Lesson 1.3: Your Wish Is My Command

Objectives

In this lesson, students will:

- Learn the broadcasting concept and practice using it in a Scratch project
- Revisit what an event is
- Learn to customize a sprite

Agenda

1. Broadcasting	15 mins
2. Student Activity: Your Wish Is My Command	20 mins
3. Wrap Up and Reflections	15 mins

Preparation

- ❑ Projector for project demonstrations
- ❑ Become familiar with the Demonstration and Sample projects
- ❑ Print student activity worksheet one per student pair

Resources & Links

- ❑ Demonstration Project: https://scratch.mit.edu/projects/284713454
- ❑ Broadcasting Activity project: https://scratch.mit.edu/projects/326153011
- ❑ Sample Project: https://scratch.mit.edu/projects/296093588
- ❑ Starter Project: https://scratch.mit.edu/projects/296217899
- ❑ Pair programming video: https://tinyurl.com/kuwoond

1. Broadcasting

In this section you will review the concept of an event and introduce the concept of broadcasting to students.

Open the following project in Scratch to help with the discussion and demonstration.
https://scratch.mit.edu/projects/284713454

Display your screen and engage students in an interactive demonstration and instruction.

Select the Abby Sprite to show Abby's script and prompt a volunteer to read the code. Then ask students to describe what programming concept is used in the code shown.

Answer: The programming concept is an **event**. An event is an action that causes something to happen.

Prompt students to name the action and the "something" that happens. (action: click on Abby; the "something": say "**Everybody Move!**")

When we click on Abby on the stage, we want the cat and dog sprites to move. Do they move? No. How can we do that? How does one sprite let the other sprites know it's time to do something?

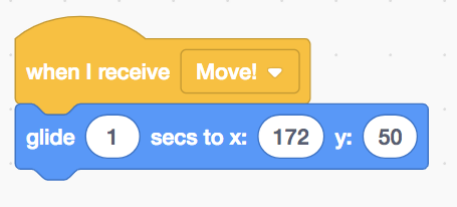
In the media, when a program is broadcasted (on the radio, or TV, social media), it is sent from one party for anybody to listen to. You can decide to hear it (passive) or to listen to it (active)

An example of a broadcast message in our daily lives could be a performer shouting to a crowd: "Hello everybody!" . You can choose to say hello back or simply go on with what you were doing.

In Scratch, sprites also communicate with each other using a broadcast message.
Demo adding the broadcast message:

To broadcast a message, you go to the **Events** category and choose the "**broadcast**" block. In the drop down menu choose New Message and type in a message, ideally one that makes sense. In the case of Abby, we will use "**Move!**"

Next, if a Sprite wants to listen to the message it adds a **When I receive <message>** event**.** For the **Cat** sprite, we add the following block to the script:



Click on Abby again to show what happens now.

What about the dog? It should move also.

Students will add the code for the dog in the next activity.

Remind students to be sure the message sent and listened to are the same.

Student Activity:

Instruct students to remix the following project and add the code to the dog sprite so it also moves when we click on Abby.
Project: 326153011

2. Student Activity: Your Wish Is My Command!

In this activity students will work in pairs to code a project to practice using events and broadcast messages.

Students are encouraged to work as a team with a partner, called pair programming. If students are already familiar with the concept of pair programming, you may choose to skip the next section. A review, however, could be beneficial.

Tell students they will be coding a project with a partner using events.

Pair Programming

When working on just about any project it is very helpful to have a partner to share ideas, ask questions, help figure out a problem together or help each other when one is stuck or does not understand something.

In many software companies like Google, Apple, Facebook and many more, computer scientists use pair programming and team programming where several programmers work together on a project. It would not even be possible to get most projects done without working in a team.

Let's watch this video to learn how pair programming works: **https://tinyurl.com/kuwoond**

When working as a pair, only one student logs into their account. If both students want to have the project in their Scratch account, tell students to share the project when they are done and immediately have the other student remix it into their own account.

Activity Description:

The project consists of giving a robot commands by clicking on various buttons that tell the robot what to do. Students are provided with a starter project that includes 3 buttons. A sample project is provided for you to demonstrate what the project is about.

Explain to students what they will be doing. **Demo** the sample project by running it and clicking on the various buttons:

https://scratch.mit.edu/projects/296093588

If they finish coding the 3 buttons, tell students to add their own button with a new command.

Note: You may choose to demonstrate how to add text, change the color and move the text inside a button sprite if students have not done this before.

Distribute the activity worksheet, one per student pair.

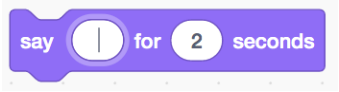
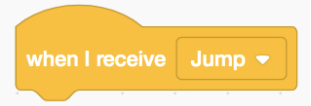
3. Wrap Up and Reflections

Remind students to share their project so partners whose account was not used can remix it into their account. Tell students to save their project and sign out of Scratch!

Reflection Points:

- What did you learn today?
- What is an event?
- How do sprites send messages to each other?
- What is a real life example of broadcasting?

Student Activity Worksheet: Your Wish Is My Command!

What to do:	Using/Details:
Remix and **save**	296217899
Create a robot sprite	
When the robot sprite receives the broadcast message "**Speak**", add a **say** block to say something for 2 second	
The ***Button2*** sprite sends a Jump broadcast message. When the **Robot** sprite receives the "**Jump**" broadcast message: the robot moves up (**change y by 60**) waits 0.4 seconds (so you can see it move) moves back down	
Add the script for the "**Dance**" command.	

Extended Activity

Add another button with a new command and write the scripts to do something new when the button is clicked.

You can duplicate one of the other Button sprites and then change the command under the **Costumes** tab.

Follow the directions to the right to change the text of your new button.