

# code jumper



**Lesson 12**  
Variables

# Code Jumper Curriculum: Lessons



## Lesson 12 Variables

American Printing House for the Blind

*Code Jumper Curriculum: Lessons*

Copyright © 2020 American Printing House for the Blind

All rights reserved. No part of this publication may be reproduced, stored in retrieval system, or transmitted in any form or by any means, except as expressly permitted under copyright law, without written permission of the publisher.

Published by American Printing House for the Blind

1839 Frankfort Avenue, Louisville, KY 40206

[www.aph.org](http://www.aph.org) | [info@aph.org](mailto:info@aph.org)

# OVERVIEW

## LESSON OBJECTIVES

Students will understand what a variable is and be able to develop programs that use variables.

## EXPECTED OUTCOMES

Students will:

- All students: Be able to explain that a variable is used to store a value for use later in a program
- Most students: Be able to use variables to play the same sound at different points in a program
- Some students: Be able to use variables to control multiple Loop and Selection pods

## LESSON PLAN STRUCTURE

- Unplugged Activity
- Guided Code Jumper Activity
- Exploration
- Standards and Check for Understanding

## RESOURCES

- Code Jumper Tutorial Videos
  - The Hub: <https://www.youtube.com/watch?v=KGb51PW9zJQ&lis=>
  - Play and Pause Pod: <https://www.youtube.com/watch?v=446jCw8qcDI&t=>
  - Plugs: <https://www.youtube.com/watch?v=hTy8moSohIE>
- Code Cards

## KEY VOCABULARY

- **Variable:** a placeholder for a piece of information that can change
- **Constant:** A fixed value in a program

# UNPLUGGED ACTIVITY

## OBJECTIVE

After this lesson, students will understand what a variable is and how it can be used in computer programming.

## MATERIALS

- 2 boxes (big enough to hold index cards)
- Index cards (two per student)

## INSTRUCTION

1. Explain to students that sometimes we want to store a value or information for use later in our programs. A variable is storage for a value that can be used again at a later point in a program.
  - a. Suggest that students think of a variable as a box where they can store a value, which can then be used again at any later point in the program.
2. Give each student two index cards. On one card, students should write or touch-type a place. On the other card, students should write down or touch-type an object.
3. Students should put the "place" cards in one box, and the "object" cards in the other box.
4. After collecting the cards in the boxes, write this sentence on the board, and available for students in near distance format: "Tomorrow I am going to (place) and I need to bring (object) with me."
5. As a group, read the sentence out loud together. Then allow students a turn to pick a card out of the box for each blank and read it at the appropriate place in the sentence as it is read aloud by the group.

6. Review that this sentence is acting like an algorithm and the “place” and “object” boxes are acting as variables.
7. Allow students time to create a new algorithm, or sentence, that uses the same “place” and/or “object” variables in their Computer Science Journals. (Example Response: “In (place) we found a treasure chest with (object) inside.”)
8. Walk to each student with the two boxes and allow them to pull out one card from each and write it in their blank. The students can put the cards back into the boxes. Continue until all students have completed their sentences.
9. Allow time for some (or all) students to read their sentence out loud.

## CLOSURE

Review with students that sometimes we want to store a value or information for use later in our programs. A variable is like the box used in class to store the index cards, but in a computer program it can store a value for use later in the program.

# CODE JUMPER GUIDED ACTIVITY

## OBJECTIVE

Students will be able to develop a Code Jumper program that uses variables.

## MATERIALS

- Code Jumper Kit
- Videos (links)
- Code Cards

Example table for recording information in Computer Science Journals:

Code Jumper Setup	Play pod 1 Sound	Play pod 2 Sound	Play pod 3 Sound
Original program			
Program with x variable			
Program with x variable with constant ___ inserted in Play pod 1			
Program with constant ___ inserted in Play pod 1 and variables in Play pods 2 and 3			

## INSTRUCTION

1. Ask students to create a simple program using three Play pods in Port 1. Choose a Sound Set, such as Nature or People sounds, and set each Sound Dial to a different sound.



[Figure Caption:] A photo of a Code Jumper program with three Play pods, one after the other, connected to the Hub at Port 1

2. Ask students to record the code they created in their Computer Science Journals. This should include the sounds and duration of each Play pod.
3. Instruct students to remove the Variable plug from the kit and take a moment to look at it and explore. Notice that it is

larger than the purple plugs and that it has a port in the top for connecting a purple plug.



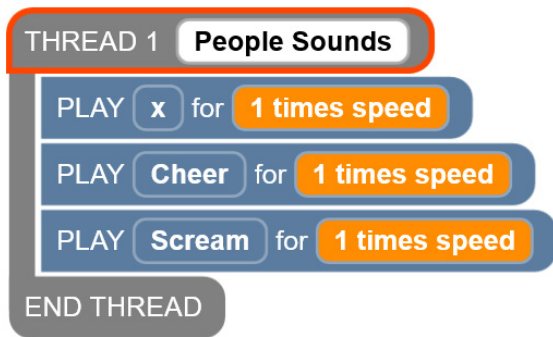
[Figure Caption:] A photo of a Variable plug

4. Have students plug the variable into the Sound Dial on the first Play pod. Call this first Variable plug the “x Variable plug.”



[Figure Caption:] This photo shows a Code Jumper program with three Play pods connected, one after the other, to the Hub at Port 1. A Variable plug is connected to the port on the first Play pod’s Sound dial.

5. Ask students to read the code. Point out that when the x Variable plug is plugged in, the app shows x instead of the sound name. How does this change the code from the original program? Compare the code that was read to the code that was written in their Computer Science Journals.



[Figure Caption:] This screenshot of the Code Jumper app displays a program in Thread 1. Under THREAD 1 People Sounds, the commands read, PLAY x for 1 times speed, PLAY Cheer for 1 times speed, PLAY Scream for 1 times speed; the commands are followed by END THREAD.

6. Next, instruct students to choose Constant plug number 4 and plug it into the x Variable plug. Explain that doing this sets the value of x. Ask students to turn the dial and see what happens. (The sound does not change, as it is set by the variable and the constant.) Again, ask students to read the code and notice the change made by adding a constant. How does the code look different?



[Figure Caption:] At the top is a screenshot of the Code Jumper app with a program in Thread 1. Under THREAD 1 People Sounds, the commands read, x = 4, PLAY x for 1 times speed, PLAY Cheer for 1 times speed, PLAY Scream for 1 times speed; the commands are followed by END THREAD.

7. Ask students to run the program and record the sounds they hear in their Computer Science Journals.



8. Have students plug another Variable plug into the Sound dial on the second Play pod, and then a third Variable plug into the Sound dial on the third Play pod. Do not plug any constants into the second and third Variable plugs. Explain that using a constant with the x Variable plug on the first Play pod ensures that the notes on all of the Play pods are always the same.

```
THREAD 1 People Sounds  
x = 4  
PLAY x for 1 times speed  
PLAY x for 1 times speed  
PLAY x for 1 times speed  
END THREAD
```



[Figure Caption:] At the top is a screenshot of the Code Jumper app with a program in Thread 1. Under THREAD 1 People Sounds, the commands read,  $x = 4$ , PLAY  $x$  for 1 times speed, PLAY  $x$  for 1 times speed, PLAY  $x$  for 1 times speed; the commands are followed by END THREAD. Below this screenshot is a Code Jumper program with three Play pods, three Variable plugs, and one Constant plug. The Constant plug is connected to the Variable plug, which is connected to the port on the first Play pod's Sound dial. A Variable plug is connected to the port on the Sound dial of the other two Play pods.

9. Ask students to try different constants in the Variable plug and run the program.
  10. Have students read the code and make observations about how the code is written with the variables in place.
- Ask students to take apart the program they just created and create a new one.
11. Ask students to create two loops of three Play pods each and plug them into different ports. Suggest they choose different Sound sets to easily distinguish between the two loops.

```

THREAD 1 People Sounds
  LOOP 4 times
    PLAY blowing_nose for 1 times speed
    PLAY Cheer for 1 times speed
    PLAY Laughing for 0.5 times speed
  END LOOP
END THREAD

THREAD 2 Percussion
  LOOP 2 times
    PLAY Hi-Hat 1 for 1 times speed
    PLAY Djembe Drum for 1 times speed
    PLAY Snare Drum for 1 times speed
  END LOOP
END THREAD

```



[Figure Caption:] At the top is a screenshot a two-thread program in the Code Jumper app. Under THREAD 1 People Sounds, the commands read, LOOP 4 times, PLAY blowing nose for 1 times speed, PLAY Cheer for 1 times speed, PLAY Laughing for 0.5 times speed, END LOOP; the commands are followed by END THREAD. Under THREAD 2 Percussion, the commands read, LOOP 2 times, PLAY Hi-Hat 1 for 1 times speed, PLAY Djembe Drum for 1 times

speed, PLAY Snare Drum for 1 times speed, END LOOP; the commands are followed by END THREAD. Below the screenshot is a photo of a Code Jumper program with two threads, each including one Loop pod and three Play pods. The Loop pods are connected directly to the Hub in Ports 1 and 2 and are comprised of three PLAY pods each.

12. In each Loop pod, ask students to plug in the Variable plug and then choose any of the constants to plug into the Variable plug.
13. Ask students: What do you think will happen when you play your program? Ask students to record the information in their Computer Science Journals. (Expected response: Both loops will be repeated for the number of times shown on the x Variable plug.)
14. Instruct students to run their program to test their predictions and then record if their predictions were correct or incorrect in their Computer Science Journals.
15. Ask students to remove one of the constants and replace it with the Random plug. Then, have students record their prediction of what will happen when it is played in their Computer Science Journals. (Expected response: Each time the program runs, it will give x a random value, and both loops will repeat that number of times.)
16. Instruct students to play the program two to three times to see if their prediction was correct. Ask students to listen carefully and count how many times the loop played each time the program ran. Then, ask students to record the number of loops for each time it is played in their Computer Science Journals.

Example table of how to record the data:

Trial	Number of Loops
1	
2	
3	

17. Create the program from Variables Code **Card 1**: Row Your Boat with Melody on a Code Jumper Kit. Play the program for the entire class.
18. Discuss with students: If you were to recreate this program, where would you start? (Expected response: By decomposing it.)
19. In their Computer Science Journals, ask students to make a list of all the different parts of Code Jumper that they think will need to recreate this program.
  - Hub
  - 2 Loop pods
  - Pause pod
  - 5 Play pods
  - Variable plug
  - Constant number 3
20. Ask students to analyze the program. How many threads are there? (Expected response: Two)
21. Ask students how using a variable in this program will ensure that each thread will run the same number of times. (Expected Response: Using the Variable plug with constant three ensures that the notes inside the loop and the words Row, Row, Row are in synch and each sequence is playing three times.



[Figure caption:] The above screenshot shows a Code Jumper program in Threads 1 and 2. Under THREAD 1 Piano, the commands read,  $x = 3$ , LOOP  $x$  times, PAUSE for a  $\frac{1}{4}$  beat, PLAY C5 for  $\frac{1}{2}$  a beat, END LOOP, PAUSE for a  $\frac{1}{4}$  beat, PLAY D5 for  $\frac{1}{2}$  a beat, PLAY E5 for  $\frac{1}{2}$  a beat. The commands are followed by END THREAD. Under THREAD 2 Row Your Boat, the commands read, LOOP  $x$  times, PLAY Row for 1 times speed, END LOOP, PLAY Your boat for 1 times speed. The commands are followed by END THREAD.

## CLOSURE

- Ask students to explain what a variable is and to give an example of how they used a variable in one of their programs.

# EXPLORATION

## OVERVIEW

In this exploration, students will create a program with a designated outcome using the Variable plug.

## INSTRUCTIONS

1. Students will create a program using the Sound Set of their choosing, with two Play pods inside a loop. Each time the program loops, it should choose a note at random and play it twice. The loop should be set to repeat six times. (See Solution 1 below.)
2. Students will create a program that generates a random note and plays the same note on both the piano and violin at the same time. The program should play eight notes. (See Solution 2 below.)

Solution 1:

```
THREAD 1 Numbers
  LOOP 6 times
    x = random
    PLAY x for 1 times speed
    PLAY x for 1 times speed
  END LOOP
END THREAD
```

Solution 2:

```
THREAD 1 Piano
  LOOP 8 times
    x = random
    PLAY x for 1 beat
```

END LOOP

END THREAD

THREAD 2 ChoirAahs

LOOP 8 times

PLAY x for 1 beat

END LOOP

END THREAD

## CSTA K-12 COMPUTER SCIENCE STANDARDS\*

- 1B-AP-09: Create programs that use variables to store and modify data.

## NATIONAL CURRICULUM OF ENGLAND\*\*:

Key Stage 1:

- Understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions
- Create and debug simple programs
- Use logical reasoning to predict the behavior of simple programs
- Use technology purposefully to create, organize, store, manipulate and retrieve digital content
- Recognize common uses of information technology beyond school
- Use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies

## Key Stage 2:

- Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content
- Use technology safely, respectfully and responsibly; recognize acceptable/unacceptable behavior; identify a range of ways to report concerns about content and contact.

## CLOSING ACTIVITIES AND CHECK FOR UNDERSTANDING

Ask students to discuss and write in their Computer Science Journals where variables can be found in their lives every day.

<b>Check for Understanding</b>	<b>Completed</b>
Students can explain in their Computer Science Journal that a variable is storage for a value that can be used again at another point in a program.	Yes / No

\*Computer Science Teachers Association (2017). CSTA K-12 Computer Science Standards, Revised 2017. Retrieved from <http://www.csteachers.org/standards>

\*\*Education, Department for. "National Curriculum in England: Computing Programmes of Study." GOV.UK, 11 Sept. 2013, [www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study](http://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study)



# CODE CARDS

## VARIABLES CODE CARD 1: ROW YOUR BOAT WITH MELODY

THREAD 1 Piano

x = 3

LOOP x times

PAUSE for a 1/4 beat

PLAY C5 for 1/2 a beat

END LOOP

PAUSE for a 1/4 beat

PLAY D5 for 1/2 a beat

PLAY E5 for 1/2 a beat

END THREAD

THREAD 2 Row Your Boat

LOOP x times

PLAY Row for 1 times speed

END LOOP

PLAY Your boat for 1 times speed

END THREAD

```

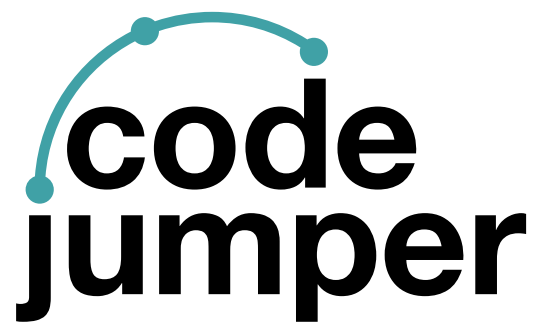
THREAD 1 Piano
  LOOP 3 times
    PAUSE for a ¼ beat
    PLAY C5 for ½ a beat
  END LOOP
  PAUSE for a ¼ beat
  PLAY D5 for ½ a beat
  PLAY E5 for ½ a beat
END THREAD

THREAD 2 Row Your Boat
  LOOP 3 times
    PLAY Row for 1 times speed
  END LOOP
  PLAY Your boat for 1 times speed
END THREAD

```



[Figure Caption:] At the top is a screenshot of a two-thread Code Jumper program. Under THREAD 1 Piano, the commands read, LOOP 3 times, PAUSE for a 1/4 beat, PLAY C5 for 1/2 beat, END LOOP, PAUSE for a 1/4 beat, PLAY D5 for 1/2 a beat, PLAY E5 for 1/2 a beat; the commands are followed by END THREAD. Under THREAD 2 Row Your Boat, the commands read, LOOP 3 times, PLAY Row for 1 times speed, END LOOP, PLAY Your boat for 1 times speed; the commands are followed by END THREAD. Below the screenshot is a photo of a Code Jumper program with two threads. Thread 1 comprises one Loop pod, two Pause pods, and three Play pods in this order: Loop pod, Pause pod (the start of the loop), Play (the end of the loop), Pause pod, Play pod, Play pod. Thread 2 consists of one Loop pod and two Play pods in this order: Loop pod containing one Play pod, Play pod.



For more resources, visit [codejumper.com](http://codejumper.com)

Copyright © 2020



**AMERICAN  
PRINTING  
HOUSE**

**1839 Frankfort Avenue  
Louisville, Kentucky 40206  
502-895-2405 • 800-223-1839  
[www.aph.org](http://www.aph.org) • [info@aph.org](mailto:info@aph.org)**