Robot Puppy Training

Summary

Description

Time to get Dash ready for the Wonder Workshop Dog Show!

Learning Procedure

In this puzzle, Dash is part robot, part puppy and loads of fun! Your students will learn to train/program Dash to perform 5 different pet tricks over the course of **5 challenges**. Each trick will be cued by a **When** block. Students will gain practice in using an array of **When** cues including: *Top Button, Picked Up, Obstacle in Front and Hear Voice*. When a condition is met, Dash will perform a command, e.g., *look left, right, up and down, turn 45 degrees left or right, make dog sound.* In the final challenge, students will be able to run the program for Dash to perform all 5 tricks! This puzzle reinforces the concept of conditions: ***When*** *this happens; that happens*. Example: Pushing the button on top of Dash's head cues a segment of code to run, i.e., Dash will Sigh, Turn Left 45° and then Turn Right 45°.

Concepts Covered

- **Stacks -** students will learn to program separate stacks of commands cued by a **When** block.
- **When -** students will edit the condition which executes the **When** command: Top Button, Obstacle in Front, Picked Up, Hear Voice.
- **Look** - students will add **Look** commands, e.g., look up, down, left and right, cued by **When** blocks.
- **Drive** - students will add Drive commands including **forward, turn left, turn right** and **turn to a voice**.
- **Angles** - students will recognize that the direction of a **Turn** or **Look** is based on angles, e.g., 45°.
- **Drive** - students will recognize that movement can be adjusted by distance or speed.
- **Sounds** - students will add pre-programmed and record new sounds in **Sound.**

In App

Vocabulary

- **When**: the When command executes when a specified event occurs

- **Look:** to face a certain direction
- **Drive:** to control the movement or direct the course of Dash
- **Sounds:** anything that can be heard by your ears
- **Stacks:** a group of commands arranged in layers

Reflection Questions

1. How is training a dog similar to programming Dash? Which do you think is harder? Why?

2. Taking care of a dog is a lot of work. What happens **when** these specific events occur in real life? (a) brushing a dog (b) feeding a dog (c) playing with a dog (c) throwing a stick, etc. (Possible answers: Your dog will have a shiny coat, the dog will be healthy, happy, the dog will return the stick.)

3. What types of **when** events occur in school? At home? In nature? What happens when these events occur? (Possible response: When you misbehave, you go to the principal's office.)

4. Can you explain why Dash's tricks were programmed as 4 separate stacks beginning with a **when** block? Why weren't they all included in one stack?

5. How is the weather like a **when** command? How does it affect your day? (Possible responses: "When it is raining. I wear my rain coat. I do not play outside." "When it is snowing, school closes and I go sledding."

Activity Extensions

1. Tricky Puppy

If only a real puppy would listen like Dash does. Challenge students to teach Dash more tricks, using the **when** command! Some are our favorite tricks are: turn around, play dead and go crazzzzzy! Just for fun, have students decorate Dash to look like their favorite kind of dog, e.g., a boxer, lab, poodle, etc. Create a collar and leash for Dash, too.

2. Doggy Functions

In this math based game, students will learn about function tables. In keeping with the dog-based theme, students will input doggy treats for Dash. After looking at a function table's inputs and outputs, students will attempt to guess the **when** event or rule. Example: **When: One dog treat is added to a bag.** After playing a few rounds, have students create their own function tables for peers. *Adapt this game to a grade based on level and complexity of operation.

WHEN INPUT # TREATS	**# OF OUTPUT TREATS**
1	2
4	5
7	?

3. Tic Tac Paw

In this game, two Dash robots will go paw-to-paw in an exciting game of Tic Tac Paw.
Each student will control his or her pet-robot using the remote control feature in the Go App. Use bulletin board paper, draw a tic-tac-toe grid large enough for Dash to fit in each square. Students take turns moving their pet-robots into a square and then claiming it with an "X" or an "O". After the game discuss which **when** events caused each player to move to a particular square. For example: "I had two in a row diagonally, but when the other player moved into the third diagonal square, I had to move to the top left square." How are possible moves affected as the game progresses? If so, how? (There are less places to move on the board.) Is it possible to program the game using the **when** block, so that one player will always win? If so, how?

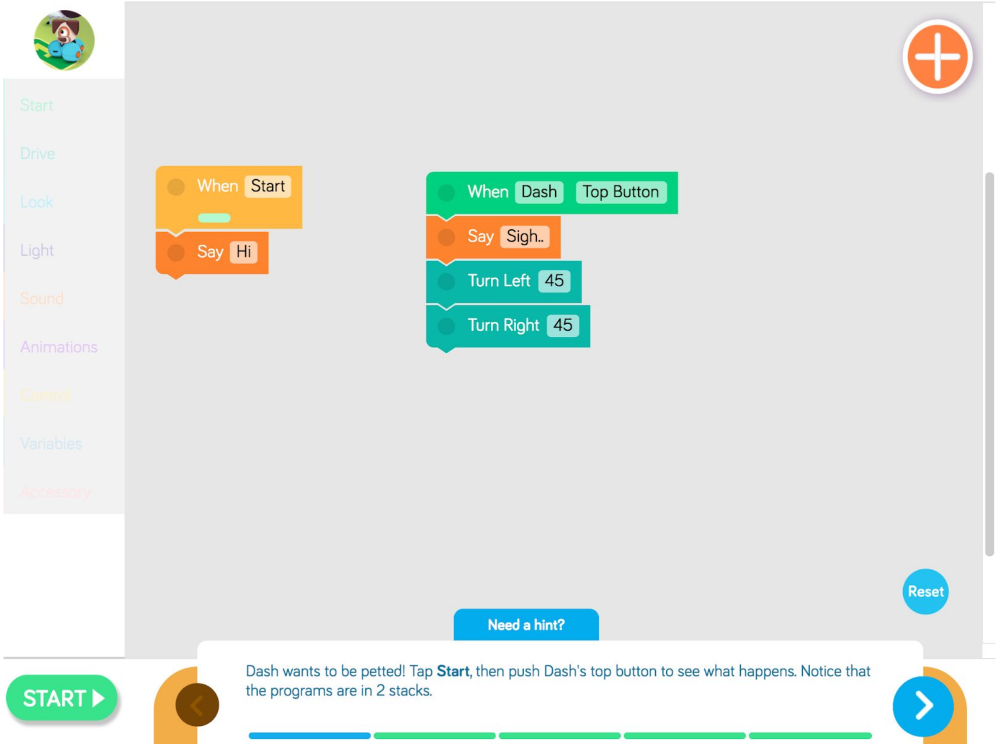
4. Cloudy Events

It's the day of the pet show, but from the look of those dark clouds overhead, we're going to get rain. In this science activity, students will learn about different types of cloud formations and compare them to **when** events. Example: "When there are cirrus clouds in the sky, a change in the weather is indicated to occur in the next 24 hours." Students will research about clouds on the internet and share the different types of clouds as a program or function table. "When this type of cloud occurs....then this happens. Types of clouds include: cirrus alto, stratus, cumulus, etc. Here's a child-friendly website students may use. Have students discuss the purpose of weather forecasting.
http://www.weatherwizkids.com/weather-clouds.htm

Solutions

Challenge 1

Dash wants to be petted. Tap **Start** then push Dash’s top button to see what happens. Notice that the programs are in 2 stacks.



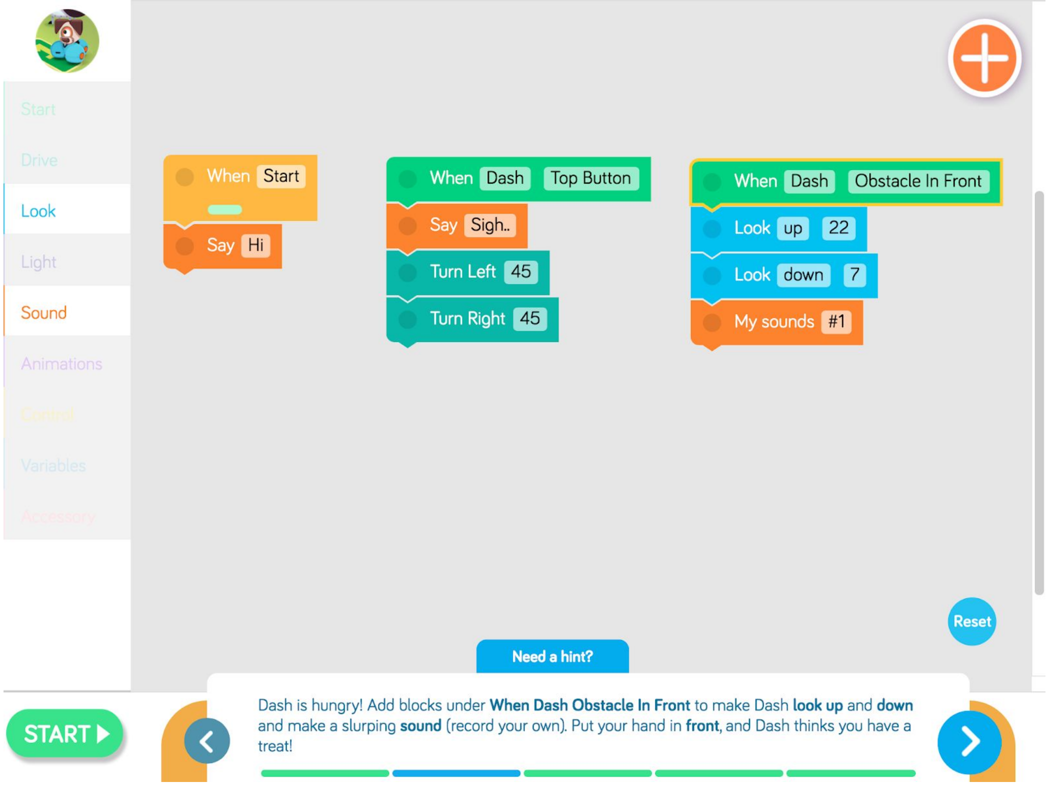
```
When Start
Say Hi

When Dash Top Button
Say Sigh..
Turn Left 45
Turn Right 45
```

Dash wants to be petted! Tap **Start**, then push Dash's top button to see what happens. Notice that the programs are in 2 stacks.

Challenge 2

Dash is hungry Add blocks under **When Dash Obstacle in Front** to make Dash **look up** and **down** and make a slurping **sound** (record your own). Put your hand in **front**, and Dash thinks you have a treat!



```
When Start
Say Hi
```

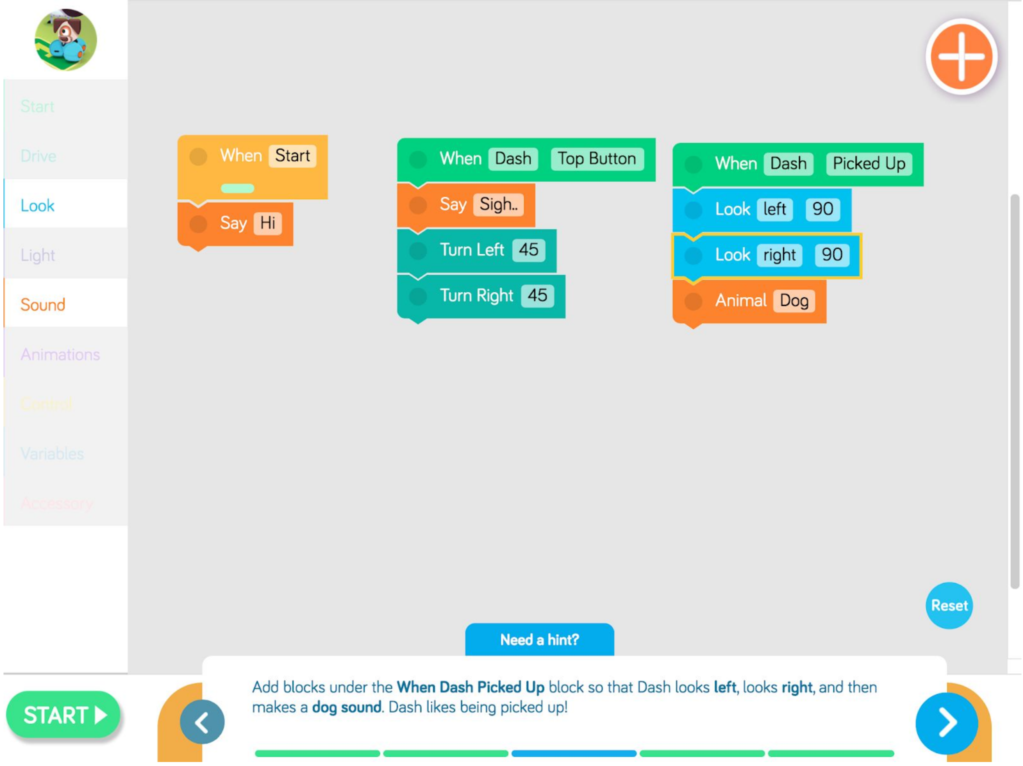
```
When Dash Top Button
Say Sigh..
Turn Left 45
Turn Right 45
```

```
When Dash Obstacle In Front
Look up 22
Look down 7
My sounds #1
```

Dash is hungry! Add blocks under **When Dash Obstacle In Front** to make Dash **look up** and **down** and make a slurping **sound** (record your own). Put your hand in **front**, and Dash thinks you have a treat!

Challenge 3

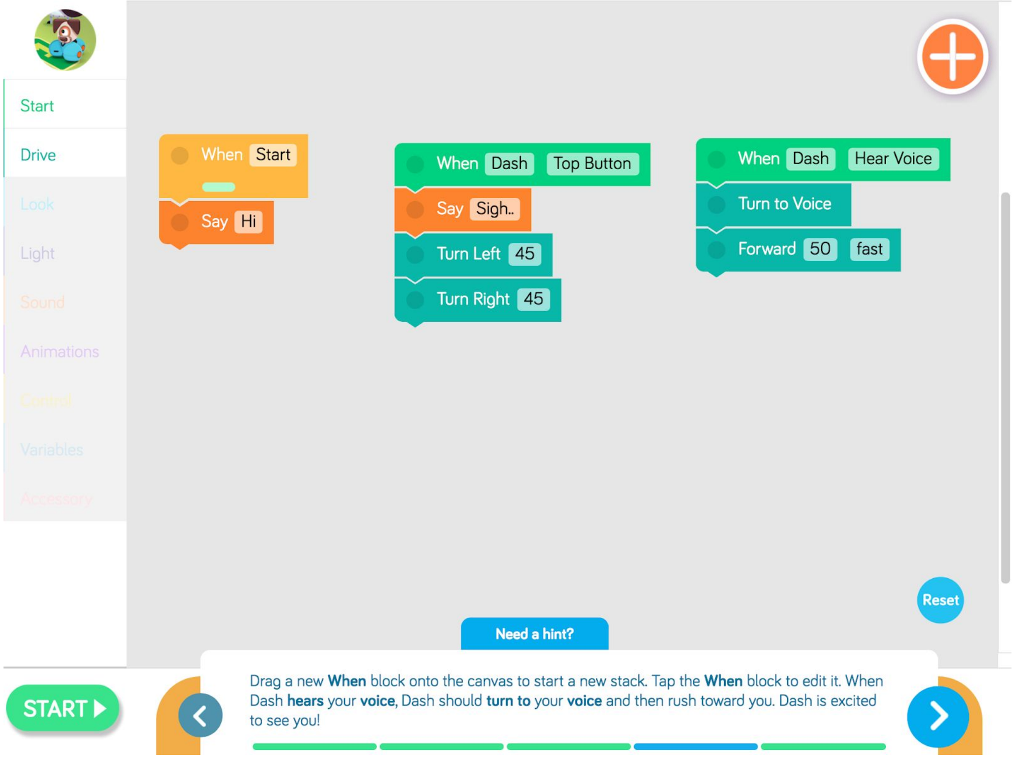
Add blocks under the **When Dash Picked Up** block so that Dash looks **left**, looks **right,** and then makes a **dog sound.** Dash like being picked up.



Add blocks under the **When Dash Picked Up** block so that Dash looks **left**, looks **right**, and then makes a **dog sound**. Dash likes being picked up!

Challenge 4

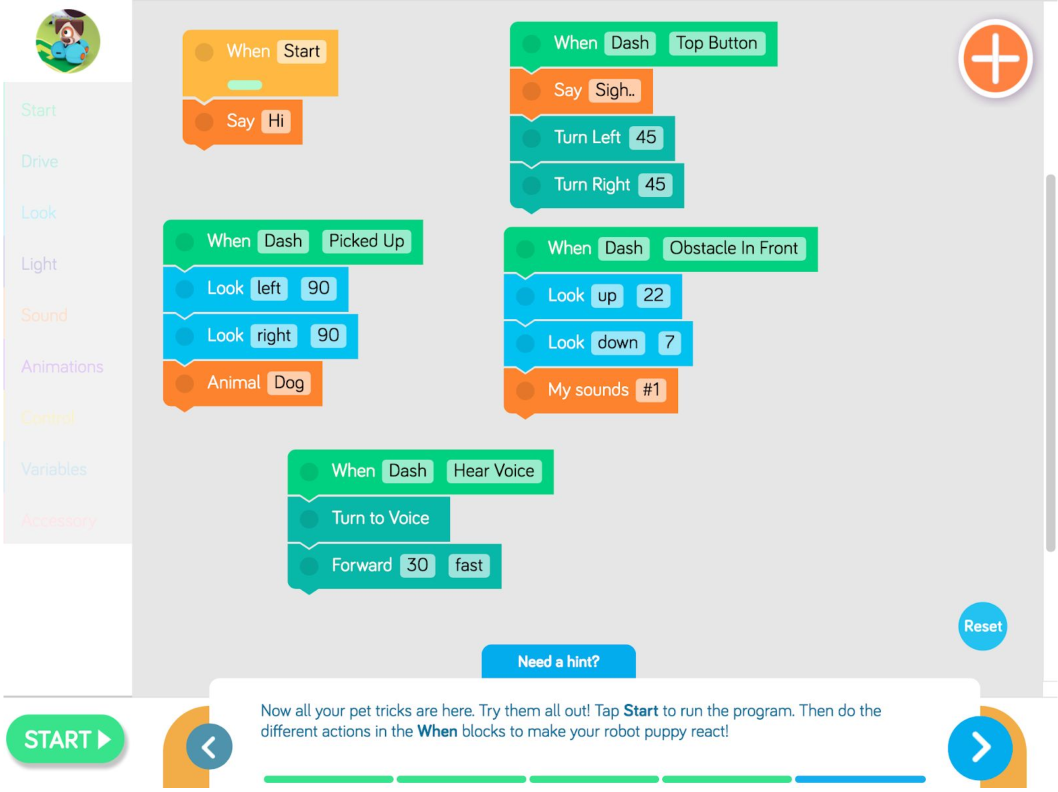
Drag a new When block onto the canvas to start a new stack. Tap the **When** block to edit it. When Dash **hears** your **voice**, Dash you turn to your voice and rush toward you. Dash is excited to see you!



Drag a new **When** block onto the canvas to start a new stack. Tap the **When** block to edit it. When Dash **hears** your **voice**, Dash should **turn to** your **voice** and then rush toward you. Dash is excited to see you!

Challenge 5

Now all of your pet tricks are here. Try them all out! Tap **Start** to run the program. Then do the different actions in the **When** block to make your puppy react.



```
When Start
Say Hi

When Dash Top Button
Say Sigh..
Turn Left 45
Turn Right 45

When Dash Picked Up
Look left 90
Look right 90
Animal Dog

When Dash Obstacle In Front
Look up 22
Look down 7
My sounds #1

When Dash Hear Voice
Turn to Voice
Forward 30 fast
```

Now all your pet tricks are here. Try them all out! Tap **Start** to run the program. Then do the different actions in the **When** blocks to make your robot puppy react!

Educational Standards

CC Mathematical Practices:
1, 2, 4, 5, 6, 7, 8

CC Math Standards:
CCSS.MATH.CONTENT.5.OA.B.3

CC Language Arts Standards
RI.2.5; R1.2.10; RI.3.4; RI.3.10; RI.4.4; RI.4.10; RI.5.4; R1.5.10

CSTA K-12 Computer Science Standards

- 1A-DA-05 Store, copy, search, retrieve, modify, and delete information using a computing device and define the information stored as data.
- 1A-AP-08 Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks.
- 1A-AP-10 Develop programs with sequences and simple loops, to express ideas or address a problem.
- 1A-AP-11 Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
- 1B-AP-08 Compare and refine multiple algorithms for the same task and determine which is the most appropriate.
- 1B-AP-10 Create programs that include sequences, events, loops, and conditionals.
- 1B-AP-11 Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.
- 1B-AP-12 Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

NGSS Science and Engineering Practices

- K-ESS2-1 Use and share observations of local weather conditions to describe patterns over time. *This standard applies to Cloudy Conditionals Activity Extension.
- K-ESS3-2 Ask questions to obtain information about the purpose of weather forecasting to prepare for, and respond to, severe weather.* *This standard applies to Cloudy Conditionals Activity Extension.
- K-2-ETS1-1 Engineering Design Ask questions, make observations, and gather information about a situation people want to change to define a simple problem that can be solved through the development of a new or improved object or tool.
- 3-5-ETS1-2 Generate and compare multiple possible solutions to a problem based on how well each is likely to meet the criteria and constraints of a problem. *Also applies to Activity Extensions