

# Cover Page

Content Area: **Sample Content Area**  
Course(s): **Computer Science (IS)**  
Time Period: **1st Marking Period**  
Length: **45 days**  
Status: **Not Published**

## **Title Page, Table of Contents, Statement of Purpose**

---

**Sayreville Public Schools Curriculum**

**Computer Science**

**7th Grade**

### **Table of Contents**

Unit 1: Computers: Software and Hardware

Unit 2: Problem Solving and Computing

Unit 3: Web Development

Unit 4: Interactive Animations and Games

### **Statement of Purpose**

Summary of the Course: Computer Science Grade 7 is a course designed for general education students and special education students to work collaboratively completing projects and assignments utilizing Code.org. Computer Science Discoveries (CS Discoveries) is an introductory computer science course that empowers students to create authentic artifacts and engage with computer science as a medium for creativity, communication, problem-solving, and fun.

In order to demonstrate a cohesive and complete implementation plan the following general suggestions are provided:

- Homework, when applicable
- Differentiated Instruction
- Varied assessments
- Modifications to the curriculum should be included that address students with Individualized Educational Plans (IEP), English Language Learners (ELL), those requiring other modifications (504 plans), as well as extension exercises for gifted learners.



# Unit 1 Computers Software and Hardware

Content Area: **Sample Content Area**  
Course(s): **Computer Science (IS)**  
Time Period: **1st Marking Period**  
Length: **1 day**  
Status: **Not Published**

## **Summary of the Unit**

---

Within this unit, students will learn that software and hardware determine a computing system's capability to store and process information. The design or selection of a computing system involves multiple considerations and potential trade-offs. Students will work complete a note-taking activity to learn about the different functions of the hardware and software that makes up the computers they use.

## **Enduring Understanding**

---

Students will distinguish between computer hardware and software

Students will discuss the purpose of each hardware part

Students will identify internal and external computer hardware parts

## **Essential Questions**

---

- What are the terms used to define storage?
- What are the characteristics of a hard disk?
- What are the differences between storage options such as hard drives, SSDs, memory cards, and cloud storage?
- What are the primary characteristics of an Operating System?
- What is the difference between firmware and operating systems?

## **Summative Assessment/Summative Criteria**

---

Note-taking document

- Students will view a teacher-made presentation on parts of a computer. Students will take notes and complete a processing activity

Edpuzzle Questions

- Students will view an Edpuzzle video with questioning

## Resources

---

Note-taking document

Google Slides Notes

Edpuzzle video

## Suggested Timeline

---

Lesson	Timeline	Idea
1- Computers: Software and Hardware	1 day	Students will view a teacher made presentation on parts of a computer. Students will take notes and complete a processing activity.
	1 Day	
		Software and hardware determine a computing system's capability to store and process information. The design or selection of a computing system involves multiple considerations and potential trade-offs.
CS.6-8.8.1.8.CS.2		Design a system that combines hardware and software components to process data.
CS.6-8.8.1.8.CS.3		Justify design decisions and explain potential system trade-offs.
		Integration of Knowledge and Ideas
LA.RL.7.7		Compare and contrast a written story, drama, or poem to its audio, filmed, staged, or multimedia version, analyzing the effects of techniques unique to each medium (e.g., lighting, sound, color, or camera focus and angles in a film).

## Suggested Modifications for Special Education, ESL and Gifted

---

- Below-level learners can be provided with graphic organizers, vocabulary cards, study guides, printed notes, and leveled readers.
- Structure lessons around questions that are authentic, relate to students' interests, social/family background and knowledge of their community.
- Provide students with multiple choices for how they can represent their understandings (e.g.

multisensory techniques-auditory/visual aids; pictures, illustrations, graphs, charts, data tables, multimedia, modeling).

- Teacher will follow IEP/504 plans, as well as make accommodations for ELL students after collaborating with ELL instructors.
- Provide multiple grouping opportunities for students to share their ideas and to encourage work among various backgrounds and cultures (e.g. multiple representation and multimodal experiences).
- Provide ELL students with multiple literacy strategies including websites with various language options.
- Consistent with individual plans, when appropriate.
- The use of manipulatives and hands on activities that incorporate multi-sensory learning.
- Graphic Organizers to promote UDL learning for all students in the classroom.
- Visuals that model exceptional behavior in situations similar to the ones the students will face on a daily basis.
- Modifications and accommodations that align with each individual IEP.
- The use of Visual aids for students who are visually impaired.
- Technology devices that better enhance lessons and increase engagement.
- Students will be grouped with peers of different abilities to draw out different strengths
- Breakdown of assignments to make the end goal more attainable
- Differentiated worksheets and materials for each diverse learner

### **Cross-Curricular/21st Century Connection**

---

LA.K-12.3.2 All Students will listen actively in a variety of situations to information from a variety of sources.

LA.K-12.3.2.2 Demonstrate comprehension of another's message through appropriate verbal or nonverbal responses.

### **Suggested Technological Innovations**

---

Students will be using Chromebooks to complete the note-taking task on Google Docs.

# Unit 2 Problem Solving and Computing

Content Area: **Sample Content Area**  
Course(s): **Computer Science (IS)**  
Time Period: **1st Marking Period**  
Length: **9 Days**  
Status: **Not Published**

## Summary of the Unit

---

Problem Solving and Computing is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. Students will practice using a problem-solving process to address a series of puzzles, challenges, and real-world scenarios. Students will learn how computers input, output, store, and process information to help humans solve problems. Students will complete a project in which they design an application that helps solve a problem of their choosing.

## Enduring Understanding

---

Students will create a maze with step-by-step instructions to show the line of code needed to solve a problem.

Students will complete activities using Code.org to learn about input/output and processing.

## Essential Questions

---

- What strategies and processes can I use to become a more effective problem solver?
- How do computers help people to solve problems?
- How do people and computers approach problems differently?
- What does a computer need from people in order to solve problems effectively?

## Summative Assessment/Criteria

---

- Paper Maze
  - Students will be creating a maze with written step-by-step instructions. Students will swap their mazes with other students and they must complete the maze by only using the written directions. Students will answer discussion questions after the activity.
- Line Up Blind

- Students will use the idea of the Design Process to line up and organize themselves as a class with different guidelines and criteria (alphabetically, age, etc)
- Word Search/Birthday Party, Animal and Games Theme
  - In this lesson, students apply the problem-solving process to three different problems in order to better understand the value of each step.
- Input/Output Activity
  - In this lesson, students consider how computers get and give information to the user through inputs and outputs by utilizing a lesson on Code.org
- Apps with Processing Activity
  - This lesson introduces students to four common types of processing: if/then (conditionals), finding a match (searching), counting, and comparing and students have to determine which type of processing an app is using.
- Apps with Storage Activity
  - This lesson introduces the final component of the unit's model of computing: storage through a storage worksheet

## Resources

---

Taco Party Video and Note-taking

Code.Org Activities

Logic Puzzles

## Unit Plan

---

Lesson	Timeline	Idea	Activities	Resources	Standards
1- Introduction to Problem Solving	1 Day	Students will be creating a maze with written step-by-step instructions. Students will swap their mazes with other students and they must complete the maze by only using the written directions. Students will answer discussion questions after the activity.	Paper Maze	<a href="#">Paper Maze</a>	<a href="#">8.2.8.ED.2:</a> - <a href="#">8.2.8.ED.3</a> <a href="#">8.2.8.ED.4</a> LA.RL7.4 LA.RL7.5 LA.RL7.7

				<a href="#">Video</a>	<a href="#">8.2.8.ED.2:</a>
					-
2- The Problem Solving Process	1 Day	This lesson introduces the formal problem-solving process that students will use over the course of the year, Define - Prepare - Try - Reflect. For a Do Now, students will “Line up Blind”	The Problem Solving Process	<a href="#">Design Process Activity</a>	<a href="#">8.2.8.ED.3</a> <a href="#">8.2.8.ED.4</a>
			Line up blind!	<a href="#">Taco Party</a>	LA.RL7.4 LA.RL7.5
				<a href="#">Line up Blind</a>	LA.RL7.7 LA.RL7.9
			Word Search	<a href="#">Word Search/Party/Redesign</a>	<a href="#">8.2.8.ED.5</a>
			Birthday Party	<a href="#">Animal Themes</a>	<a href="#">8.2.8.ED.6</a>
3- Exploring Problem Solving	4 days	In this lesson, students apply the problem-solving process to three different problems in order to better understand the value of each step.	Redesign Classroom		LA.RL7.4 LA.RL7.5
			Animal Theme	<a href="#">Game Themes</a>	LA.RL7.7 LA.RL7.9
			Game Theme	<a href="#">Logic Puzzles</a>	<a href="#">8.1.8.CS.4</a>
4- Input and Output	1 Day	In this lesson, students consider how computers get and give information to the user through inputs and outputs.	Input and Output Activity	<a href="#">Input and Output Activity</a>	LA.RL7.4 LA.RL7.5
					LA.RL7.7 LA.RL7.9 <a href="#">8.1.8.AP.1</a>
					<a href="#">8.1.8.AP.8</a>
5- Processing	1 day	This lesson introduces students to four common types of processing: if/then (conditionals), finding a match (searching), counting, and comparing.	Apps with Processing Activity	<a href="#">Apps with Processing Activity</a>	<a href="#">8.1.8.CS.2</a> LA.RL7.4
					LA.RL7.5



6- Storage	1 day	This lesson introduces the final component of the unit's model of computing: storage.	Apps with Storage	<a href="#">Apps with Storage</a>	LA.RL7.4 LA.RL7.5 LA.RL7.7 LA.RL7.9
	9 days				

LA.RL.7.9	Compare, contrast and reflect on (e.g., practical knowledge, historical/cultural context, and background knowledge) a fictional portrayal of a time, place, or character and a historical account of the same period as a means of understanding how authors of fiction use or alter history.
CS.6-8.8.1.8.AP.8	Systematically test and refine programs using a range of test cases and users.
CS.6-8.8.2.8.ED.3	Develop a proposal for a solution to a real-world problem that includes a model (e.g., physical prototype, graphical/technical sketch).
CS.6-8.8.2.8.ED.4	Investigate a malfunctioning system, identify its impact, and explain the step-by-step process used to troubleshoot, evaluate, and test options to repair the product in a collaborative team.
LA.RI.7.4	Determine the meaning of words and phrases as they are used in a text, including figurative, connotative, and technical meanings; analyze the impact of a specific word choice on meaning and tone.
CS.6-8.8.1.8.IC.1	Compare the trade-offs associated with computing technologies that affect individual's everyday activities and career options.
CS.6-8.8.2.8.ED.5	Explain the need for optimization in a design process.
LA.SL.7.5	Include multimedia components and visual displays in presentations to clarify claims and findings and emphasize salient points.
CS.6-8.8.2.8.ED.2	Identify the steps in the design process that could be used to solve a problem.
CS.6-8.8.1.8.CS.2	Design a system that combines hardware and software components to process data.
CS.6-8.8.2.8.ED.6	Analyze how trade-offs can impact the design of a product.
CS.6-8.8.1.8.CS.4	Systematically apply troubleshooting strategies to identify and resolve hardware and software problems in computing systems.
LA.RL.7.7	Compare and contrast a written story, drama, or poem to its audio, filmed, staged, or multimedia version, analyzing the effects of techniques unique to each medium (e.g., lighting, sound, color, or camera focus and angles in a film).
CS.6-8.8.1.8.AP.1	Design and illustrate algorithms that solve complex problems using flowcharts and/or pseudocode.

## Suggested Modifications for Special Education, ESL and Gifted

---

- Below-level learners can be provided with graphic organizers, vocabulary cards, study guides, printed notes, and leveled readers. Projects can be modified or leveled as needed.
- Restructure lesson using UDL principles; Structure lessons around questions that are authentic, relate to students' interests, social/family background and knowledge of their community.
- Provide students with multiple choices for how they can represent their understandings (e.g. multisensory techniques-auditory/visual aids; pictures, illustrations, graphs, charts, data tables, multimedia, modeling).
- Teacher will follow IEP/504 plans, as well as make accommodations for ELL students after collaborating with ELL instructors.
- Provide multiple grouping opportunities for students to share their ideas and to encourage work among various backgrounds and cultures (e.g. multiple representation and multimodal experiences).
- Provide ELL students with multiple literacy strategies including websites with various language options.
- Consistent with individual plans, when appropriate.
- The use of manipulatives and hands on activities that incorporate multi-sensory learning.
- Graphic Organizers to promote UDL learning for all students in the classroom.
- Visuals that model exceptional behavior in situations similar to the ones the students will face on a daily basis.
- Modifications and accommodations that align with each individual IEP.
- The use of Visual aids for students who are visually impaired.
- Technology devices that better enhance lessons and increase engagement.

## Cross Curricular/21 Century Connections

---

TECH.9.4.8.CI.3	0x	[Performance Expectation] - Examine challenges that may exist in the adoption of new ideas (e.g., 2.1.8.SSH, 6.1.8.CivicsPD.2).
TECH.9.4.8.CT.2		[Performance Expectation] - Develop multiple solutions to a problem and evaluate short- and long-term effects to determine the most plausible option
CCSS.ELA-Literacy.RST.6-8.7	2x	[Grade Level Standard] - Integrate quantitative or technical information expressed in words in a text with a version of that information expressed visually (e.g., in a flowchart, diagram, model, graph, or table).
CCSS.ELA-Literacy.RST.6-8.9	0x	[Grade Level Standard] - Compare and contrast the information gained from experiments, simulations, video, or multimedia sources with that gained from reading a text on the same topic.

## Technological Innovations

---

Students will be utilizing Chromebooks and Code.org to work through several problems in computing and problem-solving.



# Unit 3 Web Development

Content Area: **Sample Content Area**  
Course(s): **Computer Science (IS)**  
Time Period: **1st Marking Period**  
Length: **28 Days**  
Status: **Not Published**

## Summary of the Unit

---

In Web Development, students are empowered to create and share content on their own web pages. They begin by thinking about the role of the web and how it can be used as a medium for creative expression. As students develop their pages and begin to see themselves as programmers, they are encouraged to think critically about the impact of sharing information online and how to be more critical consumers of content. They are also introduced to problem-solving as it relates to programming while they learn valuable skills such as debugging, using resources, and teamwork. At the conclusion of the unit, students will have created a personal website they can publish and share.

## Enduring Understanding

---

Throughout the unit, students use their developing skills to create a multi-page website and have several opportunities to share out and engage in peer review at the end of each chapter. These projects emphasize many of the core practices of this course as students will need to tap into their creativity, problem-solving skills, and persistence to complete their websites.

## Essential Questions

---

- Why do people create websites?
- How can text communicate content and structure on a web page?
- How do I safely and appropriately make use of the content published on the internet?
- What strategies can I use when coding to find and fix issues?
- How can websites be used to address problems in the world?
- What strategies can teams use to work better together?
- How do I know what information can be trusted online?

## Summative Assessments/Criteria

---

Personal Webpage

Website for a Purpose

Code.org activities

## Resources

---

Code.org activities

WebLab

## Unit Plan

---

Lesson	Timeline	Idea	Activities	Resources	Standards
				<a href="#">Slides Presentation</a>	
1- Exploring Web Pages	1 day	In this lesson, students will start to consider the purposes a website might serve, both for the users and the creators.	Activity Guide	<a href="#">Code Journaling</a>	<a href="#">8.1.8.IC.1</a>
				<a href="#">Activity Guide</a>	
2- Intro to HTML	1 day	In this lesson students are introduced to HTML as a solution to the problem of how to communicate both the content and structure of a website to a computer.	Web Lab Practice	<a href="#">Activities on code.org</a>	<a href="#">8.1.8.AP.4</a>
3- Headings	1 day	In this lesson, students continue to use HTML to structure text on web pages, this time in pairs, with a focus on working together and debugging problems with their sites.	Headings Practice	<a href="#">Activities on Code.org</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.4</a>
4- HTML Webpage (mini page)	1 day	In this optional mini-project, students use what they have learned to create their own personal web page on a topic of their choice.		<a href="#">Web work on code studio</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.9</a>
5- Digital Footprint	1 day	This lesson takes a step back from students' work developing web pages to help them articulate what	Social Sleuth	<a href="#">Social Sleuth</a>	<a href="#">8.1.8.IC.1</a>

		personal information they choose to share digitally and with whom.			<a href="#">8.1.8.IC.2.</a>
6- Styling Text with CSS	1 day	Students learn the basic syntax for CSS rule-sets and then explore properties that impact HTML text elements.	WebLab intro to CSS	<a href="#">WebLab</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.9</a>
7- Mini Project Your Personal Style	1 day	Starting with a discussion of their personal opinions on how others should be allowed to use their work, the class explores the purpose and role of copyright for both creators and users of creative content.	Mini Project	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.9</a>
8- Intellectual Property	1 day	Students need to understand the rules governing how to legally use content they find on the web	Creative Common Solutions	<a href="#">Licensing Your Work</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.IC.1</a>
9- Using Images	1 day	This lesson introduces images, which are different from earlier tags in that they are 'self-closing' and include attributes.	Using Images	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.IC.1</a> <a href="#">8.1.8.AP.4</a>
10- Websites for Expression	1 day	This lesson emphasizes that web development, and by extension computer science, is an avenue for self expression.	Define Your Webpage	<a href="#">Define webpage</a>	<a href="#">8.2.8.ITH.1</a> <a href="#">8.2.8.ITH.2</a> <a href="#">8.2.8.ITH.5</a>
11- Styling Elements with CSS	1 day	This lesson builds on what students previously learned about CSS properties, this time looking at properties that can be used on elements.	Web Labs	<a href="#">Web Labs</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.9</a>
12- Your Webpage	1 day	This lesson reinforces the idea that students should design and plan their pages before they start coding.	Prepare Webpage	<a href="#">Prepare</a> <a href="#">Web Development Video</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.9</a>
			Personal Websites	<a href="#">Computer Science Practices</a>	<a href="#">8.1.8.AP.6</a>
13- Personal Webpage	4 days	Students get to create and share it with the rest of the world.	Peer Review	<a href="#">Peer Review</a> <a href="#">Rubric</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>
			Creator Response	<a href="#">Project Guide</a>	
				<a href="#">Checklist</a>	

14- Websites for a purpose	1 day	This lesson transitions students from thinking about websites as a form of personal expression to considering how websites might serve broader purposes for their creators.	Activity Guide	<a href="#">Purpose activity guide</a>	<a href="#">8.1.8.IC.1</a>
				<a href="#">Dealing with Disagreements</a>	
15- Team Problem Solving	1 day	This lesson explicitly addresses the challenges students may find working in a group and supports them in crafting a plan to overcome these challenges.	2 videos		<a href="#">8.1.8.AP.6</a>
			Team Plan	<a href="#">How Teamwork Works</a>	<a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.IC.1</a>
			Scavenger Hunt	<a href="#">Team Plan How Search Works</a>	
16- Sources and Research	1 day	As students are finishing up their own websites, this lesson encourages them to also think about their responsibilities as consumers of information.	Activity Guide	<a href="#">Internet Scavenger Hunt</a>	<a href="#">8.1.8.IC.1</a> <a href="#">8.1.8.IC.2</a>
			Video	<a href="#">Links and Research</a>	<a href="#">8.1.8.AP.7</a>
17- CSS Classes	1 day	This means students can single out a certain element they want to style or group together elements from one or more types of elements.	Code Studio	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>
				<a href="#">Prepare for a purpose</a>	<a href="#">8.1.8.AP.5</a>
18- Planning a Multi-page site	2 days	This lesson gives students a chance to plan as a team. They will need to make many group decisions and have effective ways of resolving disagreements.	Prepare for a purpose		<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.6</a> <a href="#">8.1.8.AP.9</a>
				<a href="#">Rubric</a>	
				<a href="#">Checklist</a>	<a href="#">8.2.8.EC.1</a> <a href="#">8.1.8.AP.6</a>
19- Linking Pages	2 days	This lesson will give students practice in using links and introduce them to good navigation practices for their sites.	Code Studio	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>

20- Project Website for 2 days a Purpose	This project emphasizes many of the core practices of this course.	Project	<a href="#">Rubric</a>	<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
			<a href="#">Checklist</a>	<a href="#">8.1.8.AP.8</a>
			<a href="#">Reflection</a>	<a href="#">8.1.8.AP.9</a>
<a href="#">Rubric</a>				
21- Peer Review and Final Touches	Peer review encourages students to leverage their peers as resources and develop effective communications skills.	Project	<a href="#">Peer Review</a>	<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
			<a href="#">Project Guide</a>	<a href="#">8.1.8.AP.9</a>
<a href="#">Student Checklist</a>				

### Cross Curricular/21st Century Connection

AAAA.K-12.1.2	0x	[Cumulative Progress Indicator] - Identify personal goals, interests, preferences and abilities related to work activities.
AAAA.K-12.2.1	4x	[Cumulative Progress Indicator] - Access and use technology-based materials for communication, solving problems, seeking information and other everyday life activities.
LA.W.6.2.D	3x	Use precise language and domain-specific vocabulary to inform about or explain the topic.
TECH.9.4.8.DC.1	0x	[Performance Expectation] - Analyze the resource citations in online materials for proper use.

### Technological Innovations

Students will be using Chromebooks and accessing Code.org to create web pages. They will be utilizing Google Forms to survey peers about the content on the web pages.



## **Suggested Modifications for Special Education, ELL and Gifted Students**

---

Below-level learners can be provided with graphic organizers, vocabulary cards, study guides, printed notes, and leveled readers. Projects can be modified or leveled as needed.

Restructure lesson using UDL principles

Structure lessons around questions that are authentic and relate to students' interests, social/family background, and knowledge of their community.

Provide students with multiple choices for how they can represent their understandings (e.g. multisensory techniques-auditory/visual aids; pictures, illustrations, graphs, charts, data tables, multimedia, modeling).

The teacher will follow IEP/504 plans, as well as make accommodations for ELL students after collaborating with ELL instructors.

Provide multiple grouping opportunities for students to share their ideas and encourage work among various backgrounds and cultures (e.g. multiple representation and multimodal experiences).

Provide ELL students with multiple literacy strategies including websites with various language options.

Consistent with individual plans, when appropriate.

The use of manipulatives and hands-on activities that incorporate multi-sensory learning.

Graphic Organizers to promote UDL learning for all students in the classroom.

Visuals that model exceptional behavior in situations similar to the ones the students will face on a daily basis.

Modifications and accommodations that align with each individual IEP.

The use of Visual aids for students who are visually impaired.

Technology devices that better enhance lessons and increase engagement.

# Unit 4 Interactive Animations and Games

Content Area: **Sample Content Area**  
Course(s): **Computer Science (IS)**  
Time Period: **1st Marking Period**  
Length: **33 Days**  
Status: **Not Published**

## Summary of the Unit

---

In the Interactive Animations and Games unit, students build on their coding experience as they create programmatic images, animations, interactive art, and games. Starting off with simple, primitive shapes and building up to more sophisticated sprite-based games, students become familiar with the programming concepts and the design process computer scientists use daily. They then learn how these simpler constructs can be combined to create more complex programs. In the final project, students develop a personalized, interactive program. Along the way, they practice design, testing, and iteration, as they come to see that failure and debugging are an expected and valuable part of the programming process.

## Enduring Understanding

---

By the end of the unit, students should be able to create an interactive animation or game that includes basic programming concepts such as control structures, variables, user input, and randomness. They should manage this task by working with others to break it down using objects (sprites) and functions. Throughout the process, they should give and respond constructively to peer feedback, and work with their teammates to complete a project. Students should leave this unit viewing themselves as computer programmers, and see programming as a fun and creative form of expression.

## Essential Questions

---

- What is a computer program?
- What are the core features of most programming languages?
- How does programming enable creativity and individual expression?
- What practices and strategies will help me as I write programs?
- How do software developers manage complexity and scale?
- How can programs be organized so that common problems only need to be solved once?
- How can I build on previous solutions to create even more complex behavior?

## Summative Assessment/Criteria

---

Interactive Card

Design a Game

Code.org activities

There are two major projects in this unit, which are at the end of each chapter on Code.org. Both offer students an opportunity to demonstrate what they've learned while leveraging creativity and peer feedback. To learn more about these projects, check out the descriptions in this unit's lesson progression.

## Resources

---

Game Lab

Code.org

## Unit Plan

---

Lesson	Timeline	Idea	Activities	Resources	<a href="#">Standards</a>
1- Programming for Entertainment	1 day	This lesson is intended to kick off this programming unit in a way that engages students of all backgrounds and interests.	Activity	<a href="#">Activity Guide</a>	<a href="#">8.1.8.IC.2</a>
2- Plotting Shapes	1 day	The primary purpose of this lesson is to introduce students to the coordinate system they will use in Game Lab.	Drawing with a Computer	<a href="#">Drawing Shapes A</a>	<a href="#">8.1.8.AP.1.</a> <a href="#">8.1.8.AP.4</a>
3- Drawing in Game Lab	1 day	The main purpose of this lesson is to give students a chance to get used to the programming environment, as well as the basic sequencing and debugging that they will use throughout the unit.	Drawing in Game Lab	<a href="#">Drawing Shapes B</a> <a href="#">Video 1</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.9</a>
				<a href="#">Video 2</a>	

4- Shapes and Parameters	1 day	This lesson gives students a chance to slightly expand their drawing skills while continuing to develop general purpose programming skills.	Programming with parameters	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
5- Variables	1 day	Students should leave this lesson knowing that variables are a way to label a value in their programs so that they can be reused or referenced later.	Video Code Studio	<a href="#">Intro Video</a> <a href="#">Code Studio</a>	<a href="#">8.1.8.AP.4.</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
6- Random Numbers	1 day	This lesson introduces randomness, which is important both as a way to make programs more interesting, but also to motivate the use of variables.	Code Studio	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
7-Sprites	1 day	Keeping track of many shapes and the different variables that control aspects of those shapes can get very complex.	Code Studio	<a href="#">Sprites video</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
8-Sprite Properties	1 Day	This lesson starts to dig into what makes sprites such a powerful programming construct--that they have properties that can be modified as a program is running.	Code Studio	<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.4.</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.3</a>
9- Text	1 day	This lesson introduces text, which students will need as they begin to build more complex programs (e.g. games with scoreboards).	Code Studio	<a href="#">Code Studio</a> <a href="#">Rubric</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>
10-Mini Project Captioned Screens	1 day	This lesson is a chance for students to get more creative with what they have learned.	Code Studio Planning	<a href="#">Planning</a>	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>
11-The Draw Loop	1 day	Students should leave the lesson understanding that the commands in the draw loop are called	Code Studio	<a href="#">Code Studio</a> <a href="#">Video</a>	<a href="#">8.1.8.AP.2</a>

		after all other code but are then called repeatedly to create animation.		<a href="#">8.1.8.AP.3</a>
			<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.4</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
			<a href="#">Video</a>	<a href="#">8.1.8.AP.2</a>
12-Sprite Movement	1 day	This lesson builds on the draw loop that students learned previously to create programs with purposeful motion.	Code Studio	<a href="#">8.1.8.AP.3</a> <a href="#">8.1.8.AP.4</a>
			<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
			<a href="#">Code Studio</a>	<a href="#">8.1.8.AP.2</a>
13- Mini Project Animation	1 day	Some students may spend more time in the animation tab drawing than programming.	Code Studio	<a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a>
			<a href="#">Rubric</a>	
			<a href="#">Activity Guide</a>	
14- Conditionals	1 day	As before, we start with using booleans directly before using booleans to trigger if statements. In the following lesson we will introduce some boolean producing blocks, such as <a href="#">keyDown()</a> , which can be used in place of simple boolean comparisons to write programs that respond to user input.	Code Studio	<a href="#">8.1.8.AP.3</a> <a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
			<a href="#">Boolean Expressions</a>	
			<a href="#">Conditional Statements</a>	
15- Keyboard Input	1 day	Therefore the introduction of conditionals and user inputs for decision making is a critical step toward creating games.	Code Studio	<a href="#">8.1.8.AP.3</a> <a href="#">8.1.8.AP.7</a> <a href="#">8.1.8.AP.8</a> <a href="#">8.1.8.AP.9</a> <a href="#">8.1.8.AP.2</a>
16-Mouse Input	1 day	Else statements are a second statement which is attached to an if statement. Else statements execute when the conditions they are attached to are false.	Code Studio	<a href="#">8.1.8.AP.3</a> <a href="#">8.1.8.AP.4</a> <a href="#">8.1.8.AP.7</a>
			<a href="#">If/Else Statements</a>	

					<a href="#">8.1.8.AP.8</a>
					<a href="#">8.1.8.AP.9</a>
				<a href="#">Reflection</a>	
					<a href="#">8.1.8.AP.2</a>
				<a href="#">Project Guide</a>	<a href="#">8.1.8.AP.3</a>
					<a href="#">8.1.8.AP.4</a>
17- Project-Interactive Card	4 days	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.	Project	<a href="#">Peer Review</a>	<a href="#">8.1.8.AP.6</a>
					<a href="#">8.1.8.AP.7.</a>
					<a href="#">8.1.8.AP.8</a>
				<a href="#">Rubric</a>	<a href="#">8.1.8.AP.9</a>
				<a href="#">Checklist</a>	
					<a href="#">8.1.8.AP.2</a>
					<a href="#">8.1.8.AP.3</a>
18- Velocity	1 day	This lesson launches a major theme of the chapter: that complex behavior can be represented in simpler ways to make it easier to write and reason about code.	Code Studio	<a href="#">Velocity Video</a>	<a href="#">8.1.8.AP.4</a>
					<a href="#">8.1.8.AP.7</a>
					<a href="#">8.1.8.AP.8</a>
					<a href="#">8.1.8.AP.9</a>
					<a href="#">8.1.8.AP.2.</a>
				<a href="#">Activity Guide A</a>	<a href="#">8.1.8.AP.3</a>
19- Collision Detection	1 day	This lesson formally introduces the use of abstractions, simple ways of representing underlying complexity.			<a href="#">8.1.8.AP.4</a>
				<a href="#">Activity Guide B</a>	<a href="#">8.1.8.AP.7</a>
					<a href="#">8.1.8.AP.8</a>
				<a href="#">Rubric</a>	<a href="#">8.1.8.AP.2</a>
20- Mini Project Side Scroller	1 day	Encourage students to spend time on parts of the activity that interest them, as long as they meet the requirements of the assignment.	Mini Project		<a href="#">8.1.8.AP.3</a>
				<a href="#">Project Guide</a>	<a href="#">8.1.8.AP.4</a>
					<a href="#">8.1.8.AP.8</a>
21- Complex Sprite Movement	1 day	It demonstrates how combining these tools, in particular the abstractions students learned in the previous two lessons, allows them to build new behaviors for their sprites.	Code Studio		<a href="#">8.1.8.AP.2</a>
					<a href="#">8.1.8.AP.3</a>

				<a href="#">8.1.8.AP.4</a>
				<a href="#">8.1.8.AP.6.</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
				<a href="#">8.1.8.AP.2.</a>
				<a href="#">8.1.8.AP.3</a>
				<a href="#">8.1.8.AP.4</a>
22- Collisions 1 day	This lesson introduces collisions, another useful abstraction that will allow students to manipulate their sprites in entirely new ways.	Code Studio		<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
				<a href="#">8.1.8.AP.2</a>
				<a href="#">8.1.8.AP.3</a>
				<a href="#">8.1.8.AP.4</a>
23- Mini-Project Flyer Game 1 day	This lesson is a chance for students to get more creative with what they have learned.	Mini Project	<a href="#">Rubric</a>	<a href="#">8.1.8.AP.3</a>
				<a href="#">8.1.8.AP.4</a>
			<a href="#">Project Guide</a>	<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
				<a href="#">8.1.8.AP.2</a>
				<a href="#">8.1.8.AP.3</a>
				<a href="#">8.1.8.AP.4</a>
24- Functions 1 day	Students will primarily use functions to break code into logical chunks that are easier to reason about.	Code studio	<a href="#">Functions video</a>	<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
				<a href="#">8.1.8.AP.2</a>
				<a href="#">8.1.8.AP.3.</a>
25- The Game Design Process 1 day	This lesson introduces multi-frame animations, and is the first in a sequence centered around the process of building software.	Defender Game	<a href="#">Defender Game</a>	<a href="#">8.1.8.AP.4</a>
				<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>

				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
				<a href="#">8.1.8.AP.2</a>
				<a href="#">8.1.8.AP.3</a>
				<a href="#">8.1.8.AP.4</a>
26- Using the Game Design 1 day Process	This lesson reviews many of those concepts while introducing them to a structured process that will help them to manage the work.	Planning Platform	<a href="#">Planning Platform Game</a>	<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
				<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
			<a href="#">Reflection</a>	
				<a href="#">8.1.8.AP.1</a>
			<a href="#">Project Guide</a>	<a href="#">8.1.8.AP.2</a>
				<a href="#">8.1.8.AP.3</a>
27- Project Design a Game	4 days	This lesson is the culmination of Unit 3 and provides students an opportunity to build a Game Lab project of their own from the ground up.	<a href="#">Peer Review</a>	<a href="#">8.1.8.AP.4</a>
				<a href="#">8.1.8.AP.6</a>
				<a href="#">8.1.8.AP.7</a>
			<a href="#">Rubric</a>	<a href="#">8.1.8.AP.8</a>
				<a href="#">8.1.8.AP.9</a>
			<a href="#">Checklist</a>	

### **Suggested Modifications for Special Education, ESL and Gifted**

Below-level learners can be provided with graphic organizers, vocabulary cards, study guides, printed notes, and leveled readers. Projects can be modified or leveled as needed.

Restructure lesson using UDL principles

Structure lessons around questions that are authentic and relate to students' interests, social/family background, and knowledge of their community.

Provide students with multiple choices for how they can represent their understandings (e.g. multisensory techniques-auditory/visual aids; pictures, illustrations, graphs, charts, data tables, multimedia, modeling).



The teacher will follow IEP/504 plans, as well as make accommodations for ELL students after collaborating with ELL instructors.

Provide multiple grouping opportunities for students to share their ideas and encourage work among various backgrounds and cultures (e.g. multiple representation and multimodal experiences).

Provide ELL students with multiple literacy strategies including websites with various language options.

Consistent with individual plans, when appropriate.

The use of manipulatives and hands-on activities that incorporate multi-sensory learning.

Graphic Organizers to promote UDL learning for all students in the classroom.

Visuals that model exceptional behavior in situations similar to the ones the students will face on a daily basis.

Modifications and accommodations that align with each individual IEP.

The use of Visual aids for students who are visually impaired.

Technology devices that better enhance lessons and increase engagement.

## Cross Curricular/21st Century Connections

---

<b>CCSS.ELA-Literacy.RST.6-8.9</b>	0x	[ <i>Grade Level Standard</i> ] - Compare and contrast the information gained from experiments, simulations, video, or multimedia sources with that gained from reading a text on the same topic.
<b>CCSS.ELA-Literacy.RST.6-8.10</b>	0x	[ <i>Grade Level Standard</i> ] - By the end of grade 8, read and comprehend science/technical texts in the grades 6–8 text complexity band independently and proficiently.
<b>LA.7.WI.j2</b>	0x	[ <i>Core Content Connector</i> ] - Organize ideas, concepts, and information (using definition, classification, comparison/contrast, and cause/effect).
<b>LA.7.HD.j1</b>	0x	[ <i>Core Content Connector</i> ] - Use information and feedback to refine own thinking.

## Technological Innovations

---

Students will use their Chromebooks and Code.org to create games and animations.