

# Lesson 26: The Game Design Process

45 minutes

## Overview

This lesson introduces students to the process they will use to design games for the remainder of the unit which is centered around a project guide that asks students to define their sprites, variables, and functions *before* they begin programming their game. Students begin by playing a game on Game Lab where the code is hidden, discussing what they think the sprites, variables, and functions would need to be to make the game. For the purposes of heavily scaffolding the software development process, students are then given a completed project guide that provides starter code and shows one way to implement the game. Students are then walked through this implementation process through a series of levels and have an opportunity to make improvements to the game to make it their own in the final level. In the subsequent lessons, students will need to complete a greater portion of the guide independently, and for the final project, they will follow this process largely independently.

**Question of the Day:** How does having a plan help to make a large project easier?

## Standards

Full Course Alignment

CSTA K-12 Computer Science Standards (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

Warm Up (5 minutes)

Play Cake Defender

Stop: Review Project Guide

Activity (35 minutes)

Implement Project Guide

Wrap Up (5 minutes)

Using the Project Guide

## Objectives

Students will be able to:

- Implement different features of a program by following a structured project guide

## Preparation

- Print copies of the the project guide if you will be giving students physical copies. Please note that this project guide is intentionally filled out. (See notes in Lesson Plan.)
- 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

- **The Game Design Process** - Slides  
▼ Make a Copy

For the students

- **Defender Game** ▼ Make a Copy

# Teaching Guide

## Warm Up (5 minutes)

### Play Cake Defender

**Transition:** This lesson begins immediately in Code Studio. On the first level, students will find a game but will not be able to see the code. They should play the game and follow the instructions which ask them to list the variables, sprites, and functions they think are necessary to create this game.

### Stop: Review Project Guide

**Discuss:** Students should have individually created a list of variables, sprites, and functions they would create to make the defender game they played. Ask students to share their lists with a neighbor before discussing as a class.

**Discussion Goal:** To help you run the conversation, you can write "Variables", "Sprites", and "Functions" on the board and record their ideas below each. Ask students to justify their decisions but don't feel the need to settle on one right answer.

### Remarks

There's usually lots of ways you can structure a program to get it to work the way you want. The important thing when writing complex or large programs is that you start with a plan, including the sprites, variables, and functions you will want in your program. Today we're going to look at how we could implement this plan to build our own defender game. By the end of the lesson you'll not only have built your game, but you'll know how to change it and make it your own. Let's get going!

**Question of the Day:** How does having a plan help to make a large project easier?

## Activity (35 minutes)

**Distribute:** Give each student or pair of students a copy of the project guide.

### Teaching Tip

**Project Guide:** The project guide is intentionally filled out for students so that they can experience using it as a reference when programming. This should give them more context when filling out their own project guide in the next two lessons.

You can give each student their own copy for reference, but you might also choose to print one copy per pair, share digital copies, or just display the guide on the projector. So long as it is available for reference, any approach will work fine.

**Prompt:** Compare the components of the game you thought would be included to the ones on this project guide. Do you notice any differences?

**Discuss:** As a class compare the list you made on the board to the list of variables, sprites, and functions on the project guide. Note the similarities. Where there are differences try to understand why. Don't approach one set as "right" vs. "wrong" but just confirm both would be able to make the game students played.

# Implement Project Guide

Students are given a large amount of starter code in this project. The sprites, variables, and functions have all already been given to them. The work of this project is writing the code for the individual functions. These levels guide students through how to implement those functions. As students move through the levels point out how the project guide is being used.

The most challenging skill students use in these levels is recognizing the need to create new functions to replace repeated code. Students need to build this skill on their own but these levels demonstrate an instance where this might happen.



1

Same Game



2

Plan Your Project



3-4

Set Up Sprites

3

4



5-7

Control Your Player

5

6

7



8-12

Sprite Interactions

8

9

10

11

12



13

Challenges

💡 Teaching Tip



This last level encourages students to make the game their own. If students have made their way to this point they have all the skills they need to progress through the curriculum, so there is no pressure to complete any of the modifications suggested in this level. If you have time, however, getting practice planning and implementing new features will be a useful skill. Even just modifying the animations of the game is an easy way students can make the game their own.

# Wrap Up (5 minutes)

## Using the Project Guide

**Prompt:** Today, you used a filled-out project guide as you completed your program.

### ✓ Assessment Opportunity ▲

As students answer the questions, make sure that they understand that the project guide is there to help them organize their work, so that they can look at smaller parts of the problem rather than thinking about it all at one time. They can use it to focus on one part of their program at a time as they code.

In the next lesson, as they fill out their own project guides, they will need to think carefully about the different parts of their program before they start coding. You may want to remind them of the "Prepare" part of the Problem Solving Process, and to think about how they might need to go back and tweak the project guide even after they have started coding.

- How did the project guide help you as you coded?
- What do you think will be important to remember when you fill out your own project guide?



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**.