

Lesson 17: Mouse Input

45 minutes

Overview

In this lesson, students continue to explore ways to use conditional statements to take user input - this time with the mouse. They will also expand their understanding of conditionals to include *else*, which allows for the computer to run a certain section of code when a condition is true, and a different section of code when it is not. This concept is introduced alongside several new mouse input commands, allowing students to gradually build up programs that use input in different ways.

Question of the Day: What are more ways that the computer can react to user input?

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

► **AP** - Algorithms & Programming

Agenda

Warm Up (5 minutes)

3-2-1 Review

Activity (35 minutes)

If/Else and More Input

Wrap Up (5 minutes)

Wrap Up

Objectives

Students will be able to:

- Respond to a variety of types of user input.
- Use an if-else statement to control the flow of a program.

Preparation

- Check the **"Teacher's Lounge"** forum for verified teachers to find additional strategies or resources shared by fellow teachers
- If you are teaching virtually, consider checking our **Virtual Lesson Modifications**

Links

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

For the teachers

- **If-Else Statements** - Resource
- **Mouse Input** - Slides ▼ Make a Copy

For the students

- **If/Else Statements** - Video (**Download**)

Vocabulary

- **Conditionals** - Statements that only run when certain conditions are true.

Introduced Code

- `if (condition) { statement1 } else { statement`
- `keyWentDown(code)`
- `keyWentUp(code)`
- `mouseDidMove()`
- `mouseDown(button)`

- `mouseWentDown(button)`
- `mouseWentUp(button)`
- `sprite.visible`

Teaching Guide

Warm Up (5 minutes)

3-2-1 Review

Prompt: What are three different things that you've been able to do with conditionals? What are two big things that everyone should remember when using conditionals? What's one thing you still want to learn how to program?

Share: Allow students to share out their responses.

Discussion Goal: Students may come up with many different specific scenarios, but make sure that they are differentiating between conditionals that respond to user input and those that respond to other changes in the program, such as a score increase, the location of a sprite, or player lives lost. Major things to remember may include putting the conditionals in the draw loop or paying attention to how the blocks are nested.

Remarks

That's great! Today we're going to look at a way to make our conditionals even more powerful, and see some new ways to get user input.

Question of the Day: What are more ways that the computer can react to user input?

Activity (35 minutes)

If/Else and More Input

Transition: Move the class to Code Studio, and have students complete the prediction level as a class or in small groups, then talk about what they found.

 Teaching Tip 

Guide to Programming Levels: Additional guidance for programming levels is provided in the [CSD Guide to Programming Levels](#). This document includes strategies and best-practices for facilitating programming levels with students.



1

Prediction

 **Video:** Show students the [If/Else Statements](#) video in the slides.

 Teaching Tip 

To encourage active engagement and reflection, use one or more of the strategies discussed in the [Guide to Curriculum Videos](#).

Questions to Consider with Video:

- What's an example of when you would need an "if/else" statement?

Discussion Goal: Make sure students are thinking of situations in which they want two different things to happen, depending on the situation. For example, they may say that they want one animation if the sprite is moving to the left and a different animation if the sprite is moving to the right. Challenge the students to think about when they would just use an `if()` block, and when an `if()/else()` block is necessary.



2-5

Skill Building

2

3

4

5



6

Practice



7



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.](#)



8

Challenges

Wrap Up (5 minutes)

Wrap Up

Question of the Day: What are more ways that the computer can react to user input?

Prompt: You now have many different ways to detect user input. With a partner, choose three different user input commands and think of an example of when you might use them. Be ready to share with the class!



Assessment Opportunity



This discussion serves as a brief review and assessment of the new user input commands. As students share out, press them to explain why their choice is better than other, similar choices (`mouseDown` / `mouseWentDown` / `mouseWentUp`). If any commands are missing after all the groups have shared out, elicit the missing ones from the group before moving on.



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](#).