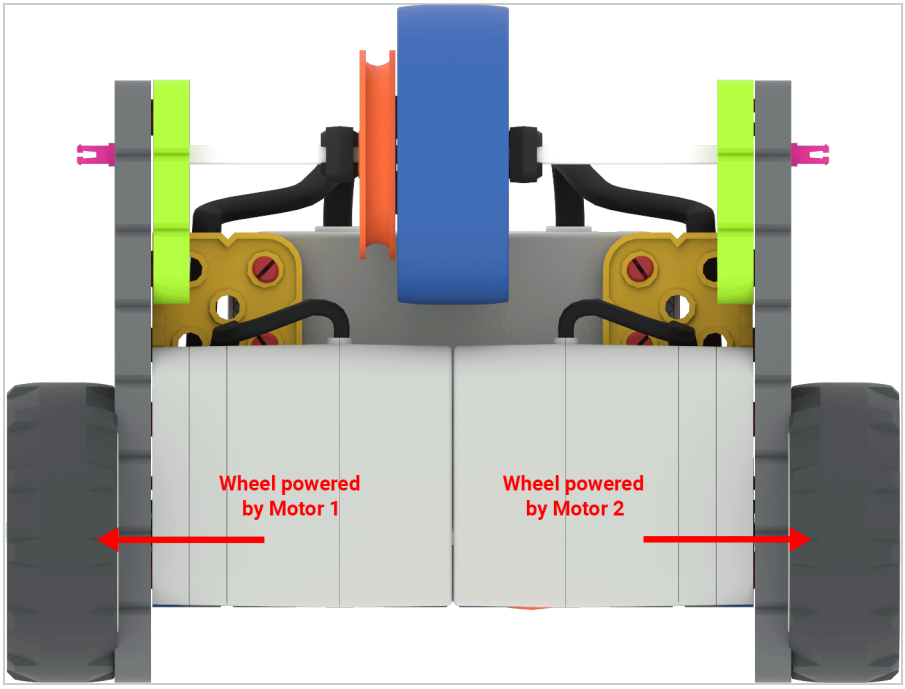
Mid-Play Break & Group Discussion

As soon as every group has completed their project, come together for a brief conversation.

- Did the Code Base robot end up where you thought it was going to? If not, how close was it to your prediction?
- How did you edit your project? What new distance did you choose?
- Did you find any difficulty when changing the distance in the [Drive for] block?

Introduce Drivetrain:

- Now that we explored how to use VEXcode GO to allow our Code Base robot to drive forward, why do you think there is a “Drivetrain” section of blocks?
- What do you think a drivetrain is? Can you explain your thinking?
- Can you show me using gestures where you think the drivetrain is on the Code Base robot?
- Can you look at the bottom of your Code Base robot and identify where the motors are in this drivetrain, and which wheels they are attached to?



Code Base Robot Drivetrain

Part 2 - Step by Step

1

Instruct

Instruct students that they will explore how to move their Code Base robot forward and in reverse!

To begin, each group should have a device, VEXcode GO, at least one placement marker, and a built Code Base. Watch the animation below to see how the Code Base moves in reverse. In the animation, the Code Base starts in the upper left corner of the tile, and drives in reverse 150mm, then stops. It then returns to the starting position and drives reverse for 75 mm.

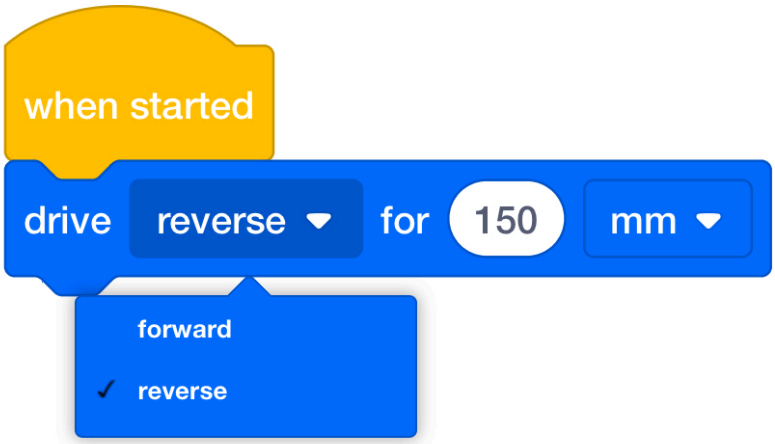
2

Model

Model for students how to launch VEXcode GO on a device and rename their project as Reverse. Show students to select ‘Save As’ to save this project separately from their first.

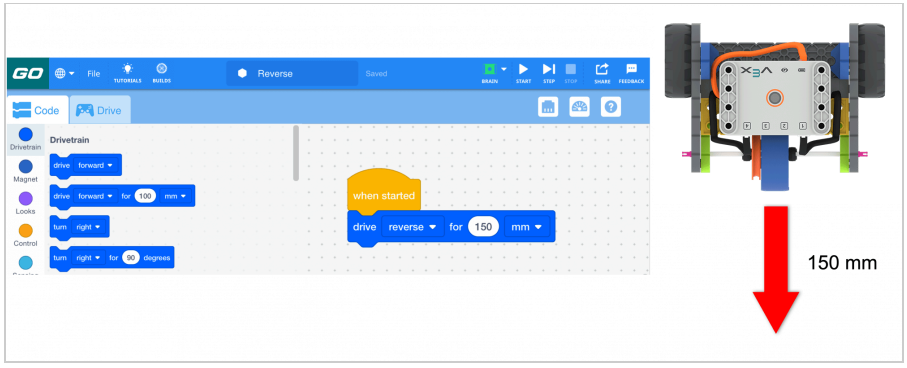
Refer to the steps in the Open and Save a Project article for more information.

- Model how to change the parameter on the [Drive for] block to have the Code Base drive in reverse.



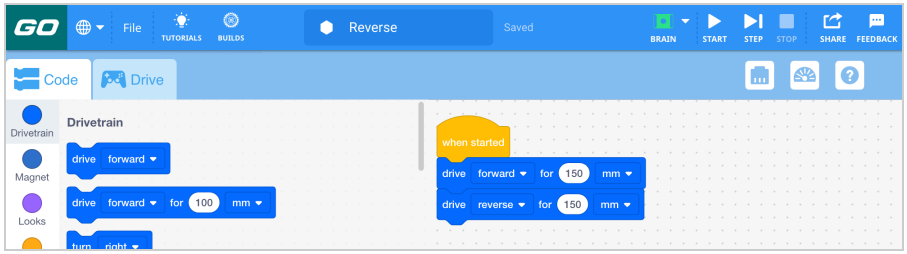
Change the parameter (reverse)

- Use the same estimation process as in Play Part 1. Have students place the Code Base in the starting position, then estimate how far the robot will move. They should place a marker where they think the Code Base will stop.
- Have students start their projects. You may need to remind them of the steps to Connect a VEX GO Brain if connection issues occur.



Reverse 150mm

- Once the students have observed the behavior of driving in reverse, model for students how to go back to their project. They should then rename their project *Forward and Reverse*. Refer to the steps in the Open and Save VEX Library article for more information.
- Model for students how to add a second [Drive for] block. One [Drive for] block should have the robot driving forward, and the second should have the robot driving in reverse. Model how to edit the parameters of the [Drive for] blocks, and then start the project again to see how the change in parameters affected the movement of the Code Base robot.



Forward and Reverse

3

Facilitate

Facilitate a discussion as students are editing their projects and observing the robot's behavior by asking the following:

- Can you show me using your hands how far you thought the Code Base robot would move before you ran the project?
- What did you change the distance parameter to and why? How far do you think the Code Base robot will travel now that the distance has been changed?
- When you added another [Drive for] block, did you set them to travel the same distance? Do they have to be the same? Why or why not?
- If my Code Base robot is coded to drive forward 100 mm, how far would I need to change the distance if I wanted it to go twice as far?

Discuss the Code Base Robot's Movement

4

Remind

Remind students may have questions when they are editing and starting their project. Remind students that learning new concepts may take multiple tries and encourage them to try again if they are unsuccessful at adding and editing the blocks in the project.

5

Ask

Ask students to think about how the Code Base robot would need to move, if they wanted it to drive to the doorway, and then back up to where it started. What types of tasks or jobs could the Code Base robot now perform that it can move forward and backward? Ask students to suggest a task that the Code Base robot could now complete using forward and backward movements.

Optional: Groups may deconstruct their Code Base robot if needed at this point in the experience. They will use the same build in the subsequent labs, so this is a teacher option.

Share

Show Your Learning

Discussion Prompts

Observing

- How did you decide where the Code Base robot would end up after the project was started?

- How do you change how far the Code Base robot moves?
- What blocks did you use in your project? Can you explain what they do?
- Can you show using gestures where the drivetrain is on the Code Base robot?

Predicting

- If you changed the direction that the Code Base robot was facing, would it change your prediction of how far it would travel? Why?
- If you wanted the Code Base robot to travel forward and in reverse the same distance, how would do that in a project? What blocks would you use and what would the distances be?

Collaborating

- How did you work within your group to create and start your project?
- Did you run into any challenges that your group helped you solve?