ENGLISH

MATH

COMPUTER SCIENCE

SCIENCE

SOCIAL STUDIES

2022
Elementary Integration Guide
KINDERGARTEN

MISSISSIPPI STATE UNIVERSITY™
CENTER FOR CYBER EDUCATION

Acknowledgements

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

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	[Code.org](#) CS Fundamentals [Kindergarten: Course A](#) [Common Sense Digital Media](#) [Kodable](#)
CS4MS website materials	[2018 Mississippi Computer Science Standards](#) [CS4MS Website](#)
Mouse practice	Online: - Alphabetical Order: https://www.abcya.com/games/alphabet - Mouse Practice: https://mouse-practice.com/ - Apple Catch - Coyote Concentration (card-matching game) - Desert Dive - Frost Bite - Helipopper - Penguin Drop - Pickle Pop - Pig Pile - Simon Sees
Keyboard practice	Online: [Astro Bubbles Keyboard Practice](#) Unplugged: - Keyboard Callout **Paper keyboard**: Using a paper keyboard, the teacher will call out letters, numbers, symbols, and/or words for students to "type" on their keyboard. **Computer with no internet:** The teacher will call out letters, numbers, symbols, and/or words. Students will use their keyboard to type into a blank document on their computer/tablet. - Keyboard Bingo - Preparation: The teacher will print squares with letters, numbers, and symbols (4-5 of each letter, 1-2 of each number/symbol). The teacher will cut out and laminate each square, then use a piece of tape or glue to adhere a magnet to each square. The teacher will distribute an equal number of random squares to students. - The teacher will project a keyboard onto a smartboard. - The teacher will call out letters, numbers, symbols, or words for students to find using their preprinted squares. - Students will raise their hands if they have the key that the teacher calls out. The teacher will choose a student to place their key on the board.
Teacher/student accounts	[Code.org](#) [Common Sense Digital Media](#) [Kodable](#)
For help with this guide	Contact Mississippi State University's Center for Cyber Education: www.tinyurl.com/ccehelpdesk

Contents by Integrated Subjects

<u>**ELA**</u>
- Week 1: L.K.1a—Print many upper- and lowercase letters
- Week 2: L.K.5a—Sort common objects into categories (e.g., shapes, foods)
- Week 3: RF.K.1d—Recognize and name all upper- and lowercase letters of the alphabet
- Week 4: RF.K.1d—Recognize and name all upper- and lowercase letters of the alphabet
- Week 6: RI.K.7—Describe the relationship between illustrations and the text
- Week 7: RF.K.2a, RF.K.2d—Demonstrate understanding of spoken words, syllables, and sounds
- Week 8: W.K.2—Compose information/explanatory texts in which they discuss a topic
- Week 9: W.K.2, RL.K.3—Identify characters, settings, and major events in a story
- Week 11: RI.K.2—Identify the main topic and retell key details of a text
- Week 13: RF.K.1—Demonstrate understanding of the organization/basic features of print
- Week 14: RF.K.1d, RF.K.3a—Produce the primary sound or many sounds of each consonant
- Week 15: RF.K.3a—Produce the primary sound or many sounds of each consonant
- Week 20: L.K.1a—Print many upper- and lowercase letters
- Week 21: RF.K.1—Demonstrate understanding of the organization/basic features of print
- Week 22: RF.K.1, RF.K.1A—Demonstrate understanding of the organization/features of print
- Week 27: RF.K.2, RF.K.2a—Demonstrate understanding of spoken words, syllables, and sounds
- Week 29: L.K.2—English capitalization, punctuation, and spelling when writing
- Week 32: SL.K.2, SL.K.3, SL.K.6—Confirm understanding of a information presented orally
- Week 33: RL.K.7—Describe the relationship between illustrations and the text
- Week 34: RL.K.7—Describe the relationship between illustrations and the text
- Week 35: RL.K.7—Describe the relationship between illustrations and the text
- Week 36: W.K.3—Narrate a single/series of events, in order, and provide a reaction
- Week 37: W.K.2—Compose information/explanatory texts in which they discuss a topic
- Week 38: W.K.3—Narrate a single/series of events, in order, and provide a reaction
- Week 40: W.K.3—Narrate a single/series of events, in order, and provide a reaction

<u>**Math**</u>
- Week 6: K.CC.1—Count to 100 by ones and by tens
- Week 7: K.CC.5—Count to answer "how many?"/Given a number from 1-20, county out that many
- Week 11: K.CC.3, K.CC.5, K.CC.6—Identify number of objects in one group as greater/less than or equal
- Week 12: K.CC.4A—Say number names in standard order, pairing each object
- Week 17: K.CC.1—Count to 100 by ones and by tens
- Week 18: K.CC.1, K.CC.2—Count forward beginning from a given number within the known sequence
- Week 19: K.CC.1, K.CC.2, K.CC.4—Understand the relationship between numbers and quantities
- Week 23: K.MD.3—Classify objects into given categories
- Week 24: K.G.5—Model objects in the world by drawing 2-D shapes and building 3-D shapes
- Week 28: K.OA.1, K.OA.2—Represent and Solve addition and subtraction within 10
- Week 20: K.OA.5—Fluently add and subtract within 5

<u>**Science**</u>
- Week 11: E.K.8B.3—Develop a device which would reduce heat from the sun
- Week 21: E.K.10.2—Develop questions to investigate ways to conserve Earth's resources
- Week 26: P.K.5A—Demonstrate an understanding of the solid and liquid states of matter
- Week 39: E.K.8A—Demonstrate an understanding of the pattern of seasonal changes on the Earth

<u>**Social Studies**</u>
- Week 5: CI.K.1, CI.K.2—Examine different roles and knowledge of how to be a good citizen
- Week 8: G.K.3.2—Identify cardinal and intermediate directions
- Week 10: G.K.1, G.K.3—Identify a sense of place, and recognize maps/graphs/representations of Earth
- Week 13: G.K.1.2, G.K.3.2—Demonstrate terms related to location, direction, size, and distance
- Week 16: CI.K.1, CI.K.3—Describe the role and responsibilities of authority figures
- Week 31: G.K3.2—Identify cardinal and intermediate directions
- Week 32: G.K3.2—Identify cardinal and intermediate directions

Contents by Topics

Coding
- Week 9
- Week 10
- Week 15
- Week 17
- Week 18
- Week 19
- Week 20
- Week 21
- Week 24
- Week 25
- Week 26

Digital Citizenship
- Week 5
- Week 16

Digital Literacy
- Week 1
- Week 2
- Week 6
- Week 7
- Week 8

Keyboarding
- Week 3
- Week 4
- Week 23

Robotics
- Week 13
- Week 14
- Week 30

Unplugged
- Week 2
- Week 3
- Week 11
- Week 18
- Week 20
- Week 22
- Week 35
- Week 37
- Week 39

- Week 27
- Week 28
- Week 29
- Week 30
- Week 31
- Week 32
- Week 33
- Week 34
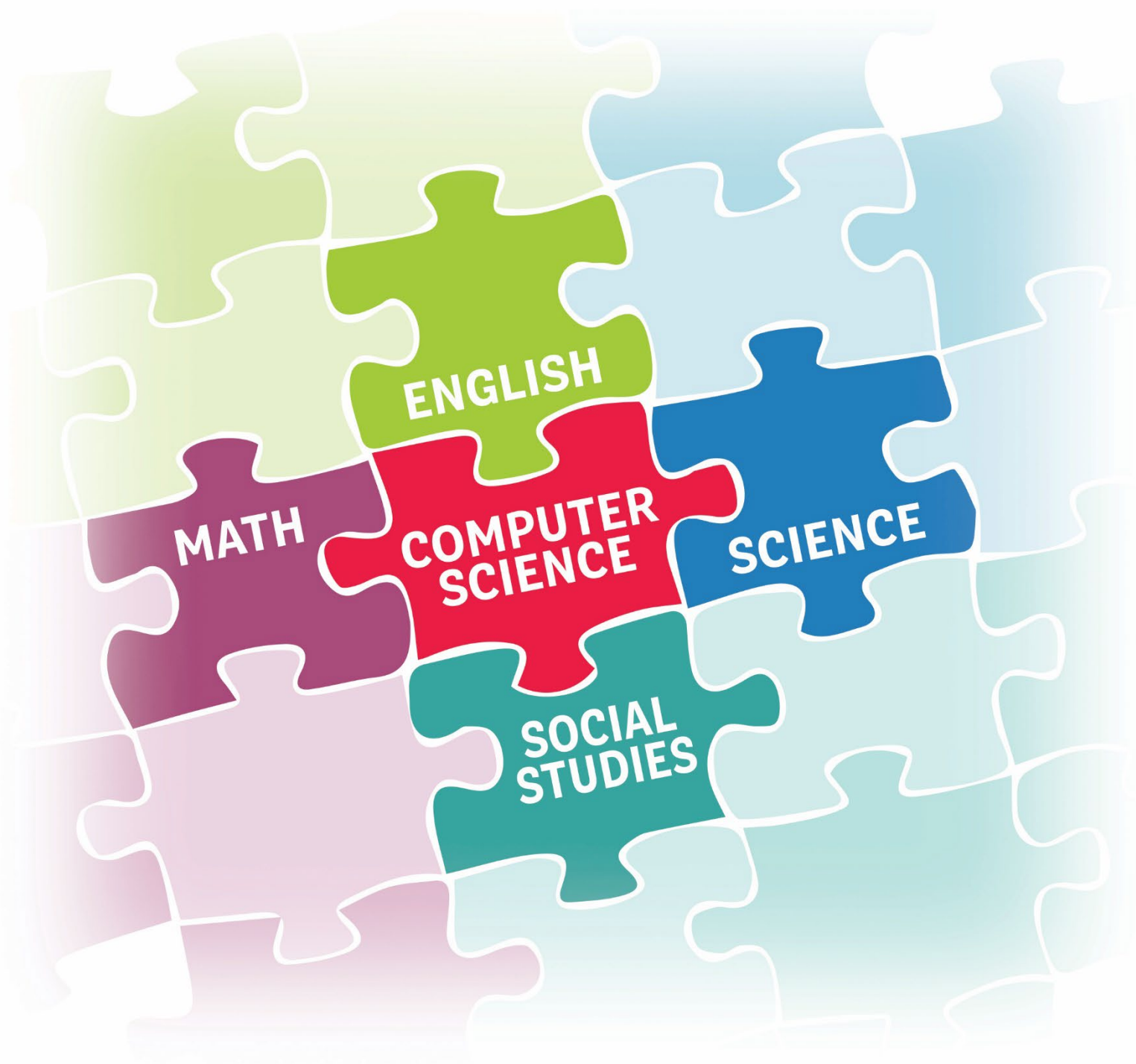- Week 35
- Week 36
- Week 38
- Week 40

Calendar/ Pacing per week:

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

Week	Title	Topics	CS Standard	Integrated Standard	Subject Integrated
1	Identifying Parts of a Computer	Digital Literacy	CS.1A.2	L.K.1a	ELA
2	Mouse Learning	Digital Literacy Unplugged	CS.1A.1	L.K.5a	ELA
3	Introduction to Keyboarding	Keyboarding Unplugged	CS.1B.1	RF.K.1d	ELA
4	Keyboarding Practice	Keyboarding	CS.1A.2 CS.1A.2a	RF.K.1d	ELA
5	Code.org: Safety in My Online Neighborhood → Account creation needed	Digital Citizenship	IC.1A.1	CI.K.1 CI.K.2	SS
6	Code.org: Learn to Drag	Digital Literacy	AP.1A.4 CS.1B.1 IC.1A.2 IC.1A.3	K.CC.1 RI.K.7	Math ELA
7	Code.org: Rhyme with That	Digital Literacy	AP.1A.4 CS.1B.1 IC.1A.2 IC.1A.3	RF.K.2.A RF.K.2.D K.CC.B.5	ELA Math
8	Code.org: Happy Maps	Digital Literacy	AP.1A.1 AP.1A.2 AP.1A.4 AP.1A.5	W.K.2 G.K.3.2	ELA SS
9	Code.org: Sequencing with Scrat	Coding	AP.1A.4	W.K.2 RL.K.3	ELA
10	Code.org: Programming with Scrat	Coding	AP.1A.2 AP.1A.4	G.K.1 G.K.3	SS
11	Code.org: Saving Scrat's Acorns	Unplugged	AP.1A.2 AP.1A.4	RI.K.2 K.CC.3 K.CC.6 K.CC.5 E.K.8B.3	ELA, Math, Science
12	Code.org: Programming with Rey and BB-8	Coding	AP.1A.2 AP.1A.4	K.CC.4A	Math
13	Graph Paper Programming	Robotics Unplugged	AP.1A.1 AP.1A.2	RF.K.1 K.CC.4A G.K.1.2 G.K.3.2	ELA, Math, SS
14	Letter Recognition Algorithm	Robotics	AP.1A.1	G.K.3.2 RF.K.1d RF.K.3a	ELA
15	Coding Beginning Sounds	Coding	AP.1A.1a	RF.K.3a	ELA

16	Digital Citizenship: Media Balance is Important	Digital Citizenship	IC.1A.1	C.I.K.1 C.I.K.3	SS
17	Code.org: Happy Loops	Coding	AP.1A.2 AP.1A.3 AP.1A.7	K.CC.1	Math
18	Number Recognition	Coding Unplugged	AP.1A.1 DA.1A.3	K.CC.1 K.CC.2	Math
19	Counting Objects	Coding	AP.1A.1 AP.1A.2	K.CC.1 K.CC.2 K.CC.4	Math
20	Coding Living and Nonliving Things	Coding Unplugged	AP.1A.1a AP.1A.1b	L.K.1A	ELA
21	Kodable: Beach Cleanup—GO GREEN! → Account creation needed	Coding	AP.1A.3 AP.1A.4 AP.1A.8	E.K.10.2 RF.K.1	Science ELA
22	Kodable: Hour of Code: ELA Integration	Unplugged	AP.1A.3a	RF.K.1 RF.K.1A	ELA
23	Drag and Drop Sorting	Keyboarding		K.MD.3	Math
24	Kodable: Maze Maker Challenge	Coding	AP.1A.3a	K.G.5	Math
25	Comparing Numbers with Alli-Gator	Coding	AP.1A.3 AP.1A.4 AP.1A.8a	K.CC.7	Math
26	Coding States of Matter	Coding	AP.1A.1 AP.1A.2	P.K.5A	Science
27	Coding Blending Sounds	Coding	AP.1A.1	RF.K.2 RF.K.2a	ELA
28	Kodable: Pizza Party	Coding	AP.1A.1 AP.1A.4 AP.1A.5	K.OA.1 K.OA.2	Math
29	Coding *The Very Hungry Caterpillar*	Coding	AP.1A.1 AP.1A.4	L.K.2	ELA
30	Coding With Magna Tiles	Coding Robotics	AP.1A.1	K.OA.5	Math
31	Code.org: Loops with Scrat	Coding	AP.1A.1 AP.1A.3	G.K3.2	Social Studies
32	Code.org: Loops with Laurel	Coding	AP.1A.2 AP.1A.3 AP.1A.4 AP.1A.7	G.K3.2	Social Studies
33	Code.org: Ocean Scene with Loops	Coding	AP.1A.2 AP.1A.3 AP.1A.4 AP.1A.7	SL.K.2 SL.K.3 SL.K.6	ELA
34	Code.org: The Big Event Jr.	Coding	AP.1A.2 AP.1A.4	RL.K.7	ELA

35	Code.org: Mini Project: On the Move with Play Lab	Coding	AP.1A.2 AP.1A.4	RL.K.7	ELA
36	Code.org: End of Course Project	Unplugged Coding	AP.1A.3	W.K.3	ELA
37	Kodable: Magic Machine	Unplugged	AP.1A.1	W.K.2	ELA
38	Kodable: Show What You Know	Coding	AP.1A.1 AP.1A.2	W.K.3	ELA
39	Kodable: If Flash, then Clap!	Unplugged	AP.1A.4	E.K.8A	Science
40	Kodable: Hour of Code: Advanced	Coding	AP.1A.3 AP.1A.5a AP.1A.8 AP.1A.8a	W.K.3	ELA

ENGLISH

MATH

COMPUTER
SCIENCE

SCIENCE

SOCIAL
STUDIES

Lessons and Activities

Week 1: Identifying Parts of a Computer

Lesson overview:	**Purpose:**
	In this lesson, students will learn the basic parts of computers, including the computer, monitor, desktop tower, keyboard, mouse, laptop, and tablet. This is an introduction to technology that they will use throughout school. Students will become familiar with the terms and how to draw letters by tracing key vocabulary words. Students will practice matching by drawing a line from the object to its name.
	Lesson:
	● Warm Up
	    ○ "What is a computer?" Let students give you the answers to what they think a computer is.
	● Watch the "What is a Computer for Kids? " video.
	● Identifying Computer Parts Activity
	    ○ Identifying Computer Parts worksheet and PowerPoint: As you are going through the PowerPoint, students will be finding and matching the pictures of the computer part on their worksheet. Once they find the picture, they will draw a line from the picture to the correct term. (Please do not rush through the slides because it allows students to see the picture and term on the board.)
	    ○ Once the PowerPoint and matching on the worksheet have been completed, the students will practice their handwriting skills by tracing each term on the worksheet.
	● Enrichment
	    ○ Computer Coloring Page: Students will color the page to reinforce the parts of the computer that they have just learned. (For additional enrichment, the students can rewrite their terms for each part onto the coloring page.)

Lesson links/ resources:	● What is a Computer for Kids? ● Identifying Computer Parts PowerPoint ● Identifying Computer Parts Worksheet ● Computer Coloring Page
CS standards addressed:	Students will be able to: ● Identify parts of a computer. Standards: ● **CS.1B.1—**Describe how internal and external parts of computing devices function to form a system.
Time needed:	**Total Time: 60 min** ● Warm Up: **5 min** ● Activity: **30 min** ● Enrichment: **25 min**
Materials needed:	Teacher: ● Projector/smartboard with sound ● Identifying Parts of a Computer PowerPoint Students: ● Identifying Parts of a Computer Worksheet ● Computer Coloring Page ● Pencil ● Crayons (or other coloring utensil)

Subject integrated:	ELA
Other standards addressed:	**L.K.1a**—Print many upper- and lowercase letters.
Vocabulary:	Computer: This is a machine that takes information, stores information, takes action on information, and gives information back. Desktop tower: This houses the "brain" of the computer called a central processing unit (CPU). Input: This is information that is put into a computer. Keyboard: This is a piece of equipment that allows you to type letters, numbers, and symbols into the computer. Laptop: This is a computer that can move from place to place, and it has a keyboard, mouse, and screen built in. Monitor: This is a screen that allows you to see letters, numbers, symbols, pictures, and videos. Mouse: This is a piece of equipment that allows you to move the cursor on your computer screen to click on pictures (icons), links, and videos. Output: This is information given by the computer or piece of equipment connected to the computer. Tablet: This is a computer that can be moved from place to place, and it has a touch screen.
Notes:	As your students are completing the worksheet, make sure that you are going through the PowerPoint to show the various images/terms. This will help matching.

Week 2: Mouse Learning

Lesson overview:	**Purpose:**
	The games help beginning computer users learn mouse skills through hand-eye coordination by dragging, dropping, clicking, double-clicking, and scrolling. Pick a different activity each day of the week to give students practice using a mouse.
	Lesson:
	● Refresher
	    ○ Review the parts of the computer with your students. Once you have gone back over the parts of the computer, show the video "How Do Computers Work?"
	● Warm Up
	    ○ Show the "Using Your Computer Mouse (for kids)" video.
	● Mouse Practice
	    ○ Online: Go to the mouse practice website. Let the students choose a game to play to practice using the mouse. (If your students are using a laptop or tablet, that's okay! They can still use the practice to operate a trackpad or touchscreen.)
	    ○ Unplugged: Provide students with coloring utensils and the Mouse Worksheet. Guide students through the completion of the worksheet.
	● Student Voice
	    ○ "How does a mouse help a person use a computer?"
	    ○ Give students the opportunity to explain, in their own words, how they think a mouse is helpful.

Lesson links/resources:	**Video Links**
	● How Do Computers Work?
	● Using Your Computer Mouse (for kids)
	Mouse Practice
	● Mouse Practice (online)
	    ○ Apple Catch
	    ○ Coyote Concentration (card-matching game)
	    ○ Desert Dive
	    ○ Frostbite
	    ○ Helipopper
	    ○ Penguin Drop
	    ○ Pickle Pop
	    ○ Pig Pile
	    ○ Simon Sees
	● Alphabetical Order
	● Mouse Practice (Unplugged)
	● Alphabet Slider
	● Alphabet Magnets

CS standards addressed:	Students will be able to: ● Operate a mouse/keypad. ● Find/select letters on a keyboard. Standards: ● **CS.1B.1—**Describe how internal and external parts of computing devices function to form a system.
Time needed:	**Total Time: 60 min** ● Refresher**: 15 min** ● Warm-Up: **5 min**

	• Mouse Practice: **15 min**
	    ◦ Choose your first option from the links/resources
	• Mouse Practice: **15 min**
	    ◦ Choose a different option from the links/ resources
	• Student Voice: **5 min**
	• Keyboarding Activity: **5 min**

Materials needed:	Teacher: • Projector/smartboard with sound • Identifying Parts of a Computer PowerPoint presentation Students: • Student devices with access to the internet • Mouse Worksheet
Subject integrated:	ELA
Other standards addressed:	**L.K.5a—**Sort common objects into categories (e.g., shapes, foods) to gain a sense of the concepts the categories represent.
Vocabulary: (please see notes)	Computer: This is a machine that takes information, stores information, takes action on information, and gives information back. Data: This is any information put into the computer or given back by the computer. Desktop tower: This houses the "brain" of the computer called a central processing unit (CPU) Input: This is information that is put into a computer. Keyboard: This is a piece of equipment that allows you to type letters, numbers, and symbols into the computer. Laptop: This is a computer that can move from place to place, and it has a keyboard, mouse, and screen built in. Monitor: This is a screen that allows you to see letters, numbers, symbols, pictures, and videos. Mouse: This is a piece of equipment that allows you to move the cursor on your computer screen to click on pictures (icons), links, and videos. Output: This is information given by the computer or piece of equipment connected to the computer. Tablet: This is a computer that can be moved from place to place, and it has a touch screen.
Notes:	• Even though this lesson is predominantly for practice using a mouse, your class can use a tablet. This will still allow students to practice using a touchscreen device. • There are several resources to use mouse practice. These can be used in center rotations, 20 min a day, separate at the teacher's discretion. • The vocabulary is for the teacher's understanding. Students are not expected to spell terms or memorize definitions.

Week 3: Introduction to Keyboarding

Lesson overview:	**Purpose:**
	Students will identify the left/right side of the keyboard, and they will learn where and how to place their hands on the keyboard.
	Lesson:
	● L / R Keyboard Coloring Sheet
	    ○ Students will identify the left/right side of the keyboard. Students will color the left side of the keyboard to coordinate with the left hand. They will color the right side of the keyboard to coordinate with the right hand.
	    ○ The teacher will call out various letters/numbers to help students practice finding the letters/numbers on a keyboard.
	● Keyboarding Practice
	    ○ Online Option: Students can spend 15 min playing Astro Bubbles keyboarding game.
	    ○ Unplugged Option: Using the keyboard coloring page, call out random letters and numbers and have your students find them and place their finger on them.

Lesson links/resources:	Online:
	● Astro Bubbles Keyboard Practice
	● Big Brown Bear
	● Read Today
	Unplugged:
	● Keyboard Callout
	    ○ **Paper keyboard**: Using a paper keyboard, the teacher will call out letters, numbers, symbols, and/or words for students to "type" on their keyboard.
	    ○ **Computer with no internet:** The teacher will call out letters, numbers, symbols, and/or words. Students will use their keyboard to type into a blank word processing document on their computer/tablet.
	● Keyboard Bingo
	    ○ Preparation: The teacher will print squares with letters, numbers, and symbols (4-5 of each letter, 1-2 of each number/symbol). The teacher will cut out and laminate each square, then use a piece of tape or glue to adhere a magnet to each square. The teacher will distribute an equal number of random squares to students.
	    ○ The teacher will project a keyboard onto a smartboard.
	    ○ The teacher will call out letters, numbers, symbols, or words for students to find using their preprinted squares.
	    ○ Students will raise their hands if they have the key that the teacher calls out. The teacher will choose a student to place their key on the board.
	● L / R Keyboard Coloring Sheet

CS standards addressed:	Students will be able to: ● Find/select letters on a keyboard. Standards: ● **CS.1B.1—**Describe how internal and external parts of computing devices function to form a system.
Time needed:	**Total Time:** **60 min** ● Coloring Sheet **15 min**

	● Keyboard Practice #1 **15 min** 　　○ Using the [L / R Keyboard Coloring Sheet](#) choose an unplugged activity (under links/resources) to implement with your students. ● Keyboard Practice #2 **15 min** 　　○ Choose a different keyboarding activity (under links/resources) for your students to complete. ● Keyboard Practice #3 **15 min** 　　○ Choose a different keyboarding activity (under links/resources) for your students to complete.
Materials needed:	Teacher: ● Smartboard/projector with sound ● Identifying Parts of a Computer PowerPoint presentation Students: ● Student devices with access to the internet ● [L / R Keyboard Coloring Sheet](#) ● Coloring utensils
Subject integrated:	ELA
Other standards addressed:	**RF.K.1d—**Recognize and name all upper- and lowercase letters of the alphabet
Vocabulary:	<u>Keyboard</u>: This is a piece of equipment that allows you to type letters, numbers, and symbols into the computer. <u>Mouse</u>: This is a piece of equipment that allows you to move the cursor on your computer screen to click on pictures (icons), links, and videos.
Notes:	

Week 4: Keyboarding Practice

Lesson overview:	**Purpose:**
	These games will help students learn where letters are located on the keyboard. The "hunt and peck" method is great at this age.
	Lesson:
	● Introduce a keyboard to students and introduce them to finding letters on the keyboard.
	● Ideas to implement this lesson:
	   ○ Teachers can divide the class into appropriate groups.
	   ○ Group 1 works on Astro Bubble, Group 2 works on Brain Game, Group 3 works on Keyboard Zoo.
	   ○ Switch after 15 min until all groups have played each game.
	   ○ Separate the games into days and allow all students to work on one game per day.
	Ex: Monday: Astro Bubble (15 min), Wednesday: Brain Game (15 min), Friday: Keyboard Zoo (15 min)

Lesson links/resources:	● Astro Bubble ● Key Memory Brain Game ● Keyboard Zoo
CS standards addressed:	Students will be able to: ● Find/select letters on a keyboard. Standards: ● **CS.1A.2**—Use appropriate terminology in identifying and describing the function of common physical components of computing systems (hardware). ● **CS.1A.2a**—Students should be able to identify and describe the function of external hardware, such as desktop computers, laptop computers, tablet devices, monitors, keyboards, mouses, and printers.

Time needed:	**Total Time: 65 min**
	● Review L-R Coloring Sheet from previous lesson: **5 min**
	● Keyboard Practice #1 **15 min**
	   ○ Choose a keyboarding activity (under links/resources) for your students to complete.
	● Keyboard Practice #2 **15 min**
	   ○ Choose a different keyboarding activity (under links/resources) for your students to complete.
	● Keyboard Practice #3 **15 min**
	   ○ Choose a different keyboarding activity (under links/resources) for your students to complete.
	● Keyboard Practice #4 **15 min**
	   ○ Choose a different keyboarding activity (under links/resources) for your students to complete.

Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet ● If your students are using a touchscreen device, they will use the keyboard that is on the game window.
Subject integrated:	ELA

Other standards addressed:	**RF.K.1d—**Recognize and name all upper- and lowercase letters of the alphabet.
Vocabulary:	<u>Keyboard</u>: This is a piece of equipment that allows you to type letters, numbers, and symbols into the computer.
Notes:	Introduce the keyboard by using the keyboard coloring sheet.

Week 5: Code.org, Course A, Lesson 1—Safety in My Online Neighborhood

Lesson overview:	**Purpose:** This lesson focuses on privacy and security. Students learn how to protect personal information and gain a deeper understanding of their data privacy rights so they can advocate for themselves and others. **Lesson:** ● Watch: My Online Neighborhood ● Read: Safety, It Rules! ● Explore: Internet Field Trip ● Pause & Think Moment ● Extended Learning (optional)
Lesson links/resources:	Lesson 1: Safety in My Online Neighborhood
CS standards addressed:	Students will be able to: ● Compare how staying safe online is like staying safe in the real world. ● Discover that the internet can be used to visit faraway places and learn new things. ● Explain rules for traveling safely on the internet. Standards: ● **IC.1A.1—**Compare how people live and work before and after the implementation or adoption of new computing technology.
Time needed:	**Total Time: 60 min** ● Keyboard Practice #1 **20 min** ○ Choose a different keyboarding activity (under links/resources) for your students to complete. ● Watch: My Online Neighborhood **5 min** ● Read: Safety, It Rules! **5 min** ● Explore: Internet Field Trip **15 min** ● Pause & Think Moment **15 min** ● Extended Learning (optional)
Materials needed:	For the teachers: ● Smartboard/projector with sound ● Safety in My Online Neighborhood: Lesson Slides - Slide Deck ● Safety in My Online Neighborhood: My Online Neighborhood - Lesson Video (Download) Students: ● Student devices with access to the internet ● Safety in My Online Neighborhood: Pause and Think Moment - Student Handout ● Safety in My Online Neighborhood: Safety: It Rules! - Poem Poster
Subject integrated:	Social Studies
Other standards addressed:	● **CI.K.1—**Examine how individuals play different roles and exercise good citizenship. ● **CI.K.2—**Demonstrate knowledge of how to be a good citizen.
Vocabulary:	Online: Using a computer, phone, or tablet to visit a website or app Website or App: A place you can visit on the internet.

Notes:

Week 6: Code.org, Course A, Lesson 2—Learn to Drag

Lesson overview:	**Purpose:**
	This skill-building lesson will give students an idea of what to expect when they head to the computer lab. This begins with a brief discussion introducing them to computer lab manners, then they will progress into using a computer to complete online puzzles.
	Lesson:
	The main goal of this lesson is to build students' experience with computers. By covering the most basic computer functions such as clicking, dragging, and dropping, we are creating a more equal playing field in the class for future puzzles.
	● Warm Up
	   ○ Introduction
	● Main Activity
	   ○ Learn to Drag and Drop
	● Wrap Up
	   ○ Reflection
	● Extended Learning
	   ○ Give students time to come up with their own puzzles.

Lesson links/resources:	Lesson 2: Learn to Drag and Drop
CS standards addressed:	Students will be able to: ● Recognize what is expected when students transition into the computer lab. ● Use appropriate terminology when referring to a computer mouse, trackpad, or touchscreen. Standards: ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **CS.1B.1**—Describe how internal and external parts of computing devices function to form a system. ● **IC.1A.2**—Work respectfully and responsibly with others online. ● **IC.1A.3**—Keep login information private and log off of devices appropriately.
Time needed:	**Total Time:** **60 min** ● Keyboarding Practice **15 min** ● Warm Up **10 min** ● Main Activity **20 min** ● Wrap Up **5 min** ● Extended Learning **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sounds ● 20/20/20 Rule - Resource ● Getting Started - Creating a Class Section - Video ● Wiggles-Go Noodle - Video Students: ● Student devices with access to the internet ● Pair Programming - Student Video
Subject integrated:	ELA Math

Other standards addressed:	ELA
	   ● **RI.K.7—**With prompting and support, describe the relationship between illustrations and the text in which they appear (e.g., what person, place, thing, or idea in the text an illustration depicts).
	Math
	   ● **K.CC.1—**Count to 100 by ones and by tens.

Vocabulary:	<u>Click</u>: Press the mouse button. <u>Drag:</u> Click your mouse button and hold as you move the mouse pointer to a new location. <u>Drop</u>: Release your mouse button to "let go" of an item that you are dragging. <u>Mouse/Trackpad</u>: Flat controller used to move the cursor and perform other functions on the computer.
Notes:	

Week 7: Rhyme With That

Lesson overview: 	**Purpose:** Students will use the computer mouse to drag and drop a letter to form Consonant-Vowel-Consonant (CVC) words. They will further identify how many rhyming words are located on each slide. **Lesson:** ● Students will practice using the mouse to drag and drop letters to form CVC words using slides 1-5. 　○ Note: Teachers may want to complete slides 1-5 as a whole class instruction so students have whole group practice practicing the initial, medial, and ending sounds of each CVC word. ● Next students will drag and drop the words to the appropriate location on the T-chart on slides 6-10. Then count how many words are in each column.
Lesson links/resources:	Rhyme with That
CS standards addressed:	Students will be able to: ● Use a mouse to drag and drop items Standards: ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **CS.1B.1**—Describe how internal and external parts of computing devices function to form a system. ● **IC.1A.2**—Work respectfully and responsibly with others online. ● **IC.1A.3**—Keep login information private and log off of devices appropriately.
Time needed:	**Total Time: 60 min** ● Keyboard practice #1 **15 min** ● Keyboard practice #2 **15 min** ● Rhyme with That **30 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet ● Rhyme With That Student Slides
Subject integrated:	ELA Math
Other standards addressed:	ELA ● **RF.K.2.A**—Recognize and produce rhyming words. ● **RF.K.2.D**—Isolate and pronounce the initial, medial vowel, and final sounds (phonemes) in three-phoneme (consonant-vowel-consonant, or CVC) words. (This does not include CVCs ending with /l/, /r/, or /x/.) Math ● **K.CC.B.5**—Count to answer "how many?" questions about as many as 20 things arranged in a line, a rectangular array, or a circle, or as many as 10 things in a scattered configuration; given a number from 1-20, count out that many objects.

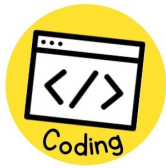|	
Vocabulary:	Click: Press the mouse button. Drag: Click your mouse button and hold as you move the mouse pointer to a new location. Drop: Release your mouse button to "let go" of an item that you are dragging. Mouse/Trackpad: Flat controller used to move the cursor and perform other functions on the computer.
Notes:	

Week 8: Code.org, Course A, Lesson 3—Happy Maps

Lesson overview:	**Purpose:**
	This context-setting lesson brings together teams with a simple task: Get the "flurb" to the fruit. Students will practice writing precise instructions as they work to translate instructions into the symbols provided.
	Lesson:
	● Warm Up
	   ○ Help the students process how to take a big problem and break it down into smaller steps.
	● Main Activity
	   ○ In this exercise, the class will get map cards that have a predefined start space (flurb) and end space (fruit). Students will need to get the flurbs to the fruit on each card using the arrows provided.
	● Wrap Up
	   ○ Discuss vocabulary learned. Have students record their "Feelings Face" in their reflection journal and draw their own map for the "flurb" to get to the fruit.

Lesson links/resources:	Lesson 3: Happy Maps
CS standards addressed:	Students will be able to: ● Decode and run a program created by someone created by someone else. ● Identify and address bugs or errors in sequenced instructions. ● Translate an algorithm into a program. Standards: ● **AP.1A.1**—Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks. ● **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **AP.1A.5**—Develop plans that describe a program's sequence of events, goals, and expected outcomes.
Time needed:	**Total Time: 65 min** ● Warm Up **5 min** ● Main Activity **40 min** ● Wrap Up **10 min** ● Keyboarding Activity **10 min**
Materials needed:	For the Teachers: ● Smartboard/projector with sound ● Happy Map Cards - Worksheet Answer Key For the Students: ● Student devices with access to the internet ● Feeling Faces - Emotion Image - Resource ● Happy Map Cards - Worksheet ● Happy Map Game Pieces - Manipulatives ● Happy Maps - Unplugged Video (Download)
Subject integrated:	ELA Social Studies

Other standards addressed:	ELA
	    ● **W.K.2**—Use a combination of drawing, dictation, and writing to compose informative/explanatory texts in which they name what they are writing about and supply some information about the topic.
	Social Studies
	    ● **G.K.3.2**—Identify cardinal and intermediate directions.

Vocabulary:	Algorithm: A list of steps to finish a task Debugging: Finding and fixing problems in an algorithm or program Program: An algorithm that has been coded into something that can be run by a machine
Notes:	

Week 9: Code.org, Course A, Lesson 4—Sequencing With Scrat

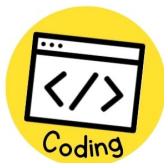| Lesson overview:	Purpose:

	In this skill-building lesson, students will develop sequential algorithms to move a squirrel character from one side of a maze to the acorn at the other side. To do this, they will stack code blocks together in a linear sequence.
	
	Lesson:
	● Warm Up
	 ○ Review expectations/norms and procedures for being in the computer lab/working with computers.
	 ○ Bridging Activity (choose one)
	  ■ **Dragging and dropping algorithms:** Project one of the maps from the "Happy Maps" activity and display it for the students to see. On a projector or in front of the class, put some direction blocks from the manipulatives in random order and practice dragging and dropping by pressing your finger on one of the paper pieces and moving it across the screen. Explain that you can "click to select this block by tapping your finger on it, or by tapping your finger on it and moving it. To drop the block, release your finger.
	  ■ **Previewing online puzzles as a class:** Project a puzzle from the lesson. Show the class how to click on a block and place it in the correct spot by dragging and dropping. Purposely make mistakes such as clicking the background or dropping the image before it's at the right spot. Ask for help from volunteers in the class when you run into these problems and help them use the skills they developed in the last unplugged lesson to make things right.
	● Main Activity
	 ○ This will teach students how to use Code.org to complete online puzzles.
	 ○ Watch the Pair Programming video with your students, then assign them to pairs. This should help students start off in the right direction.
	● Show Video
	 ○ **Programming with Blocks:** Lead the students as a whole group through the first few skills-building lessons. Allow students to work independently as understanding is achieved.
	● Wrap Up
	 ○ Ask students to draw a sequence for getting ready to go to the area of choice (e.g., computer lab, special class, playground, etc.). Make a do or don't list.
	 ○ Have them draw a "Feelings Face" that shows how they felt about today's lesson.
	● Keyboarding/Mouse Practice
	 ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Lesson 4: Sequencing with Scrat

CS standards addressed:	Students will be able to: ● Experiment with standard block-based programming actions such as clicking, dragging, dropping, etc. ● Model proper computer lab behaviors. Standards: ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Bridging Activity **10 min** ● Main Activity **20 min** ● Wrap Up **5 min** ● Keyboard/Mouse Practice **15 min**
Materials needed:	For the Teachers (links) ● Smartboard/projector with sound ● 20/20/20 Rule - Resource ● Wiggles - GoNoodle - Video For the Students (links) ● Student devices with access to the internet ● Pair Programming - Student Video ● Unplugged Blockly Blocks (Grades K-1) - Manipulatives
Subject integrated:	Math Social Studies
Other standards addressed:	Math ● **K.CC.2**—Count forward beginning from a given number within a known sequence (instead of having to begin at 1). Social Studies ● **G.K.1.2**—Demonstrate terms related to location, direction, size, and distance. ● **G.K.3.2**—Identify cardinal and intermediate directions.
Vocabulary:	Click: Press the mouse button. Double-Click: Press the mouse button very quickly. Drag: Click your mouse button and hold as you move the mouse pointer to a new location. Drop: Release your mouse button to "let go" of an item that you are dragging.
Notes:	

Week 10: Code.org, Course A, Lesson 5—Programming With Scrat

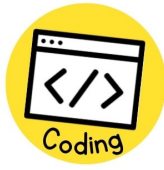|	

Lesson overview:
 	
	<u>Purpose:</u>
	In this skill-building lesson, students will continue to develop sequential algorithms. Students will develop programming and debugging skills on a computer platform. The block-based format of these puzzles helps students learn about sequences and concepts without having to worry about perfecting syntax.
	<u>Lesson:</u>
	● Warm Up
	● **Review/Teach cardinal directions:** Make the connection that "up" and "down" are the same as "north" and "south." Code.org provides a [Compass Rose](#) handout to quickly teach this if it is new to the students.
	● Main Activity
	   ○ Use a "prediction" puzzle (provided) to walk through existing code with your students to predict what Scrat will do.
	   ○ Show Video: Pair programming:
	   ○ Pair your students to complete the skills-building lessons.
	   ○ Show video: Debugging with the Step Button:
	   ○ Allow partners to complete the remaining two skills-building lessons. (Challenges are available for pairs that progress quickly)
	● Wrap Up
	   ○ Have students record their "Feelings Face" for today's activity. If time permits, have them draw their own puzzle to get Scrat to the acorn.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	[Lesson 5: Programming with Scrat](#) Cardinal Directions Activity: https://code.org/curriculum/course1/2/Teacher#Activity1
CS standards addressed:	Students will be able to: ● Build a computer program from a set of written instructions. ● Choose appropriate debugging practices when solving problems. ● Construct a program by reorganizing sequential movements. Standards: ● **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** ● Warm Up **3 min** ● Main Activity **30 min** ● Wrap Up **5 min** ● Keyboarding/Mouse Practice **22 min** (Throughout the week)
Materials needed:	Teacher: ● Smartboard/projector with sound For the Students: ● Student devices with access to the internet ● [Debugging With the Step Button - Video (Download)](#) ● [Feeling Faces Emotion Image - Resource](#)

	• [Pair Programming - Student Video](#)
Subject integrated:	Social Studies
Other standards addressed:	• **G.K.1.2—**Demonstrate terms related to location, direction, size, and distance. • **G.K.3.2—**Identify cardinal and intermediate directions
Vocabulary:	Algorithm: A list of steps to finish a task Bug: Part of a program that does not work correctly Debugging: Finding and fixing problems in an algorithm or program Program: An algorithm that has been coded into something that can be run by a machine Programming: The art of creating a program
Notes:	

Week 11: Saving Scrat's Acorns

Lesson overview: Unplugged	<u>Purpose:</u> Help Scrat retrieve his acorns before the sun melts the ice. <u>Lesson</u> Scrat loves acorns! But, beware, the sun is quickly melting and cracking the ice. Help Scrat count, compare, identify, design, and build a tent as he makes his way across the ice to gather his acorns. ○ Note: Teachers may find this activity is too long for kindergarteners to complete in one day. The different academic areas can be spread out over several instructional days if the teacher wishes to do so.
Lesson links/resources:	Saving Scrat's Acorns Saving Scrat's Acorns Student Handout
CS standards addressed:	Students will be able to: ● Build a computer program from a set of written instructions. ● Choose appropriate debugging practices when solving problems. ● Construct a program by reorganizing sequential movements. Standards: ● **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** ● Math connection activities **15 min** ● ELA connection activity **15 min** ● Science connection activity **30 min**
Materials needed:	Teacher: ● Smartboard/projector with sound For the Students: ● Student handout ● Crayons, colored pencils, or markers ● Saving Scrat's Acorns Student Handout ● Legos (optional)
Subject integrated:	ELA Math Science
Other standards addressed:	ELA ● **RI.K.2—**With prompting and support, identify the main topic and retell key details of a text. Math ● **K.CC.3—**Write numbers from 0 to 20. Represent a number of objects with a written numeral 0-20 (with 0 representing a count of no objects). ● **K.CC..6—**Identify whether the number of objects in one group is greater than, less than, or equal to the number of objects in another group, e.g., by using matching and counting strategies. ● **K.CC.5—**Count to answer "how many?" questions about as many as 20 things arranged in a line, a rectangular array, or a circle, or as many as 10 things in a scattered configuration; given a number from 1-20, count out that many objects.

	Science ● **E.K8B.3—**Develop a device (i.e., umbrella, shade structure, or hat) which would reduce heat from the sun (temperature) using an engineering design process to define the problem, design, construct, evaluate, and improve the device.
Vocabulary:	Algorithm: A list of steps to finish a task Bug: Part of a program that does not work correctly Debugging: Finding and fixing problems in an algorithm or program Program: An algorithm that has been coded into something that can be run by a machine Programming: The art of creating a program
Notes:	

Week 12: Code.org, Course A, Lesson 6—Programming With Rey and BB-8

Lesson overview:
 	
	Purpose:
	In this skill-building lesson, students will use their newfound programming skills in more complicated ways to navigate a tricky course with BB-8.
	Lesson:
	● Warm Up
	  ○ Ask the students how they felt about the last lesson. Using the questions listed under "Warm Up," form a brief review of programming and debugging. Review vocabulary learned. Introduce the lovable robot BB-8 from Star Wars to students.
	● Main Activity
	  ○ **Show video:** Programming with Rey and BB-8. Pair students to complete skills-building activities. (Challenges are available for pairs that progress quickly.)
	● Wrap Up
	  ○ Students answer, "What was today's lesson about?" Have students record their "Feelings Face" for today's activity. If time permits, have them draw the commands used to guide BB-8 through the puzzle.

Lesson links/resources:	Lesson 6: Programming with Rey and BB-8
CS standards addressed:	Students will be able to: ● Recognize problems or "bugs" in a program and develop a plan to resolve the issues. ● Sequence commands in a logical order. Standards: ● **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** ● Warm Up **15 min** ● Main Activity **30 min** ● Wrap Up **15 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet ● Feeling Faces - Emotion Image - Resource ● Pair Programming - Student Video
Subject integrated:	Math Social Studies
Other standards addressed:	Math ● **K.CC.4A—**When counting objects, say the number names in the standard order, pairing each object with one and only one number name and each number name with one and only one object. Social Studies ● **G.K.1.2—**Demonstrate terms related to location, direction, size, and distance. ● **G.K.3.2—**Identify cardinal and intermediate directions.

Vocabulary:	Algorithm: A list of steps to finish a task
	Bug: Part of a program that does not work correctly
	Debugging: Finding and fixing problems in an algorithm or program
	Program: An algorithm that has been coded into something that can be run by a machine
	Programming: The art of creating a program
Notes:	

Week 13: Graph Paper Programming

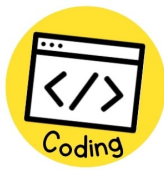| Lesson overview:	**Purpose:**
	Help students understand that writing an algorithm is just like reading a book: You follow from left to right and top to bottom. **Lesson:** ● Warm Up 　○ Choose a book to read. 　○ Suggestions: 　　■ Talk about the direction we read in (left to right) and the order (first page to last). 　　■ What happens if we read in the wrong direction or out of order? 　　■ Robots and computers must be giving instructions in the right order for them to do what you want. These directions are called these algorithms. ● Main Activity 　○ Talk about the graphing sheets and arrow cards. Teacher could use a smart board to project the pdf file found in Lesson links. 　○ We follow writing an algorithm the same way we read a book (left to right). ● Wrap Up 　○ Have the teacher choose an object in the classroom that the students are to guide the teacher to. 　○ They will work on building an algorithm as a group to guide the teacher to the object. 　○ They will have to record the algorithm from left to right.
Lesson links/resources:	[Graph Paper Programming](#)
CS standards addressed:	Students will be able to: ● Write an algorithm left to right. **Standards:** **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks. **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Time needed:	**Total Time: 60 min** ● Warm Up activity **15 min** ● Main Activity **30 min** ● Wrap Up **15 min**
Materials needed:	Teacher: ● Smartboard/projector with sound ● Storybook Students: ● Graph Paper for Programming Worksheet (see link above)
Subject integrated:	ELA Social Studies Math

Other standards addressed:	- **RF.K.1—**Demonstrate understanding of the organization and basic features of print. - **K.CC.4A—**When counting objects, say the number names in the standard order, pairing each object with one and only one number name and each number name with one and only one object. - **G.K.1.2—**Demonstrate terms related to location, direction, size, and distance. - **G.K.3.2—**Identify cardinal and intermediate directions
Vocabulary:	Algorithm: A list of steps to finish a task Bug: Part of a program that does not work correctly Debugging: Finding and fixing problems in an algorithm or program Program: An algorithm that has been coded into something that can be run by a machine Programming: The art of creating a program
Notes:	

Week 14: Letter Recognition Algorithm

Lesson overview:	Purpose:
	The students will learn how to program an algorithm to form the letters that have been learned this year.
	Lesson:
	Letter Recognition with Robot
	● Warm Up
	   ○ Talk about letters and letter sounds.
	   ○ What letters have we learned this year?
	● Main Activity
	   ○ We are going to learn how to program an algorithm to form these letters.
	   ○ The teacher will use one of the letters that have already been introduced this year to model how to write/build the algorithm.
	   ○ Use the 6x6 letter cards and have students write/build the algorithm that follows the formation of the letters.
	   ○ Students can work with partners to write the algorithm.
	   ○ Students will switch with their partner to test their algorithm.
	● Wrap Up
	   ○ What words begin with the letter sound?
	   ○ Use the various letter algorithms to review the coding process and letter recognition of various letters.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.
	*** Explain how to write the algorithm and switch with partner to test.

Lesson links/resources:	Letter Maps
CS standards addressed:	Students will be able to: ● Write an algorithm. Standards: ● **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks.
Time needed:	**Total Time: 60 min** ● Letter 1: **15 min** ● Letter 2: **15 min** ● Letter 3: **15 min** ● Keyboard Practice **15 min**
Materials needed:	Teacher: ● Smartboard/projector with sounds Students: ● 6x6 grids ● Letter maps ● Robot (optional)
Subject integrated:	ELA Social Studies
Other standards addressed:	ELA ● **RF.K.1d—**Recognize and name all upper- and lowercase letters of the alphabet.

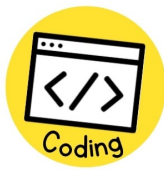|	
	● **RF.K.3a**—Demonstrate basic knowledge of one-to-one letter-sound correspondences by producing the primary sound of many of the most frequent sounds for each consonant. Social Studies ● **G.K.3.2**—Identify cardinal and intermediate directions (north, northeast, northwest, southeast, southwest, east, and west).
Vocabulary:	Algorithm: A list of steps to finish a task
Notes:	

Week 15: Coding Beginning Sounds

Lesson overview:	**Purpose:** The purpose of this activity is to identify beginning sounds and create algorithms.
	 Lesson:
	 • Warm Up:
	     ○ Review letters and sounds (with flashcards or other resources of your choosing).
	 • Main Activity:
	     ○ Using a copy of the beginning sound map, students will color the pictures that have the same sound as the letter in the corner.
	     ○ Once students have colored in the appropriate squares, they will write an algorithm to move from the starting square (top left-hand square) to all pictures that start with the beginning sound for that puzzle.
	 • Wrap Up:
	     ○ The teacher will give the student a word and the student will produce the beginning sound.
	 • Keyboarding/Mouse Practice
	     ○ If time remains, have students use resources from above to practice using the keyboard/mouse.
	 Note: Teachers may choose to only use the letters that have been covered.

Lesson links/resources:	[Beginning Sounds Map](#)
CS standards addressed:	Students will be able to: • Create an algorithm using beginning sounds. Standards: • **AP.1A.1a**—Students should be able to create and follow algorithms.
Time needed:	**Total Time:** 60 min • Warm Up **5 min** • Main Activity **30 min** • Wrap Up **10 min** • Keyboarding Practice or Letter Recognition Activity **15 min**
Materials needed:	Teacher: • Smartboard/projector with sound Students: • 6x6 sound cards • Color pencil/pencil/crayon
Subject integrated:	ELA
Other standards addressed:	**RF.K.3a**—Demonstrate basic knowledge of one-to-one letter-sound correspondences by producing the primary sound or many of the most frequent sounds for each consonant.
Vocabulary:	Code: Giving instructions to a computer or robot Algorithm: A list of steps to finish a task
Notes:	

Week 16: Digital Citizenship: Media Balance is Important

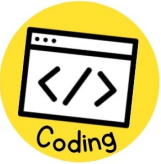|	

Lesson overview:	Purpose:
	Students consider the feelings of themselves and others when making decisions about when, where, and how to use technology.
	Lesson:
	• Warm Up:
	   ○ Before the lesson, introduce the Digital Citizens characters by having students complete the coloring book. Each character represents one of the six digital citizenship topics.
	   ○ Play the "Media Balance is Important" video.
	• Main Activity:
	   ○ The Digital Citizens showed us the ways they keep their technology use in balance. Now we are going to take a closer look at a few of the scenes from the song.
	• Wrap Up:
	   ○ Today we watched the Digital Citizens show us how they keep their online and offline activities in balance. Now I'd like you to think about what media balance means for you.
	• Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Media Balance is Important
CS standards addressed:	Students will be able to • Know when and why to take breaks from device time. • Consider the feelings of people around them, even when engaged in fun online activities. Standards: • **IC.1A.1—**Compare how people live and work before and after the implementation or adoption of new computing technology.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **10 min** • Wrap Up **5 min** • Keyboarding Activity **20 min** (Throughout the week) • Letter Recognition Practice **15 min** (See Week 14) (Throughout the week)
Materials needed:	Teacher: • Smartboard/projector with sound • Lesson Slides • Media Balance is Important Students: • Pause and Think Handout • Coloring Book
Subject integrated:	Social Studies
Other standards addressed:	**C.I.K.1—**Examine how individuals play different roles and exercise good citizenship. **C.I.K.3—**Describe the role and responsibilities of authority figures.
Vocabulary:	Balance: Having equal weight or importance of something

	Device: An electric piece of equipment like a phone, tablet, or laptop
Notes:	
	Device: An electric piece of equipment like a phone, tablet, or laptop

Week 17: Code.org, Course A, Lesson 7—Happy Loops
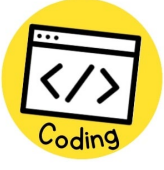

Lesson overview:	**Purpose:**
	This context-setting lesson revisits Happy Maps. This time, students will use loops to solve bigger, longer puzzles with their code. Loops allow for students to simplify their code by grouping commands that need to be repeated.
	Lesson:
	● Warm Up
	    ○ This lesson builds off the Happy Maps activity from earlier in the year. Students might benefit from a quick refresher before you hop into the difficult stuff.
	● Main Activity
	    ○ This portion of the lesson should help students see there is an easier way to handle repetitive code than to brute force a solution with dozens of the same symbols.
	● Wrap Up
	    ○ Draw a face that shows how you felt about today's lesson in the corner of your journal page.
	    ○ Have the students write or draw something in their journal that will remind them later what loops are. Prompts include:
	        ■ What does repeat mean to you?
	        ■ Draw a picture of you repeating something.
	● Keyboarding/Mouse Practice
	    ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Lesson 7: Happy Loops
CS standards addressed:	Students will be able to: ● Identify repeating code and shorten multiple actions into a single loop. ● Interpret a program with loops as a series of multiple actions. Standards: ● **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.3**—Develop programs with sequences and simple loops to express ideas or address a problem. ● **AP.1A.7**—Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.
Time needed:	**Total Time: 58 min** ● Warm Up **5 min** ● Main Activity **20 min** ● Wrap Up **8 min** ● Letter Recognition **15 min** ● Keyboarding **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet ● Feeling Faces Emotion Image - Resource ● Happy Map Cards - Worksheet ● Happy Map Cards XL - Worksheet ● Happy Map Game Pieces - Manipulatives ● Happy Map Game Pieces Bonus Pack - Manipulatives

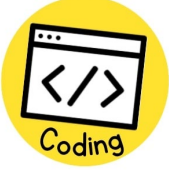| Subject integrated:	Math Social Studies
Other standards addressed:	Math ● **K.CC.1**—Count to 100 by ones and by tens. Social Studies ● **G.K.3.2**—Identify cardinal and intermediate directions (e.g., north, northeast, northwest, southeast, southwest, east, and west).
Vocabulary:	Loop: The action of doing something over and over Repeat: To do something again
Notes:	

Week 18: Number Recognition

Lesson overview:	**Purpose:**
	 This lesson allows students to practice number recognition for numbers 1-10.
	 Lesson:
	 • Warm Up
	    ○ Review counting to 10 and coding using a grid.
	 • Main Activity
	    ○ The teacher will model how to write an algorithm using the Number Recognition slides.
	    ○ Using the grid below, the students will write an algorithm using the number listed in the corner of the grid. This activity will help students with number identification. (Students will start at top left corner.)
	 • Wrap Up
	    ○ Allow students to demonstrate their code in front of the class.
	 • Keyboarding/Mouse Practice
	    ○ If time remains, have students use resources from above to practice using the keyboard/mouse.
	 Note: Teachers may spread the lesson out over several days, using 2-3 numbers a day, or teachers may divide students into groups and have each member code a number and swap to solve their partner's algorithm.

Lesson links/resources:	Number Recognition
CS standards addressed:	Students will be able to: • Use a grid and code the algorithm using a given number. Standards • **AP.1A.1**—Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks. • **DA.1A.3**—Identify and describe patterns in data visualizations, such as charts or graphs.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **30 min** • Wrap Up **5 min** • Keyboarding Activity **15 min**
Materials needed:	Teacher: • Smartboard/projector with sound • Number cards 1-10 for review Students: • Number grids
Subject integrated:	Math
Other standards addressed:	• **K.CC.1**—Count to 100 by ones and by tens. • **K.CC.2**—Count forward beginning from a given number within the known sequence (instead of having to begin at 1).
Vocabulary:	Code: Giving instructions to a computer or robot
Notes:	

Week 19: Counting Objects

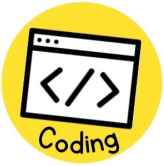|	

Lesson overview: Coding	**Purpose:** In this lesson students will learn to identify various ways/objects to represent the value of the numbers 1-10.
	 Lesson:
	 • Warm Up
	    ○ The teacher will model how to write a program using the [Number Grids](#) supplied to locate squares with the correct number.
	    ○ Review counting numbers 1-10, showing students pictures of objects, and identifying the number of objects in each picture using the smartboard.
	 • Main Activity
	    ○ Given a number, the student will write a program by using the grid to locate squares with the correct number of objects in each square. Think of the lesson from last week but look for objects instead of numbers.
	 • Wrap Up
	    ○ Did you enjoy this lesson? What did you like? What did you not like?
	    ○ Tell me about the program you wrote to find the squares that had the correct number of objects.
	 • Keyboarding/Mouse Practice
	    ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	[Number Grids](#)
CS standards addressed:	Students will be able to: • Use the grid to create an algorithm using the number of objects in each square. Standards: • **AP.1A.1**—Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks. • **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **30 min** • Wrap Up **5 min** • Keyboarding Activity **15 min**
Materials needed:	Teacher: • Smartboard/projector with sound Students: • Number grids
Subject integrated:	Math
Other standards addressed:	• **K.CC.1**—Count to 100 by ones and by tens • **K.CC.2**—Count forward beginning from a given number within the known sequence (instead of having to begin at 1) • **K.CC.4**—Understand the relationship between numbers and quantities; connect counting to cardinality.

Vocabulary:	Sequence: A set of related events, movements, or things that follow each other in a particular order Data: Any information put into the computer or given back by the computer.
Notes:	

Week 20: Coding Living and Nonliving Things

Lesson overview:	**Purpose:**
	This lesson is an unplugged activity that will be primarily pencil and paper. Students will learn about living and nonliving things while also creating an algorithm.
	Lesson:
	● Warm Up
	   ○ Show the students the video "Living and Nonliving." Discuss with the students what makes something living or nonliving and create an anchor chart displaying examples of both.
	● Main Activity
	   ○ Students will create an algorithm connecting the living and nonliving things.
	● Wrap Up
	   ○ Review the characteristics of living and nonliving things.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	● Living and Nonliving Things Grid ● Video: Living and Nonliving Things
CS standards addressed:	Students will be able to: ● Create an algorithm connecting the living and nonliving things. Standards: ● **AP.1A.1a—**Students should be able to create and follow algorithms. ● **AP.1A.2a—**Students should be able to model data storage and manipulation by using representative symbols.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **5 min** ● Keyboarding/Letter Recognition Activity (Week 14) **15 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Living Things and Nonliving Things Grid
Subject integrated:	Science
Other standards addressed:	**L.K.1A—**Students will demonstrate an understanding of living and nonliving things.
Vocabulary:	Algorithm: A list of steps to finish a task Data: This is any information put into the computer or given back by the computer.
Notes:	

Week 21: Kodable: Beach Cleanup—GO GREEN!

Lesson overview:	**Purpose:**
	For this lesson, we will think about ways to help the earth and oceans. We will do a short activity off-screen to learn a bit more about why ocean and beach pollution matters and brainstorm how technology can be used as part of the solution. Then, we will jump on-screen to practice programming our own beach cleanup solutions.
	Lesson:
	● Warm Up
	   ○ Ask students what they already know about the impact of trash/plastics on underwater life. Show the video "Tommy the SudBudz Turtle."
	● Main Activity
	   ○ Introduce Kodable and explain how the activity will work by showing the video "Introduction to Kodable Hour of Code: Beach Cleanup."
	   ○ The fuzz will only move in the directions you tell it. This means you must give the fuzz directions in the correct sequence (order). You will give the fuzz instructions by dragging and dropping commands into the command bar.
	● Wrap Up
	   ○ At the end of the lesson, lead a quick debrief of the Beach Cleanup with Kodable activity.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	● Beach Cleanup with Kodable ● Beach Cleanup Resources
CS standards addressed:	Students will be able to: ● Design and create mazes based on preexisting obstacles. ● Write simple programs to solve mazes using basic coding concepts. ● Examine ways technology can be used to solve real-world problems. ● Collaborate and communicate effectively with peers. Standards: ● **AP.1A.3**—Develop programs with sequences and simple loops to express ideas or address a problem. ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **AP.1A.8**—Using correct terminology, describe steps taken and choices made during the interactive process of program development.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **10 min** ● Keyboarding Practice **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	Science

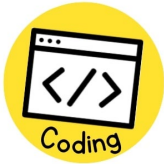|	ELA

Other standards addressed:	Science
	    ● **E.K.10.2—**With teacher guidance, develop questions to conduct a structured investigation to determine ways to conserve Earth's resources (i.e., reduce, resume, and recycle) and communicate results.
	ELA
	    ● **RF.K.1—**Demonstrate understanding of the organization and basic features of print.

Vocabulary:	<u>Program</u>: An algorithm that has been coded into something that can be run by a machine <u>Sequence</u>: A set of related events, movements, or things that follow each other in a particular order <u>Loop</u>: The action of doing something over and over <u>Debugging</u>: Finding and fixing problems in an algorithm or program
Notes:	

Week 22: Kodable: Hour of Code: ELA Integration

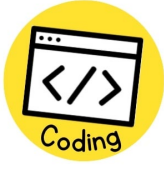|	

Lesson overview:	**Purpose:**
	We follow sequences all the time. We are going to use the sentence puzzle pieces to put words in the correct order to build a sentence that makes sense.
	Lesson:
	● Warm Up
	  ○ Read "The Kodable World." Model the skills students are working on: following words from left to right, page to page. Show them what readers do in their head as they read.
	● Main Activity
	  ○ You will practice sentence organization together as a class. Provide words and have students put them together in the correct order.
	  ○ Provide students with a sentence puzzle and have them follow on their own graphic organizers. Hand each student a copy of the Kodable sentence puzzle graphic organizer maze to fill out. Call students up to put the words in sequence and share their completed sentences at the end.
	● Wrap Up
	  ○ Students will apply what they learned from the lesson to compete Kodable's sequence sector lessons 1.1-1.5.

Lesson links/resources:	● The Kodable World ● The Kodable Word Resources
CS standards addressed:	Students will be able to: ● Apply sequencing in literacy to programming. ● Run a program by correctly sequencing sentences. Standards: ● **AP.1A.3a**—Students should be able to express ideas or address problems by developing programs with sequences and simple loops.
Time needed:	**Total Time 60 min** ● Warm Up **10 min** ● Main Activity **40 min** ● Wrap Up **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	ELA
Other standards addressed:	● **RF.K.1**—Demonstrate understanding of the organization and basic features of print. ● **RF.K.1a**—Follow words from left to right, top to bottom, and page by page.
Vocabulary:	Sequence: A set of related events, movements, or things that follow each other in a particular order Programming: Finding and fixing problems in an algorithm or program

Notes:

Week 23: Drag and Drop Sorting

Lesson overview:
 	
Keyboarding (logo)
	Purpose:
	In this lesson, students will use drag and drop to sort real-life objects that are shaped like the 2D and 3D shapes.
	Lesson:
	● Warm Up
	   ○ Review 2D and 3D shapes.
	   ○ Have students help create an anchor chart to list the shapes that are 2D and those that are 3D.
	● Main Activity
	   ○ Have students drag and drop the object on the correct column, 2D or 3D.
	● Wrap Up
	   ○ Have students give examples of 2D and 3D shapes they see at home or at school.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Sort 2D and 3D Shapes
CS standards addressed:	Students will be able to ● Use drag and drop to sort 2D and 3D shapes. Standards: ● **CS.1A.2—**Use appropriate terminology in identifying and describing the function of common physical components of computing systems (hardware). ● **CS.1A.3—**Describe basic hardware and software problems using accurate terminology.
Time needed:	**Total Time 60 min** ● Warm Up **10 min** ● Main Activity **15 min** ● Wrap Up **10 min** ● Keyboarding Activity **15 min** ● Letter Recognition (See Week 14) **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	Math
Other standards addressed:	**K.MD.3—**Classify objects into given categories; count the numbers of objects in each category and sort the categories by count. **K.G.2—**Correctly name shapes regardless of their orientations or overall size. **K.G.3—**Identify shapes as two-dimensional (lying in a plane, "Flat") or three-dimensional ("solid").
Vocabulary:	

Notes:	

Week 24: Kodable: Maze Maker Challenge

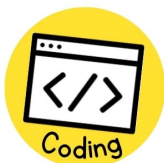| Lesson overview:
 | Purpose:
Can you build a symmetrical maze? A maze with right angles? Get creative and complete maze-building challenges with basic coding concepts.
Lesson:
● Warm Up
 ○ For this activity, we will be creating a maze using our math and coding skills.
 ○ Show the students how to access the Maze Maker activity (See the link below for those instructions.).
 ○ Once the students are set up, start them on the guided practice levels.
 ○ Model for the students how to create a maze or show the video in the link below.
● Main Activity
 ○ Students will complete their grade-level challenge.
 ○ Ask the students what shape they see.
 ○ Tell the students they will draw the shape using the blue tiles and make it part of their maze.
● Wrap Up
 ○ Ask the students to describe the shapes they used in their maze, describing the attributes.
● Keyboarding/Mouse Practice
 ○ If time remains, have students use resources from above to practice using the keyboard/mouse. |
Lesson links/resources:	● [Maze Maker Challenge](#) ● [Maze Maker Challenge Resources](#)
CS standards addressed:	Students will be able to: ● Create solvable mazes while applying grade-level geometry concepts. Standards: ● **AP.1A.3a—**Students should be able to express ideas or address problems by developing programs with sequences and simple loops.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **10 min** ● Keyboarding Activity/Letter Recognition Activity (Week 14) **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	Math
Other standards addressed:	**K.G.5—**Model shapes in the world by building shapes from components (e.g., sticks and clay balls) and drawing shapes.

Vocabulary:	Program: An algorithm that has been coded into something that can be run by a machine
	Sequence: Set of steps carried out in order
Notes:	Program: An algorithm that has been coded into something that can be run by a machine
	Sequence: Set of steps carried out in order

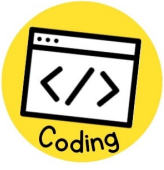|	
Week 25: Comparing Numbers With Alli-Gator	

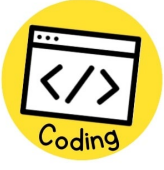|	

Lesson overview:	<u>Purpose:</u> Students will use arrows to program Alli-Gator while identifying numbers between 1 and 20.
	<u>Lesson:</u>
	Create algorithms using greater than, less than, or equal to.
	● Warm Up
	   ○ Practice identifying numbers from 1-20.
	   ○ Give two numbers and identify which is greater than, less than, or equal to.
	● Main Activity
	   ○ Have students work either in groups or alone. Have them practice the different types of number grids.
	   ○ Using the arrows as a guide, the students will help Alli-Gator find the number(s) that answer the math problem at the top of each grid.
	● Wrap Up
	   ○ Have each student or group describe the path they took.
	● Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Comparing Numbers with Alli-Gator
CS standards addressed:	Students will be able to: ● Create algorithms while comparing numbers. Standards: ● **AP.1A.3—**Develop programs with sequences and simple loops to express ideas or address a problem. ● **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **AP.1A.8a—**Students should be able to talk or write about the goals and expected outcomes of the programs they create and the choices that they made when creating programs.
Time needed:	<u>**Total Time:** 60 min</u> ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **10 min** ● Keyboarding/Letter Recognition Activity (See Week 14) **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	Math
Other standards addressed:	**K.CC.7—**Compare two numbers between 1 and 20 presented as written numerals.
Vocabulary:	<u>Sequence:</u> Set of steps carried out in order <u>Loops:</u> The action of doing something over and over

Notes:

Week 26: Coding States of Matter

Lesson overview:	**Purpose:**
	During this lesson, students will make their way through the grid. They will run into solids, liquids, or gasses along the way.
	Lesson:
	• Warm Up
	   ○ Review the definition of a solid, liquid, and gas. Create an anchor chart with pictures if needed.
	• Main Activity
	   ○ Create an algorithm to code the solids, liquids, and gasses.
	• Wrap Up
	   ○ Have students show their algorithms to the class.
	• Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.
	Teacher Note: This can be done over three days if needed. Each day can be devoted to coding solids, liquids, and gasses.

Lesson links/resources:	States of Matter
CS standards addressed:	Students will be able to: • Create an algorithm to code solids, liquids, and gasses. Standards: • **AP.1A1—**Model daily processes by creating and following algorithms (set of step-by-step instructions) to complete tasks. • **AP.1A2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Time needed:	**Total Time: 60 min** Warm Up **10 min** Main Activity **25 min** Wrap Up **10 min** Keyboarding Activity **15 min**
Materials needed:	Teacher: • Smartboard/projector with sound Students: • Coding grid • Color pencil/pencil/crayon
Subject integrated:	Science
Other standards addressed:	**P.K.5A—**Students will demonstrate an understanding of the solid and liquid states of matter.
Vocabulary:	Algorithm: A list of steps to finish a task Program: An algorithm that has been coded into something that can be run by a machine Data: This is any information put into the computer or given back by the computer.
Notes:	

Week 27: Coding Blending Sounds
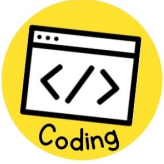

Lesson overview:	Purpose:
	In this lesson, students will use their programming and blending skills to create an algorithm.
	Lesson:
	● Warm Up
	    ○ Have students practice reading and blending CVC words.
	● Main Activity
	    ○ Using either option 1 or 2 provided in the link below, students will use arrows to create a code and blend the words as they reach them.
	● Wrap Up
	    ○ Have students create sentences aloud using the words from their algorithm.
	● Keyboarding/Mouse Practice
	    ○ If time remains, have students use resources from above to practice using the keyboard/mouse.
	Teacher Note: This activity contains two options. You may choose one or do both on two different days.

Lesson links/resources:	Blending Sounds
CS standards addressed:	Students will be able to: ● Blend CVC words while creating an algorithm. Standards: ● **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks.
Time needed:	**Total Time: 60 min** ● Warm Up **5 min** ● Main Activity **30 min** ● Wrap Up **10 min** ● Keyboarding **15 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Coding grid ● Color pencil/pencil/crayon
Subject integrated:	ELA
Other standards addressed:	● **RF.K.2—**Demonstrate understanding of spoken words, syllables, and sounds (phonemes). ● **RF.K.2a—**Recognize and produce rhyming words.
Vocabulary:	Algorithm: A list of steps to finish a task
Notes:	

Week 28: Kodable: Pizza Party
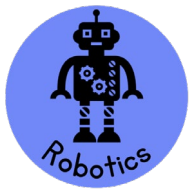

Lesson overview:	Purpose:
	For this lesson, we will learn about computer science and participate in creative problem-solving activities. We will learn about basic computer science concepts and practice problem-solving skills to solve a real-life problem.
	Lesson:
	● Warm Up
	  ○ Review vocabulary words with students. Explain who programmers are and what programming language is using the terms and definitions.
	  ○ Begin by introducing the foundational concept, sequence.
	  ○ Use chart paper to create a list of things we do in order (provide examples if needed).
	● Main Activity
	  ○ Help design a mobile app for a pizza restaurant. You will create and follow a sequence, making a simple algorithm to follow to make customized pizza orders.
	● Wrap Up
	  ○ Students will complete the K-W-L exit ticket.

Lesson links/resources:	● Kodable Pizza Party ● Kodable Pizza Party Resources
CS standards addressed:	Students will be able to: ● Create a sequence. ● Write simple numerical expressions and evaluate them in the proper sequence. Standards: ● **AP.1A.1**—Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks. ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. ● **AP.1A.5**—Develop plans that describe a program's sequence of events, goals, and expected outcomes.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Main Activity **40 min** ● Wrap Up 1**0 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Pizza-making algorithm graphic organizer ● "Ready for School" algorithm exit ticket ● Student inquiry sheet
Subject integrated:	Math
Other standards addressed:	● **K.OA.1**—Represent addition and subtraction, in which all parts and whole of the problem are within 10, with objects, fingers, mental images, drawings, sounds (e.g., clasp), acting out situations, verbal explanations, expressions, or equations. ● **K.OA.2**—Solve addition and subtraction word problems, and add and subtract within 10, e.g., by using objects or drawings to represent

	the problem.
Vocabulary:	Programmer: Person who creates a program Program: An algorithm that has been coded into something that can be run by a machine Code: Giving instructions to a computer or robot Command: A specific order from a user to the computer's operating system or to an application to perform a service Sequence: A set of related events, movements, or things that follow each other in a particular order Algorithm: A list of steps to finish a task Bug: Part of a program that does not work correctly Debugging: Finding and fixing problems in an algorithm or program
Notes:	

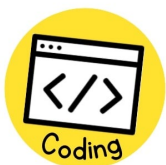# Week 29: Coding *The Very Hungry Caterpillar*

Lesson overview:	**Purpose:** Review vocabulary and print concepts. Use left, right, up, and down directions to create an algorithm helping the caterpillar make its way through the grid.
	 Lesson:
	 • Warm Up
	  ○ Read *The Very Hungry Caterpillar* By Eric Carle.
	  ○ Review the vocabulary word algorithm.
	  ○ Review direction: left, right, up, down.
	 • Main Activity
	  ○ Students will help the caterpillar make its way through [The Very Hungry Caterpillar grid](#) eating food and turning into a beautiful butterfly.
	 • Wrap Up:
	  ○ Have students present their grid and the different ways it could be completed.
	 • Keyboarding/Mouse Practice
	  ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	*[The Very Hungry Caterpillar](#)*
CS standards addressed:	Students will be able to: • Create an algorithm. **Standards:** • **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by step instructions) to complete tasks. • **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **30 min** • Wrap Up **10 min** • Keyboarding Activity **10 min**
Materials needed:	Teacher: • *The Very Hungry Caterpillar* • Smartboard/projector with sound Students: • Google Slide grid
Subject integrated:	Science
Other standards addressed:	**L.K.2—**Students will demonstrate an understanding of how living things change in form as they go through the general stages of a life cycle.
Vocabulary:	
Notes:	

Week 30: Coding With Magna Tiles

Lesson overview:	**Purpose:**
Robotics	In this lesson, students will set up a grid using magna tiles or similar manipulatives.
	Lesson:
	- Warm Up
	○ Review subtraction facts within 5.
	- Main Activity
	○ Have students create a grid using the magna tiles. The grid can be the size of your choice but be sure to mark a start and finish square.
	○ Using the subtraction flash cards, students will answer the problem and move that number of spaces in any direction using the arrows to create the algorithm.
	- Wrap Up
	○ Have students demonstrate their algorithm to the class.
	- Keyboarding/Mouse Practice
	○ If time remains, have students use resources from above to practice using the keyboard/mouse.
Lesson links/resources:	Coding Arrows
CS standards addressed:	Students will be able to: - Create a grid for a robot. Standards: - **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by step instructions) to complete tasks.
Time needed:	**Total Time: 60 min** - Warm Up **5 min** - Main Activity **30 min** - Wrap Up **10 min** - Keyboarding Activity **15 min**
Materials needed:	Teacher: - Subtraction flash cards Students: - Magna tiles (If these are not available you could use squares cut from cardstock.) - Arrow cards
Subject integrated:	Math
Other standards addressed:	**K.OA.5—**Fluently add and subtract within 5.
Vocabulary:	Algorithm: A list of steps to finish a task Program: An algorithm that has been coded into something that can be run by a machine Debugging: Finding and fixing problems in an algorithm or program
Notes:	

Week 31: Code.org, Course A, Lesson 8—Loops With Scrat
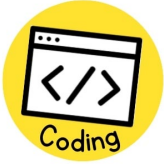
Lesson overview:

Purpose:
In this lesson, students will learn more about loops and how to implement them in Blockly code. Using loops is an important skill in programming because manually repeating commands is tedious and inefficient. With these Code.org puzzles, students will learn to add instructions to existing loops, gather repeated code into loops, and recognize patterns that need to be repeated.

Lesson:
- Warm Up: The Unplugged Foundation
  - This lesson relies on the concept of repeat loops that students learned in the previous unplugged activity, Happy Loops. It is important to bring this idea from the real world into digital form, so students understand how to use Blockly blocks to repeat a task multiple times.
- Bridging Activity (choose one):
  - Unplugged Activity Using Paper Blocks
    - Select an empty flurb map from the Happy Map Cards Worksheet and give students Unplugged Blockly Blocks (Grades K-1) prefilled with the collect command, a repeat loop, and the cardinal commands like E → (East) and W ← (West). Have students use the paper Blockly blocks from their desks to program the flurb to collect the fruit Make sure they understand that the blocks need to go from top to bottom and they all need to touch.
  - Previewing Online Puzzles as a Class
    - Pull a puzzle from the corresponding online puzzles. We recommend Puzzle 4. Using arrows, have students lay out a pattern they think will get Scrat to the acorn. Ask the students to share. See how many other students had the same answer.
- Main Activity: Preview Loops in Ice Age
  - To finish the connection, preview an online puzzle (or two) as a class.
  - Model: Reveal an entire online puzzle from the progression to come. We recommend Lesson 8, Puzzle 5. Point out the "Play Area" with Scrat and the acorn and the "Work Space" with the Blockly code. Explain that this Blockly code is now the language the class will be using to help Scrat get to the acorn. Do students see any similarities to the exercise they just did? What are the big differences?
  - Work with your class to drag code into the workspace in such a way that Scrat (eventually) gets to the acorn.
  - Transition: Students should now be ready to transition to computers to complete online puzzles on their own.
  - As students work through the puzzles, see if they can figure out how many blocks they use with a loop vs. without a loop.
- Wrap Up and Reflection
  - What was today's lesson about?
  - Draw a face that shows how you felt about today's lesson in the corner of your journal page.
  - Draw Scrat and an acorn.
  - Draw yourself using a loop to do an everyday activity like brushing your teeth.

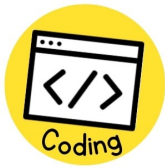| Lesson links/resources:	Lesson 8: Loops with Scrat
CS standards addressed:	The students will: ● Construct a program using structures that repeat areas of code. ● Improve existing code by finding areas of repetition and moving them into looping structures. Standards: ● **AP.1A.1a—**Students should be able to create and follow algorithms. ● **AP.1A.3a—**Students should be able to express ideas or address problems by developing programs with sequences and simple loops.
Time needed:	**Total time: 60 min** ● Warm Up **10 min** ● Bridging Activity **10 min** ● Main Activity **30 min** ● Wrap Up **10 min**
Materials needed:	Teacher: ● Smartboard/Projector with sound Students: ● Student devices with access to the internet ● Feeling Faces - Emotion Images ● Happy Map Cards - Worksheet ● Happy Map Game Pieces - Manipulatives ● Happy Map Game Pieces Bonus Pack - Manipulatives ● Pair-Programming - Video (Download) ● Unplugged Blockly Blocks (Grades K-1) - Manipulatives
Subject integrated:	Social Studies
Other standards addressed:	**G.K.3.2—**Identify cardinal and intermediate directions (e.g., north, northeast, northwest, southeast, southwest, east, and west).
Vocabulary:	Program: An algorithm that has been coded into something that can be run by a machine
Notes:	The link above gives step-by-step directions with pictures on how to complete the project.

Week 32: Code.org, Course A, Lesson 9—Loops With Laurel

Lesson overview:	**Purpose:**
	 In this skill-building lesson, students continue learning the concepts of loops. Here, students use loops to collect treasure in open cave spaces. This lesson gives students more practice with loops and introduces a new block and treasure.
	 Lesson:
	 • Warm Up
	    ○ Quickly review the definition of a loop, the action of doing something over and over again.
	    ○ What are loops? Why do we use them?
	 • Main Activity
	    ○ This activity will be combined with skill-building lessons and videos showing how to complete the skill builders. This also includes challenge, practice, and extra lessons.
	 • Wrap Up (Prompts):
	    ○ What was today's lesson about?
	    ○ Draw a face that shows how you felt about today's lesson in the corner of your journal page.
	    ○ How did loops make your program easier to write?
	    ○ Draw something that uses loops.
	 • Keyboarding/Mouse Practice
	    ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	• [Lesson 9: Loops with Laurel](#) • [Feeling Faces](#) • [Unplugged Coding Blocks](#)
CS standards addressed:	Students will be able to: • Break down a long sequence of instructions into the smallest repeatable sequence possible. • Identify the benefits of using a loop structure instead of manual repetition. Standards: • **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information. • **AP.1A.3**—Develop programs with sequences and simple loops to express ideas or address a problem. • **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. • **AP.1A.7**—Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **30 min** • Wrap Up **5 min** • Keyboarding Activity **15 min**
Materials needed:	For the teachers: • Smartboard/projector with sound • [CSF Lesson Recommendations](#) - Resource Students:

	● Student devices with access to the internet ● [Feeling Faces Emotion Image](#) - Resource ● [Unplugged Blockly Blocks (Grades K-1)](#) - Manipulatives
Subject integrated:	Social Studies
Other standards addressed:	**G.K.3.2—**Identify cardinal and intermediate directions (e.g., north, northeast, northwest, southeast, southwest, east, and west).
Vocabulary:	<u>Loop</u>: The action of doing something over and over <u>Repeat</u>: To do something again
Notes:	

Week 33: Code.org, Course A, Lesson 10—Ocean Scene With Loops

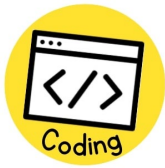|	

Lesson overview:	<u>Purpose</u>:
	In this skill-building lesson, students will learn how to draw images by looping simple sequences of instructions. Here, loops are creating patterns. At the end of this lesson, students will complete their own images.
	<u>Lesson</u>:
	● Warm Up
	  ○ Quickly review the definition of a loop, the action of doing something over and over again.
	  ○ Discuss different patterns like zigzags and stairsteps.
	    ■ How would you explain to someone how to draw that pattern? How could you draw this using a loop?
	  ○ In the artist levels, students will use 45-degree angles described as northwest, northeast, southwest, southeast. We recommend briefly discussing these directions with the class and drawing an image for students to refer.
	● Main Activity
	  ○ Video: The Artist in Code Studio, complete skill-building 2-4.
	  ○ Video: Loops in Artist, complete skill-building 6-10.
	  ○ A challenge, practice, and free play are also included in this lesson.
	● Wrap Up (Prompts):
	  ○ What was today's lesson about? Draw a face that shows how you felt about today's lesson in the corner of your journal page. Draw some stairs. Imagine the loop needed to draw this. Draw something else in your life that uses loops.
	● Keyboarding/Mouse Practice
	  ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	Lesson 10: Ocean Scene with Loops
CS standards addressed:	Students will be able to: ● Learn how to draw images by looping simple sequences of instructions. ● Use loops to create patterns. Standards: ● **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.3**—Develop programs with sequences and simple loops to express ideas or address a problem. ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions ● **AP.1A.7**—Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.
Time needed:	**Total Time:** 60 min ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **5 min** ● Keyboarding Activity **15 min**
Materials needed:	For the teachers:

	● Smartboard/projector with sound ● CSF Lesson Recommendations - Resource ● Pause and Think Online - Video Students: ● Student devices with access to the internet ● Feeling Faces Emotion Image - Resource
Subject integrated:	ELA
Other standards addressed:	● **SL.K.2—**Confirm understanding of a text read aloud or information presented orally or through other media by asking and answering questions about key details and requesting clarification if something is not understood. ● **SL.K.3—**Ask and answer questions in order to seek help, get information, or clarify something that is not understood. ● **SL.K.6—**Speak audibly and express thoughts, feelings, and ideas clearly.
Vocabulary:	Loop: The action of doing something over and over Repeat: To do something again
Notes:	

Week 34: Code.org, Course A, Lesson 11—The Big Event Jr.

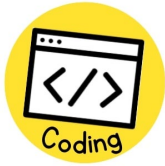|	

Lesson overview:	**Purpose**:
	In this context-setting lesson, the class will experience the concept of events through a game where they move or sort when you press buttons on a giant remote.
	Lesson:
	● Warm Up
	   ○ This lesson has a new and important vocabulary word: Event. An event is an action that causes something to happen. Let's see if we see a pattern here: When you flip a switch, the lights turn on. When you tap on a device, an app starts. When the alarm goes off, you get out of bed. In computer science, events cause other actions to happen.
	● Main Activity
	   ○ In earlier lessons, we created algorithms that allowed us to control a friend of flurb for several steps at a time. It was fun and useful, but what happens when you don't know everything that you want your friend to do in advance? This is where events come in.
	● Wrap Up
	   ○ Reflect on what students have learned through the following prompts: What did we learn? What are some examples of events?

Lesson links/resources:	Lesson 11: The Big Event Jr.
CS standards addressed:	The students will: ● Practice differentiating predefined actions and event-driven ones. ● Recognize actions of the teacher as signals to initiate commands ● Repeat commands given by an instructor. Standards: ● **AP.1A.2**—Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4**—Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** ● Warm Up **15 min** ● Main Activity **15 min** ● Wrap Up **15 min** ● Keyboarding **15 min**
Materials needed:	For the teachers: ● Smartboard/projector with sound ● The Big Event - Assessment Answer Key Students: ● Student devices with access to the internet ● Feeling Faces - Emotion Image - Resource ● The Big Event - Unplugged Video (Download) ● The Big Event - Assessment ● The Big Event (Courses A, B) - Controller Image
Subject integrated:	ELA

Other standards addressed:	**RL.K.7—**With prompting and support, describe the relationship between illustrations and the story in which they appear (e.g., what moment in a story an illustration depicts.
Vocabulary:	<u>Event</u>: An action that causes something to happen
Notes:	

Week 35: Code.org, Course A, Lesson 12—Mini Project: On the Move With Play Lab

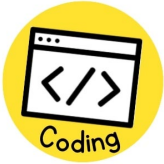|	

Lesson overview:	Purpose:
	Students will further develop their understanding of events using Play Lab today. 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.
	Lesson:
	● Warm Up
	   ○ Review "The Big Event" activity with the students.
	      ■ What did we program the button events to do?
	   ○ Now we're going to add events to our code. Specifically, we're going to have an event for when two characters touch each other.
	      ■ When have you seen two characters touch each other in an event in games?
	● Main Activity
	   ○ Bridging Activity (Choose One):
	      ■ Unplugged Activity Using Paper Blocks
	      ■ Previewing Online Puzzles as a Class
	   ○ This is the most free form plugged activity of the course. In the final stage, students have the freedom to create their own story. You may want to provide structured guidelines around what kind of story to write, particularly for students overwhelmed by too many options.
	● Wrap Up
	   ○ What was today's lesson about?
	   ○ Imagine you have a remote-controlled robot. What would the remote look like? Draw a picture of what you think you could make the robot do.

Lesson links/resources:	Lesson 12: Mini-Project—On the Move with Play Lab
CS standards addressed:	The students will: ● Create an animated, interactive story using sequence- and event-handlers. ● Identify actions that correlate to input events. ● Share a creative artifact with other students. Standards: ● **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information. ● **AP.1A.4—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions.
Time needed:	**Total Time:** 55 min ● Warm Up **10 min** ● Bridging Activity **10 min** ● Main Activity **30 min** ● Wrap Up **5 min**
Materials needed:	For the teachers: ● Smartboard/projector with sound ● CS Fundamentals Main Activity Tips ● CSF Lesson Recommendations - Resource ● Pause and Think Online - Video

	Students: ● Student devices with access to the internet ● Feeling Faces - Emotion Image - Resource ● The Big Event (Courses A, B) - Controller Image ● Unplugged Blockly Blocks (Grades K-1) - Manipulatives
Subject integrated:	ELA
Other standards addressed:	**RL.K.7—**With prompting and support, describe the relationship between illustrations and the story in which they appear (e.g., what moment in a story an illustration depicts.
Vocabulary:	Event: An action that causes something to happen
Notes:	

Week 36: Code.org, Course A, Lesson 13—End of Course Project
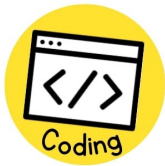
Lesson overview: 	<u>Purpose:</u> This lesson gives students one last opportunity to creatively use their new computer science skills. <u>Lesson:</u> ● Warm Up 　○ Get students excited and ready for the day's activity. Begin by asking the students to remember what they have already done in the earlier lesson using the Artist's tools. ● Main Activity 　○ Pull up the level for this lesson. 　○ Today your job is to create a path for the Artist. The Artist will be taking a trip to three places: a garden, a zoo, and outer space. 　○ Your job is to create a path for him and tell a story about what he did and saw. ● Wrap Up 　○ Allow students to showcase their projects.
Lesson links/resources:	<u>Lesson 13: End of Course Project</u>
CS standards addressed:	Students will be able to: ● Apply computer science concepts in an open-ended project. ● Overcome obstacles such as time constraints or bugs. Standards: ● **AP.1A.3—**Develop programs with sequences and simple loops to express ideas or address a problem.
Time needed:	**Total Time: 60 min** ● Warm Up **10 min** ● Main Activity **30 min** ● Wrap Up **20 min**
Materials needed:	For the teachers: ● Smartboard/projector with sound ● Artist Project Planning Guide - Answer Key Students: ● Student devices with access to the internet ● Artist Project Planning Guide - Handout
Subject integrated:	ELA
Other standards addressed:	**RL.K.7—**With prompting and support, describe the relationship between illustrations and the story in which they appear (e.g., what moment in a story an illustration depicts).
Vocabulary:	<u>Loop</u>: The action of doing something over and over
Notes:	

Week 37: Kodable: Magic Machine

Lesson overview:	**Purpose:**
	 Is technology just mysterious magic or do people control machines? Students explore what they already know about the technology they use daily.
	 Lesson:
	 • Warm Up
	    ○ Open the lesson with true or false statements to check what students already know about technology.
	    ○ Read the statements. If a statement is true, move to the left; if false, move to the right.
	 • Main Activity
	    ○ Introduce the vocabulary cards and images.
	    ○ Go over the terms and definitions and talk about what technology is and the role humans play in making machines function.
	    ○ We will work in a small group and think about how we use technology.
	    ○ Introducing the technology mind map.
	    ○ The groups will write and draw what comes to mind when they think of technology.
	 • Wrap Up
	    ○ Once groups finish brainstorming, bring the whole class together to share.

Lesson links/resources:	• [Magic Machine](#) • [Magic Machine Resources](#)
CS standards addressed:	Students will be able to: • Activate prior knowledge related to technology and coding. • Recognize the role communication plays in technology. • Identify the ways humans control machines. Standards: • **AP.1A.1—**Model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete the tasks. • **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Time needed:	**Total Time: 60 min** Warm Up **15 min** Main Activity **25-30 min** Wrap Up **5-10 min**
Materials needed:	Teacher: • Smartboard/projector with sound Students: • Student devices with access to the internet
Subject integrated:	Science ELA
Other standards addressed:	Science • **E.K.10.1—**Participate in a teacher-led activity to gather, organize, and record recyclable materials data on a chart or table using technology and communicate results.

	ELA
	● **W.K.2—**Use a combination of drawing, dictating, and writing to compose informative/explanatory texts in which they name what they are writing about and supply some information about the topic.
Vocabulary:	Programmer: Person who creates a program Program: An algorithm that has been coded into something that can be run by a machine Code: Giving instructions to a computer or robot Command: A specific order from a user to the computer's operating system or to an application to perform a service Algorithm: A list of steps to finish a task
Notes:	

Week 38: Kodable: Show What You Know

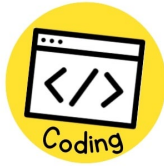|	

Lesson overview:	**Purpose:**
	Students will design, draw, and solve their own programming problems. See a sample of the Sequence Capstone lesson on the Kodable school plan.
	Lesson:
	• Warm Up
	   ○ Students will create and solve their own programming problem by designing mazes and using commands that visually represent their programming process.
	• Main Activity
	   ○ Seat students on the floor and explain they will become programmers and create and solve their own programming problem.
	   ○ Give each student the Maze Maker handout.
	   ○ Send them to their workspace to cut and glue.
	• Wrap Up
	   ○ Have students explain their design to the class before they draw it. Once they have planned out their maze and explained it, they can draw it out.
	• Keyboarding/Mouse Practice
	   ○ If time remains, have students use resources from above to practice using the keyboard/mouse.

Lesson links/resources:	• [Show What You Know](#) • [Show What You Know Resources](#)
CS standards addressed:	Students will be able to: • Create their own programming problem. • Solve programming problems. • Visually represent programming problems. • Explain programming processes. Standards: • **AP.1A.1—**Model daily process by creating and following algorithms. • **AP.1A.2—**Model the way programs store and manipulate data by using numbers or other symbols to represent information.
Time needed:	**Total Time: 60 min** • Warm Up **10 min** • Main Activity **30 min** • Wrap Up **10 min** • Keyboarding Practice **10 min**
Materials needed:	Teacher: • Smartboard/projector with sound Students: • Student devices with access to the internet
Subject integrated:	ELA
Other standards addressed:	**W.K.3—**Use a combination of drawing, dictating, and writing to narrate a single event or several loosely linked events, tell about the events in the order in which they occurred, and provide a reaction to what happened.
Vocabulary:	Program: An algorithm that has been coded into something that can be run by a machine

	Sequence: A set of related events, movements, or things that follow each other in a particular order **Command**: A specific order from a user to the computer's operating system or to an application to perform a service **Code**: Giving instructions to a computer or robot **Bug**: Part of a program that does not work correctly **Debugging**: Finding and fixing problems in an algorithm or program
Notes:	

Week 39: If Flash, then Clap!

Lesson overview:
 	
 Unplugged
	Purpose:
	 If there is lightning, then thunder will follow! Students explore conditional statements in a hands-on science activity.
	 Lesson:
	 • Warm Up:
	  ○ We are going to learn about computer science concepts and practice our problem-solving skills.
	 • Main Activity:
	  ○ An example of a condition is something we experience everyday weather!
	  ○ Let's learn about how thunder and lightning happen.
	  ○ Show the video on thunder and lightning (See the video in the lesson link.).
	  ○ The students will rub the balloon on their head for five seconds. Explain that this is creating a positive charge, like lightning.
	  ○ Students blow into their brown bags to fill them with air and twist them closed.
	  ○ Hit the bag with your free hand creating thunder.
	 • Wrap Up:
	  ○ Students can tell if they liked the thunder or lightning activity best.

Lesson links/resources:	• [If Flash, then Clap!](#) • [If Flash, then Clap! Resources](#)
CS standards addressed:	The students will: • Students will be able to determine the effect of a condition being true. • Students will be able to connect real-world conditions with "If Statements" in programming. • Students will be able to create "If Statements" to describe real world cause and effect. Standards: • **AP.1A.4** Decompose (break down) the steps needed to clove a problem into a precise sequence of instructions.
Time needed:	**Total Time: 60 min** Warm up **15 min** Main Lesson **35 min** On-screen practice **10 min**
Materials needed:	Teacher: • Smartboard/Projector with sound Students: • Student devices with access to the internet
Subject integrated:	Science
Other standards addressed:	**E.K.8A** Students will demonstrate an understanding of the pattern of seasonal changes on the Earth.
Vocabulary:	Sequence: A set of related events, movements, or things that follow each other in a particular order Program: An algorithm that has been coded into something that can be run

	by a machine Programmer: Person who creates a program Code: Giving instructions to a computer or robot Command: A specific order from a user to the computer's operating system or to an application to perform a service
Notes:	

Week 40: Kodable Hour of Code: Advanced

Lesson overview:	**Purpose:**
	Why does communication matter? How do people communicate with machines to do what we want? Test communication skills as you learn to code.
	Lesson:
	● Warm Up
	   ○ Review the vocabulary words.
	   ○ Tell the students we will learn how computers work.
	● Main Activity
	   ○ Students will act as programmers and apply basic knowledge of programming language and sequence to command a robot to move forward and jump.
	● Wrap Up
	   ○ Students will put the commands in the correct sequence to successfully direct the Kodable fuzz through the maze.

Lesson links/resources:	● Hour of Code: Advanced ● Hour of Code: Advanced Resources
CS standards addressed:	Students will be able to: ● Define sequences. ● Decompose a task into a sequence of events. ● Relate an order of events to sequencing in programming. ● Explain what a programmer does. ● Instruct a robot to move forward and jump with basic programming language. Standards: ● **AP.1A.3—**Develop programs with sequences and simple loops to express ideas or address a problem. ● **AP.1A.5a—**Students should be able to develop and visually illustrate the plan for what a program will do. ● **AP.1A.8—**Using the correct terminology, describe steps taken and choices made during the iterative process of program development. ● **AP.1A.8a—**Students should be able to talk or write about the goals and expected outcomes of the programs they create and the choices that they made when creating programs.
Time needed:	**Total time: 60 min** ● Warm Up **15 min** ● Main Activity **35 min** ● Wrap Up **10 min**
Materials needed:	Teacher: ● Smartboard/projector with sound Students: ● Student devices with access to the internet
Subject integrated:	ELA
Other standards addressed:	**W.K.3—**Use a combination of drawing, dictating, and writing to narrate a single event or several loosely linked events, tell about the events in the order in which they occurred, and provide a reaction to what happened.
Vocabulary:	Computer: A device for working with information like numbers, words, pictures, movies, or sounds

	<u>Programmer</u>: Person who creates a program <u>Code</u>: Giving instructions to a computer or robot <u>Commands</u>: A specific order from a user to the computer's operating system or to an application to perform a service <u>Sequence</u>: A set of related events, movements, or things that follow each other in a particular order
Notes:	