

Blockly, App Lab, and Your Hour of Code

In this computational thinking module, you will accomplish the following learning objectives:

- Create an app that you can distribute via Facebook or Twitter or text to any smartphone.
- Learn how to think computationally by strategizing how to decompose problems, design algorithms, manipulate data, and use abstractions as you add features to your app.
- Experience how computer scientists use named procedures to solve not only the task at hand but also other tasks to which these procedures may apply in the future.
- Reflect on how this process of building capability through the development of shared libraries of named procedures empowers people, working groups, and entire disciplines to progress and move their fields forward.

If these learning objectives sound lofty and foreboding, worry not, because the learning activities you will experience in this chapter are designed in such a way that you will enjoy building your own app and reflecting on what it means to think computationally about how the world works.

What Is Blockly?



Blockly is a block-based programming tool from Google that enables you to write computer code by dragging and connecting digital blocks onscreen. Each block represents a computer programming construct. By snapping the blocks together, you learn how computer scientists think about organizing their coding strategies.

Blockly has two views that are interchangeable. First, the Blocks view lets you create code by playing with blocks. Second, there is a Code view that shows the JavaScript code generated by Blockly when you play with its blocks. At any time, you can switch back and forth between Code view and Blocks view. You can even drag blocks into the Code window, and you can write code in the Blocks window. The code you create is real JavaScript code. In fact, Blockly itself is written in JavaScript. Therefore, the experience you gain by completing these exercises is real-world authentic practice in learning how to code.

What Is App Lab?



App Lab is a rapid prototyping environment that uses Blockly for developing web apps that run on any browser or smartphone. When you are finished developing the app, you can distribute it via Facebook or Twitter or you can text your app to any smartphone. App Lab was developed by Code.org for use in their Hour of Code™ as part of the White House initiative to help more Americans learn to code.

What Is the Hour of Code?



Hour of Code is a trademark owned by Code.org. True to its name, the Hour of Code activities are designed to last about an hour. Their purpose is to motivate more students to pursue careers in computer science. The Hour of Code made the evening news when President Barack Obama did his hour with middle school students from New Jersey [cheering him on](#).

To attract learners across a broad spectrum of interests, the Hour of Code has modules based on a variety of themes. There are dozens of activities designed to motivate different age groups. The most familiar settings are Star Wars, Frozen, Minecraft, and Moana. Your professor worked through these hours of code and selected one for use in this chapter, namely, Star Wars: Building a Galaxy with Code.

Your professor chose this Hour of Code not only because of the popularity of Star Wars, but more importantly because it progresses in a manner pedagogically appropriate for helping you understand the process of computational thinking.

Getting Your Account at Code.org

If you do not yet have an account at Code.org, [go there now](#) to get one. These accounts are totally free, and they keep track of your progress as you work through your Hour of Code. If you do not finish your Hour of Code in one sitting, for example, you can log back on later to resume at the spot where you left off. After you finish your Hour of Code and you begin using App Lab to create your own custom apps, your Code.org account lets you have multiple projects underway. When you finish a project, you can share it on Facebook, tweet it out, or text it to any smartphone.

To get your free logon, go to [code.org](#) and follow the onscreen instructions to create your own student account. This account makes you part of the code studio at [studio.code.org](#). The studio provides you with access to course materials and tools for creating apps. In this chapter, you will use your code studio login to complete your Star Wars Hour of Code and use App Lab to create your own custom app.

Beginning Your Hour of Code



When you are ready to begin your Hour of Code, follow this link to [log on at studio.code.org](#). After you get logged on, follow this link to [begin the JavaScript version](#) of the Star Wars Hour of Code. As you will see, the Hour of Code advances through fifteen stages that begin with simple constructs and progress by adding more features and tools that you can use in creating your app. At any time, you can stop working and come back later and the Hour of Code will remember where you left off. You can resume working at that spot or you can move back to review any one of your previous stages. Here is a play-by-play description of what each stage has you do:

STAGE	TASK
1	Program the droid BB-8 to gather two pieces of scrap metal as Rey the scavenger asks: "We need that scrap metal. BB-8, can you get it?"
2	The game provides more scrap metal to gather as Rey continues: "We need more scrap metal. Can you get all the metal in this area?"
3	You must move more quickly as Rey admonishes: "Go quickly, BB-8."
4	Stage 4 introduces a bandit you need to strategize how to avoid.
5	Now there are three bandits you need to avoid. This requires you to move more strategically as you think about the order in which you place your computer commands.
6	The difficulty of the game board increases and Rey needs you to find four more pieces of metal.
7	You learn how to make your computer's up and down arrow keys move R2-D2 up or down the game board as Princess Leia needs R2-D2 to get a message to one of the rebel pilots.
8	Princess Leia needs R2-D2 to get messages to each one of the rebel pilots as quickly as possible, so you must program all four of your computer's arrow keys.
9	You begin keeping score by adding points as you reach each one of the rebel pilots. You need 900 points to win.
10	Princess Leia warns you to "watch out for the Stormtroopers" which you program to subtract 100 from your score if R2-D2 encounters a Stormtrooper. Reaching a Rebel

	Pilot adds 100 points. You can also program the Mynock to give more points when R2-D2 gets the Mynock.
11	Princess Leia says “I’m seeing increased signs of activity on this planet” as you add three puffer pigs and go get them.
12	Intrigue increases as you make two or more Mynocks appear when you get a Tauntaun. In all you need to get 8 Mynocks.
13	Princess Leia says “They’re multiplying” as you make two or more Mouse Droids appear every time R2-D2 gets one Mouse Droid. The goal is to get 20 Mouse Droids.
14	Princess Leia has you visit another planet where you get new commands that let you change your droid and its speed so you can get messages to the rebel pilots.
15	As Leia says “You’re on your own now, R2-D2,” you get to create your own Star Wars game environment from an increased palette of commands and events. When you click Finish you can share your game on Facebook, tweet it out, or text it to your friends to play on their smartphones.

Video Features

Four stages begin with two-minute videos in which Star Wars engineers introduce the computer concepts you learn as you work your way through the tutorial stages. At the beginning of stage 1, Star Wars producer Kathleen Kennedy explains how the droid BB-8 is totally controlled by computer software. Industrial Light and Magic (ILM) engineer Rachel Rose explains how the tutorial uses blocks to teach JavaScript and she shows how to add as well as remove blocks from your program. At the beginning of stage 4, Rose comes back onscreen to explain how you can program by typing or by dragging blocks. After explaining how she uses the autocomplete feature when she is typing code, Rose invites you to try typing some code. At the beginning of stage 7, Walt Disney Imagineering producer Charita Carter comes onscreen to introduce the concept of events and the functions they can trigger. Finally, at stage 15, Code.org’s chief product officer Alice Steinglass explains how she designed this tutorial to teach what you need to know in order to create your own game. Then she gives you a toolset that lets you create games even better than those used in the tutorial. After you finish your game, Steinglass shows how you can share it on Facebook, tweet it out, or text it to your friends to play on their smartphones.

Timeline of Course Activities

If you don’t finish everything in one session, you can log out and come back later. When you log back in, you will have the option of resuming where you left off, or you can choose the option to View the Course and you will see a timeline of course activities. Choose any spot in the timeline to go to that stage in the course. To jump to the freeform editor that lets you create your own Star Wars app, for example, choose the last spot on the timeline:



Computational Thinking In Your Hour of Code

The following table explains the concepts of computational thinking that you practice at each stage of the Star Wars Hour of Code. You should reflect on these concepts both before and after you complete

your Hour of Code in order to reinforce within your own mind how computational thinking works. Your professor wants you to reflect on this in order that you not only do computational thinking, but also, you reflect about how you are doing it. This process of thinking about how you think is called metacognition, an important strategy for helping people learn.

STAGES	COMPUTATIONAL THINKING CONCEPTS
1-6	When a program begins to run, it can execute computer commands in a straight-through sequence. As the tutorial progresses from stages 1-6, you make your command sequences accomplish more complex tasks.
7-8	Events provide opportunities for you to execute commands in response to things that happen when a user is interacting with your program. In stages 7-8 you learn to program event handler functions that let the user control things by pressing or clicking arrow keys.
9-11	You keep score by executing arithmetic computations that add or subtract points depending on events that happen as the user moves characters around the screen. You learn to strategize how many points to add. You also get to program the sounds.
12-14	Triggered events can call upon functions that add more actors which, when encountered, execute their own code handlers.
15	The final stage provides you with a blank App Lab canvas upon which you can create your own Star Wars game. Notice how the Blocks have a Command pane as well as an Event pane. Use these expanded commands and events to create your own functions powering a game you would like to share with your professor and your friends to show what you have accomplished here.

Finishing Your Hour of Code

In the final stage of your Star Wars Hour of Code, you will be provided with open access to the App Lab tools which you can use to create your own unique app within the Star Wars environment. Use these tools to make something creative. As you do so, take creative advantage of the programming constructs you learned in the Hour of Code. To submit your app, click the Finish button to pop out the sharing options:

Share your game:

<https://studio.code.org/c/148984366>



Notice how your app will have an http web address, which is what you submit if your professor asked you to submit this assignment in your course. The Sharing options also let you share your app via Facebook, tweet it out, or text it to any smartphone.

Documenting Your Code

The App Lab tools provided in Stage 15 provide you with more commands and events than before. New commands include changing the speed of different character types, setting the background where the game is played, altering the map of obstacles encountered in the game, and ending the game as a loss or a win. New events include handling the getting of all characters or all characters of a given type, and making something happen when the droid runs into an obstacle. In order to score highly on this assignment, you need to use this expanded toolset in a computationally thoughtful manner. Your task is to create a game in which you demonstrate how you thought about organizing your code into subtasks that help you solve the problem of creating a game in which events that happen to certain characters cause other things to happen, such as the spawning of new characters whose actions are similarly handled by your code. To help your professor see how you did this, you should document your code by using the JavaScript `//` comment feature. Understand that in JavaScript, a line of code beginning with two slashes `//` is a comment that you can use to document what is happening in your code. App Lab lets you write these comments in the **Show Text** view of your code. For example, if you write an event handler that punishes the player for hitting an obstacle, you can document what it does by typing `//` comments into your code as follows:

```
function whenTouchObstacle() {
  //Sound an alert when the Droid hits an obstacle
  playSound("alert1");
  //Two stormtroopers hear the alert and
  //come onscreen to investigate
  addCharacter("Stormtrooper");
  addCharacter("Stormtrooper");
}
```

```
function whenGetStormtrooper() {
  //getting a stormtrooper adds points to your score
  addPoints(100);
  playSound("R2-D2random");
}
```

```
function whenGetAllStormtroopers() {
  //getting all the stormtroopers wins the game
  endGame("win");
  playSound("applause");
}
```

Scoring Rubric

If you have been asked to submit this assignment for a grade, be sure to study the following rubric, which your professor uses in scoring your Hour of Code submission.

DIMENSIONS	2 POINTS	1 POINT	0 POINTS
DECOMPOSITION	Comments in the code's functions explain subtask dynamics enabling the player to win or lose the game.	The game is composed of subtasks but the code lacks comments explaining how to win or lose the game.	The game has no subtasks and its code lacks comments documenting how the game is won or lost.
PROGRAMMING	The droid can move around efficiently using all four arrow keys.	All four arrow keys function but they do not all work logically.	One or more of the four arrow keys are not working.
ALGORITHMS	Functions encapsulate a logical sequence of commands responding to events that happen in the game..	Functions are present but do not encapsulate more than a single command at a time.	Functions are missing or are so ill formed that game play is incorrect or illogical
ABSTRACTION	One or more event handlers introduce new characters that add another dimension into the game.	The game has event handlers but their code introduces no new characters into the game.	Coding is linear with no event handling to make things happen other than those hard coded at the beginning of the game.
DATA	In addition to adding points for good rewards, the game subtracts points when bad things happen.	The game keeps score by adding points or subtracting points but not both.	The game does not keep score.

Using App Lab to Create Your Own Apps

After you complete your Hour of Code, you will have the option of creating your own apps with App Lab. You do this with a generic version of App Lab that lets you create your own unique design. Follow this link to studio.code.org/projects/applab where you can get a blank workspace upon which you can create your own unique design. If you need help, follow this YouTube link to the [App Lab video tutorial](#).

To see some of the interesting kinds of projects you can build with App Lab, follow these YouTube links to the topics that interest you:

- [Creating a Math Quiz](#)
- [Making a Drum Machine](#)
- [Shell Game](#)
- [Whack An Emoji](#)
- Using App Lab's [Built In Documentation](#)

After you [log on at studio.code.org](#), you can make as many different projects as you want. There is a library of apps you can remix if you want to build on the work of others. To view your projects, go to studio.code.org/projects. If you want more detailed help learning App Lab, follow this link to the App Lab course at code.org/educate/applab.

Reflecting On the Power of Computational Thinking

When you did your Hour of Code, you experienced the manner in which computer scientists organize their code into named procedures that can be called upon to perform their tasks whenever they are needed. In the Star Wars environment, you were using a library of Star Wars procedures that provided you with the capability to add characters, adjust speeds, select maps, and change backgrounds to increase the functionality of your code. You were able to build upon those objects and extend them in the making of your own game.

Being able to organize code into named procedures enables the world to leverage the power of computational thinking by creating libraries of code that can be used in virtually every discipline. To help you think about how broadly applicable this kind of computational thinking can be, follow this YouTube link to [Chris Bosch explaining how basketball players think algorithmically](#) about making plays and giving them names. Notice how Bosch compares lines of code (computer actions) to moves on the basketball court (dribbling and passing) which can be combined to form plays (such as the pick-and-roll) that get named and put into the team's library of available plays that players and coaches and even other teams can call upon at will. In your current or planned future field of work, how do you see this kind of computational thinking impacting how you might think about your work in terms of these kinds of layers of structure guiding your activity? For a remarkable example of how this kind of thinking can save lives, view this YouTube video about [data and medicine](#).