Lesson 12: Functions

45 minutes

Overview

This lesson introduces students to functions as way to organize and group repeated blocks of code together. Then, students learn how to use parameters in their functions as a way to generalize behaviors to work for different contexts. Students will use these concepts in future lessons, and it may be necessary to return to this lesson to reference the resources and videos.

Question of the Day: How can I use functions to organize and simplify my code?

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming
- ▶ **CS** - Computing Systems

Agenda

Objectives

Students will be able to:

- Use functions to organize repeated blocks of code
- Use parameters to generalize the purpose of a function

Preparation

- Prepare a board and USB cable for each pair of students
- Check the **"Teacher's Lounge"** forum for verified teachers to find additional strategies or resources shared by fellow teachers

Links

Heads Up! Please make a copy of any documents you plan to share with students.

For the teachers

- **Functions** - Slides ▾ Make a Copy

For the students

- **Functions** - Video (**Download**)
- **Functions** - Resource
- **Functions with Parameters** - Video (**Download**)

Introduced Code

- `function myFunction() { // function body, including optional "return"`
- `function myFunction(param1, param2, ..., paramN) { // function body,`
- `myFunction();`
- `myFunction(param1, ... paramN)`

Teaching Guide

Warm Up (5 minutes)

Distribute: Pass out a board and USB cable to each pair of students.

Transition to Code Studio: Have students get started in Code Studio right away

Teaching Tip

Starting Right Away: This lesson is packed with new content and includes several videos that help introduce students to these new ideas. In the interest of giving students the most amount of time to practice these new concepts, we recommend jumping straight into Code Studio and starting the lesson there.

Activity (35 minutes)

Transition: Send students to Code Studio

Teaching Tip

Guide to Programming Levels: Additional guidance for programming levels is provided in the **Guide to Programming Levels**. This document includes strategies and best-practices for facilitating programming levels with students.

Troubleshooting Devices: If student devices aren't able to connect, they can click the Help and Tips tab and then access the Maker Setup page (click the image to see more). This will let them re-reconnect their board to App Lab.

If students continue to have issues, refer to the Troubleshooting section of our **Maker Support Article**

1 Predict

Discussion Goal: Students might predict that the screen will update with button presses and the emoji and background colors will change. They may notice that `updateScreen()` appears in three places: within the two button events, and as a larger green block at the bottom of the code. It's okay if they're not completely sure how the green block is working, and they may try to describe what's happening by relating to variables - that a variable can be used in place of a value, and maybe `updateScreen()` is being used as a placeholder for the code at the bottom of the screen. Guide students to notice this is useful because the same code can be used in two different places (the left button and the right button).

It's okay to not come to a firm conclusion, since students will watch a video in the next level that explains Functions and how to use them in code.

Video: Show the **Functions** video in the slides.

Teaching Tip

Reviewing the Video: After watching the video, consider printing or displaying the corresponding Lesson Resource included as a PDF in the student resources of the lesson plan. This can be a helpful visual cue for students as they continue working through levels that require them to apply what they learned from the video, or can be used in a class discussion to summarize and clarify the content of the video. You can also remind students that these resources are available in the Help and Tips tab on individual levels.

To encourage active engagement and reflection, use one or more of the strategies discussed in the **Guide to Curriculum Videos**.

2-5 Skill Building

2 3 4 5

6 Predict

Discussion Goal: Students should notice that the left and right buttons are used, and that the function at the bottom is using the accelerometer to move something on the screen. Students may predict that the function changes depending on which button is pressed - the left button moves the purple bug, and the right button moves the orange bug. Students may also notice that the function has a variable, `bug`, that is included at the top of the function and used throughout.

In the next level, students will watch a video describing how these functions work and how `bug` is a **parameter** that can be used to change the behavior of a function when the function is called, like moving the orange bug or purple bug.

Video: Show the **Functions with Parameters** from the lesson slides.

Teaching Tip

Reviewing the Video: After watching the video, consider printing or displaying the corresponding Lesson Resource included as a PDF in the student resources of the lesson plan. This can be a helpful visual cue for students as they continue working through levels that require them to apply what they learned from the video, or can be used in a class discussion to summarize and clarify the content of the video. You can also remind students that these resources are available in the Help and Tips tab on individual levels.

7-9 Skill Building

7 8 9

10 Practice

11 Assessment

Assessment Opportunity

Formative Assessment: This level can be used as a formative assessment. A rubric is provided in the level, and written feedback can be given to students. **Click here to learn more about giving feedback to students**.

Wrap Up (5 minutes)

Journal

Prompt: *Think back to some of the programs that you have made before. What is one time that you could have used functions with parameters? What would the parameter be? How should the function's behavior change when the parameter changes?*