

Lesson 24: Functions

Overview

Question of the Day: How can programmers use functions to create their own abstractions?

Students learn how to create functions to organize their code, make it more readable, and remove repeated blocks of code. Students first think about what sorts of new blocks they would like in Game Lab, and what code those blocks would contain inside. Afterwards students learn to create functions in Game Lab. They will use functions to remove long blocks of code from their draw loop and to replace repeated pieces of code with a single function.

Purpose

In previous lessons students have learned to use a number of abstractions in their programs, including the velocity properties, `isTouching`, and collisions. These abstractions have allowed them to build much more complex programs while ignoring the details of how that behavior is created. In this lesson students learn to build abstractions of their own by creating functions.

Students will primarily use functions to break code into logical chunks that are easier to reason about. This foreshadows the chapter's transition from building technical skill to the organizational processes used to develop software.

Assessment Opportunities

1. **Create and use functions for blocks of code that perform a single high-level task within a program**

See Level 9 in Code Studio.

2. **Explain the advantages of using functions in a program.**

In the wrap up discussion, make sure students understand that functions can increase both the readability and organization of code.

3. **Explain how functions allow programmers to reason about a program at a higher level**

In the wrap up, students should make the connection between functions and abstraction, that functions allow a programmer to name a bit of code so that

Objectives

Students will be able to:

- Create and use functions for blocks of code that perform a single high-level task within a program
- Explain how functions allow programmers to reason about a program at a higher level
- Explain the advantages of using functions in a program

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
- **Functions** - Resource

For the students

- **Functions** - Video (**Download**)

Vocabulary

- **Function** - A named bit of programming instructions.

Introduced Code

- `function myFunction() { __; }`
- `myFunction()`

they can think about it at that higher level, rather than worry about all the details.

Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

► **AP** - Algorithms & Programming

Agenda

Lesson Modifications

Warm Up (5 minutes)

Your Personal Blocks

Activity (35 minutes)

Wrap Up (5 minutes)

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)

Your Personal Blocks

Prompt: What's one block you'd like to have in Game Lab? What would it do? What code would it use to work?

Think-Pair-Share: Allow students to explain their blocks to a partner before sharing out their ideas.

Discussion Goal

As students share their ideas, make sure they have a clear idea of the code that they would use to make the block, reminding them that all of the blocks they have learned this chapter (e.g. velocity, collisions) have been built from blocks that they already knew.

Remarks

Today, we're going to learn how to create our own blocks so that we decide exactly how they will work. These special blocks are called functions, and they are one of the most powerful parts of programming.

Question of the Day: How can programmers use functions to create their own abstractions?

Activity (35 minutes)

Transition: Move students into Code Studio where they will learn to create and call functions.

Questions to Consider with Video:

- Think of a time when a function might have helped you write a program.
- What code would go in the **definition** of the function?
- When would you **call** the function?
- What would you name it?



1

Video: Functions

Discussion Goal

Goal:

Key Vocabulary:

- **Function** - a named bit of programming instructions

Make sure students understand the role of the two steps in using functions, as well as seeing functions as a form of "chunking" or abstraction. The function definition should include all of the code that they want to run, and the name of the function should be a short description of the purpose of the code. The function should be called at each place in the program that the student wants that block of code to run.



2-3

Skill Building

2

3

Teaching Tip

Introducing Functions

In these first few levels students are just being shown the syntax of functions and are not asked to write or create their own. It can be useful to explain creating a function as basically "creating a new block" just like another programmer created the "isTouching" or "velocity" blocks that they've seen actually contain other more complex code.



4

Predict



5

Practice

Teaching Tip

Why Use Functions

Level 5a, 5b, and 5c introduce three uses of functions, namely removing repetition in programs, allowing code to quickly be changed at multiple points, and providing organization in code. Students will need to write more of their own functions in these levels.

☰ 6

Quick Check

🖥️ 7-9

Collector Game

7

8

9

💡 Teaching Tip

Level 7: Functions in Context

In this and the next two levels, students will use functions to organize code within a simple game. While it is not identical to the side scroller, many of the skills and uses of functions in these levels can and should be used when they complete their side scroller.

🧑‍🤖 10

Challenges

Wrap Up (5 minutes)

Key Vocabulary:

- **Function** - a named bit of programming instructions

Prompt: Why would we say that functions allow us to "create our own blocks?" Why is this something we'd want to do? Why would a function count as an abstraction?

Discuss: Have students discuss at their table before talking as a class.

📝 Assessment Opportunity

Goal: Use this first prompt to review what students learned today. When they create a function they are creating their own block that they can call or use whenever they like. They saw at least two primary motivations for creating functions today including:

- Simplifying code by breaking it into logically named chunks
- Allowing a programmer to think at a higher level by hiding the details of a particular bit of programming instructions
- Avoiding repeated code by making one block you can use multiple times

Students should review the definition of abstraction as "a simple way of representing something complex". Note that a function is an abstraction because it allows you to create one simple name for a more complex block of code.

🗣️ Remarks

Functions are a useful tool for helping us write and organize more complex pieces of code. As we start

looking to the end of the unit and your final project, being able to keep your code organized will be an important skill.

Question of the Day: How can programmers use functions to create their own abstractions?