wonder workshop

Juggernaut Challenge

Juggernaut Challenge

Club Leader Notes

Transform Dash into an unstoppable Juggernaut that you can pull back and let go!

wonder workshop

Objectives

Kids will:

- Learn new ways to program Dash to respond to their actions.
- Learn ways to add conditional statements using Wonder.
- Use measurement to determine how far Dash has traveled.

Materials & Prep

Required

Dash

- A stack of cardboard blocks or other light obstacles that Dash can drive through and knock down.

Time Required

30 minutes

Concepts Covered

- Sequences
- Conditional Statements
- Distance and Measurement
- Cues and Behaviors

Move
Move Backward
Sound: Brave
Instant
Wait For
Do Nothing
Green Lights
Yellow Lights
Red Lights
Lights Off
Stop
Spin To

Warm Up

- Practice using the **Move Forward** and **Move Backward** cues. Kids will need to physically move Dash forward and backward with their hands to trigger the cues.
- Show these videos to any Wonder beginners:
  - All About Cues
  - Creating & Deleting States
  - The Obstacle Seen cue

Juggernaut Challenge

Club Leader Notes

Transform Dash into an unstoppable Juggernaut that you can pull back and let go!

wonder workshop

Level 1

Transform Dash into a juggernaut, then pull Dash back with your hand, let go, and watch Dash smash through a wall!

- Download the starter program using this key: egfj
- Run the program. Then pull Dash backward with your hands, stop, and remove your hands. Dash will move forward like a slingshot and bash through the wall you've set up.
- Add sound to the program. You may want to direct kids to use sounds in the **Brave** category.

Level 2

Change the program so that Dash stops moving after a couple of seconds, turns around, and then smashes back through the wall from the opposite direction!

- Make Dash stop after moving forward for 2 seconds.
- Then make Dash spin 180 degrees.
- Then make Dash resume the original program and rush forward again.
- **Example solution:** gmst

Level 3

Change the program so that Dash's speed and sound are different based on how long the backward push takes. The longer you push Dash, the faster Dash should go.

- If you pull Dash backward a certain distance, Dash should make one sound, but if you pull Dash backward another distance, Dash should make a different sound, for a total of 3 different sounds.
- Dash's speed should increase in relation to the length of time you pull Dash backward.
- Dash's lights should change the longer you pull Dash backward.
- **Example solution:** z7bs

Notes:

Juggernaut Challenge

Activity Sheet

Transform Dash into an unstoppable Juggernaut that you can pull back and let go!

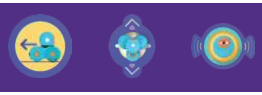
wonder workshop

Level 1

Transform Dash into a juggernaut, then pull Dash back with your hand, let go, and watch Dash smash through a wall!

- [ ] Download the starter program using this key: egfj
- [ ] Run the program! Test it by moving Dash backward with your hand and then letting go.
- [ ] Dash should move forward without your help.

Try these behaviors and cues:



Level 2

Change the program so that Dash stops moving after a couple of seconds, turns around, and then smashes back through the wall from the opposite direction!

- [ ] Add to the program to make Dash stop after 2 seconds.
- [ ] Make Dash turn around 180 degrees.
- [ ] Then make Dash resume the program and rush forward again, the same as the first time.

Try these behaviors and cues:

Level 3

Change the program so that Dash's speed and sound are dependant on how long the backward pull takes. The longer you pull Dash, the faster Dash should go.

- [ ] Program Dash to move forward in response to the **Move Backward** cue.
- [ ] Dash should make 3 different sounds depending on how long you pull Dash backward.
- [ ] Dash's speed should increase in relation to the length of time you pull Dash backward.
- [ ] The color of Dash's lights should change depending on how long you pull Dash backward.
- [ ] Run the program and test it by moving Dash **backward** with your hand and then stopping and letting go. Dash should move **forward** without your help.

Try these behaviors and cues:

Ideas

- Try loading your programs onto the robots so that you don't need to use the app to run them.