

Lesson 7: Getting Screen Inputs

45 minutes

Overview

Students learn to use several new design elements - text inputs, dropdowns, and sliders - so they can get user input from the screen of their apps. This lesson also introduces students to the `getProperty` and `getText` blocks, which allow them to access their user input in their code. Students first practice using these tools to determine what the user has input in various user interface elements. Students later use `getProperty` and `setProperty` together with the counter pattern to make elements move across the screen. A new event trigger, `change`, is also introduced to represent when a dropdown or slider changes values.

Question of the Day: How can we use the app screen for user input?

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

► AP - Algorithms & Programming

Agenda

Warm Up (5 minutes)

Journal

Activity (35 minutes)

Wrap Up (5 minutes)

Objectives

Students will be able to:

- Design apps using the text input, dropdown, and slider design elements
- Use `getProperty` and `getText` to access user input on the screen
- Use user input to update elements on the screen using code

Preparation

- Students do *not* need micro:bits for this lesson, and we recommend teaching the lesson without devices.
- 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

- [Getting Screen Inputs](#) - Slides

▼ Make a Copy

For the students

- [Taking Input with `getProperty`](#) - Resource

Introduced Code

- `getProperty(id, property)`
- `getText(id)`

Teaching Guide

Warm Up (5 minutes)

Journal

Prompt:

1. *What are all the ways we can get input from the user on a micro:bit?*
2. *What are some ways you've seen where you can get input from the user in an app?*

Share Out: Have students share their responses with a neighbor, then lead a full-class discussion.

Discussion Goal: Students should recall that they can use button presses to get input from the user on the micro:bit. For apps, they may bring up text inputs or dropdowns or buttons. They may also bring up checkboxes or sliders as other forms of inputs.

Remarks

So far we've only explored how to use the micro:bit for user input. Today we're going to look at more ways that we can get information from the screen of our app.

Activity (35 minutes)

Code Studio: 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.

 1

Predict

Discussion Goal: Two new code blocks are introduced in this level: `getProperty` and `getText`. Students may notice these new blocks are being used with variables, and may predict they are being used to get user input from the screen. They may also notice there are new design elements: text inputs, dropdowns, and sliders. Try to push students to notice that the variables from `getText` and `getProperty` are used again later in the code to update the screen using `setText` and `setProperty` blocks.

 2-7

Skill Building

2

3

4

5

6

7

 8

Practice

✓ 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)

Prompt: So far, you've seen several different types of input, some from the screen, and some from the micro:bit. Choose one type of input and answer the following questions about it.

1. What code do you need to get information from this input?
2. What's one example of when you would want to use this input?
3. What's an example of when you **wouldn't** want to use this input?

Discussion Goal: For this prompt, you can "jigsaw" the answers by assigning different students or different groups each a type of input and answering questions on each type. The different answers can then be combined for a class reference on input types.



This work is available under a [Creative Commons License \(CC BY-NC-SA 4.0\)](#).

If you are interested in licensing Code.org materials for commercial purposes [contact us](#).