

2022

Elementary Integration Guide

FIFTH GRADE



MISSISSIPPI STATE UNIVERSITY™
CENTER FOR CYBER EDUCATION

Acknowledgements

The following people assisted in the development of this integration guide:

Elementary Task Force

Melissa Atkins, Instruction Technologist, Lamar County School District
Kacy Baggett, Teacher, Rankin County School District
Heather Barry-Fenster, Teacher, Lowndes County School District
Brittany Boatman, Teacher, Houston School District
Kimberly Brammer, Teacher, Pascagoula-Gautier School District
Michelle Carter, Teacher, Picayune School District
Jana Chao, Teacher, Clinton Public School District
Lerenda Dixon, Teacher, McComb School District
Samantha Elizondo, Teacher, Tupelo Public School District
Angie Frazier, Teacher, Rankin County School District
Tammy Hale, Teacher, Tate County School District
Kayla Hathcock, Teacher, Amory School District
Ashley Hawkins, Teacher, Biloxi Public School District
Vicky Johnson, Teacher, Franklin County School District
Dinah Lachney, Teacher, Madison County School District
Ashley Matthews, Administrator, Lowndes County School District
Kayla Moore, Teacher, Enterprise School District
Olivia Moore, Teacher, DeSoto County School District
Beth Neese, Teacher, Western Line School District
Hannah Padgett, Teacher, Tate County School District
Dr. Lee Pambianchi, Administrator, Rankin County School District
Kristen Phillips, Teacher, Oxford School District
Brittney Price, Teacher, South Panola School District
Melissa Sundberg, Teacher, Ocean Springs School District
Melissa Tingle, Teacher, Lauderdale School District
Susie Williams, Curriculum Coordinator, Leland School District

Mississippi Department of Education Team

Wendy Clemons, executive director, Office of Secondary Education
Louella Webster, supervisor for computer science/STEM

Center for Cyber Education Team

Shelly Hollis, Director
Lizzie Brandon, Project Manager
Amanda Taylor, Project Manager

Research and Curriculum Unit Team

Brock Turnipseed, Marketing and Communications Manager
Chris McMillen, Communications Coordinator
Heather Craig, Editor
Will Graves, Project Coordinator

Funding for the development of this guide was provided by:



Introduction

In March 2021, The Mississippi Computer Science and Cyber Education Equality Act ([House Bill 633](#)) was passed requiring all districts to offer computer science content and courses by the 2024-2025 school year. The bill allows for a phased-in approach as listed below:

- 2022-2023: All middle schools offer at least one (1) course in computer science, and 50% of elementary schools offer a minimum of one (1) hour of instruction in computer science each week at each grade level.
- 2023-2024: All elementary schools offer a minimum of one (1) hour of instruction in computer science each week at each grade level, and 50% of high schools offer at least one (1) course in computer science.
- 2024-2025: All schools will offer instruction in computer science.

To make the integration of computer science content as seamless as possible for elementary teachers, a task force of elementary teachers, principals, the Mississippi Department of Education, and the Mississippi State University Center for Cyber Education was formed to write an integration guide for each grade level, kindergarten through fifth grade. These guides provide plans for a minimum of 40, 60-minute lessons covering six computer science topics: coding, robotics, digital literacy, digital citizenship, keyboarding, and unplugged activities.

Each guide contains a breakdown of content by integrated subjects, content by computer science topics, and a calendar/pacing guide. Teachers may choose to start at the beginning and teach each lesson once a week in chronological order or teach the lesson that integrates with another core subject topic at a more relevant time. In addition to a lesson overview and links to required resources, each lesson plan maps to a Mississippi Computer Science Standard and a core subject area standard. A suggestion on how to break the lesson into smaller segments to be covered throughout the week is also provided in the "Time needed" section.

There are several resources available in each integration guide. Some may require the creation of accounts, but all resources referenced are free. The pacing guide notes lessons requiring account creation so teachers can plan ahead. A list of sites used is provided for technology departments to whitelist or unblock. All resources may be used on any internet-capable device, including Chromebooks and tablets.

Resources

Computing resources	<ul style="list-style-type: none">• Code.org CS Fundamentals<ul style="list-style-type: none">◦ Course F—Fifth Grade• Common Sense Digital Media• Scratch Educator Guide
CS4MS resources	<ul style="list-style-type: none">• 2018 Mississippi Computer Science Standards• CS4MS Website
Keyboard practice	<ul style="list-style-type: none">• Astro Bubbles Keyboard Practice• Nitrotype• Typing.com
Teacher/student accounts	<p>If you will be teaching computer science in a subject rotation, please see your content area to determine the accounts you and your students will need.</p> <ul style="list-style-type: none">• Code.org• Common Sense Digital Media• Scratch
Help with this guide	<p>Contact Mississippi State University's Center for Cyber Education: www.tinyurl.com/ccehelpdesk</p>

Contents by Topics

Social Studies

- Week 4: G.5.1, G.5.2, G.5.3—Physical features, Maps
- Week 5: H.5.6—Pre-Columbian civilizations
- Week 6: H.5.2—The New World
- Week 7: H.5.3—Colonization of North America
- Week 8: CI.5.2, CR.5.1—American Revolution
- Week 9: H.5.4—Articles of Confederation
- Week 38: G.5.1, G.5.2, G.5.3—Exploration of Maps

English

- Week 1: RI.5.6—Analyzing Accounts on the Same Topic: Point of View
- Week 2: RI.5.7—Multiple Print Sources
- Week 3: SL.5.1—Comprehension and Collaboration
- Week 10: RL.5.1—Inferences/Text Evidence
- Week 11: RL.5.2—Theme
- Week 12: RL.5.2—Summarizing
- Week 13: RL.5.3—Compare/Contrast Text Features
- Week 14: RL.5.4—Language/Meaning
- Week 15: RL.5.3, RL.5.5—Understanding Literary Structure
- Week 16: RL.5.5, RL.5.6—Point of View
- Week 17: RL.5.7—Analyzing Virtual Elements
- Week 18: RL.5.9—Compare/Contrast Stories in the Same Genre
- Week 19: RI.5.4, RI.5.7—Language/Meaning, Multiple Print Sources
- Week 20: RI.5.2—Main Ideas and Summarizing
- Week 21: RI.5.3—Explain Relationships in Scientific/Technical/Historical Texts
- Week 22: RI.5.4—Unfamiliar Words/Contextual Clues
- Week 23: RI.5.5—Text Structure

Science

- Week 24: L.5.3A—Photosynthesis
- Week 25: L.5.3B—Ecosystems
- Week 26: P.5.5A—Physical Properties
- Week 27: P.5.5B—Mixtures and Solutions
- Week 28: P.5.5C—Physical and Chemical Change
- Week 29: P.5.6—Newton's Laws of Motion
- Week 30: E.5.8A—Astronomy
- Week 39: P.5.6—Newton's Laws of Motion

Math

- Week 31: 5.NF.1, 5.NF.6, 5.NF.7—Fractions
- Week 32: 5.NBT.7—Add and Subtract Decimals
- Week 33: 5.NBT.3—Read Decimals
- Week 34: 5.NBT.4—Place Value
- Week 35: 5.MD.1—Conversions
- Week 36: 5.OA.2—Simple Expressions
- Week 37: 5.G.2—Coordinate Planes
- Week 40: 5.MD.4, 5.MD.5—Measuring Prisms

Contents by Topic

Coding

- Week 4
- Week 5
- Week 6
- Week 7
- Week 8
- Week 9
- Week 10
- Week 11
- Week 12
- Week 13
- Week 14
- Week 15
- Week 16
- Week 17
- Week 18
- Week 19
- Week 20
- Week 22
- Week 23
- Week 24
- Week 25
- Week 26
- Week 27
- Week 28
- Week 29
- Week 30
- Week 31
- Week 32
- Week 33
- Week 34
- Week 35
- Week 36
- Week 37
- Week 38
- Week 39
- Week 40

Digital Citizenship

- Week 1
- Week 2
- Week 3

Digital Literacy

- Week 21

Keyboarding

- Use as a filler to build typing fluency
 - See resources

Robotics

- Week 8
- Week 37
- Week 38
- Week 39
- Week 40

Unplugged

- Week 8
- Week 37
- Week 38
- Week 39
- Week 40

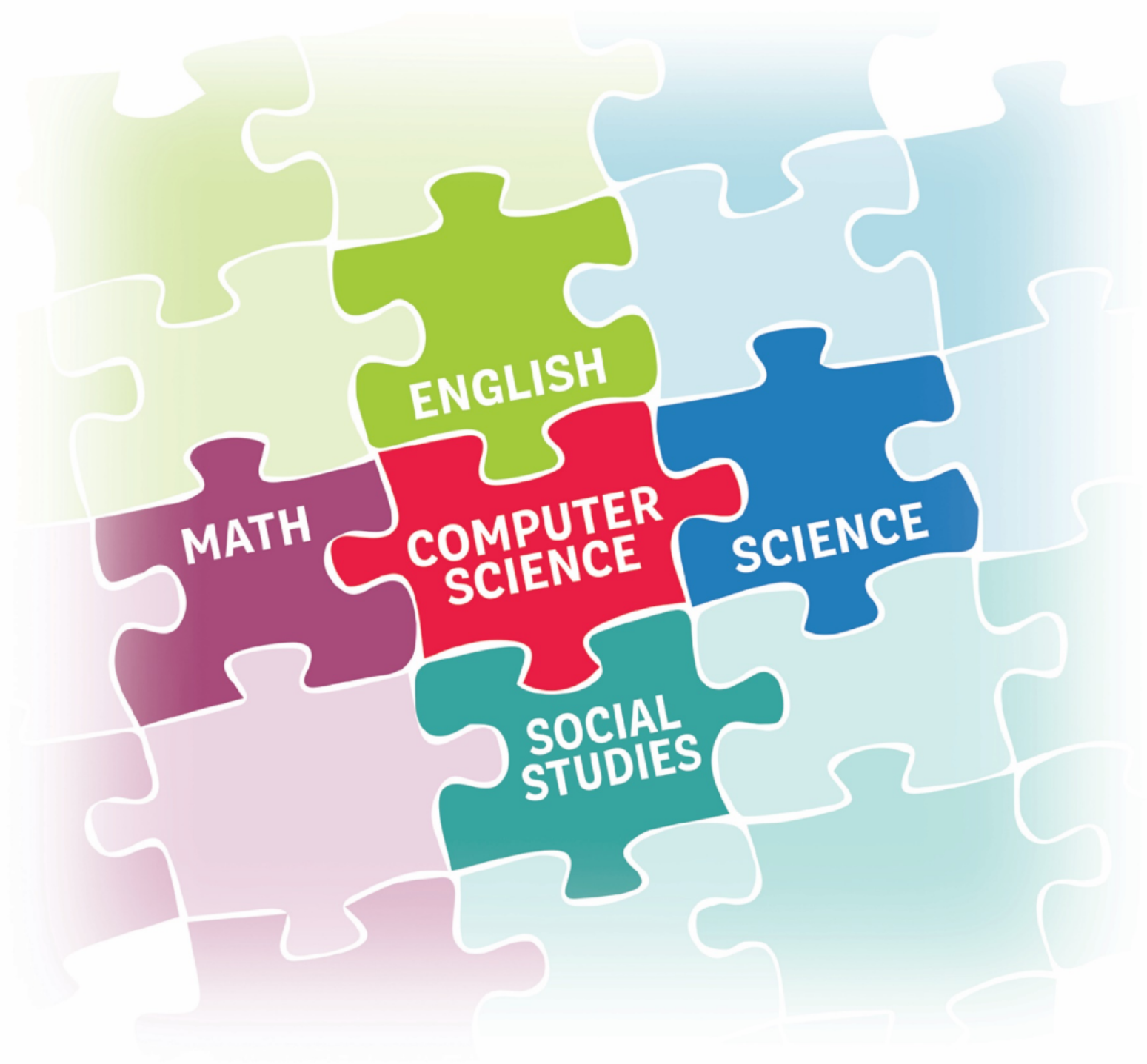
Calendar/Pacing Per Week:

→ Teachers will need to create a FREE **teacher and/or student account** (See the notes section of the lesson.).

Week	Title	Topics	CS Standard	Standard	Subject Integrated
1	Digital Friendships → Account creation needed	Digital Citizenship	NI.1B.2 IC.1B.3a	RI.5.6	ELA
2	Let's Face It	Digital Citizenship	IC.1B.3 IC.1B.3a	RI.5.7	ELA
3	Media Balance	Digital Citizenship	IC.1B.1	SL.5.1	ELA
4	Oceans and Continents Clicker Game → Account creation needed	Coding	AP.1B.2 AP.1B.4	G.5.1 G.5.2 G.5.3	Social Studies
5	Storytelling—Ancient Civilizations of Central and South America → Account creation needed	Coding	AP.1B.3	W.5.2 H.5.6	ELA / Social Studies
6	Imagine a World with European Explorers	Coding	AP.1B.4	W.5.2 H.5.2	ELA / Social Studies
7	13 Colonies Greeting Cards	Coding	AP.1B.9	W.5.2 H.5.3	ELA / Social Studies
8	Road to the Revolutionary War Using Robots	Coding, Robotics, Unplugged	AP.1B.8	H.5.4 H.5.5	Social Studies
9	Come See Our New Nation Advertisement	Coding	AP.1B.3	CI.5.1	Social Studies
10	Surprise Ending Inferences	Coding	DA.1B.2	RL.5.1	ELA
11	The Theme of Gumball	Coding	AP.1B.3	RL.5.2	ELA
12	Story (re)Telling	Coding	AP.1B.6b AP.1B.3	RL.5.2	ELA
13	Code.org: Hello World	Coding	AP.1B.3 AP.1B.4 AP.1B.5	RL.5.3	ELA
14	Figurative Conversations	Coding	AP.1B.3 AP.1B.4	RL.5.4	ELA
15	Create-a-Character	Coding	AP.1B.3 AP.1B.4	RL.5.3 RL.5.5	ELA
16	Two Points of View	Coding	IC.1B.3a AP.1B.3 AP.1B.4 AP.1B.9	RL.5.3 RL.5.5 RL.5.6	ELA

17	Code.org: Analyze Visuals	Coding	AP.1B.3 AP.1B.5 AP.1B.8	RL.5.4 RL.5.7	ELA
18	Adventure on the High Seas	Coding	IC.1B.3a AP.1B.3 AP.1B.9	RL.5.9	ELA
19	Game Design Choices	Coding	IC.1B.2a	RI.5.4 RI.5.7	ELA
20	Scratch-a-Summary	Coding	AP.1B.2	RI.5.2	ELA
21	Reading News Online	Digital Literacy	DA.1B.1	RI.5.3	ELA
22	Escape With Context Clues	Coding	AP.1B.2 AP.1B.5	RI.5.4	ELA
23	Variables and Text Structures	Coding	AP.1B.3a	RI.5.5	ELA
24	Photosynthesis Animation	Coding	DA.1B.1	L.5.3A	Science
25	Code.org: Coding Ecosystems → Account creation needed	Coding	AP.1B.3 AP.1B.4 AP.1B.5	L.5.3B	Science
26	Matter Matters Animation	Coding	AP.1B.3	P.5.5A	Science
27	Mix and Match Mixtures Game	Coding	AP.1B.1 AP.1B.3 AP.1B.5	P.5.5B	Science
28	Catch the Change Game	Coding	AP.1B.3 AP.1B.5	P.5.5C	Science
29	Animating Motion	Coding	AP.1B.3	P.5.6	Science
30	Traveling Home with Polaris Game	Coding	AP.1B.1a AP.1B.3	E.5.8A.4	Science
31	Animating Fractions	Coding	AP.1B.3	5.NF.1 5.NF.6 5.NF.7	Math
32	Animating Decimals	Coding	AP.1B.3	5.NBT.7	Math
33	Place Value Pop-Up	Coding	AP.1B.2	5.NBT.3	Math
34	Digital Roundup	Coding	AP.1B.3	5.NBT.4	Math
35	Creating Conversions	Coding	AP.1B.2 AP.1B.9	5.MD.1	Math
36	Escape the Missing Variable	Coding	AP.1B.2	5.OA.2	Math


37	Plotting Points	Coding Robotics Unplugged	AP.1B.1	5.G.2	Math
38	Oceans and Continents Trivia	Coding Robotics Unplugged	AP.1B.1	5.1.1	Social Studies
39	Newton's Robot Maze	Coding Robotics Unplugged	AP.1B.1	P.5.6	Science
40	Directions Through the Prism	Coding Robotics Unplugged	AP.1B.1	5.MD.4 5.MD.5	Math



Lessons and Activities


FIFTH GRADE

Week 1: Digital Friendships

<p>Lesson overview:</p> 	<p>Purpose: Students will determine similarities and differences between multiple accounts of the same event or topic. They will demonstrate how an author's point of view affects his or her account of an event or topic through learning about cyberbullying and different points of view through digital friendships.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Warm Up: Who Do You Chat With? • Compare and Contrast: Two Scenarios • Evaluate: Benefits and Risks • Wrap Up: Finish the Story
<p>Lesson links/resources:</p>	<p>Digital Friendships Lesson Plan—Common Sense Media Lesson slides Heart's Online Friendships handout—Teacher version Finish the Story handout Lesson quiz</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Compare and contrast different kinds of online-only friendships • Describe the benefits and risks of online-only friendships • Describe how to respond to an online-only friend if the friend asks something that makes them uncomfortable <p>Standards:</p> <ul style="list-style-type: none"> • NI.1B.2—Discuss real-world cybersecurity problems and how personal information can be protected. • IC.1B.3a—Students will collaborate and receive feedback from others.
<p>Time needed:</p>	<p>Total time: 55 min</p> <ul style="list-style-type: none"> • Warm Up: Who Do You Chat With? 5 min • Compare and Contrast: Two Scenarios 20 min • Evaluate: Benefits and Risks 20 min • Wrap Up: Finish the Story 10 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Lesson slides • Heart's Online Friendships handout—Teacher version • Finish the Story handout • Lesson quiz • Common Sense account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Common Sense account
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<p>RI.5.6—Analyze multiple accounts of the same event or topic, noting important similarities and differences in the point of view they represent.</p>

Vocabulary:	<p><u>Benefit</u>: Something positive that results from a situation</p> <p><u>Private information</u>: Information about you that can be used to identify you because it is unique to you (e.g., your full name or your address)</p> <p><u>Risk</u>: Something negative or dangerous that could come from a situation</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.</p>

Week 2: Let's Face It

<p>Lesson overview:</p> 	<p>Purpose: Some online spaces can be full of negative, rude, or downright mean behavior. But what counts as cyberbullying? Students learn what is—and what isn't—cyberbullying and give them the tools they will need to combat the problem.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Warm Up: What is Cyberbullying? • Expand: Who is involved? • Analyze: Sondra's Story • Wrap Up: Upstander Cards
<p>Lesson links/resources:</p>	<p>Is It Cyberbullying? Lesson Plan—Common Sense Media</p> <p>Lesson slides</p> <p>What's Cyberbullying? video</p> <p>Sondra's Story</p> <p>Handout—Teacher version</p> <p>Lesson quiz</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Recognize similarities and differences between in-person bullying, cyberbullying, and being mean • Empathize with the targets of cyberbullying • Identify strategies for dealing with cyberbullying and ways they can be an upstander for those being bullied <p>Standards:</p> <ul style="list-style-type: none"> • IC.1B.3—Seek diverse perspectives for the purpose of improving computational artifacts. • IC.1B.3a—Students will collaborate and receive feedback from others.
<p>Time needed:</p>	<p>Total Time: 50 min</p> <ul style="list-style-type: none"> • Warm Up: What is Cyberbullying? 5 min • Expand: Who is involved? 15 min • Analyze: Sondra's Story 15 min • Wrap Up: Upstander Cards 15 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Lesson slides • What's Cyberbullying? video • Sondra's Story • Handout—Teacher version • Lesson quiz • Common Sense account <p>Students:</p> <ul style="list-style-type: none"> • Blank index cards • Markers and crayons • Common Sense account
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<p>RI.5.7—Draw on information from multiple print or digital sources, demonstrating the ability to locate an answer to a question quickly or to solve a problem efficiently.</p>

Vocabulary:	<p><u>Bully</u>: The person who is doing the bullying</p> <p><u>Bullying</u>: Unwanted and aggressive verbal, social, or physical behavior towards another</p> <p><u>Bystander</u>: Someone who sees a bullying or cyberbullying situation, but doesn't do anything to stop it</p> <p><u>Cyberbullying</u>: Using digital devices, sites, and apps to intimidate, harm, and upset someone</p> <p><u>Empathy</u>: To imagine the feelings that someone else is experiencing</p> <p><u>Target</u>: The person who is on the receiving end of the bullying</p> <p><u>Upstander</u>: A person who supports and stands up for someone else</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media</p>

Week 3: Media Balance

Lesson overview:



Purpose:

Helping kids learn what makes different media choices healthy or not is a good start, but how do we help them actually make responsible choices in the real world? Students will have the opportunity to create a personalized media plan.

Lesson:

- Distribute the [My Perfect Day handout](#) and allow students 10 minutes to complete it. (Slide 4)
- Invite students to share out. For each student who shares, follow up by highlighting any activities involving:
 - Digital media (TV, movies, video games, texting, etc.)
 - Non-digital media (books, magazines, etc.)
 - Friends and/or family
 - The outdoors
 - Alone time
 - Hobbies
- Ask: "Now imagine that your perfect day will be granted—it's going to happen—but it has to happen for a week straight. For seven straight days, you will do the same thing—all the things in your perfect day. Given that, would you change anything in your perfect day? If so, what? Share your ideas with your partner." (Slide 5)
- Invite students to share out. Highlight any changes that result in more balance between the types of activities listed in step 2. For students whose perfect days involve only one or two types of activities, follow up by asking how they'd feel missing out on the other types. For example, how would they feel not seeing their family for a week? Or not having any alone time? Or not spending time outdoors?

Lesson links/resources:

[Finding My Media Balance Lesson Plan](#)- Common Sense Media
[Lesson slides](#)
[My Media Balance video](#)
[My Perfect Day handout](#)
[My Media Plan handout](#)
[Lesson quiz](#)

CS standards addressed:

Students will be able to:

- Reflect on how balanced they are in their daily lives
- Consider what "media balance" means and how it applies to them
- Create a personalized plan for healthy and balanced media use

Standards:

- **1C.1B.1**—Discuss computing technologies that have changed the world and express how those technologies influence and are influenced by cultural practices.

Time needed:

Total time: 50 min

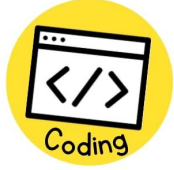
- Warm Up: My Perfect Day **15 min**
- Watch: What is Media Balance? **20 min**
- Create: My Media Plan **15 min**

Materials needed:

Teacher:

	<ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Lesson slides • My Media Balance video • My Perfect Day handout • My Media Plan handout • Lesson quiz • Common Sense account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Common Sense account
Subject integrated:	ELA
Other standards addressed:	SL.5.1 —Engage effectively in a range of collaborative discussions (one-on-one, in groups, and teacher-led) with diverse partners on grade 5 topics and texts, building on others' ideas and expressing their own clearly.
Vocabulary:	<p>Balance: All of the parts are in the correct—though not necessarily equal—proportions</p> <p>Media: All of the ways that large groups of people get and share information (TV, books, internet, newspapers, phones, etc.)</p> <p>Media balance: Using media in a way that feels healthy and in balance with other life activities (family, friends, school, hobbies, etc.)</p>
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media .

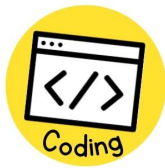
Week 4: Oceans and Continents Clicker Game

<p>Lesson overview:</p> 	<p>Purpose: The student will be able to locate physical features on a map and differentiate between landforms and bodies of water while using Scratch.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • The teacher will review the continents and oceans of the world using a blank map projected onto a screen. • The teacher will introduce the students to Scratch with the “Getting Started” tutorial video and create an account. • The teacher will show the “Make a Clicker Game” tutorial video to the students. • The students will create their own clicker game after watching the tutorials. • The teacher will put students in groups to exchange devices and play each other's games. • The teacher will project a blank map again and have students review continents and oceans.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Scratch Tutorial
<p>CS standards addressed:</p>	<p>The students will be able to:</p> <ul style="list-style-type: none"> • Differentiate between landforms and bodies of water using a Scratch game • Create a sprite for their game • Change the background of a Scratch game • Create a clicker game using Scratch • <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data. • AP.1B.4—Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.
<p>Time needed:</p>	<p>Total time: 70-80 min</p> <ul style="list-style-type: none"> • Continent/ocean review 5 min • Getting Started tutorial 10 min • Make a Clicker Game tutorial 10 min • Create and test a clicker game 15-30 min • Playing others' games to practice finding oceans and continents 15-20 min • Continent/ocean review 5 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Social Studies</p>

Other standards addressed:	<ul style="list-style-type: none"> ● G.5.1—Locate on a map the physical features of America prior to exploration. ● G.5.2—Describe physical features of the environment ● G.5.3—Recognize maps, graphs, and other representations of the Earth.
Vocabulary:	<p><u>Coding</u>: Process of creating instructions for computers using programming language</p> <p><u>Algorithms</u>: A list of steps to complete a task</p> <p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>User</u>: Someone who uses an object, including software and hardware</p> <p><u>Background</u>: Picture underneath the sprite and other objects</p>
Notes:	<p>Other sources to use for map practice if you don't want to use Scratch:</p> <ul style="list-style-type: none"> ● Program a Code and Go Mouse to get from one continent to another crossing specific oceans. <ul style="list-style-type: none"> ○ You will need a large map (at least poster sized if not larger.) <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 5: Storytelling—Ancient Civilizations of Central and South America

Lesson overview:



Purpose:

The students will take the information they have learned about the Aztecs, Incas, and Mayas to create a story as if “walking” through one of the ancient civilizations that they will choose. They will explain important facts about the ancient civilization that they have chosen.

Lesson:

Students watch [videos and create a project](#) in Scratch.

- Videos: Introduction to Scratch. (Use smartboard to show videos in attached [slide](#) or assign the slide to students in google classroom for them to view independently.) *Youtube has some additional instructional videos if needed.
- Demonstrate on smart board how to open the [Dialogue Example Project](#).
 - As students move through these tutorials, they will be setting the scene for the ancient civilization they choose.
- Let the students watch the Introduction to Dialogue and Sequencing tutorial.
- Setting the Scene
 - Students will choose a story starter, add two sprites, make the sprites face each other, add a backdrop, and say a starter phrase. The starter phrase may be a fact about the ancient civilization they have chosen, or an introduction to the civilization (e.g., “This is the ancient Mayan civilization.”).
- Speaking and Responding
 - Students will code their project to have sprites respond to one another discussing facts about civilization.
- Add-on
 - Please review the six options for Add-on to determine which is most beneficial for your students.

[Add a Title Screen](#)

[Costume Animation 1 of 2](#)

[Costume Animation 2 of 2](#)

[Adding Motion \(challenge\)](#)

[Add a Third Character](#)

[Add a Second Scene to your Story \(Challenge\)](#)

Lesson links/resources:

- [Storytelling Lesson Plan Slide](#)

CS standards addressed:

The students will be able to:

- Make a storyboard using [Scratch](#)
- Create a sprite and background using [Scratch](#)

Standards:

- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.

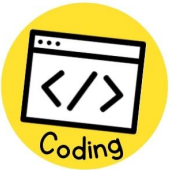
Time needed:

Total time: 70-90 min

- Dialogue [Dialogue Example Project](#) 10 min
- [Example Project](#)

	<ul style="list-style-type: none"> • Setting the Scene 20 min • Speaking and Responding 20 min • Add-on 20 min
Materials needed:	<p>Teachers:</p> <ul style="list-style-type: none"> • Storytelling Lesson Plan Slide • Computer • Projector/smartboard with sound • Scratch account • Additional Storytelling Resources <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
Subject integrated:	<p>ELA</p> <p>Social Studies</p>
Other standards addressed:	<p>ELA</p> <ul style="list-style-type: none"> • W.5.2—Write informative/explanatory texts to examine a topic and convey ideas and information clearly. <p>Social Studies</p> <ul style="list-style-type: none"> • H.5.6—Differentiate among pre-Columbian civilizations.
Vocabulary:	<p><u>Programs</u>: An algorithm that has been coded into something that can be run by a machine</p> <p><u>Sequences</u>: A set of logical steps carried out in order</p> <p><u>Events</u>: Actions that cause something to happen</p> <p><u>Loops</u>: The action of doing things over and over again</p> <p><u>Conditionals</u>: Programming language commands</p>
Notes:	<ul style="list-style-type: none"> • You may want to use student groups for this activity. • You may choose to project the tutorial videos on your smartboard/projector to help guide students. • You can add an Unplugged Story planning time. <ul style="list-style-type: none"> ◦ Use a storyboard to sketch what they want for each part of the story. 15-20 min <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

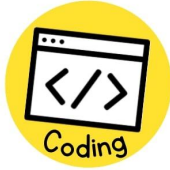
Week 6: Imagine a World With European Explorers

<p>Lesson overview:</p> 	<p>Purpose: The students will imagine a world and create a moving scene using Scratch to show what the New World would have looked like to the European settlers. The students will upload a background picture of the new world and add to the scene to create what the settlers would have seen.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • The teacher will show the Imagine a World tutorial video to the students. • The students will create a written outline of their story. • The students will click create (top left tab on the home page) on Scratch to write a story about the new world through the eyes of the European settlers. <ul style="list-style-type: none"> ◦ The students will choose a background and sprite to represent their character and setting. ◦ The students will use the information on their outline to put into their story. • The students will share their stories with the whole class.
<p>Lesson links/resources:</p>	<p>Scratch Tutorials: Imagine a World</p>
<p>CS standards addressed:</p>	<p>The students will be able to:</p> <ul style="list-style-type: none"> • Describe the New World using Scratch • Create a sprite using Scratch • Create a moving background using Scratch <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.4—Decompose problems into smaller, manageable subproblems to facilitate the program development process.
<p>Time needed:</p>	<p>Total time: 65 min</p> <ul style="list-style-type: none"> • Imagine a World tutorial video to explain how to make a moving story 5 min • Story planning using outline 15-20 min • Create a story on Scratch 30 min • Sharing story with class and or desk partners 10 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch teacher account <p>Students:</p> <ul style="list-style-type: none"> • Computer/ tablet with access to internet • Scratch student accounts
<p>Subject integrated:</p>	<p>Social Studies</p>
<p>Other standards addressed:</p>	<p>ELA</p> <ul style="list-style-type: none"> • W.5.2—Write informative/explanatory texts to examine a topic and convey ideas and information clearly. <p>Social Studies</p> <ul style="list-style-type: none"> • H.5.2—Examine the reasons and impact for exploration of the New World.

Vocabulary:	<u>Program development</u> : Coding of an individual activity <u>Upload</u> : Moving one file from a computer system to another
Notes:	<p>The students will need to include the sprites for animals and people to put with the uploaded photo. Make sure to have your sprites move through the scene.</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 7: 13 Colonies Greeting Cards

Lesson overview:



Purpose:

The students will create a “Wish You Were Here” greeting card from one of the 13 colonies to send back to England with a partner. The post card will include a background picture of the colony/area, along with details of the geography, economy, religion, native civilization, and daily life. Students will explain the colony in a postcard fashion to someone who has never seen it before.

Lesson:

- The teacher will review how to create a background and let students choose a background to represent their colony by using the [Create a Backdrop](#) tutorial (if needed).
- The teacher will review information of the 13 colonies with students and show students how to find information listed above for their colony.
 - This will be information provided by individual teachers.
- The teacher will introduce greeting cards on [Scratch](#).
 - Please see the “Notes” section.
- The students will access [Scratch Create](#) and create a sprite.
 - The teacher can review the [Add a Sprite](#) tutorial if needed.
 - The students will create the greeting card and have the sprite tell the information about the colony.
- The teacher will introduce the voice over for the sprites by using the [Record a Sound tutorial](#).

Lesson links/resources:

- [Scratch Create](#)
- [Create a Backdrop](#)
- [Add a Sprite](#)
- [Record a Sound tutorial](#)

CS standards addressed:

The students will be able to:

- Create a sprite using [Scratch](#)
- Create a greeting card with facts about the colony using [Scratch](#)

Standards:

- **AP.1B.9**—Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.

Time needed:

Total time: 80 min

- Review creating backdrop/students add backdrop **10 min**
- Gather background information for chosen colony and write it down **20 min**
- [Record a Sound tutorial](#) **10 min**
- Introduce students to using voice over for animations **10 min**
- Create postcards using background and sprites including recording of information **30 min**

Materials needed:

Teachers:

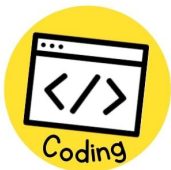
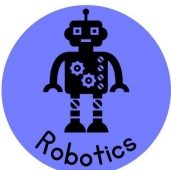

- Computer
- Projector/smartboard with sound
- [Scratch](#) account

Students:

- Computer/tablet with internet access
- [Scratch](#) account

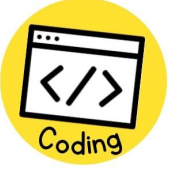
Subject integrated:	ELA Social Studies
Other standards addressed:	ELA <ul style="list-style-type: none"> ● W.5.2—Write informative/explanatory texts to examine a topic and convey ideas and information clearly. Social Studies <ul style="list-style-type: none"> ● H.5.3—Describe reasons for colonization in North America.
Vocabulary:	<u>Design</u> : A way to author computer applications using a combination of text, graphics, and style elements
Notes:	<ul style="list-style-type: none"> ● Here are some examples of greeting cards to share with students: <ul style="list-style-type: none"> ○ Scratch Card Examples ● Here is a YouTube tutorial for Scratch Greeting Cards →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 8: Road to Revolutionary War Using Robots

<p>Lesson overview:</p> <div style="text-align: center;">  <p>Coding</p>  <p>Robotics</p>  <p>Unplugged</p> </div>	<p>Purpose: The students will use a codable robot to see important battle sites of the Revolutionary War.</p> <p>Lesson:</p> <ul style="list-style-type: none"> The teacher will provide information about each battle. <ul style="list-style-type: none"> This will be information provided by individual teachers. The teacher will place a U.S. map on the floor and have the important battles of the war marked on the map. This needs to be a large map that is at least poster sized. <ul style="list-style-type: none"> Students will code the robot to reach each battlefield. Students will write the code on paper to show proof the students have been to each battlefield. Once they reach a battlefield, provide a link for them to go on a virtual field trip of that battle. <u>Unplugged option:</u> If you don't have robots, do it "unplugged" and have students use coding cards to create the code they would use to program a robot to reach each battle site.
<p>Lesson links/resources:</p>	<p>Links to virtual tours of American battlefields</p>
<p>CS standards addressed:</p>	<p>The students will be able to:</p> <ul style="list-style-type: none"> Program a robot to locate various battles of the Revolutionary War <p>Standards:</p> <ul style="list-style-type: none"> AP.1B.8—Test and debug a program algorithm to ensure it runs as intended.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> Explain activity to students 5 min Review coding arrows how to program robots 10 min Programming robots to reach battleground activity 45 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> Large U.S. map Codable robot (Please see the "notes" section.) <p>Students:</p> <ul style="list-style-type: none"> Computer/tablet with internet access
<p>Subject integrated:</p>	<p>Social Studies</p>
<p>Other standards addressed:</p>	<ul style="list-style-type: none"> H.5.4—Explain major events of the American Revolution and their outcomes. H.5.5—Chart the causes and events leading to the American Revolution.
<p>Vocabulary:</p>	<p><u>Debug:</u> Find the exact point where a mistake was made in coding <u>Unplugged:</u> Creating codes without using a computer program</p>
<p>Notes:</p>	<p>Use QRCode Monkey for free QR codes to make it easier for students to access the virtual tour sites.</p>

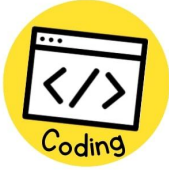
--	--

Week 9: Come See Our New Nation Advertisement

<p>Lesson overview:</p> 	<p>Purpose: The students will create an advertisement for their new country to get people to come visit. Let them tell everyone all the good things about their country using Scratch. They will create a sprite to tell about the country and the new government that has been created. They will choose a background to represent the new country.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • The teacher will review the Articles of Confederation and the U.S. Constitution. • The students will review Scratch if needed, then the teacher will explain the lesson. <ul style="list-style-type: none"> ◦ The teacher will show example advertisements to students. • The students will then create an advertisement for a new nation including the name of the new country, type of government, and interesting information about the country. • The students will also include a background to represent the new country. • Students may present advertisements to class if the teacher chooses.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Examples of advertisements to show students • The American Revolution for Kids - Articles of Confederation • The Constitution of the United States for Kids and Teachers - FREE Lesson Plans & Games for Kids
<p>CS standards addressed:</p>	<p>The students will be able to:</p> <ul style="list-style-type: none"> • Create an advertisement using Scratch describing the new country of America • Create a talking sprite using Scratch • Create a background representing the new America using Scratch <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
<p>Time needed:</p>	<p>Total time: 75 min</p> <ul style="list-style-type: none"> • Review the Articles of Confederation and U.S. Constitution 10 min • Show example advertisements 5 min • Create advertisement for new nation 40 min • Students present advertisements 20 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Social Studies</p>
<p>Other standards addressed:</p>	<p>CI.5.1—Explain how weaknesses of the Articles of Confederation led to the Constitution.</p>

Vocabulary:	<p><u>Coding</u>: Process of creating instructions for computers using programming language</p> <p><u>Algorithms</u>: A list of steps to complete a task</p> <p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>User</u>: Someone who uses an object, including software and hardware</p> <p><u>Background</u>: Picture underneath the sprite and other objects</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

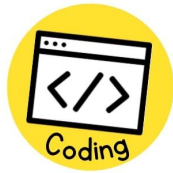
Week 10: Surprise Ending Inferences

<p>Lesson overview:</p> 	<p>Purpose: Students will listen to the beginning of a story provided by the teacher (or created by another student). Students will attempt to infer the object of their partner's story. Students will create a code with a surprise at the end to show their inference about the story.</p> <p>Lesson: Students watch videos and create a project in Scratch.</p> <ul style="list-style-type: none"> • Videos: Introduction to Scratch. (Use smartboard to show videos in attached slide or assign the slide to students in google classroom for them to view independently.) *Youtube has some additional instructional videos if needed. • Demonstrate on smart board how to open the project in Scratch • Assign partners among the students. • Direct the students to log in to their Scratch account. • Students will watch videos and create Scratch account. • Have partners watch the introduction video and decide on the adventure type: Jungle Adventure Basketball Adventure Underwater Adventure Outer Space Adventure • The students will create an adventure to present their inferences. • Instruct students to show their projects to a classmate. • Discuss the lesson and facilitate a brief discussion about what students learned and experienced. <ul style="list-style-type: none"> ○ Tell me about the program you made today. <ul style="list-style-type: none"> ■ <i>What was your favorite part of this lesson?</i> ■ <i>What did you learn about computer science and coding?</i> ■ <i>What was the most challenging part of this lesson?</i>
<p>Lesson links/resources:</p>	<p>Lesson Plan Slide An Unusual Discovery An Unusual Discovery Lesson Plan</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Use event blocks (like “when flag clicked”) to trigger a series of code • Sequence at least three “say” blocks between two sprites (characters) to construct a dialogue • Program a conditional so that the computer can make a decision based on a user response • Produce repeated movements by applying control blocks to their program <p>Standards:</p> <ul style="list-style-type: none"> • DA.1B.2—Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Watch the introduction video 5 min • Open an adventure project link and add characters and dialogue 30 min • Choose Add-ons 15 min

	<ul style="list-style-type: none"> • Share projects 10 min
Materials needed:	<p>Teachers:</p> <ul style="list-style-type: none"> • An Unusual Discovery Lesson Plan • Shoutout sheet (optional) • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Headphones • Scratch account
Subject integrated:	ELA
Other standards addressed:	RL.5.1 —Quote accurately from a text when explaining what the text says explicitly and when drawing inferences from the text.
Vocabulary	<p><u>Control structures</u>: Sections of code that order the direction or flow of how a program functions; Control structures include conditionals and loops</p> <p><u>Debugging</u>: The process of identifying and fixing error(s) in a program when it is not functioning as expected</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 11: The Theme of Gumball

Lesson overview:



Purpose:

The student will create a story using Gumball. The student will then present their story. Their peers will write down what they believe the theme of the story to be.

Lesson:

Direct students to log in to their [Scratch](#) account.

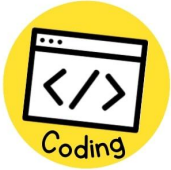
-
- Students will watch videos and create a story using Gumball. Students watch [videos and create a project](#) in Scratch.
 - Videos: Introduction to Scratch. (Use smartboard to show videos in attached [slide](#) or assign the slide to students in google classroom for them to view independently.) *Youtube has some additional instructional videos if needed.
 - Demonstrate on smart board how to open the [project in Scratch](#).
- When there are five minutes left in class, instruct students to watch the wrap-up video.
- Discuss the lesson and facilitate a brief discussion about what students learned and experienced.
 - "Tell me about the program you made today."
 - "What was your favorite part of this activity?"
 - "What did you learn about computer science and coding?"
 - "What was the most challenging part of this activity?"

Lesson
links/resources:

[Gumball's Coding Adventure Lesson Plan Slide](#)
[The Theme of Gumball lesson plan](#)
[Solution sheets](#)

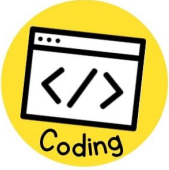
CS standards addressed:	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Watch a series of videos and create one coding project with opportunities to personalize their work using Add-ons, which are mini-coding challenges that build on top of the core project <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
Time needed:	<p><u>Total time: 60 min</u></p> <ul style="list-style-type: none"> • Introduction video 5 min • Foreshadow the Glitch with Dialogue 10 min • The Glitch and a Response 10 min • Create a New Scene 5 min • Reuse the Glitch 5 min • Conclude the Story and Choose and Add-on 10 min • The Amazing World of Gumball and Wrap Up 15 min
Materials needed:	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • The Theme of Gumball lesson plan • Solution sheets • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Headphones • Scratch account
Subject integrated:	ELA
Other standards addressed:	RL.5.2 —Determine the theme of the story, drama, or poem from details in the text. Summarize the text.
Vocabulary:	<p><u>Sequences</u>: A set of related events, movements, or things that follow each other in a particular order</p> <p><u>Events</u>: An action that causes something to happen</p> <p><u>Loops</u>: The action of doing something over and over again</p> <p><u>Conditionals</u>: Statements that only run under certain conditions</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 12: Story (re)Telling

<p>Lesson overview:</p> 	<p><u>Purpose:</u> The teacher will read a text. The student will go to Scratch to create a background, sprites, and dialogue to retell the story.</p> <p><u>Lesson:</u></p> <ul style="list-style-type: none"> • Have students log in to the Scratch website using their student accounts. • Demonstrate on smart board how to open the pro Students watch videos and create a project in Scratch. • Videos: Introduction to Scratch. (Use smartboard to show videos in attached slide or assign the slide to students in google classroom for them to view independently.) *Youtube has some additional instructional videos if needed. • Students will watch videos and begin a project in Scratch. • Check in with students as they watch the videos and complete their projects. • Students choose Add-ons to enhance their project. • Encourage students to also show their projects to a classmate. • Discuss the lesson and facilitate a brief discussion about what students learned and experienced. <ul style="list-style-type: none"> ◦ "What was your favorite part of this lesson?" ◦ "What story did you tell?" ◦ "What blocks did you use, and what did they do?"
<p>Lesson links/resources:</p>	<p>Storytelling Lesson Plan Slide Storytelling Lesson Plan Storytelling Solution Sheets Scratch</p>
<p>CS standards addressed:</p>	<p>Students will be able to: Learn about how Scratch works, then tell a story using Scratch where a character walks through a scene describing what they see</p> <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.6b—Students should document the plan as, for example, a storyboard, flowchart, pseudocode, or story map. • AP.1B.3—Create programs that include sequences, events, loops and conditionals.
<p>Time needed:</p>	<p><u>Total time: 55 min</u></p> <ul style="list-style-type: none"> • CS First Survey 5 min • Introduction to Dialogue and Sequencing 5 min • Setting the Scene 15 min • Speaking and Responding 10 min • Choose an Add-on 5 min • Reflection, Wrap Up, Next Steps 15 min
<p>Materials needed:</p>	<p><u>Teacher:</u></p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Storytelling Lesson Plan Slide • Storytelling Lesson Plan • Storytelling Solution Sheets • Scratch account

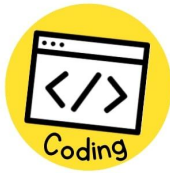
	Students: <ul style="list-style-type: none"> • Computer/tablet with internet access • Microphone • Headphones • Scratch account
Subject integrated:	ELA
Other standards addressed:	RL.5.2 —Determine the theme of the story, drama, or poem from details in the text. Summarize the text.
Vocabulary:	<p><u>Storyboard</u>: A graphic organizer that provides the viewer with a high-level view of a project</p> <p><u>Flowchart</u>: A type of diagram that represents an algorithm, workflow or process</p> <p><u>Pseudocode</u>: An artificial and informal language that helps programmers develop algorithms</p> <p><u>Story map</u>: A web map that has been thoughtfully created, given context, and provided with supporting information so it becomes a stand-alone resource</p>
Notes:	<p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 13: Code.org—Hello World, Lesson 1

<p>Lesson overview:</p> 	<p>Purpose: Events are very common in most computer programs. In this activity, students will use events to make a character move around the screen, make noises, and change backgrounds based on user-initiated events.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Main Activity <ul style="list-style-type: none"> ◦ Code.org Lesson 1- Hello World Lesson Plan
<p>Lesson links/resources:</p>	<p>Code.org Lesson 1- Hello World Lesson Plan</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create an interactive animation using sprites and events • Create new sprites and assign them costumes and locations <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals. • AP.1B.4—Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. • AP.1B.5—Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features.
<p>Time needed:</p>	<p>Total time: 55 min</p> <ul style="list-style-type: none"> • Code.org Lesson 1- Hello World 55 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Code.org account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Code.org account
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<p>RL.5.3—Compare and contrast two or more characters, settings, or events in a story or drama, drawing on specific details in the text.</p>
<p>Vocabulary:</p>	<p><u>Event</u>: An action that causes something to happen <u>Decompose</u>: Break a problem down into smaller pieces</p>
<p>Notes:</p>	

Week 14: Figurative Conversations

Lesson overview:



Purpose:

Students will use sprites and dialogue to show their understanding of literal and figurative meaning. Students will learn about computer science and then create a project that explores figurative language.

Lesson:

Students watch [videos and create a project](#) in Scratch.

- Videos: Introduction to Scratch. (Use smartboard to show videos in attached [slide](#) or assign the slide to students in google classroom for them to view independently.) *Youtube has some additional instructional videos if needed.
- Demonstrate on smart board how to open [the project in Scratch.](#)

1. Present the first two videos of [Figurative Language.](#)
2. Navigate students to the [Figurative Language](#) lesson.
3. Reiterate the video instructions:
 - a. Open the Starter Project
 - b. Add your figurative language
 - c. Show what the language seems to mean, and what the writer actually means using "say" blocks.
4. Release students to work. While they're working:
 - a. Check that students are on-task. Look for signs that students might not know what to do, like adding a lot of sprites or dragging out a lot of unrelated blocks.
 - b. Talk with students one-on-one about how they can show the literal and figurative meanings of a piece of figurative language through what their character says and does.

Lesson links/resources:

[Figurative Language Lesson Plan Slide](#)
[Figurative Language Lesson Plan](#)

CS standards addressed:

Students will be able to:

- Apply their knowledge of figurative language to programming
- To code their story and show the literal and figurative meaning of their text

Standards:

- **AP.1B.3**—Create programs that include sequence, events, loops, and conditionals.
- **AP.1B.4**—Decompose problems into smaller, manageable subproblems to facilitate the program development process.

Time needed:

Total time: 55 min

- Introduction to Figurative Language **5 min**
- Explore Your Figurative Language **15 min**
- Choose Add-ons **10 min**
- Figurative Language survey **10 min**
- Wrap Up: Next Steps **10 min**

Materials needed:

Teacher:

- Computer
- Projector/smartboard with sound
- [Figurative Language Lesson Plan](#)
- [Scratch](#) account

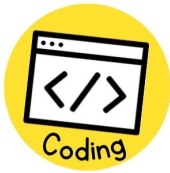
Students:

- Computer/tablet with internet access
- Headphones

	<ul style="list-style-type: none"> • Scratch account
Subject integrated:	ELA
Other standards addressed:	RL.5.4 —Determine the meaning of words and phrases as they are used in a text, including figurative language such as metaphors and similes.
Vocabulary:	<p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Loops</u>: The action of doing something over and over again</p> <p><u>Conditionals</u>: Statements that only run under certain conditions</p> <p><u>Decompose</u>: Break a problem down into smaller pieces</p> <p><u>Program</u>: An algorithm that has been coded into something that can be run by a machine.</p> <p><u>Parallelism</u>: The process of events happening at the same time, either independently or interdependently.</p> <p><u>Debugging</u>: The process of identifying and fixing error(s) in a program when it is not functioning as expected.</p> <p><u>Control structures</u>: Sections of code that order the direction or flow of how a program functions; the control structure in this lesson is focused on loops.</p>
Notes:	<p>Unplugged activity: Programmer says</p> <ul style="list-style-type: none"> • Programmer says reinforces the concepts of sequencing instructions and conditionals. <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 15: Create-a-Character

Lesson overview:



Purpose:

Students tell the story of a character who does not have a school assignment when it's due. Students will describe this character through their actions, thoughts and words. This lesson can also be used to compare and contrast chapters, scenes, or stanzas.

Lesson:

Students watch [videos and create a project](#) in Scratch.

- Videos: Introduction to Scratch. (Use smartboard to show videos in attached [slide](#) or assign the slide to students in google classroom for them to view independently.) *YouTube has some additional instructional videos if needed.
- Demonstrate on smart board how to open [the project in Scratch.](#)
- Introduce the lesson by reading aloud the following prompts:
 - "In this lesson, you will use code to tell the story of someone who needs to turn in an assignment for school but hasn't completed it yet."
 - "What are some reasons that someone might not have an assignment done? What are some wacky, fun, or fantastical reasons someone might not have their homework? What might someone do when they find out that they don't have their assignment?"
- Review the Characterization ELA concept by reading aloud the following prompt:
 - "There are lots of ways to tell a story about someone not having an assignment. You will use what your character says, thinks, or does to tell readers about your character. Are they smart, brave, resilient, or creative? You will get to decide."
- Before computer time, consider using a pre-writing lesson, like brainstorming character traits, to connect this to your classroom learning objectives.
- Tell students that they will apply their knowledge of characterization to code a story in Scratch. They will follow videos to code their story and develop their character.

Lesson links/resources:

- [Characterization Lesson Plan Slide](#)
- [Characterization Lesson Plan](#)

CS standards addressed:

Students will be able to:

- Tell the story of a character who does not have a school assignment when it's due
- Describe this character through their actions, thoughts and words
 - This [example project](#) shows what a student can create if they complete all the Add-ons in the lesson.

Standards:

- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.
- **AP.1B.4**—Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

Time needed:

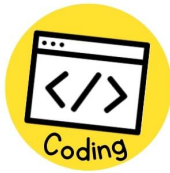
Total time: 50 min

- Introduction to Characterization **10 min**

	<ul style="list-style-type: none"> • What is Your Character Thinking? 10 min • Choose an Add-on 10 min • Characterization Survey 10 min • Wrap Up: Next Steps 10 min
Materials needed:	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Characterization Lesson Plan • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Headphones • Scratch account
Subject integrated:	ELA
Other standards addressed:	<ul style="list-style-type: none"> • RL.5.3—Compare and contrast two or more characters, settings, or events in a story or drama, drawing on specific details in the text (e.g., how characters interact). • RL.5.5—Explain how a series of chapters, scenes, or stanzas fits together to provide the overall structure or a particular story, dramas, or poem.
Vocabulary:	<p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Loops</u>: The action of doing something over and over again</p> <p><u>Conditionals</u>: Statements that only run under certain conditions</p> <p><u>Decompose</u>: Break a problem down into smaller pieces</p> <p><u>Program</u>: An algorithm that has been coded into something that can be run by a machine</p>
Notes:	<p>Unplugged activity: Draw like a robot</p> <ul style="list-style-type: none"> • In this activity, students practice giving very specific instructions to a partner and compare their characters. <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 16: Two Points of View

Lesson overview:



Purpose:

Students work with a partner to tell the same story from two different points of view.

Lesson:

Students watch [videos and create a project](#) in Scratch.

- Videos: Introduction to Scratch. (Use smartboard to show videos in attached [slide](#) or assign the slide to students in google classroom for them to view independently.) *YouTube has some additional instructional videos if needed.
 - Demonstrate on smart board how to open the [project in Scratch](#).
 - Review the ELA concept by reading aloud the following prompts:
 - "The narrator describes what is happening in the story but is not directly involved in it. They tell the story from the third-person point of view and use words like "he," "she," and "them." If a character within the story is describing the action themselves, they use words like "I," "we," and "us." This is called telling the story from the first-person point of view."
 - "In this lesson, one person in your group will tell a story from the first-person, and the other will tell the story from the third person."
 - "Before computer-time, consider using a partner pre-writing lesson, like brainstorming a scene from a story they would like to tell, to connect this to your classroom learning objectives."
 - Tell students that they will apply their knowledge of narration to code a story. They will follow videos to code their story in the first- or third-person point of view.

Lesson links/resources:

- [Narration overview Lesson Plan Slide](#)
- [Narration Lesson Plan](#)

CS standards addressed:

Students will be able to:

- Sequence "say" blocks to tell a story in the first-person or third-person
- Use event blocks (like "when flag clicked") to trigger a series of code
- Use wait blocks to construct a dialogue between two sprites or think blocks to describe a new sprite
- Use motion blocks (e.g., "glide to X Y"), sound effects, and animation to personalize their story

Standards:

- **IC.1B.3a**—Students will collaborate and receive feedback from others.
- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.
- **AP.1B.4**—Decompose problems into smaller, manageable subproblems to facilitate the program development process.
- **AP.1B.9**—Take on varying roles, with teacher guidance when collaborating with peers during the design, implementation, and review stages of program development.

Time needed:

Total time: 50 min

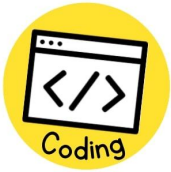
- Introduction to Narration **10 min**
- Start Your Story **15 min**
- Narration Survey **10 min**
- Wrap Up: Next Steps **15 min**

Materials needed:	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Narration Lesson Plan • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Headphones • Journal • Scratch account
Subject integrated:	ELA
Other standards addressed:	<ul style="list-style-type: none"> • RL.5.3—Compare and contrast two or more characters, settings, or events in a story or drama, drawing on specific details in the text (e.g., how characters interact). • RL.5.5—Explain how a series of chapters, scenes, or stanzas fits together to provide the overall structure or a particular story, dramas, or poem. • RL 5.6—Describe how a narrator's or speaker's point of view influences how events are described.
Vocabulary:	<p><u>Design</u>: A way to author computer applications using a combination of text, graphics, and style elements in a unified code-space</p> <p><u>Program</u>: An algorithm that has been coded into something that can be run by a machine</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 17: Code.org—Analyze Visuals

Lesson 1, Level 1

Lesson overview:



Purpose:

The student will create a poem/story and an image to go with their poem. The student will present their picture and poem/story to show their connection. The student should comment on their picture with an explanation of their story.

Lesson:

- Introduce the day's activity by reading aloud the following prompts:
 - "Today we are going to use our creativity while we learn to code. There are lots of ways to be creative. What are some ways you like to be creative?"
 - "Did you know that computer science can be very creative? In fact, you can make art with code. As a reminder, code is a set of instructions that a computer can understand. Just like choosing which colors of paint to use, choosing what code you write can be an opportunity to express your creativity too!"
 - "We're going to use poetry to inspire our creative coding today."
 - "Does anyone know what poetry is?"
 - "Do you have a favorite poem or poet?"
- Main Activity
 - Read poetry together—Read a few poems together as a class and practice naming the mood of each poem. Encourage students to share their interpretation, even if it's different from someone else's.
- Wrap Up
 - Check for understanding with two sticky notes. On one sticky note, have your students write one thing they learned about coding today. On the second, write one thing you have learned about poetry.

Lesson links/resources:

[Poem Art - Code.org - Lesson plan](#)

CS standards addressed:

Students will be able to:

- Develop programs that respond to timed events
- Illustrate shifts in the mood of a poem
- Represent figurative and literal language in a text

Standards:

- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.
- **AP.1B.5**—Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features.
- **AP.1B.8**—Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.

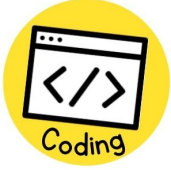
Time needed:

Total time: 55 min

- Warm Up **5 min**
- Main Activity (Poem Art) **45 min**
- Wrap Up **5 min**


Materials needed:	<u>Teachers:</u> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Poem Art - Code.org - Lesson plan • Code.org account <u>Students</u> <ul style="list-style-type: none"> • Computer/tablet with internet access • Headphones • Sticky notes • Code.org account
Subject integrated:	ELA
Other standards addressed:	<ul style="list-style-type: none"> • RL.5.4—Determine the meaning of words and phrases as they are used in a text, including figurative language such as metaphors and similes. • RL.5.7—Analyze how visual and multimedia elements contribute to the meaning, tone, or beauty of a text.
Vocabulary:	<u>Code:</u> To write instructions for a computer <u>Event:</u> An action that causes something to happen <u>Program:</u> A series of steps that have been coded into something that can be run by a machine
Notes:	<p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 18: Adventure on the High Seas

<p>Lesson overview:</p> 	<p>Purpose: Create the beginning of a story and use details to help infer. Share with your partner, and they will finish the story using your details.</p> <p>Lesson:</p> <p>Students watch videos and create a project in Scratch.</p> <ul style="list-style-type: none"> • Videos: Introduction to Scratch. (Use smartboard to show videos in attached slide or assign the slide to students in google classroom for them to view independently.) *YouTube has some additional instructional videos if needed. • Demonstrate on smart board how to open the project in Scratch. • Warm Up <ul style="list-style-type: none"> ◦ Have students write the beginning of a story (setting, characters, plot, etc.). Have students share with a partner. • Activity <ul style="list-style-type: none"> ◦ Watch video and open Scratch to begin creating the end of the story. Have students share their story with their partner or with the class. • Wrap Up <ul style="list-style-type: none"> ◦ Have students write about how their story and their partner's story were similar and different.
<p>Lesson links/resources:</p>	<p>Adventure on the High Seas Lesson Plan Slide Adventure on the High Seas Lesson Plan</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Collaborate and receive feedback from others • Collaborate with peers during the design, implementation, and review stages of program development <p>Standards:</p> <ul style="list-style-type: none"> • IC.1B.3a—Students will collaborate and receive feedback from others. • AP.1B.3—Create programs that include sequences, events, loops and conditionals. • AP.1B.9—Take on varying roles, with teacher guidance when collaborating with peers during the design, implementation, and review stages of program development.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Warm Up 5 min • Adventure Writing 50 min • Wrap Up 5 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Adventure on the High Seas Lesson Plan • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>ELA</p>

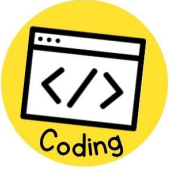
Other standards addressed:	RL.5.9 —Compare and contrast stories in the same genre on their approaches to similar themes and topics.
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Dialogue</u>: Conversation between two or more people as a feature of a book, play, or movie</p> <p><u>Scene</u>: The place where an incident in real life or fiction occurs or occurred.</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 19: Game Design Choices

<p>Lesson overview:</p> 	<p><u>Purpose:</u> Students will learn basic video game coding concepts by making different types of games, including racing, platform, and launching. The links to these gaming slides are below.</p> <p><u>Lesson:</u> Students watch videos and create a project in Scratch.</p> <ul style="list-style-type: none"> • Videos: Introduction to Scratch. (Use smartboard to show videos in attached slide or assign the slide to students in google classroom for them to view independently.) *YouTube has some additional instructional videos if needed. • Demonstrate on smart board how to open the project in Scratch.
<p>Lesson links/resources:</p>	<p>Lesson Plan Slide Scratch</p>
<p>CS standards addressed:</p>	<p>Students will be able to: Standards:</p> <ul style="list-style-type: none"> • IC.1B.2a—Students will demonstrate an understanding of diversity in ability and interests by developing artifacts and tools that use different methods of communication and/or appeal to different users.
<p>Time needed:</p>	<p><u>Total time: 50 min</u></p> <ul style="list-style-type: none"> • Warm Up: Gender Stereotypes 10 min • Analyze: True! Or Is It? 10 min • Create: Stereotype Avatar 20 min • Wrap Up: "Just Because" Poem 10 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Lesson Plan Slide • Scratch <p>Students:</p> <ul style="list-style-type: none"> • Lesson Plan Slide • Scratch
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<ul style="list-style-type: none"> • RI.5.4—Determine the meaning of general academic and domain-specific words and phrases in a text relevant to a grade 5 topic or subject area. • RI.5.7—Draw on information from multiple print or digital sources, demonstrating the ability to locate an answer to a question quickly or to solve a problem efficiently.
<p>Vocabulary:</p>	<p><u>Avatar</u>: An image or character that represents a person online</p>


Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts in Scratch.</p> <p>This lesson was recreated using resources from CS First which is no longer available online.</p>

Week 20: Scratch-a-Summary

<p>Lesson overview:</p> 	<p>Purpose: Students will determine when a text has two or more main ideas. They will identify all of the main ideas or points presented by the author. Students will identify key details from the text and show how they support each of the main ideas. Students will use details and the main idea and create a Scratch summary of the informational text.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Students will gather as a group and discuss main ideas and spark ideas. Teachers will provide the text. • Participants will create projects working at their own pace. • Gather together and share and reflect.
Lesson links/resources:	Scratch
CS standards addressed:	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Use details and the main idea to create a Scratch summary of the informational text <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data.
Time needed:	<p>Total time: 50 min</p> <ul style="list-style-type: none"> • Warm Up: Students will gather as a group and discuss main ideas and spark ideas. Teachers will provide the text. 15 min • Main Activity: Participants will create projects working at their own pace. 30 min • Wrap Up: Gather together and share and reflect. 10 min
Materials needed:	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • www.scratch.mit.edu/ideas • Scratch account
Subject integrated:	ELA
Other standards addressed:	RI.5.2 —Determine two or more main ideas of a text and explain how they are supported by key details; summarize the text.
Vocabulary:	<p><u>Variable</u>: A label for a piece of information used in a program</p> <p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p>
Notes:	<p>Alternate site: Lesson 6—Blank Space Stories</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

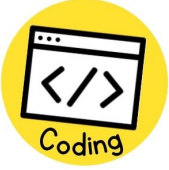
--	--

Week 21: Reading News Online

<p>Lesson overview:</p> 	<p>Purpose: Kids find and read news in lots of different ways. But studies show they are not very good at interpreting what they see. How can we help them get better? Teaching your students about the structure of online news articles is an important place to start.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Warm Up: The News You Know • Explore: The Parts of a News Site • Watch: How to Read News Online • Wrap Up: Label a News Page
<p>Lesson links/resources:</p>	<p>Reading News Online - Common Sense Media</p> <p>Lesson slides</p> <p>Video: Reading News Online</p> <p>Mix & Match Cards handout</p> <p>Label a News Page handout</p> <p>Lesson quiz</p>
<p>CS standards addressed:</p>	<p>Students should be able to:</p> <ul style="list-style-type: none"> • Understand the purposes of different parts of an online news page • Identify the parts and structure of an online news article • Learn about things to watch out for when reading online news pages, such as sponsored content and advertisements <p>Standards:</p> <ul style="list-style-type: none"> • DA.1B.1—Organize and present collected data visually to highlight relationships and support a claim.
<p>Time needed:</p>	<p>Total time: 50 min</p> <ul style="list-style-type: none"> • Warm Up: The News You Know 10 min • Explore: The Parts of a News Site 20 min • Watch: How to Read News Online 10 min • Wrap Up: Label a News Page 10 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with internet • Lesson slides • Video: Reading News Online • Mix & Match Cards handout • Label a News Page handout • Lesson quiz • Common Sense account <p>Students:</p> <ul style="list-style-type: none"> • Scissors • Glue • Blank paper • Common Sense account
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<p>RI.5.3—Explain the relationships or interactions between two or more individuals, events, ideas, or concepts in a historical, scientific, or technical text based on specific information in the text.</p>

Vocabulary:	<p><u>Article</u>: A written story in a newspaper, magazine, or online news site</p> <p><u>Commercial</u>: An advertisement intended to make money</p> <p><u>News</u>: New information about recent or important events</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.</p>

Week 22: Escaping With Context Clues

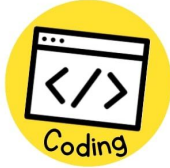
<p>Lesson overview:</p> 	<p>Purpose: Students will use Scratch to create an escape game where peers must use context clues to identify unknown words to escape.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary and create and identify context clues to be used in the escape room. ◦ Allow students to explore Scratch Escape Room Examples to brainstorm/discuss ideas to create games. • Animation: <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share or allow classmates to play.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Scratch Escape Room Examples
<p>CS standards addressed:</p>	<p>Student will be able to:</p> <ul style="list-style-type: none"> • Create programs using variables <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data. • AP.1B.5—Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Identify context clues and unknown words to be used in the escape room 10 min • Create the animation 20 min • Wrap up activity/present 20 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>ELA</p>
<p>Other standards addressed:</p>	<p>RI.5.4—Determine the meaning of general academic and domain-specific words and phrases in a text relevant to a grade 5 topic or subject area.</p>
<p>Vocabulary:</p>	<p><u>Event</u>: An action that causes something to happen <u>Sequence</u>: A set of logical steps carried out in order <u>Conditionals</u>: Expressions that evaluate to either true or false <u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>

Notes:

→Teachers will need to create FREE teacher and/or student accounts (when applicable) at [Scratch](#).

Week 23: Variables and Text Structures

Lesson overview:



Purpose:

Students will use [Scratch](#) to create a game show to teach a younger person about all text structures. Students will choose a sprite and add a variable with each of the text structure types: compare/contrast, chronological, cause/effect, descriptive, and problem/solution.

Lesson:

- Gather as a group to discuss the game and spark ideas on how to design a game that teaches about types of text structures.
- Students will individually create projects working at their own pace.
- Gather together and share and reflect.

- [Scratch](#)
- [Catch Game tutorial](#)
- [Clicker Game tutorial](#)
- [Scratch Game ideas](#)

CS standards addressed:

Students will be able to:

- Create a game show format explaining text structures
 - Example: Players must "catch" examples of a particular text structure
- Create sprites and add variables for each type of text structure

Standards:

- **AP.1B.3a**—Students should be able to create programs that include sequences, events, loops, and conditionals.

Time needed:

Total time: 60 min

- Review text structures **5 min**
- Model how to choose a sprite and build a game **10 min**
- Have students create their own game to review text structures **30 min**
- Swap games with another student to review **15 min**

Materials needed:

Teacher:

- Computer
- Projector/smartboard with sound
- [Scratch](#) account
- [Scratch Game ideas](#)

Students:

- Computer/tablet with internet access
- Paper
- Pencil
- [Scratch](#) account

Subject integrated:

ELA

Other standards addressed:

RI.5.5—Compare and contrast the overall structure (e.g., chronology, comparison, cause/effect, problem/solution) of events, ideas, concepts, or information in two or more texts.

Vocabulary:

Sequence: A set of logical steps carried out in order

Loops: The action of doing something over and over again

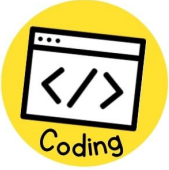
Conditionals: Statements that only run under certain conditions

Decompose: Break a problem down into smaller pieces

Program: An algorithm that has been coded into something that can be run by a machine

	<u>Events</u> : An action that causes something to happen
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch .

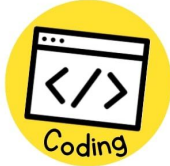
Week 24: Photosynthesis Animation

<p>Lesson overview:</p> 	<p>Purpose: The students will utilize Scratch to create an animation explaining photosynthesis.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Review photosynthesis and create a drawing explaining the process. • Create <ul style="list-style-type: none"> ◦ Students will create a background and animated sprite to showcase the process of photosynthesis. • Share <ul style="list-style-type: none"> ◦ Students will share and present their animation to the classroom.
<p>Lesson links/resources:</p>	<p>Scratch</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create and animate a sprite. • Add text or audio to a sprite to share information. • Create a background that reflects the information being presented. <p>Standard:</p> <ul style="list-style-type: none"> • DA.1B.1—Organize and present collected data visually to highlight relationships and support a claim.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to activity/draw photosynthesis process 10 min • Background creation 10 min • Create and animate sprite 15 min • Add photosynthesis text or audio 15 min • Wrap up activity/present 10 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Paper • Pencil • Scratch account
<p>Subject integrated:</p>	<p>Science</p>
<p>Other standards addressed:</p>	<p>L.5.3A—Students will demonstrate an understanding of photosynthesis and the transfer of energy from the sun into chemical energy necessary for plant growth and survival.</p>
<p>Vocabulary:</p>	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance <u>Costume</u>: An image that gives a sprite its appearance <u>Event</u>: An action that causes something to happen</p>
<p>Notes:</p>	<p>A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch to help students with the creation of their animated presentation.</p>

	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>
--	---

Week 25: Coding Ecosystems

Lesson overview:



Purpose:

The students will utilize their knowledge of ecosystems and food webs to recreate a selected biome with an animated food web demonstration that showcases all trophic levels of that biome.

Lesson:

- Introduction
 - Follow the prompts on the [CSC Project: Ecosystems using Sprite Lab](#). (Students completed Hello World in Week 13)
 - Predict - *Display the Predict slide in front of the class. Distribute the code PDF printouts to students. Do not run the code during this stage.*
 - Which sprites will appear before the program runs?
 - How are we meant to use the program?
 - What will happen when we use it?
 - Run - Open the [CSC Starter Program- Ecosystem](#) press “Run,” and invite suggestions from students about how to interact with it. Prompt students to observe the output of the running program and compare this to the predictions they made.
- Investigate
 - Students will go to the [CSC Starter Program- Ecosystem](#) to access the example of Meadow Ecosystem code.
- Create
 - Students will create a new ecosystem of their choice with food web connections. *Instruct students to click the Remix button on the starter program. This will create their own editable copy.*
- Students will share and present.

Lesson links/resources:

- [Code.org](#)
- [CSC Project: Ecosystems using Sprite Lab](#)

CS standards addressed:

Students will be able to:

- Create code to illustrate a new ecosystem using block code
- Create sprites that represent different plants and animals

Standard:

- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.
- **AP.1B.4**—Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.
- **AP.1B.5**—Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

Time needed:

Total time: 60 min

- Introduction to activity **10 min**
- Investigate and discuss **10 min**
 - [Meadow Ecosystem](#) code example
- Create new ecosystem **15 min**
- Add food web connections for the sprites in new ecosystem **15 min**
- Wrap up activity/present **10 min**

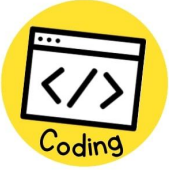
Materials needed:

Teachers:

- Computer

	<ul style="list-style-type: none"> • Projector/smartboard with sound • Code.org account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Paper • Pencil • Code.org account
Subject integrated:	Science
Other standards addressed:	L.5.3B —Students will demonstrate an understanding of a healthy ecosystem with a stable web of life and the roles of living things within a food chain and/or food web, including producers, primary and secondary consumers, and decomposers.
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Costume</u>: An image that gives a sprite its appearance</p> <p><u>Event</u>: An action that causes something to happen</p>
Notes:	<p>Students need to have the basic understanding of creating a sprite for this activity. Access the following link for a tutorial:</p> <ul style="list-style-type: none"> • Hello World sprite creation tutorial <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Code.org.</p>

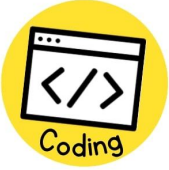
Week 26: Matter Matters Animation

<p>Lesson overview:</p> 	<p>Purpose: Students will create an animation on Scratch demonstrating the movement of molecular structure through the different states of matter.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Review the states of matter. ◦ Have students create a sprite of matter that can change into different states (e.g., water). • Create <ul style="list-style-type: none"> ◦ Students will create the different scenarios needed to change their chosen sprite to a solid, liquid, and gas. • Students will share and present.
Lesson links/resources:	Scratch
CS standards addressed:	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Use sequences, events, loops, and conditionals <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
Time needed:	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to activity 10 min • Background creation 10 min • Create and animate sprite 15 min • Add audio or text to animation 15 min • Wrap up activity/present 10 min
Materials needed:	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
Subject integrated:	Science
Other standards addressed:	P.5.5A.2 —Collect, analyze, and interpret data from measurements of the physical properties of solids, liquids, and gasses (e.g., volume, shape, movement, and spacing of particles).
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Costume</u>: An image that gives a sprite its appearance</p> <p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p> <p><u>Algorithms</u>: A procedure or formula used for solving a problem</p>

Notes:

A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on [CS First](#) to help students with the creation of their animated presentation.
→Teachers will need to create FREE teacher and/or student accounts (when applicable) at [Scratch](#).

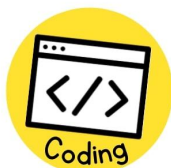
Week 27: Mix and Match Mixtures Game

<p>Lesson overview:</p> 	<p>Purpose: Students will design an interactive animation on Scratch that builds an interactive game or demonstration combining different materials to create examples of mixtures and solids, depending on the student's chosen algorithms for their activity.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Showcase a Scratch Catching Game, then have the students watch a Catching Game tutorial. • Create <ul style="list-style-type: none"> ◦ Students will design a game that allows for users to catch items to create a mixture or solution depending on the students choice of focus for the game. • Share and allow students in the class to play and critique other students' games. <ul style="list-style-type: none"> ◦ How accurate was the game? ◦ Did the game correctly catch items? ◦ Did the students' choices correctly reflect their choice of mixture or solution?
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Catching Game tutorial
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create an interactive game to reflect the theme • Create algorithms that allow them to accurately catch chosen items • Create multiple animated sprites <p>Standard:</p> <ul style="list-style-type: none"> • AP.1B.1—Compare and refine multiple algorithms for the same task and determine which is most appropriate. • AP.1B.3—Create programs that include sequences, events, loops, and conditionals. • AP.1B.5—Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to activity: Catching Game tutorial 10 min • Create game background 10 min • Create items for mixtures to catch 15 min • Create game code 15 min • Wrap up activity/present 10 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Paper • Pencil • Scratch account
<p>Subject integrated:</p>	<p>Science</p>

Other standards addressed:	P.5.5B —Students will demonstrate an understanding of mixtures and solutions.
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Costume</u>: An image that gives a sprite its appearance</p> <p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p> <p><u>Algorithms</u>: A procedure or formula used for solving a problem</p>
Notes:	<p>A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. There are also many variations of game tutorials that students may use as a starting point in the creation of their mixture activity.</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 28: Catch the Change Game

Lesson overview:



Purpose:

The students will modify and create an interactive game on [Scratch](#) that others may use that will focus on distinguishing between a chemical and physical change.

Lesson:

- Introduction
 - Review the Catching Game tutorial.
- Create
 - The students will click on create, then they will design and create a game that allows for users to catch examples of either a physical or chemical change.
 - Students can make sprites for both physical and chemical change examples. In the game, users could score points if they catch the correct example and lose points if they catch the wrong example.
 - Students will create a visual background and multiple sprites to reflect their chosen theme and choice of catching a physical or chemical change.
- Students will share and allow others to play and critique their created games.
 - How accurate was the game?
 - Did the game correctly catch items?
 - Did the game correctly catch items?

Lesson links/resources:

- [Scratch](#)
- [Catching Game tutorial](#)

CS standards addressed:

Students will be able to:

- Create an interactive game to reflect the theme
- Create algorithms that allow them to accurately catch chosen items
- Create multiple animated sprites

Standard:

- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.
- **AP.1B.5**—Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features.

Time needed:

Total time: 60 min

- Introduction to activity **10 min**
- Create game background **10 min**
- Create chemical/physical change **15 min**
- Create game code **15 min**
- Wrap up activity/present **10 min**

Materials needed:

Teacher:

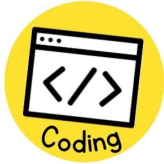
- Computer
- Projector/smartboard with sound
- [Scratch](#) account

Students:

- Computer/tablet with internet access
- Paper
- Pencil
- [Scratch](#) account

Subject integrated:	Science
Other standards addressed:	P.5.5C —Students will demonstrate an understanding of the difference between physical and chemical changes.
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Costume</u>: An image that gives a sprite its appearance</p> <p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p> <p><u>Algorithms</u>: A procedure or formula used for solving a problem</p>
Notes:	<p>A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. There are also many variations of game tutorials that students may use as a starting point in the creation of their mixture activity.</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

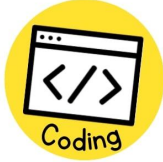
Week 29: Animating Motion

<p>Lesson overview:</p> 	<p>Purpose: The students will create an animated presentation on Scratch that demonstrates Newton's Laws of Motion.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Review Newton's three laws of motion. ◦ Have students choose one of Newton's laws to animate and present. • Create <ul style="list-style-type: none"> ◦ Have the students choose and create a sprite of their choice that can be animated to show motion. ◦ Have the students create a background that will allow the motion reflected in that law. • Let students present their project and have others in the class critique. <ul style="list-style-type: none"> ◦ Did the animation showcase the chosen Newton's law? ◦ Did the created background reflect the motion being animated?
<p>Lesson links/resources:</p>	<p>Scratch</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create an animated sprite to represent an object in motion • Create a background that reflects the chosen sprite and it's motion <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to activity 10 min • Create background 10 min • Create sprites 15 min • Add audio or text to sprites 15 min • Wrap up activity/present 10 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Paper • Pencil • Scratch account
<p>Subject integrated:</p>	<p>Science</p>
<p>Other standards addressed:</p>	<p>P.5.6—Students will demonstrate an understanding of the factors that affect the motion of an object through a study of Newton's laws of motion.</p>
<p>Vocabulary:</p>	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance <u>Costume</u>: An image that gives a sprite its appearance <u>Event</u>: An action that causes something to happen</p>

	<u>Sequence:</u> A set of logical steps carried out in order
Notes:	<p>A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation.</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 30: Traveling Home With Polaris Game

Lesson overview:



Purpose:

The students will create an interactive map or setting on [Scratch](#) that will have students using the North Star (also called the Pole Star, or Polaris) in order to successfully travel from a set starting point to their new desired location. The focus is on how Polaris is used to help with find true geographic north and how this is possible without the use of GPS or other directional tools such as a compass.

Lesson:

- Introduction
 - Showcase a [Travel Game example](#), then have students watch a [Scratch Moving Game tutorial](#).
- Create
 - The students will click on "[Create](#)," then they will design a game that creates a ship or boat that will travel through the created background using Polaris as the guide to their ending point.
 - Students will create a visual background and multiple sprites to reflect their chosen theme and obstacles they must avoid during their travels. Students need to choose the best way to make their sprite move through the maze of obstacles to reach the end.
- Reflect
 - Allow students to share their game and critique other students' games.
 - How accurate is the game?
 - Did the game correctly reflect the theme?

Lesson links/resources:

- [Scratch](#)
- [Travel Game example](#)
- [Scratch Moving Game tutorial](#)

CS standards addressed:

Students will be able to:

- Create an interactive game to reflect the theme
- Create code that allows for movement and direction from set keys when used
- Solve a problem by completing the maze traveling task with the best solution

Standard:

- **AP.1B.1a**—Students should be able to look at different ways to solve the same task and decide which would be the best solutions.
- **AP.1B.3**—Create programs that include sequences, events, loops, and conditionals.

Time needed:

Total time: 60 min

- Introduction to activity **10 min**
- Create background **10 min**
- Create sprite(s) **15 min**
- Add game animation **15 min**
- Wrap up activity/present **10 min**

Materials needed:

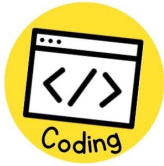
Teacher:

- Computer
- Projector/smartboard with sound
- [Scratch](#) account

Students:

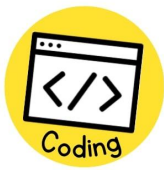
	<ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
Subject integrated:	Science
Other standards addressed:	E.5.8A.4 —Construct scientific arguments to support claims about the importance of astronomy in navigation and exploration, including the use of telescopes, compasses, and star charts.
Vocabulary:	<p><u>Sprite</u>: A graphic on the screen with a location, size, and appearance</p> <p><u>Costume</u>: An image that gives a sprite its appearance</p> <p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p> <p><u>Algorithms</u>: A procedure or formula used for solving a problem</p>
Notes:	<p>A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. Students may code by using arrows or mouse clicks to create the ability to change directions based on Polaris's location throughout the activity.</p> <p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 31: Animating Fractions

<p>Lesson overview:</p> 	<p>Purpose: Students will create an animation on Scratch that will describe the steps of how to add, subtract, multiply, or divide fractions with unlike denominators. Next they will animate the process.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over vocabulary and create or identify fractions to be used in the animation and solve them. • Animation <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share and allow other students to play.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Clicker Game tutorial • Clicker Game examples
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create sequence, events, loops, and conditionals <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Identify fractions to be used in the animation and solve 15 min • Create the animation 20 min • Wrap up activity/present 15 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account • Educator guides <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Math</p>
<p>Other standards addressed:</p>	<ul style="list-style-type: none"> • 5.NF.1—Add and subtract fractions with unlike denominators (including mixed numbers) by replacing given fractions with equivalent fractions in such a way as to produce an equivalent sum or difference of fractions with like denominators. • 5.NF.6—Solve real-world problems involving multiplication of fractions and mixed numbers, e.g., by using visual fraction models or equations to represent the problem. • 5.NF.7—Apply and extend previous understandings of division to divide unit fractions by whole numbers and whole numbers by unit fractions.
<p>Vocabulary:</p>	<p><u>Event</u>: An action that causes something to happen <u>Sequence</u>: A set of logical steps carried out in order</p>

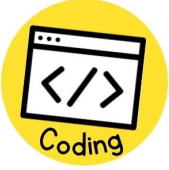
	<p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 32: Animating Decimals

<p>Lesson overview:</p> 	<p>Purpose: Students will create an animation on Scratch that will describe the steps of how to add, subtract, multiply, and divide decimals. Next they will animate the process.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary, create or identify decimal numbers to be used in the animation, and solve them. • Animation <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, students can share or allow classmates to play.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Clicker Game tutorial • Decimal Game example
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create sequence, events, loops, and conditionals <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops, and conditionals.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Identify decimal problems to be used in the game, and solve 15 min • Create the animation 20 min • Wrap up activity/present 15 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Mathematics</p>
<p>Other standards addressed:</p>	<p>5.NBT.7—Add, subtract, multiply, and divide decimals to hundredths, using concrete models (to include, but not limited to base ten blocks, decimal tiles, etc.) or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used.</p>

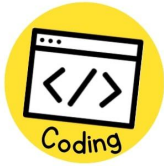
Vocabulary:	<p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 33: Place Value Pop-Up

<p>Lesson overview:</p> 	<p>Purpose: Students will create a game on Scratch that displays a multi-digit decimal number to the millions and thousandth's place. Once the number pops up, it will have an underlined digit. The player will then click the correct place value from a list of three options.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary, create, and identify decimal numbers to be used in the animation, and underline one digit in each decimal number. • Animation <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share or allow other students to play.
Lesson links/resources:	<ul style="list-style-type: none"> • Scratch • Clicker Game tutorial
CS standards addressed:	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Use variables <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data.
Time needed:	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Create the decimal numbers being used and the place value forms that will pop up on the screen for or each number 15 min • Create the animation 20 min • Wrap up activity/present 15 min
Materials needed:	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
Subject integrated:	Math
Other standards addressed:	5.NBT.3 —Read decimals to thousandths.
Vocabulary:	<u>Variables</u> : A label for a piece of information used in a program
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch .

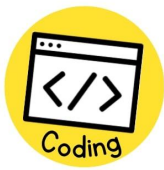
--	--

Week 34: Digital Roundup

<p>Lesson overview:</p> 	<p>Purpose: Students will create a game on Scratch that displays a multi-digit decimal number to the millions and thousandth's place. Once the number pops up it will have an underlined digit. The player will then click the correct rounded place value from a list of three options.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary, create and identify decimal numbers to be used in the animation, and underline one digit in each decimal number. • Animation <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share or allow classmates to play.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Clicker Game tutorial
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Create sequences, events, loops, and conditionals <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.3—Create programs that include sequences, events, loops and conditionals.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Create the multi-digit numbers being used in the activity and decide which digits will be underlined 15 min • Create the animation 20 min • Wrap up activity/present 15 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Mathematics</p>
<p>Other standards addressed:</p>	<p>5.NBT.4—Use place value understanding to round decimals to any place.</p>
<p>Vocabulary:</p>	<p><u>Event</u>: An action that causes something to happen <u>Sequence</u>: A set of logical steps carried out in order <u>Conditionals</u>: Expressions that evaluate to either true or false <u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>
<p>Notes:</p>	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

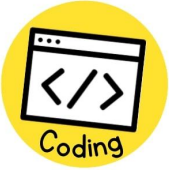
--	--

Week 35: Creating Conversions

<p>Lesson overview:</p> 	<p><u>Purpose:</u> Students will create a tutorial on Scratch with a partner on how to teach Metric and US Customary Conversions using length, weight, capacity, meter, liter, and gram.</p> <p><u>Lesson:</u></p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary, create, and identify metric and U.S. customary conversions measuring for length, weight, capacity, meter, liter, and gram. ◦ Students can be divided into groups so that students would have either Metric or US Customary measurements. • Animation <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share or allow classmates to play.
<p>Lesson links/resources:</p>	<p>Scratch</p>
<p>CS standards addressed:</p>	<p>Students will be able to:</p> <ul style="list-style-type: none"> • Collaborate with peers during the design, implementation, and review stages of program development <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data. • AP.1B.9—Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.
<p>Time needed:</p>	<p><u>Total time: 60 min</u></p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Create the steps of the tutorial 15 min • Create the animation 20 min • Wrap up activity/present 15 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Math</p>
<p>Other standards addressed:</p>	<p>5.MD.1—Convert among different-sized standard measurement units within a given measurement system (customary and metric) (e.g., convert 5 cm to 0.05 m), and use these conversions in solving multi-step, real-world problems.</p>

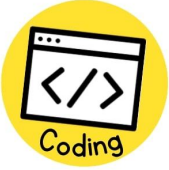
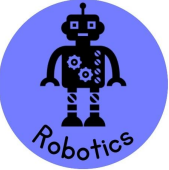

Vocabulary:	<p><u>Event</u>: An action that causes something to happen</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>
Notes:	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

Week 36: Escape the Missing Variable

<p>Lesson overview:</p> 	<p>Purpose: Students will create an escape room on Scratch. The students will need to find the missing variables to escape, display the digit that is represented by the variable, and solve.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Go over the vocabulary and create and identify algebraic expressions to be used in the escape room. • Animation: <ul style="list-style-type: none"> ◦ Create the characters, background, animation, and dialogue. • Wrap Up <ul style="list-style-type: none"> ◦ When the animation is complete, the students can share or allow classmates to play.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Scratch • Math Escape Room example
<p>CS standards addressed:</p>	<p>Student will be able to:</p> <ul style="list-style-type: none"> • Create programs using variables <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.2—Create programs that use variables to store and modify data.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to the activity 10 min • Identify algebraic expressions to be used in the escape room 10 min • Create the animation 20 min • Wrap up activity/present 20 min
<p>Materials needed:</p>	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Scratch account <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Scratch account
<p>Subject integrated:</p>	<p>Math</p>
<p>Other standards addressed:</p>	<p>5.OA.2—Write simple expressions that record calculations with numbers and interpret numerical expressions without evaluating them.</p>
<p>Vocabulary:</p>	<p><u>Event</u>: An action that causes something to happen <u>Sequence</u>: A set of logical steps carried out in order <u>Conditionals</u>: Expressions that evaluate to either true or false <u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p>
<p>Notes:</p>	<p>→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.</p>

--	--

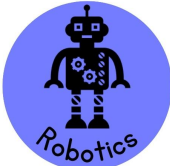
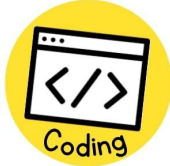
Week 37: Plotting Points

<p>Lesson overview:</p> <div style="text-align: center;">    </div>	<p>Purpose: The students will utilize a programmable robot in order to correctly traverse and graph the first quadrant plane to plot coordinates on the x- and y- axis.</p> <p>Lesson:</p> <ul style="list-style-type: none"> • Introduction <ul style="list-style-type: none"> ◦ Group students (three to five per group) to introduce the activity and review how to operate the codable robot. • Allow students to take turns practicing the use of the robot's features. • Have students mark the coordinates on their paper. <ul style="list-style-type: none"> ◦ Include some coordinates that you want students to reach with their robot, as well as some coordinates to avoid (obstacles). <ul style="list-style-type: none"> ■ You can also place physical obstacles on the graph such as pencils, erasers, sticky notes, etc. • Let students write the code for the robots to reach each coordinate while avoiding all obstacles. <ul style="list-style-type: none"> ◦ Coding cards are a great tool to help students think out their codes. • Allow students to practice running the code and debugging. • Wrap Up <ul style="list-style-type: none"> ◦ Review the lesson and allow groups to show their coding skills off by showing other groups how their robot reaches the coordinates without hitting obstacles. <p>Non-codable robot option:</p> <ul style="list-style-type: none"> • If you do not have a robot, students can still code! Print off a picture of a robot and allow them to move the paper robot using the squares on the graph paper as a measurable distance. You may want to use coding cards for the unplugged version of this activity.
<p>Lesson links/resources:</p>	<ul style="list-style-type: none"> • Coding cards • Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley)
<p>CS standards addressed:</p>	<p>The student will be able to:</p> <ul style="list-style-type: none"> • Plot coordinates using a programmable robot <p>Standards:</p> <ul style="list-style-type: none"> • AP.1B.1—Compare and refine multiple algorithms for the same task and determine which is the most appropriate.
<p>Time needed:</p>	<p>Total time: 60 min</p> <ul style="list-style-type: none"> • Introduction to activity 10 min • Build/modify robot 10 min • Code robot for obstacle course 15 min • Practice runs/code improvements 15 min • Wrap up activity/present 10 min
<p>Materials needed:</p>	<p>Teachers:</p> <ul style="list-style-type: none"> • Large (at least poster sized) four-quadrant graph paper (one per group) • Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) • List of coordinates (provided by teacher) <p>Students:</p> <ul style="list-style-type: none"> • Pencil • Quadrant graph paper • Optional: Coding cards

Subject integrated:	Math
Other standards addressed:	5.G.2 —Represent real-world and mathematical problems by graphing points in the first quadrant of the coordinate plane and interpret coordinate values of points in the context of the situation.
Vocabulary:	<u>Robot</u> : A standalone computer system that performs physical and computational activities <u>Coding</u> : How we communicate with computers <u>Algorithms</u> : A specific procedure for solving a computational problem <u>Program</u> : A specific set of ordered operations for a computer to perform
Notes:	

Week 38: Oceans and Continents Trivia

Lesson overview:



Purpose:

The students will each write a trivia question and answer on a notecard for each continent and ocean. They will then work with teams and place the cards on the corresponding oceans and continents. The students will program the robot to get to a card on the map. The students will answer the trivia question on the card and keep the cards they get correct. The goal is to have a card for each continent and ocean.

Lesson:

- Introduction
 - Explain the activity to the students
- Trivia Questions
 - Have each student write trivia questions on a notecard for each continent and ocean using these links:
 - [The 7 Continents of the World](#)
 - [Ocean Facts For Kids](#)
- Robot Review
 - Have students review how to use the codable robot.
- Game
 - Have students (one at a time) choose from a shuffled deck of notecards, decide on an answer, and program the robot to travel from a starting point (that the teacher decides) to the continent or ocean that answers the trivia card.

Non-codable robot option:

- If you do not have a robot, students can still code! Print off a picture of a robot and allow them to move the paper robot. Make sure to show the students the measurable distance that the robot can move per step; this can be the length of the robot picture, or the length of a notecard. You may want to use [coding cards](#) for the unplugged version of this activity.

Lesson links/resources:

- [The 7 Continents of the World](#)
- [Ocean Facts For Kids](#)
- [Coding cards](#)
- Codable robot (e.g., [Code and Go Mouse](#), [Dash Robot](#), [Botley](#))

CS standards addressed:

The students will be able to:

- Program a robot to travel to the oceans and continents on a large map
- Create a trivia question of each of the continents and oceans
- Answer questions about each of the continents and oceans to gain points during the game

Standards:

- **AP.1B.1**—Compare and refine multiple algorithms for the same task and determine which is the appropriate.

Time needed:

Total time: 60 min

- Explanation of activity **5 min**
- Creating trivia questions **15 min**
- Review codable robots **5 min**
- Playing games **40 min**

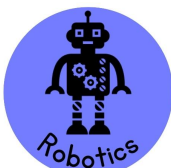
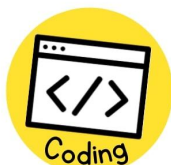
Materials needed:

Teacher:

	<ul style="list-style-type: none"> • Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) • Large world map <p>Students:</p> <ul style="list-style-type: none"> • Computer/tablet with internet access • Optional: Coding cards
Subject integrated:	Social Studies
Other standards addressed:	5.1.1 —Map the seven continents and five oceans.
Vocabulary:	<p><u>Robot</u>: A standalone computer system that performs physical and computational activities</p> <p><u>Coding</u>: How we communicate with computers</p> <p><u>Algorithms</u>: A specific procedure for solving a computational problem</p> <p><u>Program</u>: A specific set of ordered operations for a computer to perform</p>
Notes:	<p>Have students research the oceans and continents using the links listed above and write down 5 interesting facts about each ocean and continent to use for the programming lesson. This is a great review game.</p>

Week 39: Newton's Robot Maze

Lesson overview:



Purpose:

The students will create and build a robot that will use their knowledge of Newton's Laws of Motion to traverse a pre-made obstacle course that focuses on these laws.

Lesson:

- The teacher will introduce the teacher-created obstacle course with multiple areas that students will need to avoid in an attempt to complete the course the fastest.
 - Teachers: The area to be avoided can be anything that could slow motion, such as net force or slow motion.
- The students will work in pairs or groups to write code that will be used to program their codable robot to complete the course in the fastest time without hitting any obstacles that may slow them down.

Non-codable robot option:

- If you do not have codable robots, let the students act as the robot. One student will read the code aloud while another student acts as the robot.
- Once the pairs or groups complete their code, they should walk through the program with the robot in order to test the accuracy of their program and debug as necessary.
- Teachers should have the students run the course three times and record the time for each attempt. Students should also have a data chart to record their maze attempt times.
 - Which team was the fastest?
 - Did your group complete the obstacle course with your written program?
 - What changes, if any, would you make with your program to make it more successful?
 - What issues did your group have during this process?

Lesson links/resources:

[Teach Engineering—What Are Newton's Laws?](#) (Teacher Resource)
[Newton's Laws of Motion](#) (video for students)

CS standards addressed:

The students will be able to:

- Create code to traverse a set obstacle course avoiding objects and areas that slow motion
- Problem solve to improve any coding issues for the robot

Standard:

- **AP.1B.1**—Compare and refine multiple algorithms for the same task and determine which is the most appropriate.

Time needed:

Total time: 60 min

- Introduction to activity **10 min**
- Build/modify robot **10 min**
- Code robot for obstacle course **15 min**
- Practice runs and code improvements **15 min**
- Wrap up activity/timed attempts **10 min**

Materials needed:

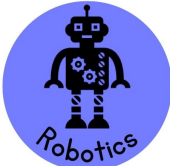
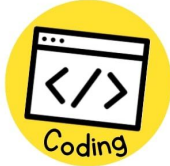
Teachers:

- Codable robot (e.g., [Code and Go Mouse](#), [Dash Robot](#), [Botley](#))
- Obstacle course with obstacles that slow down motion
- Paper/pencil (to record time data)

	<ul style="list-style-type: none"> • Timer • Time sheets <p>Students:</p> <ul style="list-style-type: none"> • Pencil • Time sheets • Paper • Timer
Subject integrated:	Science
Other standards addressed:	P.5.6 —Students will demonstrate an understanding of the factors that affect the motion of an object through a study of Newton's Laws of Motion.
Vocabulary:	<p><u>Direction</u>: A course along which something moves</p> <p><u>Motion</u>: The action or process of moving or being moved</p> <p><u>Position</u>: Put or arrange (someone or something) in a particular place or way</p> <p><u>Speed</u>: The rate at which someone or something is able to move or operate</p> <p><u>Sequence</u>: A set of logical steps carried out in order</p> <p><u>Loops</u>: A sequence of instructions that is continually repeated until a certain condition is reached</p> <p><u>Conditionals</u>: Expressions that evaluate to either true or false</p> <p><u>Algorithms</u>: A procedure or formula used for solving a problem</p>
Notes:	

Week 40: Directions Through the Prism

Lesson overview:



Purpose:

The student will draw a rectangular prism using grid paper. The student will determine the volume of the prism. Then the student will create a number grid—with one number being the correct answer—and will program a codable robot to reach the correct answer from a start location.

Lesson:

- Introduction
 - Review volume measurements (cubic centimeters, cubic inches, cubic feet) and formulas (length x width x height).
 - This activity is not an introduction to volume. Students must have prior knowledge and practice with finding volume using cubes.
- Create
 - Students will draw a rectangular prism(s) on grid paper.
 - Then students need to find the volume of the prism(s).
 - They need to fill in a table with various numbers, including answers to the volume of the prism(s).
- Programming
 - Once the student draws a rectangular prism on paper and determines the volume, they must then use [coding cards](#) to write a program that a robot would read to locate the volume on a number grid.
 - If you have a codable robot, you may use it in place of coding cards. An example of the grid is listed below.

Example:

The student draws a rectangular prism 2 in. x 6 in. x 2 in. = 24 cu in.

Start	35 cu in.	74 cc	48 cu in.
36 cu in.	24 cc	92 cu in.	14 cu in.
93 cu in.	36 cc	24 cu in.	74 cu in.

- Share
 - Students will swap their individual projects to let other students calculate the formula and program a robot to reach the correct answer.

Lesson links/resources:

- [Sketching Rectangular Prism Video](#)
- [Coding cards](#)

CS standards addressed:

The student will be able to:

- Create directions on creating a prism

Standards:

- **AP.1B.1**—Compare and refine multiple algorithms for the same task and determine which is the most appropriate.

Time needed:

Total time: 60 min

- Introduction **10 min**
- Create **15 min**
- Program **10 min**

	<ul style="list-style-type: none"> • Share 25 min
Materials needed:	<p>Teacher:</p> <ul style="list-style-type: none"> • Computer • Projector/smartboard with sound • Optional: Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) <p>Students:</p> <ul style="list-style-type: none"> • Pencil • Graph paper • Answer grid
Subject integrated:	Math
Other standards addressed:	<ul style="list-style-type: none"> • 5.MD.4—Measure volumes by counting unit cubes, using cubic centimeters, cubic inches, cubic feet, and improvised units. • 5.MD.5—Relate volume to the operations of multiplication and addition and solve real-world and mathematical problems involving volume.
Vocabulary:	<p><u>Robot</u>: A stand-alone computer system that performs physical and computational activities</p> <p><u>Coding</u>: How we communicate with computers</p> <p><u>Algorithms</u>: A specific procedure for solving a computational problem</p> <p><u>Program</u>: A specific set of ordered operations for a computer to perform the exact point where a mistake was made in coding</p>
Notes:	<p>Option: Have students create different prisms with different measurements using rulers to draw their prisms. Then students could exchange papers with other students to figure the volumes of different prisms instead of just their own.</p>