

Lesson 17: Project - Interactive Card

Overview

Question of the Day: What skills and practices are important when creating an interactive program?

In this culminating project for Chapter 1, students plan for and develop an interactive greeting card using all of the programming techniques they've learned to this point.

Purpose

This end of chapter assessment is a good place for students to bring together all the pieces they have learned (drawing, variables, sprites, images, conditionals, user input) in one place. Students should still be working with code that is easily readable and doesn't involve very many high level abstractions. Giving students the opportunity to really be creative after learning all these new concepts will help to engage them further as they head into Chapter 2.

Assessment Opportunities

Use the project rubric attached to this lesson to assess student mastery of the learning goals of this unit.

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

- **AP** - Algorithms & Programming

Agenda

Lesson Modifications

Warm Up (5 minutes)

Journal

Activity (215 minutes)

Demo Project Exemplars (Level 1)

Unplugged: Interactive Card Planning

Levels: Implementing Interactive Card (Level 2 - 7)

Peer Review

Iterate - Update Code

Reflect

Objectives

Students will be able to:

- Apply an iterator pattern to variables or properties in a loop
- Sequence commands to draw in the proper order
- Use conditionals to react to keyboard input or changes in variables / properties

Preparation

- Read the Forum
- Print out a copy of the project guide for each student

Links

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

For the teachers

- **CSD Unit 3 - Interactive Animations and Games** - Slides
- **Extra Code in Challenge Levels** - Resource

For the students

- **Computer Science Practices** - Reflection
- **Interactive Card** - Project Guide
- **Interactive Card** - Peer Review
- **Interactive Card** - Rubric
- **Interactive Card - Student Checklist**

Teaching Guide

Lesson Modifications



Attention, teachers! If you are teaching virtually or in a socially-distanced classroom, please **click here** to access modifications that can be used during this lesson.

Warm Up (5 minutes)

Journal

Prompt: Think of one time you gave or received a card from someone. Who was that person? What was the purpose of the card? What about the card made it specific to that purpose?

Share: Have students share out their ideas.

Discussion Goal

This discussion introduces the chapter project: creating an interactive card. This is a good chance to tie in the 'card' concept to the problem-solving process, by defining the "problem" as cheering someone up, or letting them know that we are thinking about them.

Remarks

We're going to start designing our own cards today. Because we have learned how to program, our cards are going to be interactive. At the end of the project, you'll be able to email or text your card to someone.

Question of the Day: What skills and practices are important when creating an interactive program?

Activity (215 minutes)

Demo Project Exemplars (Level 1)

Goal: Students see an example of a final project and discuss the different elements that went into making it.

Display: Show students the sample program.



Demo App: Interactive Card

Discuss: Have students share their observations and analyses of the program.

Encourage the class to consider that there are multiple approaches to programming anything, but that there may be clues as to how something was created. In particular, when they are sharing their thoughts ask them to specify the following:

- Clues that suggest a sprite was used
- Clues that suggest a conditional was used
- Clues that suggest an iterator pattern was used

Display: Show students the rubric. Review the different components of the rubric with them to make sure they understand the components of the project.

Question of the Day: What skills and practices are important when creating an interactive program?

Unplugged: Interactive Card Planning

Goal: Students should plan out what they want to create before they head to the computer so that once they get to the computer they are just executing the plan.

Distribute: Hand out the project guide to students. Students will use this to scope out their projects before creating them. Give students some time to brainstorm the type of card they want to create and who the recipient will be.

Also distribute a copy of the Student Checklist to each student. Students can use this to self-assess and reflect as they develop their project.

Steps

1. The first layer of the interactive card is a background drawn with just the commands in the Drawing drawer. The front of the Activity Guide provides a grid for students to lay out their background, a reference table of drawing commands, and an area for students to take notes and write pseudocode.
2. Next, students think through the sprites they'll need, filling out a table with each sprite's label, images, and properties.
3. Finally, students consider the conditionals they'll need in order to make their card interactive.

Levels: Implementing Interactive Card (Level 2 - 7)

Transition: Once students have completed their planning sheet, it's time to head to the Code.org website. The short level sequence asks students to complete each element of their project.



Make an Interactive Card

💡 Teaching Tip

New Code and Previous Challenge Levels: Throughout the unit, students may have learned additional code in the challenge levels of different lessons. As students approach this project, they may want to revisit the code they learned in these levels or visit them for the first time to learn new codes to use in this project. To help guide students back to previous levels, you can use the **Extra Code in Challenge Levels** as a resource to quickly find where new codes were introduced in earlier challenge levels.

**3-7**

Project Work

3**4****5****6****7**

Peer Review

Distribute: Give each student a copy of the peer review guide.

Students should spend 15 minutes reviewing the other student's card and filling out the peer review guide.

Iterate - Update Code

Circulate: Students should complete the peer review guide's back side and decide how to respond to the feedback they were given. They should then use that feedback to improve their cards.

Students should use the rubric and student checklist to self-assess and make sure their project matches with the rubric.

💡 Teaching Tip

Rubric and Checklist: Students have two resources they can use for self-reflection and making sure they are on the right track: the rubric and the student checklist. We recommend having students use the checklist for their own self-assessment and reflection, since it may be easier to digest and understand when reviewing their own project. However, we recommend teachers use the full rubric for evaluating projects to give more accurate feedback to students. You can see examples of this with the Sample Marked Rubrics resource at the top of the lesson plan (only visible to verified teachers)

Reflect

Using the rubric, students should assess their own project before submitting it.

Send students to Code Studio to complete their reflection on their attitudes toward computer science. Although their answers are anonymous, the aggregated data will be available to you once at least five students have completed the survey.

 **8**

Reflection

Wrap Up (5 minutes)

Reflection

Question of the Day: What skills and practices are important when creating an interactive program?

Prompt: Have students reflect on their development of the five practices of CS Discoveries (Problem Solving, Persistence, Creativity, Collaboration, Communication).

- Choose one of the five practices which you demonstrated growth in during this lesson. Write something you did that showed this practice.

- Choose one practice you think you can continue to grow in. What's one thing you'd like to do better?
- Choose one practice you thought was especially important for this project. What made it so important?