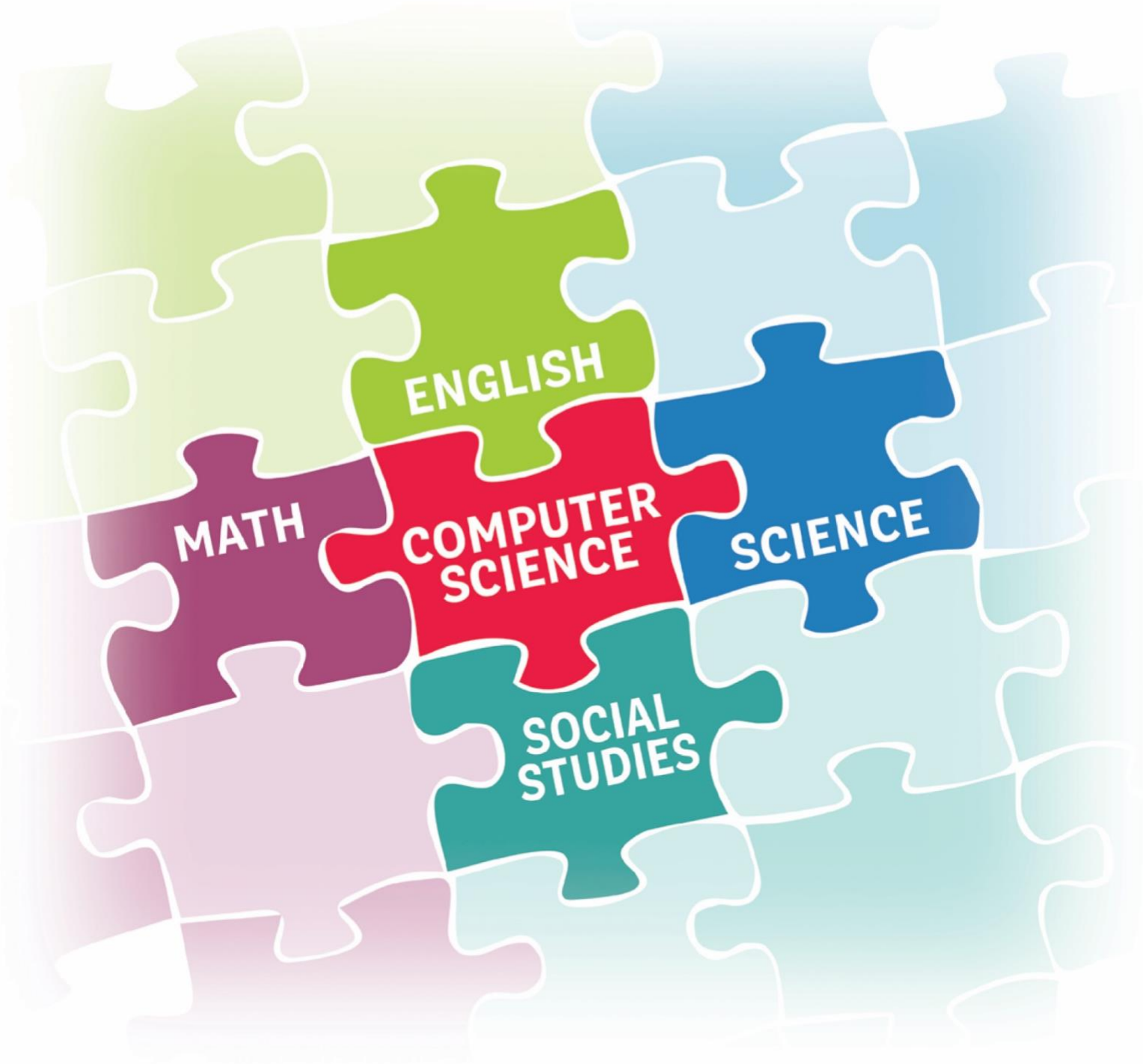


2022

Elementary Integration Guide

FIFTH GRADE

MISSISSIPPI STATE UNIVERSITY™
CENTER FOR CYBER EDUCATION

Acknowledgements

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

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

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

Center for Cyber Education Team
Shelly Hollis, Director
Lizzie Brandon, Project Manager
Amanda Taylor, Project Manager

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

Funding for the development of this guide was provided by:

Introduction

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

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

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

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

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

Resources	
Computing resources	• Code.org CS Fundamentals ○ Course F—Fifth Grade • Common Sense Digital Media • CS First Guide • Scratch Educator Guide
CS4MS resources	• 2018 Mississippi Computer Science Standards • CS4MS Website
Keyboard practice	• Astro Bubbles Keyboard Practice • Nitrotype • Typing.com
Teacher/student accounts	If you will be teaching computer science in a subject rotation, please see your content area to determine the accounts you and your students will need. • Code.org • Common Sense Digital Media • CS First • Scratch
Help with this guide	Contact Mississippi State University's Center for Cyber Education: www.tinyurl.com/ccehelpdesk

Contents by Topics

Social Studies

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

English

- Week 1: RI.5.6—Analyzing Accounts on the Same Topic: Point of View
- Week 2: RI.5.7—Multiple Print Sources
- Week 3: RI.5.8—Understanding Supporting Elements
- Week 10: RL.5.1—Inferences/Text Evidence
- Week 11: RL.5.2—Theme
- Week 12: RL.5.2—Summarizing
- Week 13: RL.5.3—Compare/Contrast Text Features
- Week 14: RL.5.4—Language/Meaning
- Week 15: RL.5.3, RL.5.5—Understanding Literary Structure
- Week 16: RL.5.5, RL.5.6—Point of View
- Week 17: RL.5.7—Analyzing Virtual Elements
- Week 18: RL.5.9—Compare/Contrast Stories in the Same Genre
- Week 19: RI.5.1—Using Details to Support Inferences
- Week 20: RI.5.2—Main Ideas and Summarizing
- Week 21: RI.5.3—Explain Relationships in Scientific/Technical/Historical Texts
- Week 22: RI.5.4—Unfamiliar Words/Contextual Clues
- Week 23: RI.5.5—Text Structure

Science

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

Math

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

Contents by Topic

Coding

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

Digital Citizenship

- Week 1
- Week 2
- Week 3
- Week 19

Digital Literacy

- Week 18
- Week 21

Keyboarding

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

Robotics

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

Unplugged

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

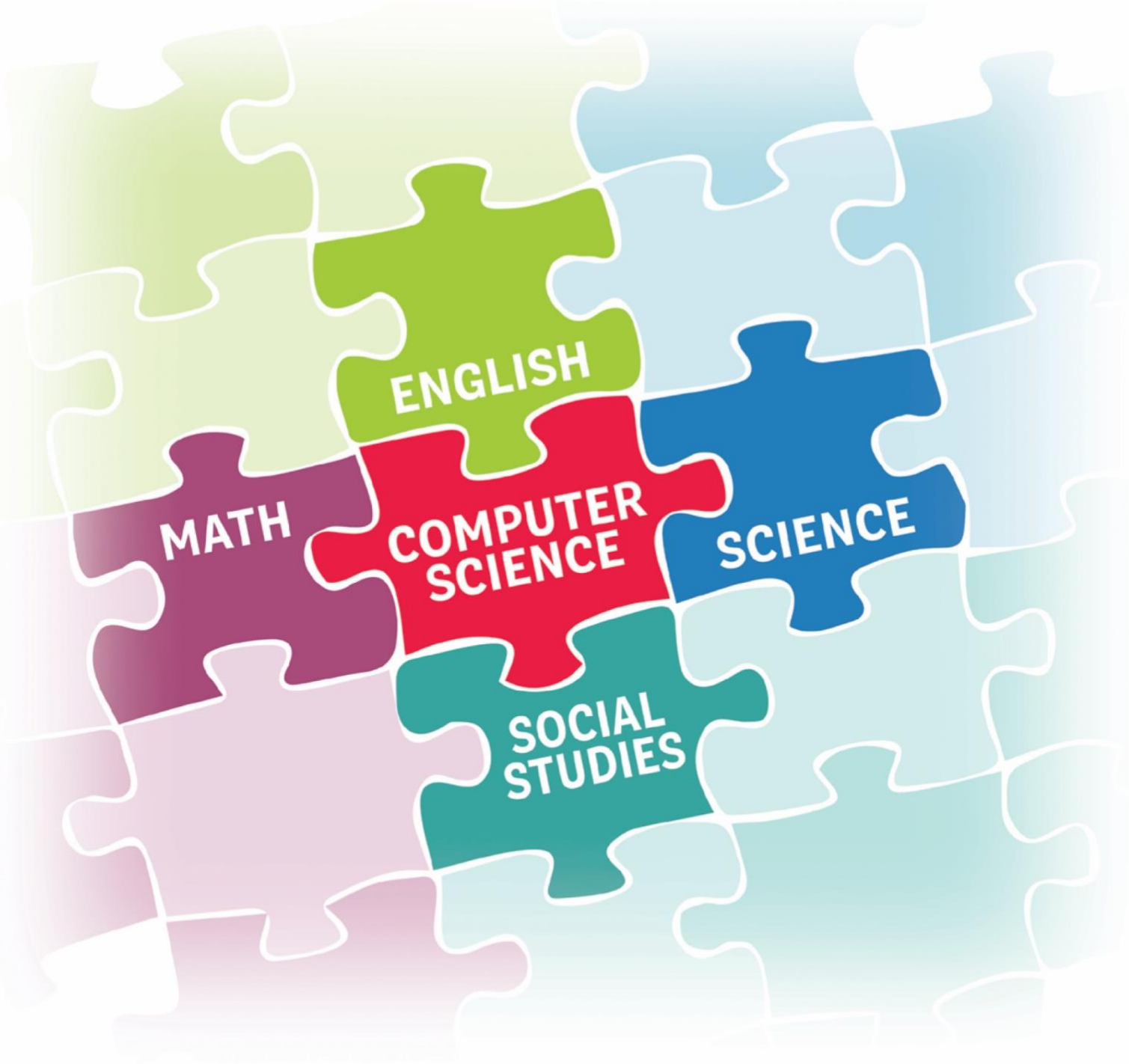
Calendar/Pacing Per Week:

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

Week	Title	Topics	CS Standard	Standard	Subject Integrated
1	Digital Friendships → Account creation needed	Digital Citizenship	IC.1B.3 IC.1B.3a	RI.5.6	ELA
2	Let's Face It	Digital Citizenship	IC.1B.3 IC.1B.3a	RI.5.7	ELA
3	Media Balance	Digital Citizenship	CS.1B.3	RI.5.8	ELA
4	Oceans and Continents Clicker Game → Account creation needed	Coding	AP.1B.1	G.5.1 G.5.2 G.5.3	Social Studies
5	Storytelling—Ancient Civilizations of Central and South America → Account creation needed	Coding	AP.1B.3	W.5.2 H.5.6	ELA / Social Studies
6	Imagine a World With European Explorers	Coding	AP.1B.4	W.5.2 H.5.2	ELA / Social Studies
7	13 Colonies Greeting Cards	Coding	AP.1B.9	W.5.2 H.5.3	ELA / Social Studies
8	Road to the Revolutionary War Using Robots	Coding, Robotics, Unplugged	AP.1B.8	H.5.4 H.5.5	Social Studies
9	Come See Our New Nation Advertisement	Coding	AP.1B.3	CI.5.1	Social Studies
10	Surprise Ending Inferences	Coding	DA.1B.2	RL.5.1	ELA
11	The Theme of Gumball	Coding	AP.1B.3	RL.5.2	ELA
12	Story (re)Telling	Coding	AP.1B.6b	RL.5.2	ELA
13	Code.org: Hello World	Coding	1A.AP.09 1A.AP.11 AP.1B.3 AP.1B.5 IC.1A.2	RL.5.3	ELA
14	Figurative Conversations	Coding	AP.1B.10 AP.1B.11	RL.5.4	ELA
15	Create-a-Character	Coding	AP.1B.10 AP.1B.11	RL.5.3 RL.5.5	ELA
16	Two Points of View	Coding	IC.1B.3a AP.1B.9	RL.5.5 RL.5.6	ELA
17	Code.org: Analyze Visuals	Coding	AP.1B.10a	RL.5.4 RL.5.7	ELA

18	Adventure on the High Seas	Coding / Digital Literacy	IC.1B.3a AP.1B.9	RL.5.9	ELA
19	Beyond Gender Stereotypes	Digital Citizenship	IC.1B.2a	RI.5.1	ELA
20	Scratch-a-Summary	Coding	1B.AP.09	RI.5.2	ELA
21	Reading News Online	Digital Literacy	DA.1B.1	RI.5.3	ELA
22	Escape With Context Clues	Coding	AP.1B.3 AP.1B.5	RI.5.4	ELA
23	Variables and Text Structures	Coding	AP.1B.3a	RI.5.5	ELA
24	Photosynthesis Animation	Coding	DA.1B.1	L.5.3A	Science
25	Code.org: Coding Ecosystems → Account creation needed	Coding	1B.AP.10 1B.AP.11 1B.AP.12	L.5.3B	Science
26	Matter Matters Animation	Coding	AP.1B.3	P.5.5A	Science
27	Mix and Match Mixtures Game	Coding	AP.1B.1	P.5.5B	Science
28	Catch the Change Game	Coding	AP.1B.5	P.5.5C	Science
29	Animating Motion	Coding	AP.1B.3	P.5.6	Science
30	Traveling Home With Polaris Game	Coding	AP.1B.1a	E.5.8A.4	Science
31	Animating Fractions	Coding	AP.1B.3	5.NF.1 5.NF.6 5.NF.7	Math
32	Animating Decimals	Coding	AP.1B.3	5.NBT.7	Math
33	Place Value Pop-Up	Coding	AP.1B.2	5.NBT.3	Math
34	Digital Roundup	Coding	AP.1B.3	5.NBT.4	Math
35	Creating Conversions	Coding	AP.1B.9	5.MD.1	Math
36	Escape the Missing Variable	Coding	AP.1B.2	5.OA.2	Math
37	Plotting Points	Coding Robotics Unplugged	AP.1B.1	5.G.2	Math
38	Oceans and Continents Trivia	Coding	AP.1B.1	G.5.1	Social Studies

		Robotics Unplugged		G.5.2 G.5.3	
39	Newton's Robot Maze	Coding Robotics Unplugged	AP.1B.3	P.5.6	Science
40	Directions Through the Prism	Coding Robotics Unplugged	AP.1B.3	5.MD.4 5.MD.5	Math



Lessons and Activities

FIFTH GRADE

Week 1: Digital Friendships

Lesson overview: 	**Purpose:** Students will determine similarities and differences between multiple accounts of the same event or topic. They will demonstrate how an author's point of view affects his or her account of an event or topic through learning about cyberbullying and different points of view through digital friendships. **Lesson:** • Warm Up: Who Do You Chat With? • Compare and Contrast: Two Scenarios • Evaluate: Benefits and Risks • Wrap Up: Finish the Story
Lesson links/resources:	Digital Friendships—Common Sense Media
CS standards addressed:	Students will be able to: • Compare and contrast different kinds of online-only friendships • Describe the benefits and risks of online-only friendships • Describe how to respond to an online-only friend if the friend asks something that makes them uncomfortable Standards: • **IC.1B.3—**Seek diverse perspectives for the purpose of improving computational artifacts. • **IC.1B.3a—**Students will collaborate and receive feedback from others.
Time needed:	**Total time: 55 min** • Warm Up: Who Do You Chat With? **5 min** • Compare and Contrast: Two Scenarios **20 min** • Evaluate: Benefits and Risks **20 min** • Wrap Up: Finish the Story **10 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Lesson slides • Heart's Online Friendships handout—Teacher version • Finish the Story handout • Lesson quiz • Common Sense account Students: • Computer/tablet with internet access • Common Sense account
Subject integrated:	ELA
Other standards addressed:	**RI.5.6—**Analyze multiple accounts of the same event or topic, noting important similarities and differences in the point of view they represent.
Vocabulary:	Benefit: Something positive that results from a situation Private information: Information about you that can be used to identify you

	because it is unique to you (e.g., your full name or your address) Risk: Something negative or dangerous that could come from a situation
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.

Week 2: Let's Face It

Lesson overview: 	**Purpose:** Some online spaces can be full of negative, rude, or downright mean behavior. But what counts as cyberbullying? Students learn what is—and what isn't—cyberbullying and give them the tools they will need to combat the problem. **Lesson:** ● Warm Up: What is Cyberbullying? ● Expand: Who is involved? ● Analyze: Sondra's Story ● Wrap Up: Upstander Cards
Lesson links/resources:	Is It Cyberbullying?—Common Sense Media
CS standards addressed:	Students will be able to: ● Recognize similarities and differences between in-person bullying, cyberbullying, and being mean ● Empathize with the targets of cyberbullying ● Identify strategies for dealing with cyberbullying and ways they can be an upstander for those being bullied Standards: ● **IC.1B.3—**Seek diverse perspectives for the purpose of improving computational artifacts. ● **IC.1B.3a—**Students will collaborate and receive feedback from others.
Time needed:	**Total Time: 50 min** ● Warm Up: What is Cyberbullying? **5 min** ● Expand: Who is involved? **15 min** ● Analyze: Sondra's Story **15 min** ● Wrap Up: Upstander Cards **15 min**
Materials needed:	Teacher: ● Lesson slides ● What's Cyberbullying? video ● Sondra's Story ● Handout—Teacher version ● Lesson quiz ● Common Sense account Students: ● Blank index cards ● Markers and crayons ● Common Sense account
Subject integrated:	ELA
Other standards addressed:	**RI.5.7—**Draw on information from multiple print or digital sources, demonstrating the ability to locate an answer to a question quickly or to solve a problem efficiently.

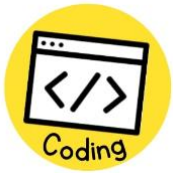
Vocabulary:	Bully: The person who is doing the bullying Bullying: Unwanted and aggressive verbal, social, or physical behavior towards another Bystander: Someone who sees a bullying or cyberbullying situation, but doesn't do anything to stop it Cyberbullying: Using digital devices, sites, and apps to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

Week 3: Media Balance

Lesson overview: 	**Purpose:** Helping kids learn what makes different media choices healthy or not is a good start, but how do we help them actually make responsible choices in the real world? Students will have the opportunity to create a personalized media plan. **Lesson:** ● Distribute the My Perfect Day handout and allow students 10 minutes to complete it. (Slide 4) ● Invite students to share out. For each student who shares, follow up by highlighting any activities involving: ○ Digital media (TV, movies, video games, texting, etc.) ○ Non-digital media (books, magazines, etc.) ○ Friends and/or family ○ The outdoors ○ Alone time ○ Hobbies ● Ask: "Now imagine that your perfect day will be granted—it's going to happen—but it has to happen for a week straight. For seven straight days, you will do the same thing—all the things in your perfect day. Given that, would you change anything in your perfect day? If so, what? Share your ideas with your partner." (Slide 5) ● Invite students to share out. Highlight any changes that result in more balance between the types of activities listed in step 2. For students whose perfect days involve only one or two types of activities, follow up by asking how they'd feel missing out on the other types. For example, how would they feel not seeing their family for a week? Or not having any alone time? Or not spending time outdoors?
Lesson links/resources:	Finding My Media Balance- Common Sense Media
CS standards addressed:	Students will be able to: ● Reflect on how balanced they are in their daily lives ● Consider what "media balance" means and how it applies to them ● Create a personalized plan for healthy and balanced media use Standards: ● **CS.1B.3—**Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies. [TROUBLESHOOTING] (P6.2)
Time needed:	**Total time: 50 min** ● Warm Up: My Perfect Day **15 min** ● Watch: What is Media Balance? **20 min** ● Create: My Media Plan **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Lesson slides ● My Media Balance video ● My Perfect Day handout

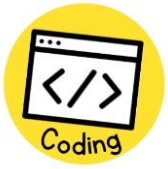
	• My Media Plan handout • Lesson quiz • Common Sense account Students: • Computer/tablet with internet access • Common Sense account
Subject integrated:	ELA
Other standards addressed:	**RI.5.8—**Explain how an author uses reasons and evidence to support particular points in a text, identifying which reasons and evidence support which point(s).
Vocabulary:	Balance: All of the parts are in the correct—though not necessarily equal—proportions Media: All of the ways that large groups of people get and share information (TV, books, internet, newspapers, phones, etc.) Media balance: Using media in a way that feels healthy and in balance with other life activities (family, friends, school, hobbies, etc.)
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.

Week 4: Oceans and Continents Clicker Game

Lesson overview:  	**Purpose:** The student will be able to locate physical features on a map and differentiate between landforms and bodies of water while using Scratch. **Lesson:** • The teacher will review the continents and oceans of the world using a blank map projected onto a screen. • The teacher will introduce the students to Scratch with the "Getting Started" tutorial video and create an account. • The teacher will show the "Make a Clicker Game" tutorial video to the students. • The students will create their own clicker game after watching the tutorials. • The teacher will put students in groups to exchange devices and play each other's games. • The teacher will project a blank map again and have students review continents and oceans.
Lesson links/resources:	• Scratch • Scratch Tutorial
CS standards addressed:	The students will be able to: • Differentiate between landforms and bodies of water using a Scratch game • Create a sprite for their game • Change the background of a Scratch • game • Create a clicker game using Scratch • Standards: • **AP.1B.1—**Compare and refine multiple algorithms for the same task and determine which is the appropriate.
Time needed:	**Total time: 70-80 min** • Continent/ocean review **5 min** • Getting Started tutorial **10 min** • Make a Clicker Game tutorial **10 min** • Create and test a clicker game **15-30 min** • Playing others' games to practice finding oceans and continents **15-20 min** • Continent/ocean review **5 min**
Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Scratch account Students: • Computer/tablet with internet access • Scratch account
Subject integrated:	Social Studies
Other standards	• **G.5.1—**Locate on a map the physical features of America prior to exploration.

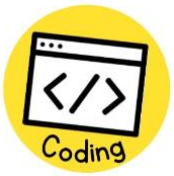
addressed:	• **G.5.2—**Describe physical features of the environment • **G.5.3—**Recognize maps, graphs, and other representations of the Earth.
Vocabulary:	Coding: Process of creating instructions for computers using programming language Algorithms: A list of steps to complete a task Sprite: A graphic on the screen with a location, size, and appearance User: Someone who uses an object, including software and hardware Background: Picture underneath the sprite and other objects
Notes:	Other sources to use for map practice if you don't want to use Scratch: • Program a Code and Go Mouse to get from one continent to another crossing specific oceans. ○ You will need a large map (at least poster sized if not larger.) →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

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

Lesson overview:  	**Purpose:** The students will take the information they have learned about the Aztecs, Incas, and Mayas to create a story as if "walking" through one of the ancient civilizations that they will choose. They will explain important facts about the ancient civilization that they have chosen. **Lesson:** ● Students will go to this link: Storytelling overview - CS First ○ Your students should log on to their CS First account. If they do not have an account, their progress cannot be saved. ● Your students will be completing the Lesson 1: Dialogue of Storytelling overview - CS First ○ Students should find "1. Dialogue" and click the blue start button. ○ As students move through these tutorials, they will be setting the scene for the ancient civilization they will choose. ● Let the students watch the Introduction to Dialogue and Sequencing tutorial. ○ They will see the starter project link to the right. Students need to select that link to begin their assignment. ● Setting the Scene ○ Students will choose a story starter, add two sprites, make the sprites face each other, add a backdrop, and say a starter phrase. The starter phrase may be a fact about the ancient civilization they have chosen, or an introduction to the civilization (e.g., "This is the ancient Mayan civilization."). ● Speaking and Responding ○ Students will code their project to have sprites respond to one another discussing facts about the civilization. ● Add-on ○ Please review the six options for Add-on to determine which is most beneficial for your students.
Lesson links/resources:	● CS First Starter Guide ● Storytelling overview - CS First (step-by-step instructions for students)
CS standards addressed:	The students will be able to: ● Make a storyboard using Scratch ● Create a sprite and background using Scratch Standards: ● **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 70-90 min** ● Dialogue: starter project **10 min** ● Setting the Scene **20 min** ● Speaking and Responding **20 min** ● Add-on **20 min**
Materials needed:	Teachers: ● Storytelling overview - CS First

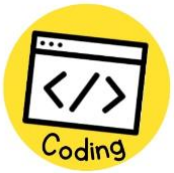
	• Computer • Projector/smartboard with sound • CS First account Students: • Computer/tablet with internet access • CS First account
Subject integrated:	ELA Social Studies
Other standards addressed:	ELA • **W.5.2—**Write informative/explanatory texts to examine a topic and convey ideas and information clearly. Social Studies • **H.5.6—**Differentiate among pre-Columbian civilizations.
Vocabulary:	Programs: An algorithm that has been coded into something that can be run by a machine Sequences: A set of logical steps carried out in order Events: Actions that cause something to happen Loops: The action of doing things over and over again Conditionals: Programming language commands
Notes:	• You may want to use student groups for this activity. • You may choose to project the tutorial videos on your smartboard/projector to help guide students. • You can add an Unplugged Story planning time. ○ Use a storyboard to sketch what they want for each part of the story. **15-20 min** →Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 6: Imagine a World With European Explorers

Lesson overview:  	**Purpose**: The students will imagine a world and create a moving scene using Scratch to show what the New World would have looked like to the European settlers. The students will upload a background picture of the new world and add to the scene to create what the settlers would have seen. **Lesson**: ● The teacher will show the Imagine a World tutorial video to the students. ● The students will create a written outline of their story. ● The students will click create (top left tab on the home page) on Scratch to write a story about the new world through the eyes of the European settlers. ○ The students will choose a background and sprite to represent their character and setting. ○ The students will use the information on their outline to put into their story. ● The students will share their stories with the whole class.
Lesson links/resources:	Scratch Tutorials: Imagine a World
CS standards addressed:	The students will be able to: ● Describe the New World using Scratch ● Create a sprite using Scratch ● Create a moving background using Scratch Standards: ● **AP.1B.4—**Decompose problems into smaller, manageable subproblems to facilitate the program development process.
Time needed:	**Total time: 65 min** ● Imagine a World tutorial video to explain how to make a moving story **5 min** ● Story planning using outline **15-20 min** ● Create a story on Scratch **30 min** ● Sharing story with class and or desk partners **10 min**
Materials needed:	Teachers: ● Computer ● Projector/smartboard with sound ● Scratch teacher account Students: ● Computer/ tablet with access to internet ● Scratch student accounts
Subject integrated:	Social Studies
Other standards addressed:	ELA ● **W.5.2—**Write informative/explanatory texts to examine a topic and convey ideas and information clearly. Social Studies ● **H.5.2—**Examine the reasons and impact for exploration of the New World.

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

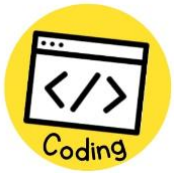
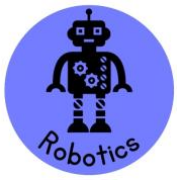
Week 7: 13 Colonies Greeting Cards

Lesson overview:
 
	Purpose:
	The students will create a "Wish You Were Here" greeting card from one of the 13 colonies to send back to England with a partner. The post card will include a background picture of the colony/area, along with details of the geography, economy, religion, native civilization, and daily life. Students will explain the colony in a postcard fashion to someone who has never seen it before.
	Lesson:
	• The teacher will review how to create a background and let students choose a background to represent their colony by using the Create a Backdrop tutorial (if needed).
	• The teacher will review information of the 13 colonies with students and show students how to find information listed above for their colony.
	  ○ This will be information provided by individual teachers.
	• The teacher will introduce greeting cards on Scratch.
	  ○ Please see the "Notes" section.
	• The students will access Scratch Create and create a sprite.
	  ○ The teacher can review the Add a Sprite tutorial if needed.
	  ○ The students will create the greeting card and have the sprite tell the information about the colony.
	• The teacher will introduce the voice over for the sprites by using the Record a Sound tutorial.

Lesson links/resources:	• Scratch Create • Create a Backdrop • Add a Sprite • Record a Sound tutorial
CS standards addressed:	The students will be able to: • Create a sprite using Scratch • Create a greeting card with facts about the colony using Scratch Standards: • **AP.1B.9—**Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.
Time needed:	**Total time: 80 min** • Review creating backdrop/students add backdrop **10 min** • Gather background information for chosen colony and write it down **20 min** • Record a Sound tutorial **10 min** • Introduce students to using voice over for animations **10 min** • Create postcards using background and sprites including recording of information **30 min**
Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Scratch account Students:

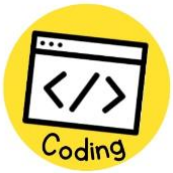
	• Computer/tablet with internet access • Scratch account
Subject integrated:	ELA Social Studies
Other standards addressed:	ELA • **W.5.2—**Write informative/explanatory texts to examine a topic and convey ideas and information clearly. Social Studies • **H.5.3—**Describe reasons for colonization in North America.
Vocabulary:	Design: A way to author computer applications using a combination of text, graphics, and style elements
Notes:	• Here are some examples of greeting cards to share with students: ○ Scratch Card Examples • Here is a YouTube tutorial for Scratch Greeting Cards →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 8: Road to Revolutionary War Using Robots

Lesson overview:
 
 
 
	Purpose:
	The students will use a codable robot to see important battle sites of the Revolutionary War.
	Lesson:
	• The teacher will provide information about each battle.
	  ○ This will be information provided by individual teachers.
	• The teacher will place a U.S. map on the floor and have the important battles of the war marked on the map. This needs to be a large map that is at least poster sized.
	  ○ Students will code the robot to reach each battlefield.
	  ○ Students will write the code on paper to show proof the students have been to each battlefield.
	  ○ Once they reach a battlefield, provide a link for them to go on a virtual field trip of that battle.
	• Unplugged option: If you don't have robots, do it "unplugged" and have students use coding cards to create the code they would use to program a robot to reach each battle site.

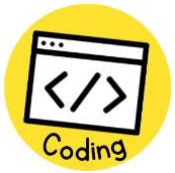
Lesson links/resources:	Links to virtual tours of American battlefields
CS standards addressed:	The students will be able to: • Program a robot to locate various battles of the Revolutionary War Standards: • **AP.1B.8—**Test and debug a program algorithm to ensure it runs as intended.
Time needed:	**Total time: 60 min** • Explain activity to students **5 min** • Review coding arrows how to program robots **10 min** • Programming robots to reach battleground activity **45 min**
Materials needed:	Teachers: • Large U.S. map • Codable robot (Please see the "notes" section.) Students: • Computer/tablet with internet access
Subject integrated:	Social Studies
Other standards addressed:	• **H.5.4—**Explain major events of the American Revolution and their outcomes. • **H.5.5—**Chart the causes and events leading to the American Revolution.
Vocabulary:	Debug: Find the exact point where a mistake was made in coding Unplugged: Creating codes without using a computer program
Notes:	Use QRCode Monkey for free QR codes to make it easier for students to access the virtual tour sites.

Week 9: Come See Our New Nation Advertisement

Lesson overview:  	**Purpose**: The students will create an advertisement for their new country to get people to come visit. Let them tell everyone all the good things about their country using Scratch. They will create a sprite to tell about the country and the new government that has been created. They will choose a background to represent the new country. **Lesson:** • The teacher will review the Articles of Confederation and the U.S. Constitution. • The students will review Scratch if needed, then the teacher will explain the lesson. ○ The teacher will show example advertisements to students. • The students will then create an advertisement for a new nation including the name of the new country, type of government, and interesting information about the country. • The students will also include a background to represent the new country. • Students may present advertisements to class if the teacher chooses.
Lesson links/resources:	• Scratch • Examples of advertisements to show students • The American Revolution for Kids - Articles of Confederation • The Constitution of the United States for Kids and Teachers - FREE Lesson Plans & Games for Kids
CS standards addressed:	The students will be able to: • Create an advertisement using Scratch describing the new country of America • Create a talking sprite using Scratch • Create a background representing the new America using Scratch Standards: • **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 75 min** • Review the Articles of Confederation and U.S. Constitution **10 min** • Show example advertisements **5 min** • Create advertisement for new nation **40 min** • Students present advertisements **20 min**
Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Scratch account Students: • Computer/tablet with internet access • Scratch account
Subject integrated:	Social Studies
Other standards addressed:	**CI.5.1—**Explain how weaknesses of the Articles of Confederation led to the Constitution.

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

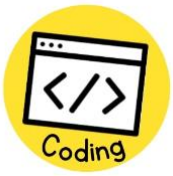
Week 10: Surprise Ending Inferences

Lesson overview:
 
 	
	Purpose:
	Students will listen to the beginning of a story provided by the teacher (or created by another student). Students will attempt to infer the object of their partner's story. Students will create a code with a surprise at the end to show their inference about the story.
	Lesson:
	• Assign partners among the students.
	• Direct the students to log in to their CS First account.
	• Students will watch videos and create an An Unusual Discovery project in Scratch for CS First.
	  ○ The projects of students in your class are automatically shared with your teacher account.
	• Have partners watch the introduction video and decide on the adventure type:
	  ○ Jungle Adventure
	  ○ Basketball Adventure
	  ○ Underwater Adventure
	  ○ Outer space Adventure
	• The students will create an adventure to present their inferences.
	• Instruct students to show their projects to a classmate.
	• When there are five minutes left in class, instruct students to find the wrap-up page and complete the short survey.
	• Discuss the lesson and facilitate a brief discussion about what students learned and experienced.
	  ○ Tell me about the program you made today.
	  ■ *What was your favorite part of this lesson?*
	  ■ *What did you learn about computer science and coding?*
	  ■ *What was the most challenging part of this lesson?*

Lesson links/resources:	An Unusual Discovery overview - CS First
CS standards addressed:	Students will be able to: • Use event blocks (like "when flag clicked") to trigger a series of code • Sequence at least three "say" blocks between two sprites (characters) to construct a dialogue • Program a conditional so that the computer can make a decision based on a user response • Produce repeated movements by applying control blocks to their program Standards: • **DA.1B.2—**Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.
Time needed:	**Total time: 60 min** • Watch the introduction video **5 min** • Open an adventure project link and add characters and dialogue **30 min** • Choose Add-ons **15 min** • Share projects **10 min**
Materials needed:	Teachers: • An Unusual Discovery Lesson Plan

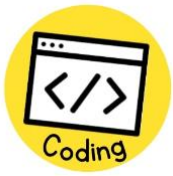
	• Shoutout sheet (optional) • CS First account Students: • Computer/tablet with internet access • Headphones • CS First account
Subject integrated:	ELA
Other standards addressed:	**RL.5.1—**Quote accurately from a text when explaining what the text says explicitly and when drawing inferences from the text.
Vocabulary	Control structures: Sections of code that order the direction or flow of how a program functions; Control structures include conditionals and loops Debugging: The process of identifying and fixing error(s) in a program when it is not functioning as expected
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 11: The Theme of Gumball

Lesson overview:  	**Purpose:** The student will create a story using Gumball. The student will then present their story. Their peers will write down what they believe the theme of the story to be. **Lesson:** ● Direct students to log in to their CS First account and select Gumball's Coding Adventure project in Scratch for CS First. ● Students will watch videos and create a story using Gumball. ○ The projects of students in your class are automatically shared with your teacher account. ● When there are five minutes left in class, instruct students to find the wrap-up page and complete the short survey. ● Discuss the lesson and facilitate a brief discussion about what students learned and experienced. ○ "Tell me about the program you made today." ○ "What was your favorite part of this activity?" ○ "What did you learn about computer science and coding?" ○ "What was the most challenging part of this activity?"
Lesson links/resources:	Gumball's Coding Adventure overview - CS First
CS standards addressed:	Students will be able to: ● Watch a series of videos and create one coding project with opportunities to personalize their work using Add-ons, which are mini-coding challenges that build on top of the core project Standards: ● **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** ● Introduction video **5 min** ● Foreshadow the Glitch with Dialogue **10 min** ● The Glitch and a Response **10 min** ● Create a New Scene **5 min** ● Reuse the Glitch **5 min** ● Conclude the Story and Choose and Add-on **10 min** ● The Amazing World of Gumball and Wrap Up**15 min**
Materials needed:	Teachers: ● Computer ● Projector/smartboard with sound ● The Theme of Gumball lesson plan ● Solution sheets ● Digital materials ● CS First account Students: ● Computer/tablet with internet access ● Headphones ● CS First account
Subject integrated:	ELA
Other standards addressed:	**RL.5.2—**Determine the theme of the story, drama, or poem from details in the text. Summarize the text.

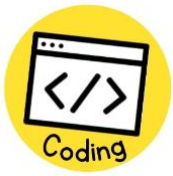
Vocabulary:	Sequences: A set of related events, movements, or things that follow each other in a particular order Events: An action that causes something to happen Loops: The action of doing something over and over again Conditionals: Statements that only run under certain conditions
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 12: Story (re)Telling

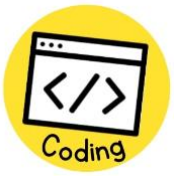
Lesson overview:  	**Purpose:** The teacher will read a text. The student will go to Scratch to create a background, sprites, and dialogue to retell the story. **Lesson:** ● Have students log in to the CS First website using their student accounts. ● Students will watch videos and create a Check It Out project in Scratch for CS First. ● Check in with students as they watch the videos and complete their projects. ○ Video 1: Students should use the tabs on their internet browser to switch between the Scratch for CS First tab and CS First tab. ● Students choose Add-ons to enhance their project. ● When there are five minutes left in class, instruct students to find the wrap-up page and complete the short survey. ● Your students' projects are automatically shared with your teacher account. Encourage students to also show their projects to a classmate. ● Discuss the lesson and facilitate a brief discussion about what students learned and experienced. ○ "What was your favorite part of this lesson?" ○ "What story did you tell?" ○ "What blocks did you use, and what did they do?"
Lesson links/resources:	Storytelling overview - CS First
CS standards addressed:	Students will be able to: ● Learn about how CS First works, then tell a story using Scratch for CS First where a character walks through a scene describing what they see Standards: ● **AP.1B.6b—**Students should document the plan as, for example, a storyboard, flowchart, pseudocode, or story map.
Time needed:	**Total time: 55 min** ● CS First Survey **5 min** ● Introduction to Dialogue and Sequencing **5 min** ● Setting the Scene **15 min** ● Speaking and Responding **10 min** ● Choose an Add-on **5 min** ● Reflection, Wrap Up, Next Steps **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Storytelling lesson plan ● Check It Out Solution Sheet ● CS First account Students: ● Computer/tablet with internet access ● Microphone ● Headphones ● CS First account

Subject integrated:	ELA
Other standards addressed:	**RL.5.2—**Determine the theme of the story, drama, or poem from details in the text. Summarize the text.
Vocabulary:	Storyboard: A graphic organizer that provides the viewer with a high-level view of a project Flowchart: A type of diagram that represents an algorithm, workflow or process Pseudocode: An artificial and informal language that helps programmers develop algorithms Story map: A web map that has been thoughtfully created, given context, and provided with supporting information so it becomes a stand-alone resource
Notes:	**Unplugged activity: Draw like a robot** • In this activity, students practice giving very specific instructions to a partner and compare their characters. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

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

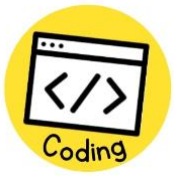
Lesson overview:  	**Purpose:** Events are very common in most computer programs. In this activity, students will use events to make a character move around the screen, make noises, and change backgrounds based on user-initiated events. **Lesson:** • Main Activity ○ Code.org Lesson 1- Hello World
Lesson links/resources:	Code.org Lesson 1- Hello World
CS standards addressed:	Students will be able to: • Create an interactive animation using sprites and events • Create new sprites and assign them costumes and locations Standards: • **1A.AP.09—**Model the way programs store and manipulate data by using numbers or other symbols to represent information. • **1A.AP.11—**Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. • **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals. • **AP.1B.5—**Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features. • **IC.1A.2—**Work respectfully and responsibly with others online.
Time needed:	**Total time: 55 min** • Code.org Lesson 1- Hello World **55 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Code.org account Students: • Computer/tablet with internet access • Code.org account
Subject integrated:	ELA
Other standards addressed:	**RL.5.3**—Compare and contrast two or more characters, settings, or events in a story or drama, drawing on specific details in the text.
Vocabulary:	Event: An action that causes something to happen Decompose: Break a problem down into smaller pieces
Notes:	

Week 14: Figurative Conversations

Lesson overview:  	**Purpose:** Students will use sprites and dialogue to show their understanding of literal and figurative meaning. Students will learn about computer science and then create a project that explores figurative language. **Lesson:** 1. Present the first two videos of Figurative Language from CS First to the entire class. 2. Navigate students to the Figurative Language lesson. 3. Reiterate the video instructions: a. Open the Starter Project b. Add your figurative language c. Show what the language seems to mean, and what the writer actually means using "say" blocks. 4. Release students to work. While they're working: a. Check that students are on-task. Students should have two browser tabs open: CS First videos and Scratch for CS First Coding Editor. Look for signs that students might not know what to do, like adding a lot of sprites or dragging out a lot of unrelated blocks. b. Talk with students one-on-one about how they can show the literal and figurative meanings of a piece of figurative language through what their character says and does.
Lesson links/resources:	CS First Figurative Language
CS standards addressed:	Students will be able to: • Apply their knowledge of figurative language to programming • Follow CS First videos to code their story and show the literal and figurative meaning of their text Standards: • **AP.1B.10—**Create programs that include sequence, events, loops, and conditionals. • **AP.1B.11—**Decompose problems into smaller, manageable subproblems to facilitate the program development process.
Time needed:	**Total time: 55 min** • Introduction to Figurative Language **5 min** • Explore Your Figurative Language **15 min** • Choose Add-ons **10 min** • Figurative Language survey **10 min** • Wrap Up: Next Steps **10 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Figurative Language Lesson Plan • CS First account Students: • Computer/tablet with internet access • Headphones • CS First account
Subject integrated:	ELA
Other standards	**RL.5.4—**Determine the meaning of words and phrases as they are used in a text, including figurative language such as metaphors and similes.

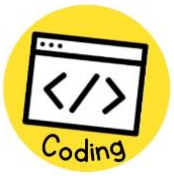
addressed:	
Vocabulary:	Sequence: A set of logical steps carried out in order Loops: The action of doing something over and over again Conditionals: Statements that only run under certain conditions Decompose: Break a problem down into smaller pieces Program: An algorithm that has been coded into something that can be run by a machine. Parallelism: The process of events happening at the same time, either independently or interdependently. Debugging: The process of identifying and fixing error(s) in a program when it is not functioning as expected. Control structures: Sections of code that order the direction or flow of how a program functions; the control structure in this lesson is focused on loops.
Notes:	**Unplugged activity: Programmer says** • Programmer says reinforces the concepts of sequencing instructions and conditionals. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 15: Create-a-Character

Lesson overview:  	**Purpose:** Students tell the story of a character who does not have a school assignment when it's due. Students will describe this character through their actions, thoughts and words. This lesson can also be used to compare and contrast chapters, scenes, or stanzas. **Lesson:** ● Introduce the Characterization lesson by reading aloud the following prompts: ○ "In this lesson, you will use code to tell the story of someone who needs to turn in an assignment for school but hasn't completed it yet." ○ "What are some reasons that someone might not have an assignment done? What are some wacky, fun, or fantastical reasons someone might not have their homework? What might someone do when they find out that they don't have their assignment?" ● Review the Characterization ELA concept by reading aloud the following prompt: ○ "There are lots of ways to tell a story about someone not having an assignment. You will use what your character says, thinks, or does to tell readers about your character. Are they smart, brave, resilient, creative? You will get to decide." ● Before computer time, consider using a pre-writing lesson, like brainstorming character traits, to connect this to your classroom learning objectives. ● Tell students that they will apply their knowledge of characterization to code a story in Scratch for CS First. They will follow CS First videos to code their story and develop their character.
Lesson links/resources:	● CS First Characterization ● Characterization Lesson Plan
CS standards addressed:	Students will be able to: ● Tell the story of a character who does not have a school assignment when it's due ● Describe this character through their actions, thoughts and words ○ This example project shows what a student can create if they complete all of the Add-ons in the lesson. Standards: ● **AP.1B.10—**Create programs that include sequences, events, loops, and conditionals. ● **AP.1B.11—**Decompose problems into smaller, manageable subproblems to facilitate the program development process.
Time needed:	**Total time: 50 min** ● Introduction to Characterization **10 min** ● What is Your Character Thinking? **10 min** ● Choose an Add-on **10 min** ● Characterization Survey **10 min** ● Wrap Up: Next Steps **10 min**
Materials needed:	Teachers: ● Computer ● Projector/smartboard with sound ● Characterization Lesson Plan ● CS First account Students:

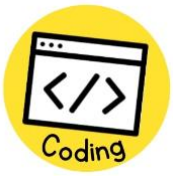
	• Computer/tablet with internet access • Headphones • CS First account
Subject integrated:	ELA
Other standards addressed:	• **RL.5.3—**Compare and contrast two or more characters, settings, or events in a story or drama, drawing on specific details in the text (e.g., how characters interact). • **RL.5.5—**Explain how a series of chapters, scenes, or stanzas fits together to provide the overall structure or a particular story, dramas, or poem.
Vocabulary:	Sequence: A set of logical steps carried out in order Loops: The action of doing something over and over again Conditionals: Statements that only run under certain conditions Decompose: Break a problem down into smaller pieces Program: An algorithm that has been coded into something that can be run by a machine
Notes:	**Unplugged activity: Programmer says** • Programmer says reinforces the concepts of sequencing instructions and conditionals. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 16: Two Points of View

Lesson overview:  	**Purpose:** Students work with a partner to tell the same story from two different points of view. **Lesson:** ● Review the Narration ELA concept by reading aloud the following prompts: ○ "The narrator describes what is happening in the story but is not directly involved in it. They tell the story from the third-person point of view and use words like "he," "she," and "them." If a character within the story is describing the action themselves, they use words like "I," "we," and "us." This is called telling the story from the first-person point of view." ○ "In this lesson, one person in your group will tell a story from the first-person, and the other will tell the story from the third person." ○ "Before computer-time, consider using a partner pre-writing lesson, like brainstorming a scene from a story they would like to tell, to connect this to your classroom learning objectives." ● Tell students that they will apply their knowledge of narration to code a story. They will follow CS First videos to code their story in the first- or third-person point of view.
Lesson links/resources:	● Narration overview ● Narration Lesson Plan
CS standards addressed:	Students will be able to: ● Sequence "say" blocks to tell a story in the first-person or third-person ● Use event blocks (like "when flag clicked") to trigger a series of code ● Use wait blocks to construct a dialogue between two sprites or think blocks to describe a new sprite ● Use motion blocks (e.g., "glide to X Y"), sound effects, and animation to personalize their story Standards: ● **IC.1B.3a—**Students will collaborate and receive feedback from others. ● **AP.1B.9—**Take on varying roles, with teacher guidance when collaborating with peers during the design, implementation, and review stages of program development.
Time needed:	**Total time: 50 min** ● Introduction to Narration **10 min** ● Start Your Story **15 min** ● Narration Survey **10 min** ● Wrap Up: Next Steps **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Narration Lesson Plan ● CS First account Students: ● Computer/tablet with internet access ● Headphones ● Journal ● CS First account
Subject integrated:	ELA

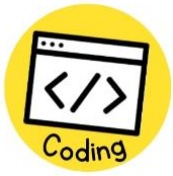
Other standards addressed:	• **RL.5.5—**Explain how a series of chapters, scenes, or stanzas fits together to provide the overall structure or a particular story, dramas, or poem. • **RL 5.6—**Describe how a narrator's or speaker's point of view influences how events are described..
Vocabulary:	Design**:** A way to author computer applications using a combination of text, graphics, and style elements in a unified code-space Program: An algorithm that has been coded into something that can be run by a machine
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 17: Code.org—Analyze Visuals
Lesson 1, Level 1

Lesson overview:  	**Purpose:** The student will create a poem/story and an image to go with their poem. The student will present their picture and poem/story to show their connection. The student should comment on their picture with an explanation of their story. **Lesson:** ● Introduce the day's activity by reading aloud the following prompts: ○ "Today we are going to use our creativity while we learn to code. There are lots of ways to be creative. What are some ways you like to be creative?" ○ "Did you know that computer science can be very creative? In fact, you can make art with code. As a reminder, code is a set of instructions that a computer can understand. Just like choosing which colors of paint to use, choosing what code you write can be an opportunity to express your creativity too!" ○ "We're going to use poetry to inspire our creative coding today." ○ "Does anyone know what poetry is?" ○ "Do you have a favorite poem or poet?" ● Main Activity ○ Read poetry together—Read a few poems together as a class and practice naming the mood of each poem. Encourage students to share their interpretation, even if it's different from someone else's. ● Wrap Up ○ Check for understanding with two sticky notes. On one sticky note, have your students write one thing they learned about coding today. On the second, write one thing you have learned about poetry.
Lesson links/resources:	Poem Art - Code.org
CS standards addressed:	Students will be able to: ● Develop programs that respond to timed events ● Illustrate shifts in the mood of a poem ● Represent figurative and literal language in a text Standards: ● **AP.1B.10a—**Students should explain code choices using comments within the code, presentations, and demonstrations.
Time needed:	**Total time: 55 min** ● Warm Up **5 min** ● Main Activity (Poem Art) **45 min** ● Wrap Up **5 min**
Materials needed:	Teachers: ● Computer ● Projector/smartboard with sound ● Lesson 1: Poem Art ● CS First account

	Students • Computer/tablet with internet access • Headphones • Sticky notes • CS First account
Subject integrated:	ELA
Other standards addressed:	• **RL.5.4—**Determine the meaning of words and phrases as they are used in a text, including figurative language such as metaphors and similes. • **RL.5.7—**Analyze how visual and multimedia elements contribute to the meaning, tone, or beauty of a text.
Vocabulary:	Code**:** To write instructions for a computer Event: An action that causes something to happen Program**:** A series of steps that have been coded into something that can be run by a machine
Notes:	Additional sites: Art overview - CS First

Week 18: Adventure on the High Seas	
Lesson overview:	**Purpose:**

 	Create the beginning of a story and use details to help infer. Share with your partner, and they will finish the story using your details. **Lesson:** ● Warm Up ○ Have students write the beginning of a story (setting, characters, plot, etc.). Have students share with a partner. ● Activity ○ Watch video and open Scratch to begin creating the end of the story. Have students share their story with their partner or with the class. ● Wrap Up ○ Have students write about how their story and their partner's story were similar and different.
Lesson links/resources:	Adventure on the High Seas overview - CS First
CS standards addressed:	Students will be able to: ● Collaborate and receive feedback from others ● Collaborate with peers during the design, implementation, and review stages of program development Standards: ● **IC.1B.3a—**Students will collaborate and receive feedback from others. ● **AP.1B.9—**Take on varying roles, with teacher guidance when collaborating with peers during the design, implementation, and review stages of program development.
Time needed:	**Total time: 60 min** ● Warm Up **5 min** ● Adventure Writing **50 min** ● Wrap Up **5 min**
Materials needed:	Teachers: ● Computer ● Projector/smartboard with sound ● Adventure on the High Seas Lesson Plan ● CS First account Students: ● Computer/tablet with internet access ● CS First account
Subject integrated:	ELA
Other standards addressed:	**RL.5.9—**Compare and contrast stories in the same genre on their approaches to similar themes and topics.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Dialogue**:** Conversation between two or more people as a feature of a book, play, or movie Scene**:** The place where an incident in real life or fiction occurs or occurred.

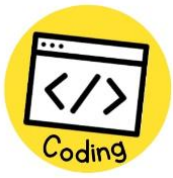
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 19: Beyond Gender Stereotypes

Lesson overview: 	**Purpose:** Students will encounter all kinds of stereotypes in the media. But are kids always aware of what they are seeing? Help your students think critically about how gender stereotypes can affect the ways they view themselves and others. **Lesson:** • Warm Up: Gender Stereotypes • Analyze: True! Or is It? • Create: Stereotype Avatar • Wrap Up: "Just Because" poem
Lesson links/resources:	Beyond Gender Stereotypes - Common Sense Education
CS standards addressed:	Students will be able to: • Define "gender stereotypes" and describe how they can be present online • Create an avatar and a poem that show how gender stereotypes impact who they are Standards: • **IC.1B.2a—**Students will demonstrate an understanding of diversity in ability and interests by developing artifacts and tools that use different methods of communication and/or appeal to different users.
Time needed:	**Total time: 50 min** • Warm Up: Gender Stereotypes **10 min** • Analyze: True! Or Is It? **10 min** • Create: Stereotype Avatar **20 min** • Wrap Up: "Just Because" Poem **10 min**
Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Lesson slides • Stereotype Avatar handout • "Just Because" Poem • Lesson quiz • Common Sense account Students: • Computer/tablet with internet access • Common Sense account
Subject integrated:	ELA
Other standards addressed:	**RI.5.1—**Quote accurately from a text when explaining what the text says explicitly and when drawing inferences from the text.
Vocabulary:	Avatar: An image or character that represents a person online

Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.

Week 20: Scratch-a-Summary

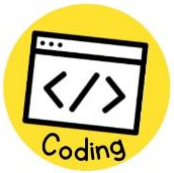
Lesson overview:  	**Purpose:** Students will determine when a text has two or more main ideas. They will identify all of the main ideas or points presented by the author. Students will identify key details from the text and show how they support each of the main ideas. Students will use details and the main idea and create a Scratch summary of the informational text. **Lesson:** ● Students will gather as a group and discuss main ideas and spark ideas. Teachers will provide the text. ● Participants will create projects working at their own pace. ● Gather together and share and reflect.
Lesson links/resources:	Scratch
CS standards addressed:	Students will be able to: ● Use details and the main idea to create a Scratch summary of the informational text Standards: ● **1B.AP.09—**Create programs that use variables to store and modify data.
Time needed:	**Total time: 50 min** ● Warm Up **15 min** ○ Favorite Meal ○ Party Games ● Main Activity **30 min** ● Create Your Own Story **20 min** ● Wrap Up **10 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● www.scratch.mit.edu/ideas ● Scratch account
Subject integrated:	ELA
Other standards addressed:	**RI.5.2—**Determine two or more main ideas of a text and explain how they are supported by key details; summarize the text.
Vocabulary:	Variable: A label for a piece of information used in a program Sprite: A graphic on the screen with a location, size, and appearance
Notes:	Alternate site: Lesson 6—Blank Space Stories →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 21: Reading News Online

Lesson overview: 	**Purpose:** Kids find and read news in lots of different ways. But studies show they are not very good at interpreting what they see. How can we help them get better? Teaching your students about the structure of online news articles is an important place to start. **Lesson:** • Warm Up: The News You Know • Explore: The Parts of a News Site • Watch: How to Read News Online • Wrap Up: Label a News Page
Lesson links/resources:	Reading News Online - Common Sense Media
CS standards addressed:	Students should be able to: • Understand the purposes of different parts of an online news page • Identify the parts and structure of an online news article • Learn about things to watch out for when reading online news pages, such as sponsored content and advertisements Standards: • **DA.1B.1—**Organize and present collected data visually to highlight relationships and support a claim.
Time needed:	**Total time: 50 min** • Warm Up: The News You Know **10 min** • Explore: The Parts of a News Site **20 min** • Watch: How to Read News Online **10 min** • Wrap Up: Label a News Page **10 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with internet • Lesson slides • Video: Reading News Online • Mix & Match Cards handout • Label a News Page handout • Lesson quiz • Common Sense account Students: • Scissors • Glue • Blank paper • Common Sense account
Subject integrated:	ELA
Other standards addressed:	**RI.5.3—**Explain the relationships or interactions between two or more individuals, events, ideas, or concepts in a historical, scientific, or technical text based on specific information in the text.
Vocabulary:	Article**:** A written story in a newspaper, magazine, or online news site Commercial**:** An advertisement intended to make money News**:** New information about recent or important events

Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Common Sense Digital Media.

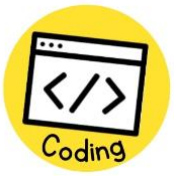
Week 22: Escaping With Context Clues

Lesson overview:
 
 	
	Purpose:
	Students learn basic video game coding concepts by making different types of games. In this lesson students will create an escape game where peers must use context clues to escape.
	Lesson:
	• Direct students to CS First, have them login to their account, and watch the first video.
	• Students watch videos and create a Gaming Story project using Scratch for CS First.
	• Video 2: Students who are exploring the costumes or sounds tab may not understand the coding aspect of this lesson. Remind them of the task and try to answer any questions they have.
	• Students choose Add-ons to enhance their project.
	• When there are five minutes left in class, instruct students to find the wrap-up page and complete the short survey.
	• Your students' projects are automatically shared with your teacher account. Encourage students to also show their projects to a classmate.
	• Discuss the lesson and facilitate a brief discussion about what students learned and experienced.
	  ○ "What story did you talk about gaming?"
	  ○ "What was your favorite part of this lesson?"
	  ○ "What blocks did you use, and what did they do?"

Lesson links/resources:	Game Design overview - CS First
CS standards addressed:	Students will be able to: • Learn basic video game coding concepts by making different types of games, including racing, platform, launching, and more Standards: • **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals. • **AP.1B.5—**Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features.
Time needed:	**Total time: 50 min** • CS First survey **5 min** • What is Computer Science? **5 min** • Choose a Character **10 min** • Tell a Story **5 min** • Add-on **5 min** • Reflection **5 min** • Wrap Up: Gaming Story **10 min** • Next Steps **5 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Game Design Lesson Plan • Solution sheets • CS First account Students: • Computer/tablet with internet access • CS First account

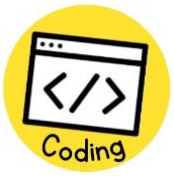
Subject integrated:	ELA
Other standards addressed:	**RI.5.4—**Determine the meaning of general academic and domain-specific words and phrases in a text relevant to a grade 5 topic or subject area.
Vocabulary:	Sequence: A set of logical steps carried out in order Loops: The action of doing something over and over again Conditionals: Statements that only run under certain conditions Decompose: Break a problem down into smaller pieces Program: An algorithm that has been coded into something that can be run by a machine Events: An action that causes something to happen
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at CS First.

Week 23: Variables and Text Structures

Lesson overview:  	**Purpose:** Students will use Scratch to create a game show to teach a younger person about all text structures. Students will choose a sprite and add a variable with each of the text structure types: compare/contrast, chronological, cause/effect, descriptive, and problem/solution. **Lesson:** • Gather as a group to discuss the game and spark ideas on how to design a game that teaches about types of text structures. • Students will individually create projects working at their own pace. • Gather together and share and reflect.
Lesson links/resources:	• Scratch • Catch Game tutorial • Clicker Game tutorial • Scratch Game ideas
CS standards addressed:	Students will be able to: • Create a game show format explaining text structures • Create sprites and add variables for each type of text structure Standards: • **AP.1B.3a—**Students should be able to create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** • Review text structures **5 min** • Model how to choose a sprite and build a game **10 min** • Have students create their own game to review text structures **30 min** • Swap games with another student to review **15 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Scratch account • Scratch Game ideas Students: • Computer/tablet with internet access • Paper • Pencil • Scratch account
Subject integrated:	ELA
Other standards addressed:	**RI.5.5—**Compare and contrast the overall structure (e.g., chronology, comparison, cause/effect, problem/solution) of events, ideas, concepts, or information in two or more texts.
Vocabulary:	Sequence: A set of logical steps carried out in order Loops: The action of doing something over and over again Conditionals: Statements that only run under certain conditions Decompose: Break a problem down into smaller pieces Program: An algorithm that has been coded into something that can be run by a machine Events: An action that causes something to happen

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

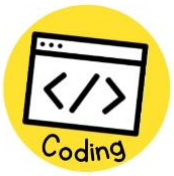
Week 24: Photosynthesis Animation

Lesson overview:
 
 	
	Purpose:
	The students will utilize Scratch to create an animation explaining photosynthesis.
	Lesson:
	• Introduction
	  ○ Review photosynthesis and create a drawing explaining the process.
	• Create
	  ○ Students will create a background and animated sprite to showcase the process of photosynthesis.
	• Share
	  ○ Students will share and present their animation to the classroom.

Lesson links/resources:	Scratch
CS standards addressed:	Students will be able to: • Create and animate a sprite. • Add text or audio to a sprite to share information. • Create a background that reflects the information being presented. Standard: • **DA.1B.1—**Organize and present collected data visually to highlight relationships and support a claim.
Time needed:	**Total time: 60 min** • Introduction to activity/draw photosynthesis process **10 min** • Background creation **10 min** • Create and animate sprite **15 min** • Add photosynthesis text or audio **15 min** • Wrap up activity/present **10 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Scratch account Students: • Computer/tablet with internet access • Paper • Pencil • Scratch account
Subject integrated:	Science
Other standards addressed:	**L.5.3A—**Students will demonstrate an understanding of photosynthesis and the transfer of energy from the sun into chemical energy necessary for plant growth and survival.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen

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

Week 25: Coding Ecosystems

Lesson overview:
 
	Purpose:
	The students will utilize their knowledge of ecosystems and food webs to recreate a selected biome with an animated food web demonstration that showcases all trophic levels of that biome.
	Lesson:
	• Introduction
	  ○ Introduce the students to creating a sprite using the Hello World tutorial discussing the sprites' purpose.
	• Investigate
	  ○ Five students will go to the CSC Starter Program- Ecosystem to access the example of Meadow Ecosystem code.
	• Create
	  ○ Students will create a new ecosystem of their choice with food web connections.
	• Students will share and present.

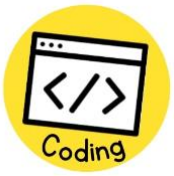
Lesson links/resources:	• Code.org • CSC Project: Ecosystems using Sprite Lab
CS standards addressed:	Students will be able to: • Create code to illustrate a new ecosystem using block code • Create sprites that represent different plants and animals Standard: • **1B.AP.10—**Create programs that include sequences, events, loops, and conditionals. • **1B.AP.11—**Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. • **1B.AP.12—**Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

Time needed:	**Total time: 60 min**
	• Introduction to activity **10 min**
	  ○ Hello World sprite creation tutorial
	• Investigate and discuss **10 min**
	  ○ Meadow Ecosystem code example
	• Create new ecosystem **15 min**
	• Add food web connections for the sprites in new ecosystem **15 min**
	• Wrap up activity/present **10 min**

Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Code.org account Students: • Computer/tablet with internet access • Paper • Pencil • Code.org account
Subject integrated:	Science

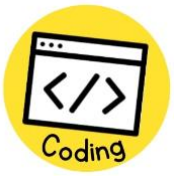
Other standards addressed:	**L.5.3B—**Students will demonstrate an understanding of a healthy ecosystem with a stable web of life and the roles of living things within a food chain and/or food web, including producers, primary and secondary consumers, and decomposers.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen
Notes:	Students need to have the basic understanding of creating a sprite for this activity. Access the following link for a tutorial: • Hello World sprite creation tutorial →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Code.org.

Week 26: Matter Matters Animation

Lesson overview:  	**Purpose:** Students will create an animation on Scratch demonstrating the movement of molecular structure through the different states of matter. **Lesson:** ● Introduction ○ Review the states of matter. ○ Have students create a sprite of matter that can change into different states (e.g., water). ● Create ○ Students will create the different scenarios needed to change their chosen sprite to a solid, liquid, and gas. ● Students will share and present.
Lesson links/resources:	Scratch
CS standards addressed:	Students will be able to: ● Use sequences, events, loops, and conditionals Standards: ● **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** ● Introduction to activity **10 min** ● Background creation **10 min** ● Create and animate sprite **15 min** ● Add audio or text to animation **15 min** ● Wrap up activity/present **10 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Scratch account
Subject integrated:	Science
Other standards addressed:	**P.5.5A.2—**Collect, analyze, and interpret data from measurements of the physical properties of solids, liquids, and gasses (e.g., volume, shape, movement, and spacing of particles).
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached Algorithms: A procedure or formula used for solving a problem

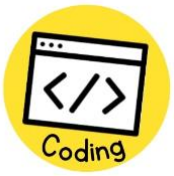
Notes:	A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 27: Mix and Match Mixtures Game

Lesson overview:  	**Purpose:** Students will design an interactive animation on Scratch that builds an interactive game or demonstration combining different materials to create examples of mixtures and solids, depending on the student's chosen algorithms for their activity. **Lesson:** ● Introduction ○ Showcase a Scratch Catching Game, then have the students watch a Catching Game tutorial. ● Create ○ Students will design a game that allows for users to catch items to create a mixture or solution depending on the students choice of focus for the game. ● Share and allow students in the class to play and critique other students' games. ○ How accurate was the game? ○ Did the game correctly catch items? ○ Did the students' choices correctly reflect their choice of mixture or solution?
Lesson links/resources:	● Scratch ● Catching Game tutorial
CS standards addressed:	Students will be able to: ● Create an interactive game to reflect the theme ● Create algorithms that allow them to accurately catch chosen items ● Create multiple animated sprites Standard: ● **AP.1B.1—**Compare and refine multiple algorithms for the same task and determine which is most appropriate.
Time needed:	**Total time: 60 min** ● Introduction to activity: Catching Game tutorial **10 min** ● Create game background **10 min** ● Create items for mixtures to catch **15 min** ● Create game code **15 min** ● Wrap up activity/present **10 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Paper ● Pencil ● Scratch account
Subject integrated:	Science
Other standards addressed:	**P.5.5B—**Students will demonstrate an understanding of mixtures and solutions.

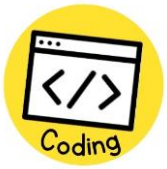
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false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

Week 28: Catch the Change Game

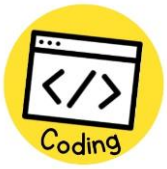
Lesson overview:  	**Purpose:** The students will modify and create an interactive game on Scratch that others may use that will focus on distinguishing between a chemical and physical change. **Lesson:** ● Introduction ○ Review the Catching Game tutorial. ● Create ○ The students will click on create, then they will design and create a game that allows for users to catch examples of either a physical or chemical change. ○ Students will create a visual background and multiple sprites to reflect their chosen theme and choice of catching a physical or chemical change. ● Students will share and allow others to play and critique their created games. ○ How accurate was the game? ○ Did the game correctly catch items? ○ Did the game correctly catch items?
Lesson links/resources:	● Scratch ● Catching Game tutorial
CS standards addressed:	Students will be able to: ● Create an interactive game to reflect the theme ● Create algorithms that allow them to accurately catch chosen items ● Create multiple animated sprites Standard: ● **AP.1B.5—**Modify, remix, or incorporate portions of an existing program into one's own work to develop something new or add more advanced features.
Time needed:	**Total time: 60 min** ● Introduction to activity **10 min** ● Create game background **10 min** ● Create chemical/physical change **15 min** ● Create game code **15 min** ● Wrap up activity/present **10 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Paper ● Pencil ● Scratch account
Subject integrated:	Science
Other standards addressed:	**P.5.5C—**Students will demonstrate an understanding of the difference between physical and chemical changes.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance

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

Week 29: Animating Motion

Lesson overview:  	**Purpose:** The students will create an animated presentation on Scratch that demonstrates Newton's Laws of Motion. **Lesson:** ● Introduction ○ Review Newton's three laws of motion. ○ Have students choose one of Newton's laws to animate and present. ● Create ○ Have the students choose and create a sprite of their choice that can be animated to show motion. ○ Have the students create a background that will allow the motion reflected in that law. ● Let students present their project and have others in the class critique. ○ Did the animation showcase the chosen Newton's law? ○ Did the created background reflect the motion being animated?
Lesson links/resources:	Scratch
CS standards addressed:	Students will be able to: ● Create an animated sprite to represent an object in motion ● Create a background that reflects the chosen sprite and it's motion Standards: ● **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** ● Introduction to activity **10 min** ● Create background **10 min** ● Create sprites **15 min** ● Add audio or text to sprites **15 min** ● Wrap up activity/present **10 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Paper ● Pencil ● Scratch account
Subject integrated:	Science
Other standards addressed:	**P.5.6—**Students will demonstrate an understanding of the factors that affect the motion of an object through a study of Newton's laws of motion.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen Sequence: A set of logical steps carried out in order

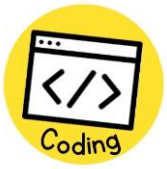
Notes:	A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.
Week 30: Traveling Home With Polaris Game	
Lesson overview:	**Purpose:**

	The students will create an interactive map or setting on Scratch that will have students using the North Star (also called the Pole Star, or Polaris) in order to successfully travel from a set starting point to their new desired location. The focus is on how Polaris is used to help with find true geographic north and how this is possible without the use of GPS or other directional tools such as a compass.
	Lesson:
	• Introduction
	  ○ Showcase a Travel Game example, then have students watch a Scratch Moving Game tutorial.
	• Create
	  ○ The students will click on "Create," then they will design a game that creates a ship or boat that will travel through the created background using Polaris as the guide to their ending point.
	  ○ Students will create a visual background and multiple sprites to reflect their chosen theme and obstacles they must avoid during their travels. Students need to choose the best way to make their sprite move through the maze of obstacles to reach the end.
	• Reflect
	  ○ Allow students to share their game and critique other students' games.
	  ▪ How accurate is the game?
	  ▪ Did the game correctly reflect the theme?

Lesson links/resources:	• Scratch • Travel Game example • Scratch Moving Game tutorial
CS standards addressed:	Students will be able to: • Create an interactive game to reflect the theme • Create code that allows for movement and direction from set keys when used • Solve a problem by completing the maze traveling task with the best solution Standard: • **AP.1B.1a—**Students should be able to look at different ways to solve the same task and decide which would be the best solutions.
Time needed:	**Total time: 60 min** • Introduction to activity **10 min** • Create background **10 min** • Create sprite(s) **15 min** • Add game animation **15 min** • Wrap up activity/present **10 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Scratch account Students: • Computer/tablet with internet access • Scratch account

Subject integrated:	Science
Other standards addressed:	**E.5.8A.4—**Construct scientific arguments to support claims about the importance of astronomy in navigation and exploration, including the use of telescopes, compasses, and star charts.
Vocabulary:	Sprite: A graphic on the screen with a location, size, and appearance Costume: An image that gives a sprite its appearance Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached Algorithms: A procedure or formula used for solving a problem
Notes:	A tutorial for the creation of backgrounds, sprites, and animation can be found on Scratch as well as on CS First to help students with the creation of their animated presentation. Students may code by using arrows or mouse clicks to create the ability to change directions based on Polaris's location throughout the activity. →Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

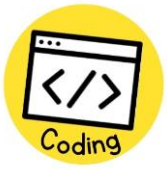
Week 31: Animating Fractions	
Lesson overview:	**Purpose:**

	Students will create an animation on Scratch that will describe the steps of how to add, subtract, multiply, or divide fractions with unlike denominators. Next they will animate the process.
	Lesson:
	• Introduction
	  ○ Go over vocabulary and create or identify fractions to be used in the animation and solve them.
	• Animation
	  ○ Create the characters, background, animation, and dialogue.
	• Wrap Up
	  ○ When the animation is complete, the students can share and allow other students to play.

Lesson links/resources:	• Scratch • Clicker Game tutorial • Clicker Game examples
CS standards addressed:	Students will be able to: • Create sequence, events, loops, and conditionals Standards: • **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** • Introduction to the activity **10 min** • Identify fractions to be used in the animation and solve **15 min** • Create the animation **20 min** • Wrap up activity/present **15 min**
Materials needed:	Teachers: • Computer • Projector/smartboard with sound • Scratch account • Educator guides Students: • Computer/tablet with internet access • Scratch account
Subject integrated:	Math
Other standards addressed:	• **5.NF.1—**Add and subtract fractions with unlike denominators (including mixed numbers) by replacing given fractions with equivalent fractions in such a way as to produce an equivalent sum or difference of fractions with like denominators. • **5.NF.6—**Solve real-world problems involving multiplication of fractions and mixed numbers, e.g., by using visual fraction models or equations to represent the problem. • **5.NF.7—**Apply and extend previous understandings of division to divide unit fractions by whole numbers and whole numbers by unit fractions.
Vocabulary:	Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached

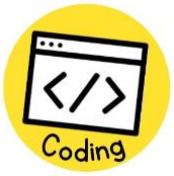
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 32: Animang Decimals	
Lesson overview:	**Purpose:**

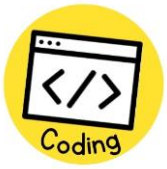
	Students will create an animation on Scratch that will describe the steps of how to add, subtract, multiply, and divide decimals. Next they will animate the process.
	Lesson:
	• Introduction
	  ○ Go over the vocabulary, create or identify decimal numbers to be used in the animation, and solve them.
	• Animation
	  ○ Create the characters, background, animation, and dialogue.
	• Wrap Up
	  ○ When the animation is complete, students can share or allow classmates to play.

Lesson links/resources:	• Scratch • Clicker Game tutorial • Decimal Game example
CS standards addressed:	Students will be able to: • Create sequence, events, loops, and conditionals Standards: • **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** • Introduction to the activity **10 min** • Identify decimal problems to be used in the game, and solve **15 min** • Create the animation **20 min** • Wrap up activity/present **15 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Scratch account Students: • Computer/tablet with internet access • Scratch account
Subject integrated:	Mathematics
Other standards addressed:	**5.NBT.7—**Add, subtract, multiply, and divide decimals to hundredths, using concrete models (to include, but not limited to base ten blocks, decimal tiles, etc.) or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used.
Vocabulary:	Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached

Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.
Week 33: Place Value Pop-Up	
Lesson overview:	**Purpose:**

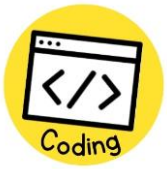
	Students will create a game on Scratch that displays a multi-digit decimal number to the millions and thousandth's place. Once the number pops up, it will have an underlined digit. The player will then click the correct place value from a list of three options. **Lesson:** ● Introduction ○ Go over the vocabulary, create, and identify decimal numbers to be used in the animation, and underline one digit in each decimal number. ● Animation ○ Create the characters, background, animation, and dialogue. ● Wrap Up ○ When the animation is complete, the students can share or allow other students to play.
Lesson links/resources:	● Scratch ● Clicker Game tutorial
CS standards addressed:	Students will be able to: ● Use variables Standards: ● **AP.1B.2—**Create programs that use variables to store and modify data.
Time needed:	**Total time: 60 min** ● Introduction to the activity **10 min** ● Create the decimal numbers being used and the place value forms that will pop up on the screen for or each number **15 min** ● Create the animation **20 min** ● Wrap up activity/present **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Scratch account
Subject integrated:	Math
Other standards addressed:	**5.NBT.3—**Read decimals to thousandths.
Vocabulary:	Variables: A label for a piece of information used in a program
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 34: Digital Roundup	
Lesson overview:	**Purpose:**

	Students will create a game on Scratch that displays a multi-digit decimal number to the millions and thousandth's place. Once the number pops up it will have an underlined digit. The player will then click the correct rounded place value from a list of three options. **Lesson:** ● Introduction ○ Go over the vocabulary, create and identify decimal numbers to be used in the animation, and underline one digit in each decimal number. ● Animation ○ Create the characters, background, animation, and dialogue. ● Wrap Up ○ When the animation is complete, the students can share or allow classmates to play.
Lesson links/resources:	● Scratch ● Clicker Game tutorial
CS standards addressed:	Students will be able to: ● Create sequences, events, loops, and conditionals Standards: ● **AP.1B.3—**Create programs that include sequences, events, loops and conditionals.
Time needed:	**Total time: 60 min** ● Introduction to the activity **10 min** ● Create the multi-digit numbers being used in the activity and decide which digits will be underlined **15 min** ● Create the animation **20 min** ● Wrap up activity/present **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Scratch account
Subject integrated:	Mathematics
Other standards addressed:	**5.NBT.4—**Use place value understanding to round decimals to any place.
Vocabulary:	Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached

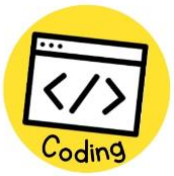
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 35: Creating Conversions

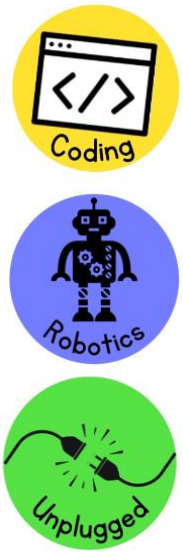
Lesson overview:  	**Purpose:** Students will create a tutorial on Scratch with a partner on how to teach Metric and US Customary Conversions using length, weight, capacity, meter, liter, and gram. **Lesson:** ● Introduction ○ Go over the vocabulary, create, and identify metric and U.S. customary conversions measuring for length, weight, capacity, meter, liter, and gram. ● Animation ○ Create the characters, background, animation, and dialogue. ● Wrap Up ○ When the animation is complete, the students can share or allow classmates to play.
Lesson links/resources:	Scratch
CS standards addressed:	Students will be able to: ● Collaborate with peers during the design, implementation, and review stages of program development Standards: ● **AP.1B.9—**Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.
Time needed:	**Total time: 60 min** ● Introduction to the activity **10 min** ● Create the steps of the tutorial **15 min** ● Create the animation **20 min** ● Wrap up activity/present **15 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Scratch account
Subject integrated:	Math
Other standards addressed:	**5.MD.1—**Convert among different-sized standard measurement units within a given measurement system (customary and metric) (e.g., convert 5 cm to 0.05 m), and use these conversions in solving multi-step, real-world problems.
Vocabulary:	Event: An action that causes something to happen

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

Week 36: Escape the Missing Variable

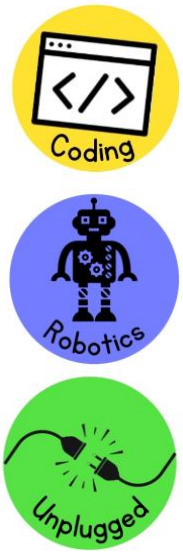
Lesson overview:  	**Purpose:** Students will create an escape room on Scratch. The students will need to find the missing variables to escape, display the digit that is represented by the variable, and solve. **Lesson:** ● Introduction ○ Go over the vocabulary and create and identify algebraic expressions to be used in the escape room. ● Animation: ○ Create the characters, background, animation, and dialogue. ● Wrap Up ○ When the animation is complete, the students can share or allow classmates to play.
Lesson links/resources:	● Scratch ● Math Escape Room example
CS standards addressed:	Student will be able to: ● Create programs using variables Standards: ● **AP.1B.2—**Create programs that use variables to store and modify data.
Time needed:	**Total time: 60 min** ● Introduction to the activity **10 min** ● Identify algebraic expressions to be used in the escape room **10 min** ● Create the animation **20 min** ● Wrap up activity/present **20 min**
Materials needed:	Teacher: ● Computer ● Projector/smartboard with sound ● Scratch account Students: ● Computer/tablet with internet access ● Scratch account
Subject integrated:	Math
Other standards addressed:	**5.OA.2—**Write simple expressions that record calculations with numbers and interpret numerical expressions without evaluating them.
Vocabulary:	Event: An action that causes something to happen Sequence: A set of logical steps carried out in order Conditionals: Expressions that evaluate to either true or false Loops: A sequence of instructions that is continually repeated until a certain condition is reached
Notes:	→Teachers will need to create FREE teacher and/or student accounts (when applicable) at Scratch.

Week 37: Plotting Points

Lesson overview: 	**Purpose:** The students will utilize a programmable robot in order to correctly traverse and graph the first quadrant plane to plot coordinates on the x- and y- axis. **Lesson:** ● Introduction ○ Group students (three to five per group) to introduce the activity and review how to operate the codable robot. ● Allow students to take turns practicing the use of the robot's features. ● Have students mark the coordinates on their paper. ○ Include some coordinates that you want students to reach with their robot, as well as some coordinates to avoid (obstacles). ■ You can also place physical obstacles on the graph such as pencils, erasers, sticky notes, etc. ● Let students write the code for the robots to reach each coordinate while avoiding all obstacles. ○ Coding cards are a great tool to help students think out their codes. ● Allow students to practice running the code and debugging. ● Wrap Up ○ Review the lesson and allow groups to show their coding skills off by showing other groups how their robot reaches the coordinates without hitting obstacles. **Non-codable robot option:** ● If you do not have a robot, students can still code! Print off a picture of a robot and allow them to move the paper robot using the squares on the graph paper as a measurable distance. You may want to use coding cards for the unplugged version of this activity.
Lesson links/resources:	● Coding cards ● Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley)
CS standards addressed:	The student will be able to: ● Plot coordinates using a programmable robot Standards: ● **AP.1B.1—**Compare and refine multiple algorithms for the same task and determine which is the most appropriate.
Time needed:	**Total time: 60 min** ● Introduction to activity **10 min** ● Build/modify robot **10 min** ● Code robot for obstacle course **15 min** ● Practice runs/code improvements **15 min** ● Wrap up activity/present **10 min**
Materials needed:	Teachers: ● Large (at least poster sized) four-quadrant graph paper (one per group) ● Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) ● List of coordinates (provided by teacher) Students:

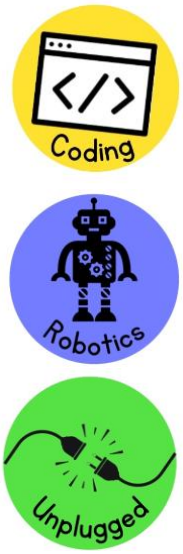
	• Pencil • Quadrant graph paper • Optional: Coding cards
Subject integrated:	Math
Other standards addressed:	**5.G.2—**Represent real-world and mathematical problems by graphing points in the first quadrant of the coordinate plane and interpret coordinate values of points in the context of the situation.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

Week 38: Oceans and Continents Trivia

Lesson overview: 	**Purpose:** The students will each write a trivia question and answer on a notecard for each continent and ocean. They will then work with teams and place the cards on the corresponding oceans and continents. The students have to program the robot to get the cards on the map. The students keep the cards they get correct, and the goal is to have a card for each continent and ocean. **Lesson:** ● Introduction ○ Explain the activity to the students ● Trivia Questions ○ Have each student write trivia questions on a notecard for each continent and ocean using these links: ■ Continent Facts ■ Ocean Facts For Kids ● Robot Review ○ Have students review how to use the codable robot. ● Game ○ Have students (one at a time) choose from a shuffled deck of notecards, decide on an answer, and program the robot to travel from a starting point (that the teacher decides) to the continent or ocean that answers the trivia card. **Non-codable robot option:** ● If you do not have a robot, students can still code! Print off a picture of a robot and allow them to move the paper robot. Make sure to show the students the measurable distance that the robot can move per step; this can be the length of the robot picture, or the length of a notecard. You may want to use coding cards for the unplugged version of this activity.
Lesson links/resources:	● Continent Facts ● Ocean Facts For Kids ● Coding cards ● Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley)
CS standards addressed:	The students will be able to: ● Program a robot to travel to the oceans and continents on a large map ● Create a trivia question of each of the continents and oceans ● Answer questions about each of the continents and oceans to gain points during the game Standards: ● **AP.1B.1—**Compare and refine multiple algorithms for the same task and determine which is the appropriate.
Time needed:	**Total time: 60 min** ● Explanation of activity **5 min** ● Creating trivia questions **15 min** ● Review codable robots **5 min** ● Playing games **40 min**

Materials needed:	Teacher: ● Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) ● Large world map Students: ● Computer/tablet with internet access ● Optional: Coding cards
Subject integrated:	Social Studies
Other standards addressed:	● **G.5.1—**Locate on a map the physical features of America prior to exploration. ● **G.5.2—**Describe physical features of the environment. ● **G.5.3—**Recognize maps, graphs, and other representations of the Earth.
Vocabulary:	Robot: A standalone computer system that performs physical and computational activities Coding: How we communicate with computers Algorithms: A specific procedure for solving a computational problem Program: A specific set of ordered operations for a computer to perform
Notes:	Have students research the oceans and continents using the links listed above and write down 5 interesting facts about each ocean and continent to use for the programming lesson. This is a great review game.

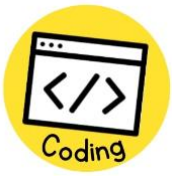
Week 39: Newton's Robot Maze

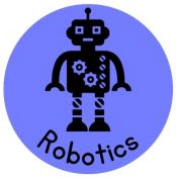
Lesson overview: 	**Purpose:** The students will create and build a robot that will use their knowledge of Newton's Laws of Motion to traverse a pre-made obstacle course that focuses on these laws. **Lesson:** ● The teacher will introduce the teacher-created obstacle course with multiple areas that students will need to avoid in an attempt to complete the course the fastest. ○ Teachers: The area to be avoided can be anything that could slow motion, such as net force or slow motion. ● The students will work in pairs or groups to write code that will be used to program their codable robot to complete the course in the fastest time without hitting any obstacles that may slow them down. **Non-codable robot option:** ● If you do not have codable robots, let the students act as the robot. One student will read the code aloud while another student acts as the robot. ● Once the pairs or groups complete their code, they should walk through the program with the robot in order to test the accuracy of their program and debug as necessary. ● Teachers should have the students run the course three times and record the time for each attempt. Students should also have a data chart to record their maze attempt times. ○ Which team was the fastest? ○ Did your group complete the obstacle course with your written program? ○ What changes, if any, would you make with your program to make it more successful? ○ What issues did your group have during this process?
Lesson links/resources:	Teach Engineering—What Are Newton's Laws?
CS standards addressed:	The students will be able to: ● Create code to traverse a set obstacle course avoiding objects and areas that slow motion ● Problem solve to improve any coding issues for the robot Standard: ● **AP.1B.3—**Create programs that include sequences, events, loops, and conditionals.
Time needed:	**Total time: 60 min** ● Introduction to activity **10 min** ● Build/modify robot **10 min** ● Code robot for obstacle course **15 min** ● Practice runs and code improvements **15 min** ● Wrap up activity/timed attempts **10 min**
Materials needed:	Teachers:

	• Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) • Obstacle course with obstacles that slow down motion • Paper/pencil (to record time data) • Timer • Time sheets Students: • Pencil • Time sheets • Paper • Timer
Subject integrated:	Science
Other standards addressed:	**P.5.6—**Students will demonstrate an understanding of the factors that affect the motion of an object through a study of Newton's Laws of Motion.
Vocabulary:	Direction: A course along which something moves Motion: The action or process of moving or being moved Position: Put or arrange (someone or something) in a particular place or way Speed: The rate at which someone or something is able to move or operate Sequence: A set of logical steps carried out in order Loops: A sequence of instructions that is continually repeated until a certain condition is reached Conditionals: Expressions that evaluate to either true or false Algorithms: A procedure or formula used for solving a problem
Notes:	

Week 40: Directions Through the Prism

Lesson overview:





Purpose:

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

Lesson:

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

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

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

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

Lesson links/resources:

- Sketching Rectangular Prism Video
- Coding cards

CS standards addressed:

The student will be able to:
- Create directions on creating a prism

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

Time needed:

Total time: 60 min

	• Introduction **10 min** • Create **15 min** • Program **10 min** • Share **25 min**
Materials needed:	Teacher: • Computer • Projector/smartboard with sound • Optional: Codable robot (e.g., Code and Go Mouse, Dash Robot, Botley) Students: • Pencil • Graph paper • Answer grid
Subject integrated:	Math
Other standards addressed:	• **5.MD.4—**Measure volumes by counting unit cubes, using cubic centimeters, cubic inches, cubic feet, and improvised units. • **5.MD.5—**Relate volume to the operations of multiplication and addition and solve real-world and mathematical problems involving volume.
Vocabulary:	Robot: A stand-alone computer system that performs physical and computational activities Coding: How we communicate with computers Algorithms: A specific procedure for solving a computational problem Program: A specific set of ordered operations for a computer to perform the exact point where a mistake was made in coding
Notes:	

Appendix A: Code.org

I'd like to start using Code.org in my classroom. How should I start?

https://support.code.org/hc/en-us/articles/228116468-I-d-like-to-start-using-Code-org-in-my-classroom-How-should-I-start-

How to create a teacher account:

https://support.code.org/hc/en-us/articles/228116468-I-d-like-to-start-using-Code-org-in-my-classroom-How-should-I-start-

How to create a classroom section:

https://support.code.org/hc/en-us/articles/115000488132-Creating-a-classroom-section

Finding curriculum and lesson plans:

https://support.code.org/hc/en-us/articles/115001595051-Finding-curriculum-and-lesson-plans

Code.org Support

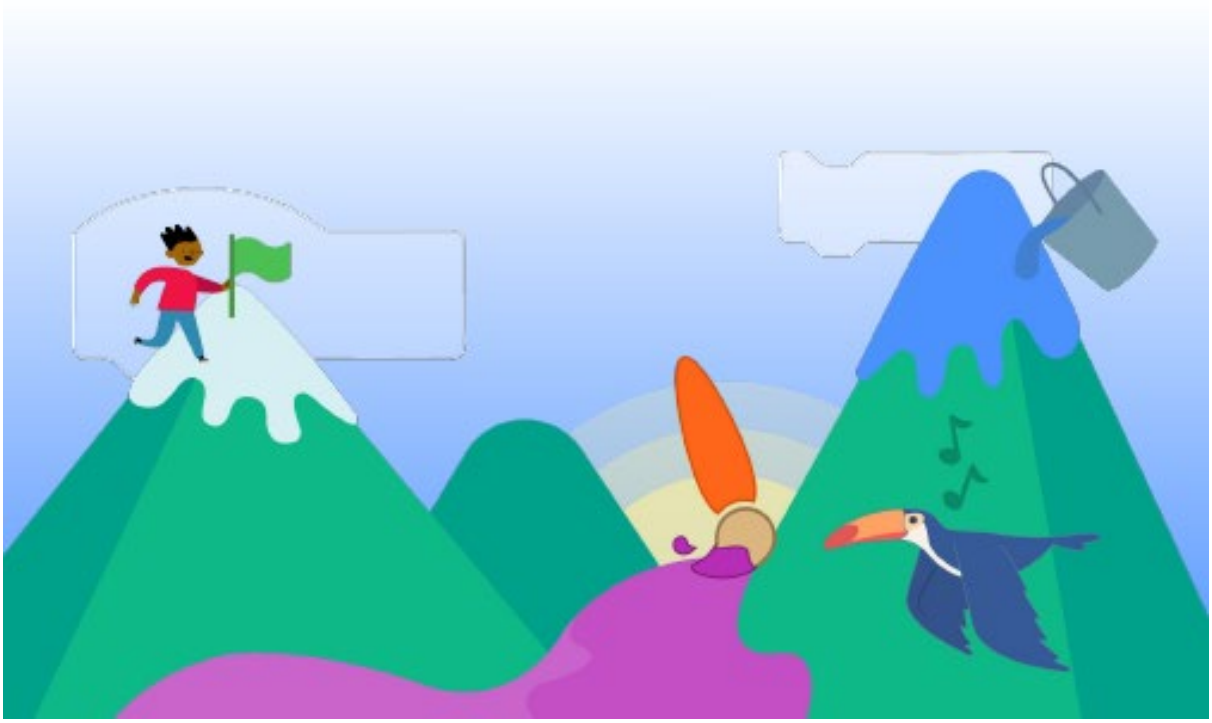
https://support.code.org/hc/en-us

Appendix B: Scratch

SCRATCH

Educator's Guide

- Teacher Accounts
- Beginner's Guide
- Lesson Guides



Teacher Accounts

As an educator, you can request a Scratch Teacher Account. A Scratch Teacher Account provides educators with additional features to manage student participation on Scratch, including the ability to create student accounts, organize student projects into studios, and monitor student comments. This guide will walk you through creating an account, creating a class, adding and managing your students, and creating class studios. You can also see our Scratch for Educators page and our Teacher Account FAQ page for additional information.

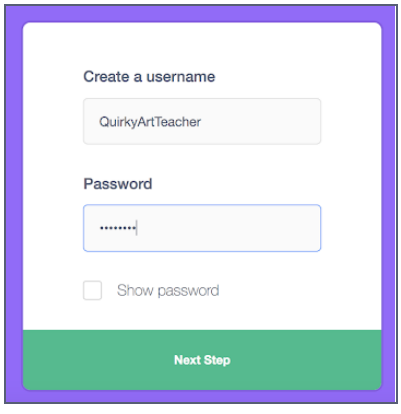
Create Your Teacher Account

Visit this link to get started: https://scratch.mit.edu/educators/register

You'll be prompted to create a username and password. ***Make sure that your username does not contain your name or personal information***, like your school, location, or email address.

Within the Scratch community, all users are asked to refrain from sharing personal information through their usernames. ***It's important that both you and your students follow these guidelines. Accounts that do not adhere to these guidelines will be deleted.***

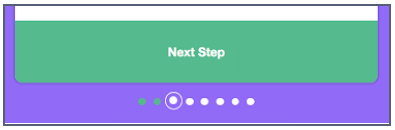
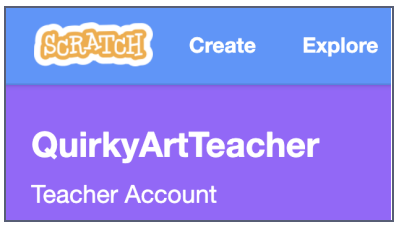
Creating your teacher account



Tips for making your username

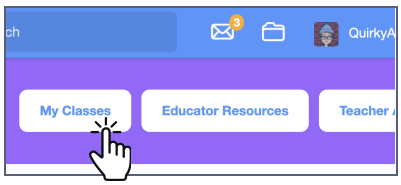
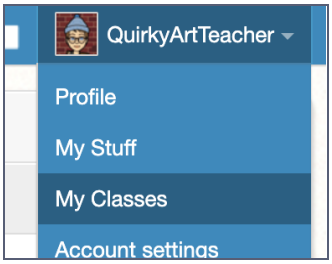
- **Incorporate the name of the subject you teach**
  - ex: QuirkyArtTeacher
- **Use a tool or term from the subject you teach**
  - ex: MetamorphicRocks
- **Add an important date, be unique**
  - ex: Bibliophile1440
- **Make it memorable with a pun or an alliteration!**
  - ex: TyranoTeacher

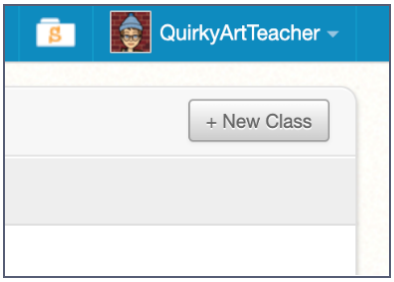
Be sure to make a note of your username and password.

	Click through each step to **complete registration**.
	Log into your email and confirm your email address. Check your spam folder if you do not see the email. Once you have **confirmed your email address**, we'll review your account.
	Once your account has been reviewed and approved, **you will receive a welcome email**. Then, you can **log into your teacher account at scratch.mit.edu**!

Create a Class

Creating classes allows you to manage groups of students, and create studios where your students can add their projects.

Creating your class	
 	Once you have successfully logged into your Teacher Account, if you are looking at the homepage, there will be a bar at the top of the screen with three options. Select "**My Classes**." You can also access your classes from the dropdown under your username.

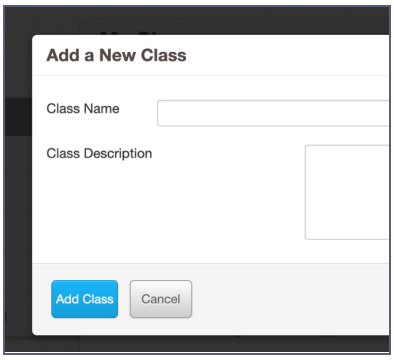


To create a class, click the "**+ New Class**" button at the top right of the page.

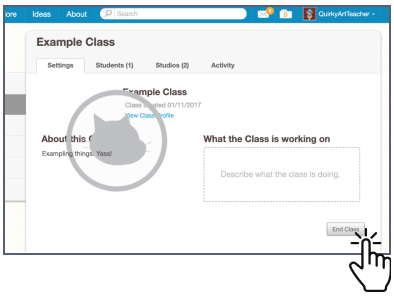
Enter the class name and description.

Warning: Do not include real names and locations, like the name of your school or city/town.

Once you have created a class, you can add students.



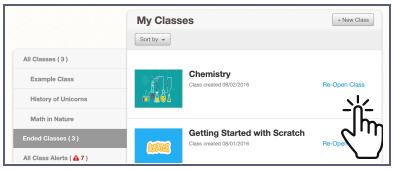
Ending your class



To end a class, under "My Classes," choose your class and on the Settings tab, click the "End Class" button.

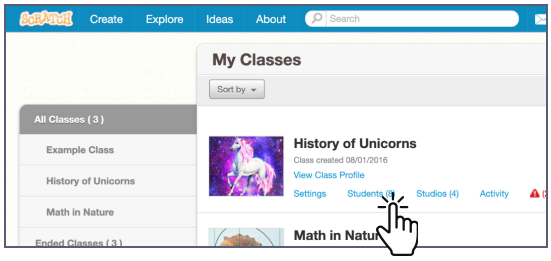
When you end a class, your class profile page will be hidden and your students will no longer be able to log in (but their projects and the class studios will still be visible on the site).

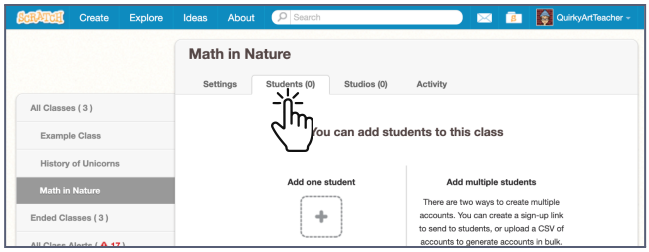
You may re-open the class at any time. By going to the "Ended Classes" tab and clicking the "Re-Open Class" link near the class you want to reopen.



Add Students to Your Class

While on “My Classes,” select the class and then click on “Students” (either the link under the class name or the Students tab). Once created, your student accounts will appear here.

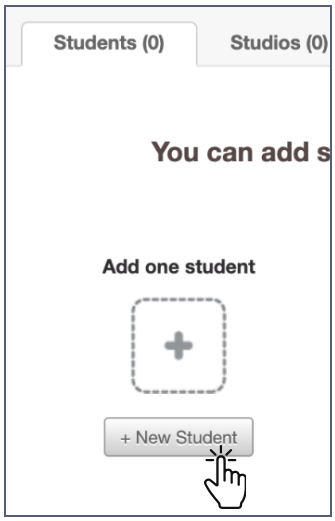




There are three ways to add students to your class. The first method allows you to add an individual student to a class. Methods 2 and 3 allow you to add multiple students to a class.

Tip: Create a naming convention as a guideline for generating usernames. For example, you may want each name to include an abbreviation for the course name, the class section, and the student’s number on your roster (ex: VisArts-02-17). Use the Student Username List we have created to record the usernames and passwords your students have created.

Method 1: Add Individual Students



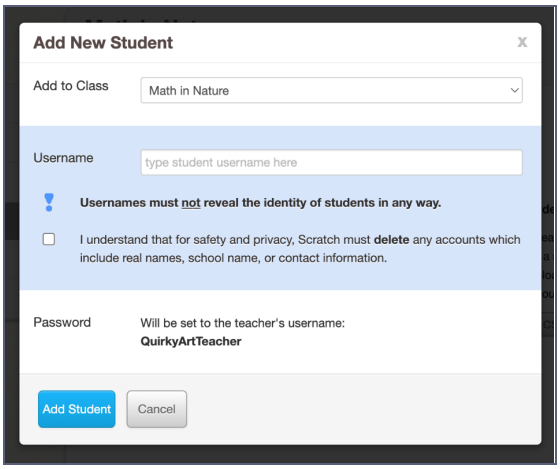
Click the “+ New Student” button to add students individually.

Confirm the correct class is showing in the “Add to Class” dropdown menu.

You will be prompted to create a username for this student.

Warning: Make sure that the usernames you create do not contain identifying information about yourself, your students, or your school. Accounts that do not adhere to these guidelines will be deleted.

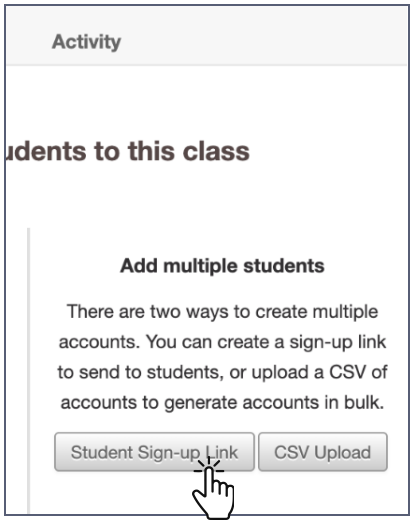
The password for this student username will automatically be set as the username of your teacher account.

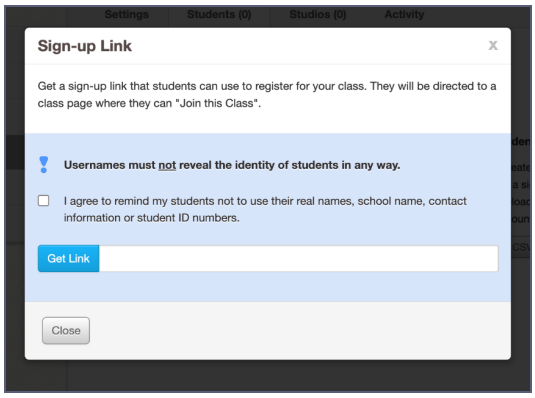


Have students log into their accounts and change their passwords as soon as possible.

Tip: It is not possible to add an existing Scratch account to a classroom. You will need to create a new Student Account for them using your Teacher Account. A student can only be a part of one class, and it is not possible to transfer students from one class or teacher to another.

Method 2: Student Sign-up Link



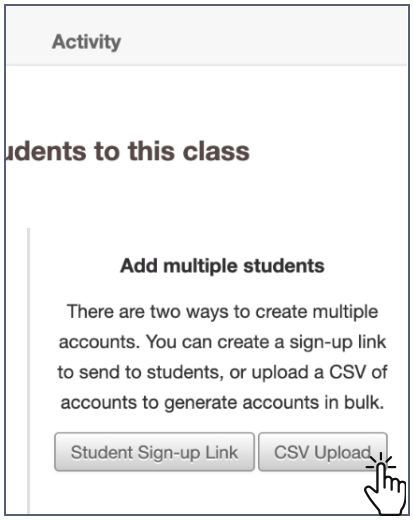


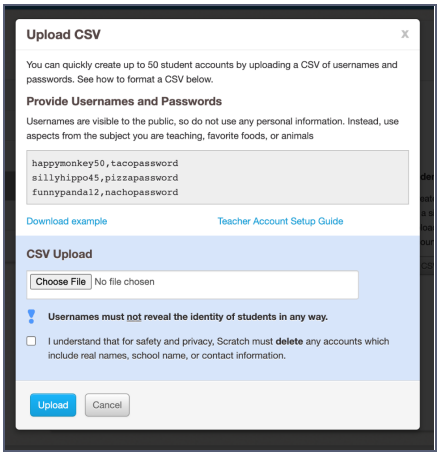
Clicking the "Student Sign-Up Link" button brings you to another window and clicking the "Get Link" button will generate a link that will allow your students to join the class you have just created. The link will start with "http://scratch.mit.edu/signup…"

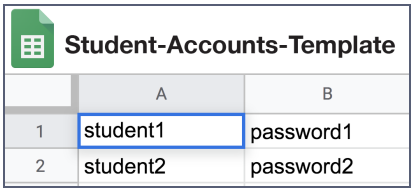
Students can then create their own usernames and passwords.

Warning: Remind your students that, when making their usernames, the username should not contain identifying information about themself, their teacher, or their school. Accounts that do not adhere to these guidelines will be deleted.

Method 3: CSV Upload







Click the “CSV Upload” button on the class page.

Using the template provided by clicking the “Download example” link, create a username and password for each of your students. You can use the template provided or create your own spreadsheet with student usernames in column A and passwords in column B. To upload your own template, you’ll need to save the file as a CSV file.

Once you’ve created usernames and passwords for each student and saved the file, click the “Choose file” button to locate the file, then click the “Upload” button.

It is not possible to add more than 250 students to a single class. You can, however, create a new class and add another 250 student accounts to each new class.

Warning: Make sure that the usernames you create do not contain identifying information about yourself, your students, or your school. Accounts that do not adhere to these guidelines will be deleted.

You can add students via any of these methods at any time under the “Students” tab.

Creating Studios for Student Work

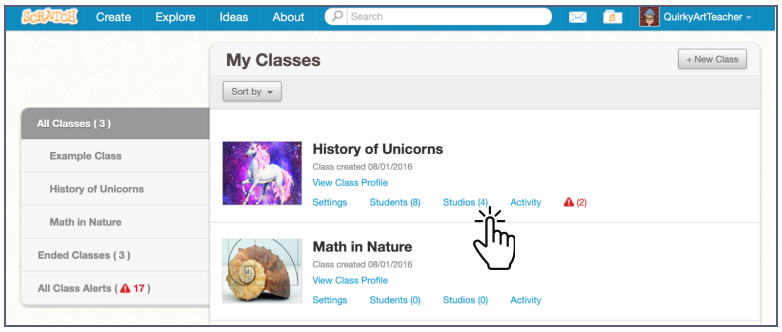
Studios allow you to create collections of student projects for specific classes or assignments. This makes it easier for you to view their projects throughout their creative process. It also makes it easier for students to collaborate and be inspired by each other's work.

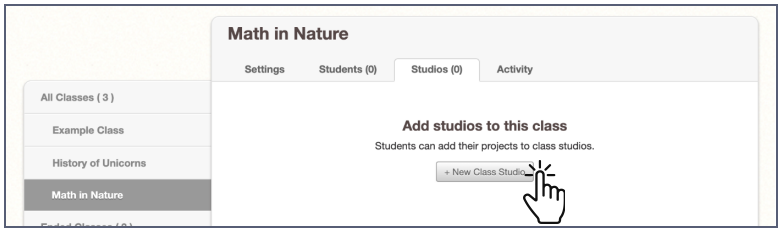
Scratcher status is required in order to create a studio, and the person who created the studio is automatically assigned the role of "host." There is only one host per studio, and only studio hosts can edit the title, thumbnail, and description.

Studios are immediately public, even those created in the context of a class. Unlike Scratch projects, there is no share/unshare option for studios. Everyone can follow a studio, see studio comments and projects, and leave a comment or add a project (unless commenting or the ability to add projects is turned off).

There are two ways to create a studio on a teacher account. Method one creates studios that automatically add all students in a class as curators. Method two creates studios without automatically adding students as curators, and students or any Scratcher can be individually added as curators.

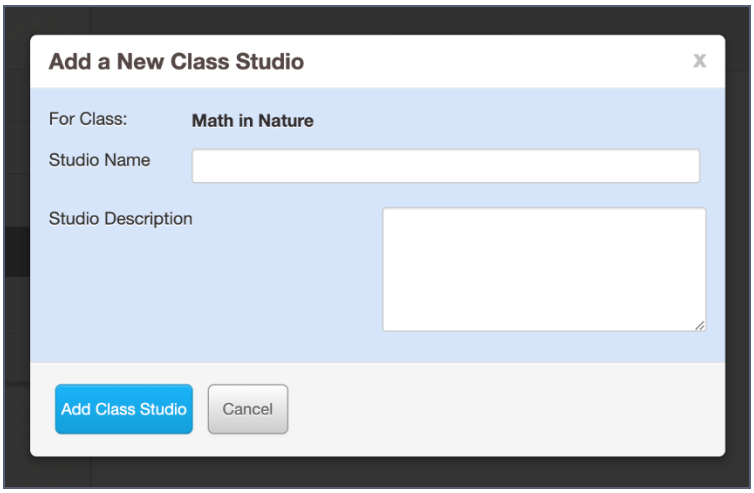
Method 1: Create a studio that automatically adds all students in a class as curators



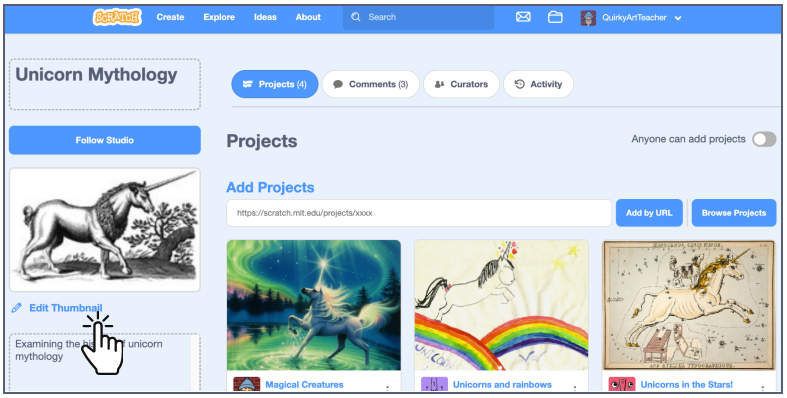


Once logged into your Scratch account, go to "My Classes."

Choose the class to assign the studio to, then click on "Studios" (either the link under the class name or the Studios tab). Then click the "+ New Class Studio" button.



On the window that appears, you will be asked to **give the studio a name and description**. (These can always be adjusted in the studio later.) In the description, you can share the theme of the studio, what kinds of projects you are looking to include… Just be sure your title and description don't reveal any personal information (like school name or first and last name).

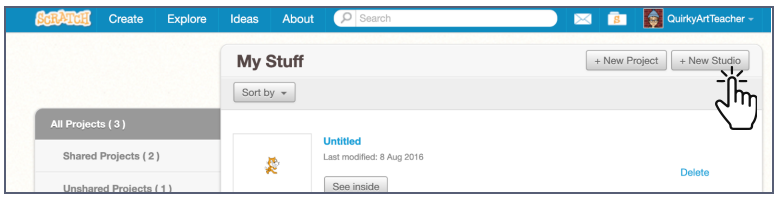


Then, click the "Add Class Studio" button.

Once in the studio, click the "Edit Thumbnail" button to change the default gray cat image in the upper left-hand corner. **Upload your own studio thumbnail image**. The maximum file size for a thumbnail is 512 KB and your image must be less than 500x500 pixels.

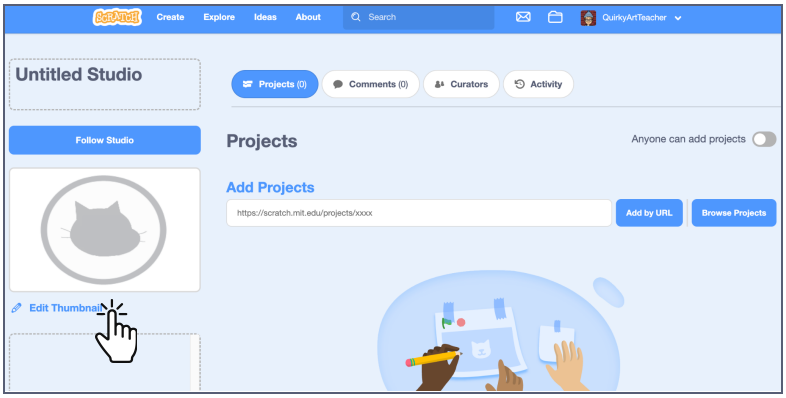
When you click on the "Curators" tab, you should see all the class students have been set as studio curators.

Method 2: Create a studio without automatically adding students as curators



Once logged into your Scratch account, go to "My Stuff."

Choose the "**+ New Studio**" button at the top right.



Click on “Untitled Studio” to **give your studio a name and description**. In the description, you can share the theme of the studio, what kinds of projects you are looking to include… Just be sure your title and description don’t reveal any personal information (like school name or first and last name).

Click the “Edit Thumbnail” button to change the default gray cat image in the upper left-hand corner. **Upload your own studio thumbnail image**. The maximum file size for a thumbnail is 512 KB and your image must be less than 500x500 pixels.

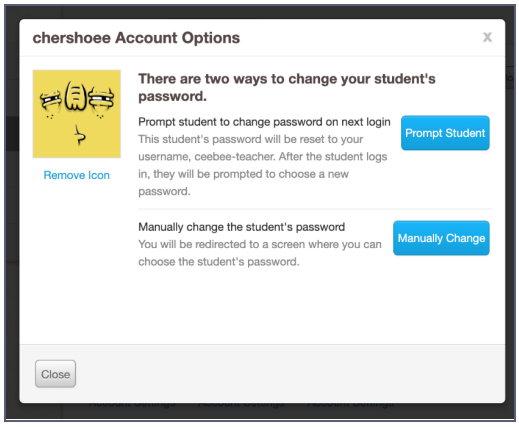
When you click on the “Curators” tab, you should see no curators have been assigned yet.

See our Studio Guide for detailed information on:

- Studio Definitions
- How to Manage a Studio
- How to Add Projects to a Studio

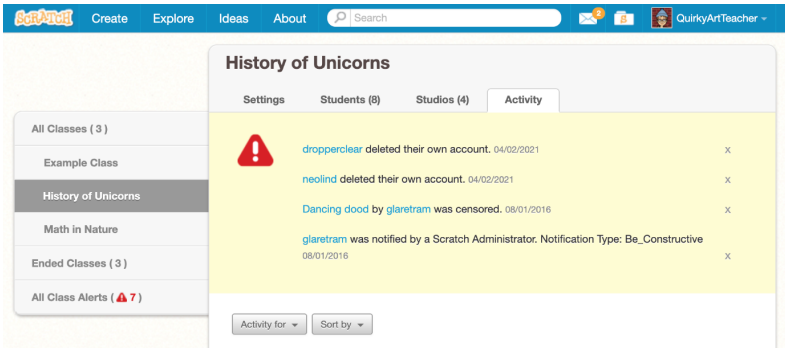
Managing Your Students

Managing a student



You can manually **reset a student password** from within your Scratch Teacher Account. First, navigate to "My Classes" and choose the class and go to the "Students" tab. Then click on the "Account Settings" link below the student's account.

You cannot delete a student's account by using a Teacher Account, but students can delete their own account.



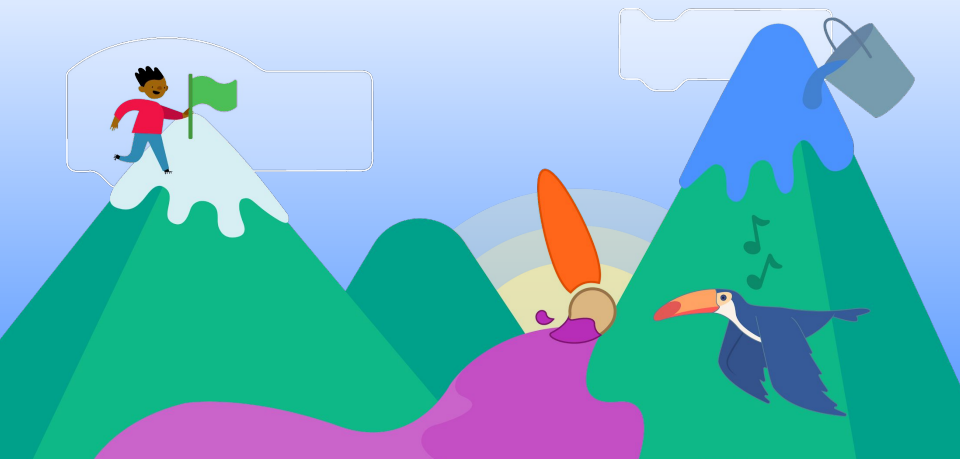
You can see alerts about notifications your students receive on the "Activity" tab of a class or the "All Class Alerts" tab.

Tip: If you'd like to translate this guide, **click here to make a copy** of this Google doc.

Getting Started with SCRATCH

Beginner's Guide

Create your own games, animations, interactive stories, and more.

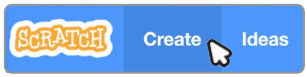


GETTING STARTED

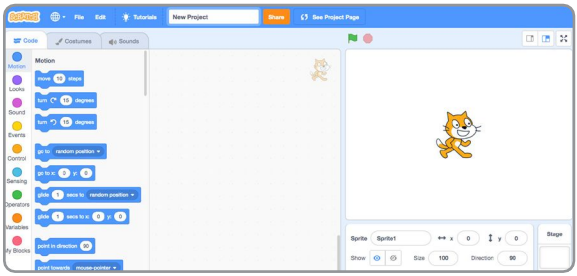
You can use Scratch online at: **scratch.mit.edu**

scratch.mit.edu

Once you've navigated to scratch.mit.edu, click **Create**.



This will bring you to the **Scratch Editor,** where you can start creating projects.



If your computer uses an older operating system, or your internet connection is unreliable, you can download Scratch and use it offline.

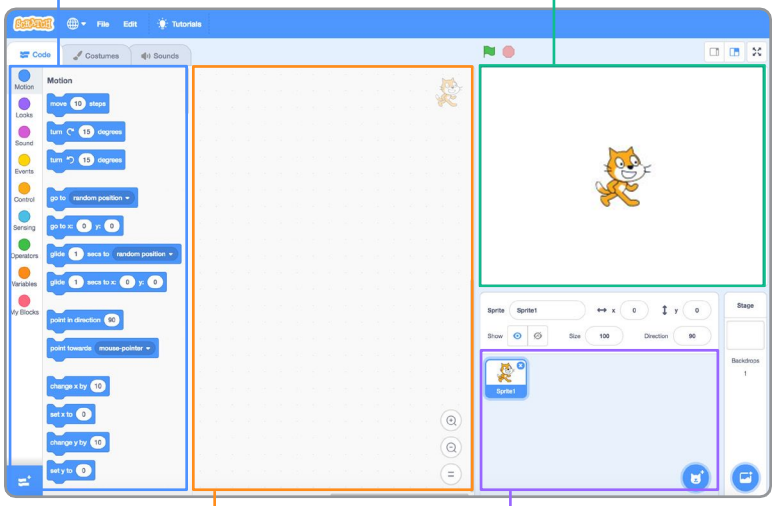
Visit: **https://scratch.mit.edu/download** for information on downloading and installing the Scratch app.

THE SCRATCH EDITOR

The Scratch Editor is where you create projects in Scratch. Here are its main parts:

Blocks Palette
Blocks for coding your projects

The Stage
Where your creations come to life

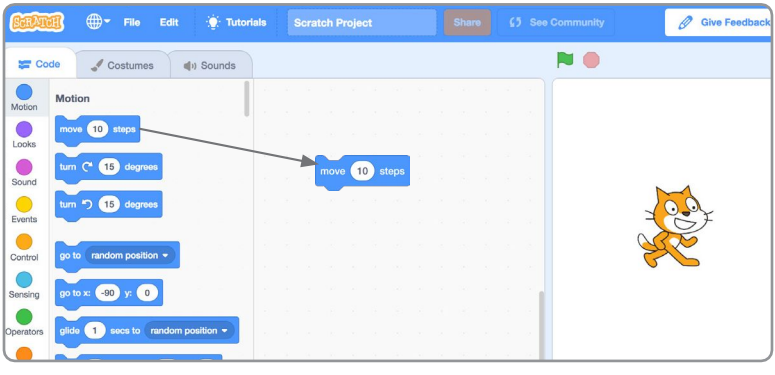


Coding Area
Drag in blocks and snap them together to code your sprites

Sprite List
Click the thumbnail of a sprite to select it

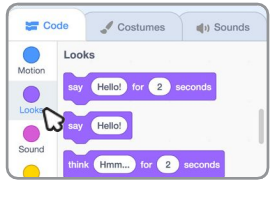
LET'S CODE!

To code projects in Scratch, you snap together blocks.
Start by dragging out a **move** block.

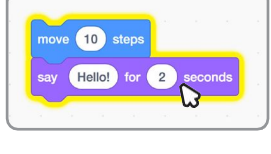


Click the block to try it.
Does your cat move?

Now say something!
Click the **Looks** category.

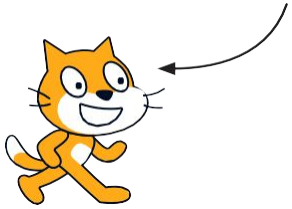


Drag out a **say** block.
Snap it onto the **move** block. Click on your blocks to try them.



WHAT IS A SPRITE?

In Scratch, any character or object is called a sprite. Every new project in Scratch starts with the Cat sprite.



Want to choose a different sprite?

Click the New Sprite icon.

Or, hover over the **New Sprite** icon to see more options.

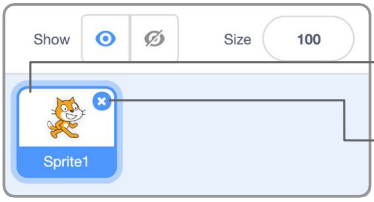
Upload an image from your computer.

Draw your own sprite.

Click for a surprise sprite!

Choose a sprite from the library.

Want to **delete a sprite** from your project?



First, select the sprite by clicking on its thumbnail in the Sprite List.

Then, click here to delete the sprite.

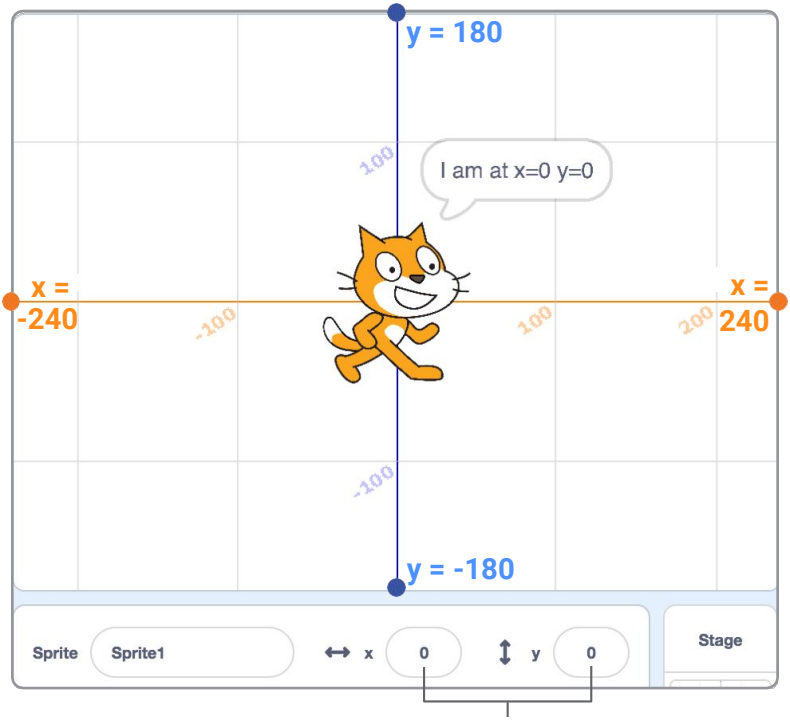
WHERE IS YOUR SPRITE?

Every sprite has an **x** and **y** position on the Stage.

x is the position of the sprite from left-to-right.

y is the position from top-to-bottom.

At the very center of the stage, **x** is 0 and **y** is 0.

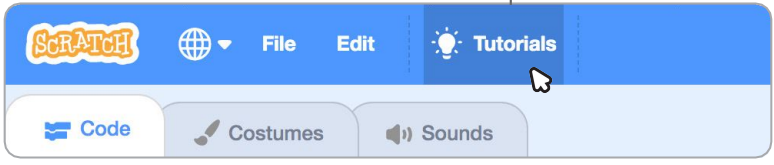


When you move your sprite, you can see its **x** and **y** position change.

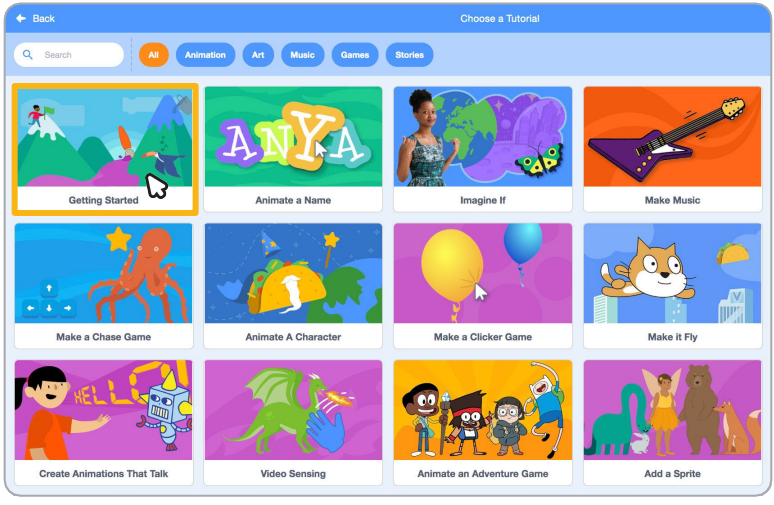
TUTORIALS

There are a range of tutorials available in the Scratch **Tutorials Library**, which guide learners in creating projects with Scratch. Students can get started making their own stories, animations, and games.

You can get to the Tutorials Library from the Scratch Editor by clicking the **Tutorials** button.



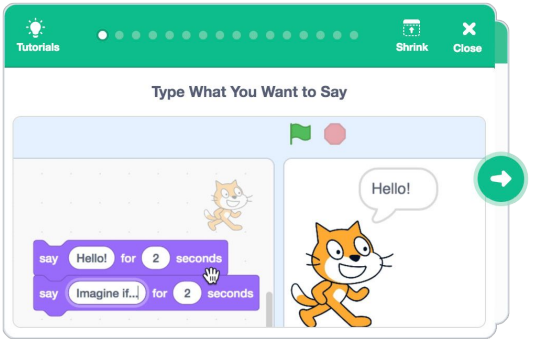
The **Getting Started** tutorial will walk you through the basics.



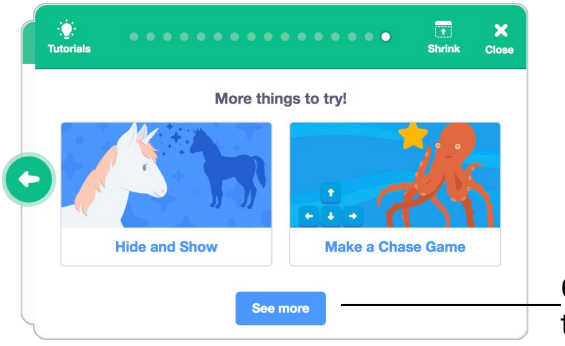
TUTORIALS

Once you've selected the tutorial, it will open in the Scratch Editor.

Click the green arrow to see each step.



When you've reached the end of a tutorial you can select another tutorial, and keep adding to your project.



Click here to see all the Tutorials.

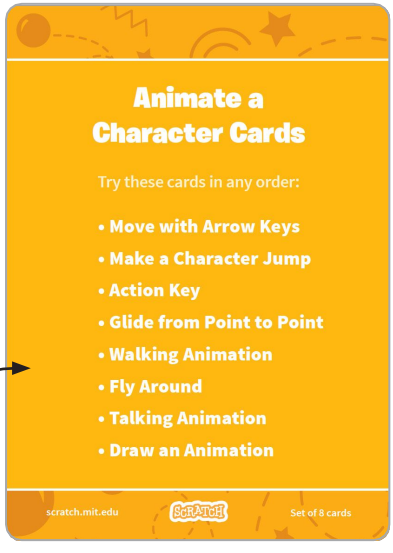
CODING CARDS

The Scratch **Coding Cards** provide another way to learn to create projects with Scratch. Download the cards at **scratch.mit.edu/ideas**.

Each set of cards starts with a title card, which shows you what you can create.

The **Animate a Character** cards are a great set to start with.

Examples of what you can create

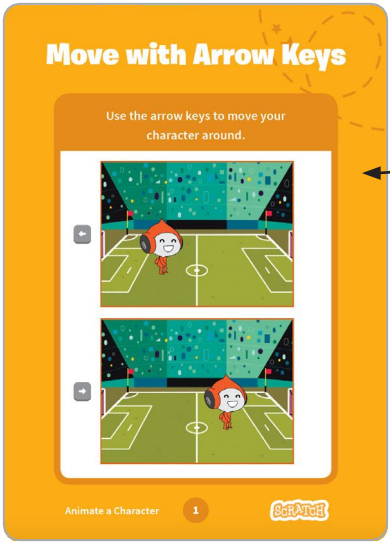


A list of all the cards in this set

USING THE CODING CARDS

After each title card is a series of cards walking you through each step of creating a project.

Add your own sprites, backdrops and more!



The front of each card shows you what you can create.

The back shows you how to do it.

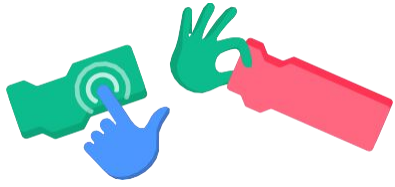
GET CREATIVE!

Encourage students to use their imagination as you create projects. There are many different ways they can make their Scratch projects unique.

You can choose or draw your own characters.

Choose a sound or record your own.

Try changing numbers or adding blocks to your code to see what happens.

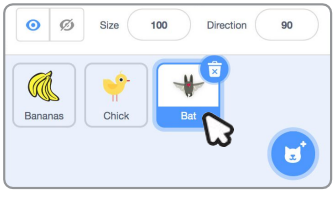


Experiment and customize your project however you want!

GET CREATIVE WITH SPRITES!

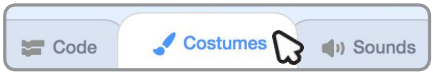
Scratch has its own paint tools, which allow you to customize sprites from the library, or even create sprites of your own.

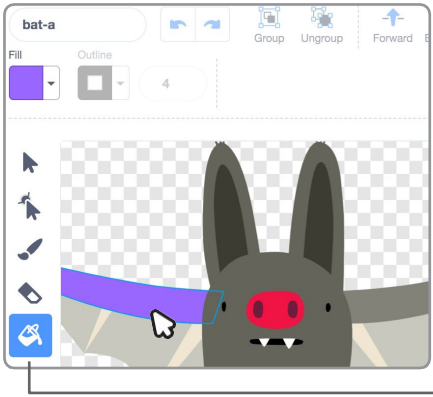
Let's start by editing a sprite from the library.



Select a sprite to edit by clicking on it in the Sprite list.

Click the **Costumes** tab at the top left to see the paint tools.





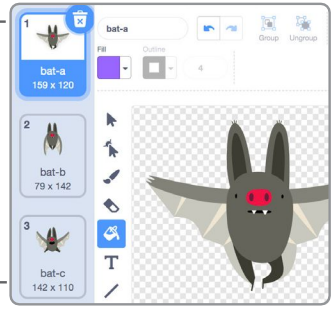
The paint tools allow you to recolor sprites, add to them with a paint brush, and change them in a variety of ways.

You can use the **paint bucket** tool to recolor different parts of a sprite.

GET CREATIVE WITH SPRITES!

Some sprites, like the Bat sprite have multiple costumes, or poses.

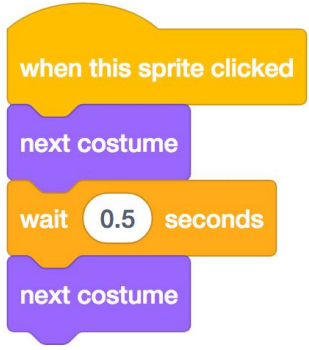
You can see a sprite's costumes by clicking the **Costumes** tab.



If your sprite only has one costume, right click on the costume to duplicate it (On Mac control + click).

Now you can modify the second costume using the paint tools, so your sprite has two different poses or facial expressions.

Click the **Code** tab, then try adding these blocks.

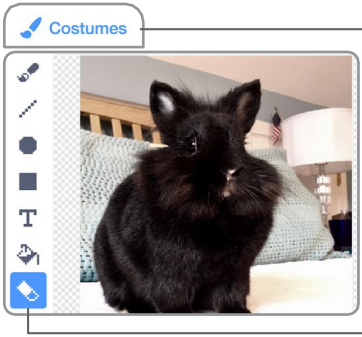


ADD YOUR OWN PHOTOS

There are many ways to create your own sprites and artwork using the Scratch paint tools.

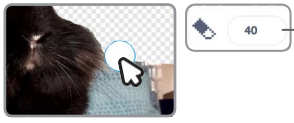
You can create your own sprites by uploading photos or images and erasing the background.

Hover over the New Sprite button, then select **Upload Sprite**.



Next click the **Costumes** tab. You will see bitmap tools for editing your image.

Click the **eraser** icon and use the eraser tool to remove the background from your photo.



Tip: to adjust the size of the eraser, type a larger or smaller number.

There are two modes for drawing in Scratch:

1. **Bitmap Mode** allows you to edit photos and paint with pixels.
2. **Vector Mode** allows you to create and edit shapes.

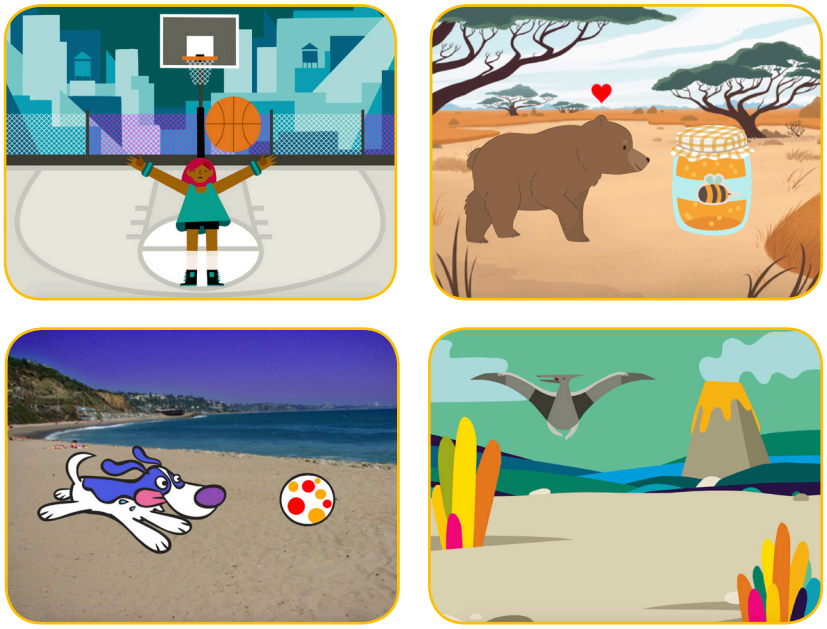
Tip: If you'd like to remix and customize this guide, **click here to make your own copy** of the Google Slides template.

COLOR SCHEME

EDUCATOR GUIDE

Animate a Character

With this guide, you can plan and lead a 55-minute lesson using Scratch. Students will gain experience with coding as they bring characters to life with animation.



ANIMATE A CHARACTER / EDUCATOR GUIDE

Lesson Outline

Objective: Students will become familiar with the Scratch environment by animating a character.

IMAGINE
10 minutes

First, gather as a group to introduce the theme and spark ideas.

CREATE
40 minutes

Next, help students as they animate characters, working at their own pace through the tutorial.

SHARE
5 minutes

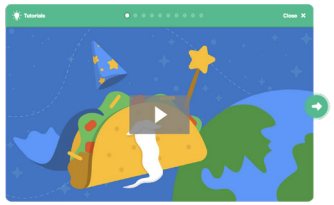
At the end of the session, gather together to share and reflect.

Get Ready for the Lesson

Use this checklist to prepare for the lesson.

☐ Preview the Tutorial

The *Animate a Character* tutorial shows students how to create their own projects. Preview the tutorial before your lesson and try the first few steps: scratch.mit.edu/tutorials



☐ Print the Activity Cards (optional)

Print a few sets of *Animate a Character* cards to have available for students during the lesson.
scratch.mit.edu/ideas

☐ Make sure students sign into their Scratch accounts

Have students sign into their own Scratch accounts at scratch.mit.edu.

☐ Set up computers or laptops

Arrange computers so that students can work individually or in pairs.

☐ Set up a computer with projector or large monitor

You can use a projector to show examples and demonstrate how to get started.

ANIMATE A CHARACTER / EDUCATOR GUIDE

Imagine

IMAGINE

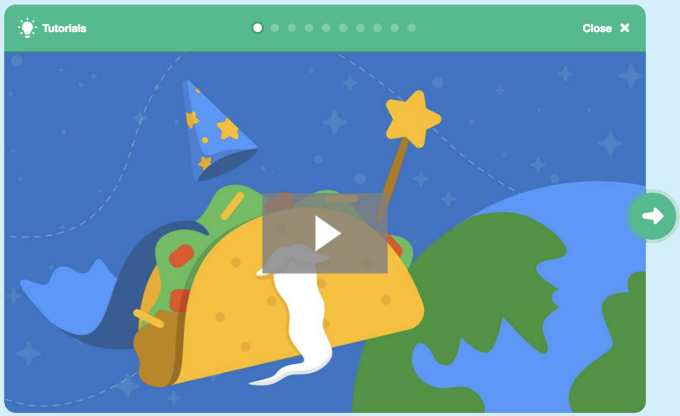
Begin by gathering the students to introduce the theme and spark ideas for projects.

Warm-up Activity: Favorite Characters

Gather the group in a circle. Ask each student to say their name, then share a favorite character from a book, movie, or TV show, and one or two of their favorite things about that character.

Provide Ideas and Inspiration

To spark ideas, watch the Animate a Character video at the start of the tutorial. The video shows a variety of projects to spark ideas and inspiration.



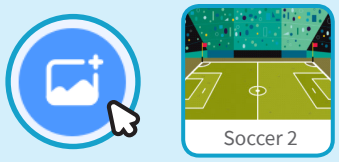
View the scratch.mit.edu/ideas

Demonstrate the First Steps

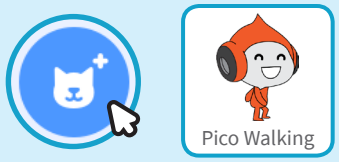
IMAGINE

Demonstrate the first few steps of the tutorial so students can see how to get started.

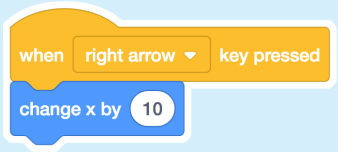
Choose a backdrop.



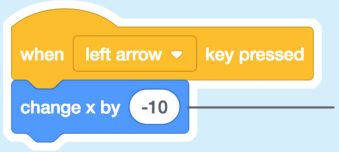
Choose a character to animate.



Make your sprite move right and left with arrow keys:



Choose right arrow from the menu.

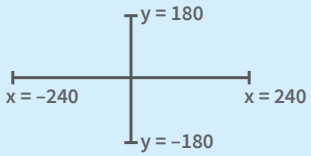


Choose left arrow from the menu.

Type a minus sign to move left.

Press the left arrow and right arrow keys on your keyboard to move.

Helpful Hint: Understanding x y coordinates will help students figure out how to move sprites around the stage.



y is the position on the Stage from top to bottom.

x is the position on the Stage from right to left.

Create

CREATE

Support students as they create animated Scratch projects.

Start with Prompts

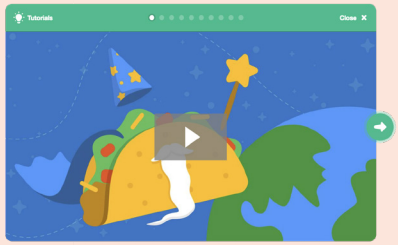
Ask students questions to get started

Which character would you like to animate?

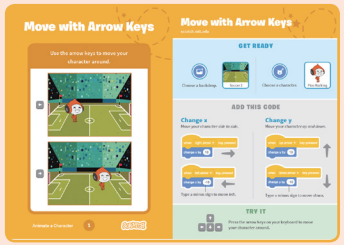
What do you want your character to do?

Provide Resources

Offer options for getting started



Some students may want to follow the online tutorial: scratch.mit.edu/animate



Others may want to explore using the activity cards: scratch.mit.edu/ideas

Suggest Ideas for Starting

- Choose a character to animate.
- Animate your character: make it jump, fly, glide or talk!
- Choose a backdrop.

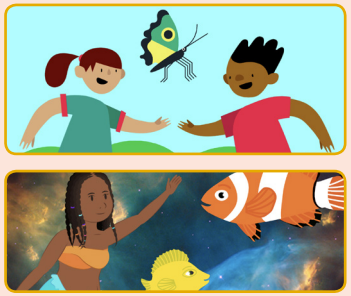
CREATE

More Things to Try

- Try combining more than one kind of animation.
- If you're not sure what to do, pick a card and try something new.
- Add a second character or object to animate.

Support collaboration

- When someone gets stuck, connect them to another participant who can help.
- See a cool idea? Ask the creator to share with others.



Encourage experimenting

The Animate a Character activity can be done in any order, with a range of different character and object sprites.

Encourage students to try new things:

What will your character do next?

How can you make your animation interactive?

Share

SHARE

Have students share their project with their neighbors.

Ask questions they can discuss:

What do you like best about the project you made?

What was the hardest part?

If you had more time, what would you add or change?

What's Next?

Students can use the ideas and concepts from this lesson to create a wide variety of projects. Encourage them to continue developing their projects into games, stories or interactive art with the resource listed below.

Video Sensing

Interact with characters and objects in Scratch with video sensing.

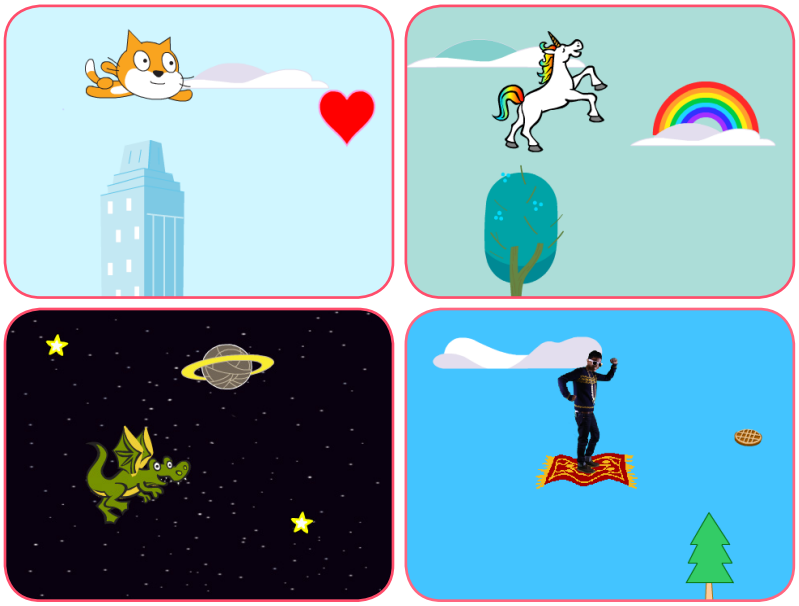
Find this project and more in the Tutorials library: **scratch.mit.edu/ideas**

Scratch is a project of the Lifelong Kindergarten Group at the MIT Media Lab.

EDUCATOR GUIDE

Make It Fly

With this guide, you can plan and lead a 55-minute lesson using Scratch. Students will choose a character and program it to fly.



MAKE IT FLY / EDUCATOR GUIDE

Lesson Outline

Objective: Students will create an animation with the illusion of a flying character.

IMAGINE
10 minutes

First, gather as a group to introduce the theme and spark ideas.

CREATE
40 minutes

Next, help students as they create a flying animation, working at their own pace through the tutorial.

SHARE
5 minutes

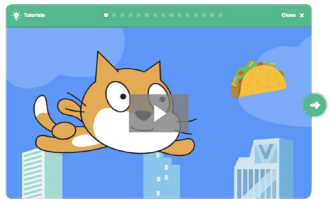
At the end of the session, gather together to share and reflect.

Get Ready for the Lesson

Use this checklist to prepare for the lesson.

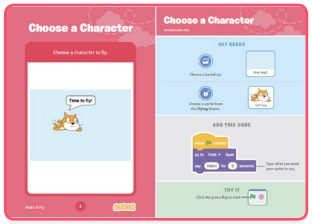
☐ **Preview the Tutorial**

The *Make It Fly* tutorial shows students how to create their own projects. Preview the tutorial before your lesson and try the first few steps: scratch.mit.edu/fly



☐ **Print the Activity Cards (optional)**

Print a few sets of *Make It Fly* cards to have available for students during the lesson.
scratch.mit.edu/fly/cards



☐ **Make sure students sign into their Scratch accounts**

Have students sign into their own Scratch accounts at scratch.mit.edu.

☐ **Set up computers or laptops**

Arrange computers so that students can work individually or in pairs.

☐ **Set up a computer with projector or large monitor**

You can use a projector to show examples and demonstrate how to get started.

Imagine

IMAGINE

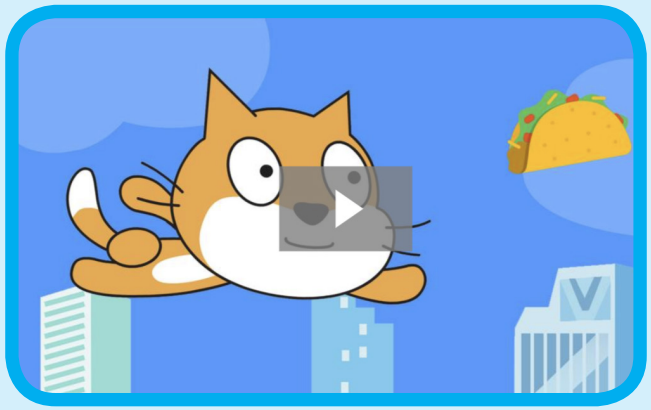
Begin by gathering the students to introduce the theme and spark ideas for projects.

Warm-up Activity: If I Could Fly...

Gather the group in a circle and ask, "If you could fly, where would you want to go?" Suggest that they close their eyes and imagine flying through their favorite place. Ask, "Where are you? What kinds of things do you see below you?" If there's time, have each person say where they imagined flying or something they saw on their flight.

Provide Ideas and Inspiration

Show the introductory video for the *Make It Fly* tutorial. The video shows a variety of projects for ideas and inspiration.



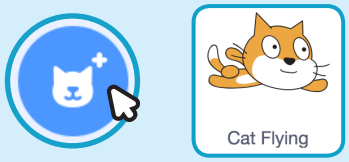
View at scratch.mit.edu/fly or vimeo.com/llk/fly

Demonstrate the First Steps

IMAGINE

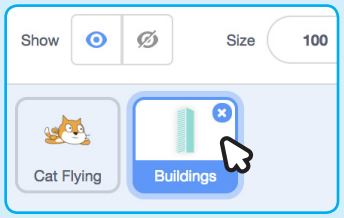
Demonstrate the first few steps of the tutorial so students can see how to get started.

In Scratch, click Create.
Choose a flying sprite from the library:



Choose a new sprite for your character to fly past:

Make the building move across the stage to make your character look like it's flying:



Create

CREATE

Support students as they make a flying animation.

Start with Prompts

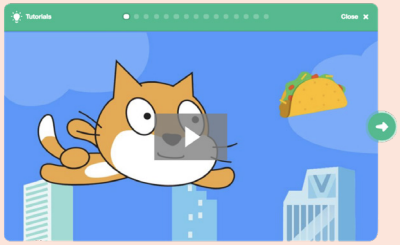
Ask students questions to get started

What character would you like to make fly?

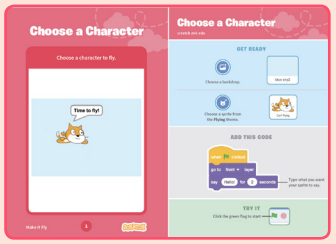
Where will your character go flying?

Provide Resources

Offer options for getting started



Some students may want to follow the online tutorial: scratch.mit.edu/fly



Others may want to explore using the activity cards: scratch.mit.edu/fly/cards

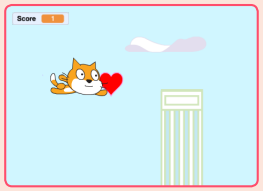
Suggest Ideas for Starting

- Choose a character
- Make the character say something
- Choose buildings or other scenery
- Make the scenery move

CREATE

More Things to Try

- Switch costumes to change the scenery.
- Make your character move when you press a key.
- Add clouds and other floating objects.
- Score points when touching an object.



Encourage Debugging

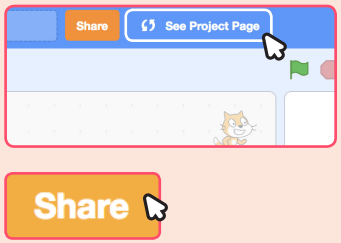
Here are some strategies to suggest to help students fix any bugs or difficulties they encounter:

- When stuck, talk out what you're working on with someone.
- Try out small bits of code at a time to figure out what's happening at each step.
- Look closely at the blocks on the tutorial or activity cards to see if they are the same or different from the blocks you're using.
- Remember that bugs always arise when creating a computer program. Debugging is a helpful skill to know not just in coding, but throughout life.

Prepare to Share

To add instructions and credits to a project, click the button: "See project page".

Give your project a title, add instructions and credits, then click Share.



Share

SHARE

Share projects with others in the room. Organize a flying character showcase. Ask half the room show their projects, while the others view them. Then switch.

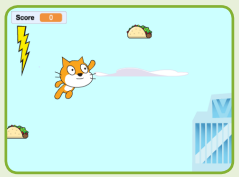
Suggest that they ask each other questions, such as:

What do you like best about the project you made?

What might you like to change or make next?

What's Next?

Students can use the ideas and concepts from this lesson to create other projects. Here are a couple of variations on the flying character project you could suggest.



Flying Game

Make a game where you avoid some objects and try to catch others. Add or subtract points based on what your character touches.

Flying Stories

Tell a story about your flying characters. You can record your voice and play sound clips. Or, use say blocks to make voice bubbles.

EDUCATOR GUIDE

Make a Chase Game

With this guide, you can plan and lead a 55-minute lesson using Scratch. Students will make a game that includes a variable to keep score.

Lesson Outline

Objective: Students will create a game using sensing.

IMAGINE
10 minutes

First, gather as a group to introduce the theme and spark ideas.

CREATE
40 minutes

Next, help students as they make chase games, working at their own pace through the tutorial.

SHARE
5 minutes

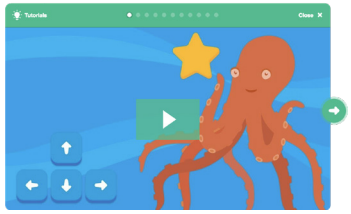
At the end of the session, gather together to share and reflect.

Get Ready for the Lesson

Use this checklist to prepare for the lesson.

☐ Preview the Tutorial

The *Make a Chase Game* tutorial shows students how to create their own projects. Preview the tutorial before your lesson and try the first few steps,



☐ Print the Activity Cards (optional)

Print a few sets of *Chase Game* cards to have available for students during the lesson. You can download the cards at: scratch.mit.edu/ideas

☐ Make sure students sign into their Scratch accounts

Have students sign into their own Scratch accounts at scratch.mit.edu.

☐ Set up computers or laptops

Arrange computers so that students can work individually or in pairs.

☐ Set up a computer with projector or large monitor

You can use a projector to show examples and demonstrate how to get started.

Imagine

IMAGINE

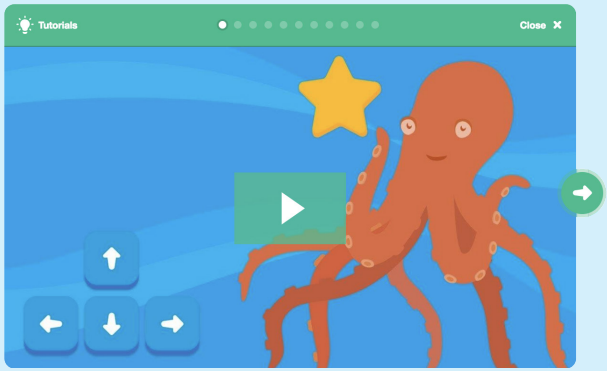
Begin by gathering the students to introduce the theme and spark ideas for projects.

Warm-up Activity: Imaginary Chase

Gather the students in a circle. Start by giving an example of one thing chasing another, such as "The dog is chasing the dinosaur." The next person adds on, such as, "The dinosaur is chasing a donut." The following person adds on by saying, "The donut is chasing a duck." or whatever creature or object they choose. Continue until each person has added on to this imaginary game of chase.

Provide Ideas and Inspiration

To spark ideas, watch the Make a Chase Game video at the start of the tutorial.



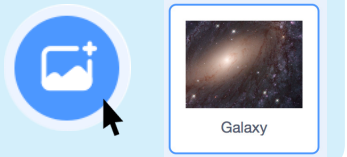
View the video at scratch.mit.edu/chase

Demonstrate the First Steps

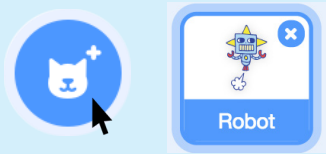
IMAGINE

Demonstrate the first few steps of the tutorial so students can see how to get started.

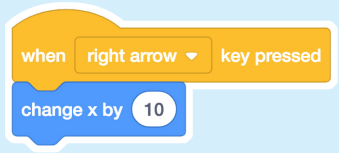
Choose a backdrop.



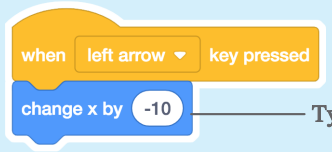
Choose a Sprite, like Robot.



Make your sprite move right and left with arrow keys.



Choose right arrow from the menu.

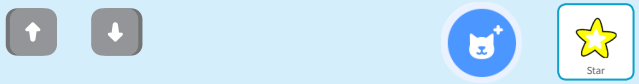


Choose left arrow from the menu.

Type a minus sign to move left.

Press the left arrow and right arrow keys on your keyboard to move.

Discuss next steps they can try, such as coding the sprite to move up and down and adding a sprite to chase.



Create

CREATE

Support students as they create catch games. Suggest working in pairs.

Start with Prompts

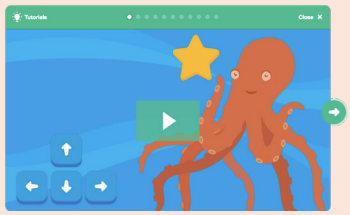
Ask students questions to get started

Which backdrop would you like to choose for your game?

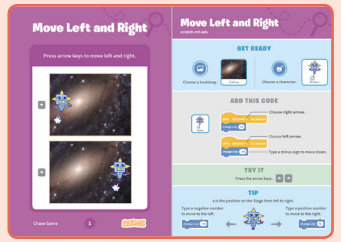
Who do you want as the main character in your game? What will it chase?

Provide Resources

Offer options for getting started



Some students may want to follow the online tutorial: scratch.mit.edu/chase



Others may want to explore using the printed cards: scratch.mit.edu/ideas

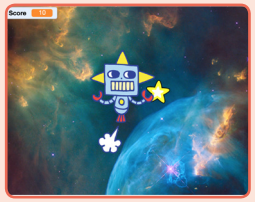
Suggest Ideas for Starting

- Choose a backdrop
- Choose or draw a main character
- Make it move with arrow keys.
- Select an object to chase.

CREATE

More Things to Try

- Code the star or other sprite to chase
- Add a variable to keep score
- Add sounds
- Add a level
- Show a message when reaching the new level



Encourage Tinkering

- Encourage students to feel comfortable trying combinations of blocks and seeing what happens.
- Suggest students look inside other chase games to see the code.
- If they find code they like, they can drag the scripts or sprites into the backpack to reuse in their own project.

Prepare to Share

To add instructions and credits to a project, click the button: *"See project page"*.

Share

SHARE

Have students share their projects with their neighbors.

Ask questions that encourage reflection:

What do you like best about your game?

If you had more time, what would you add or change?

What's Next?

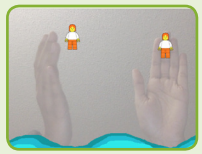
Chase Game projects provide an introduction to creating interactive games in Scratch. Here are a few ways that learners can build on the concepts they learned from this project.

Add Obstacles

For a more complex game, add obstacles to avoid. Subtract points when you hit the obstacles.

Make a Two-Player Game

For an extra challenge, make a version of the game that allows two players to play.



Video Sensing

If the computers have a web camera attached or built-in, learners can make a game that they interact by moving their bodies. See the Video Sensing tutorial and educator guide for support.

Created by the Scratch Team

EDUCATOR GUIDE

Pong Game

With this guide, you can plan and lead a 55-minute lesson using Scratch. Students will gain experience with coding as they design a bouncing ball game.

PONG GAME / EDUCATOR GUIDE

Lesson Outline

Objective: Students will develop an interactive game using variables to keep score.

IMAGINE
10 minutes

First, gather as a group to introduce the theme and spark ideas.

CREATE
40 minutes

Next, help students as they make games, working at their own pace through the tutorial.

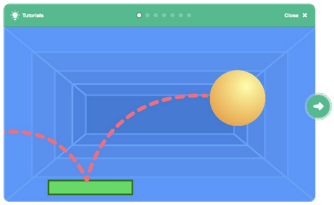
SHARE
5 minutes

At the end of the session, gather together to share and reflect.

Get Ready for the Lesson

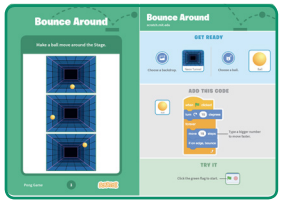
Use this checklist to prepare for the lesson.

☐ Preview the Tutorial



The *Pong Game* tutorial shows students how to create their own projects. Preview the tutorial before your lesson and try the first few steps: scratch.mit.edu/pong

☐ Print the Activity Cards (optional)



Print a few sets of *Pong Game* cards to have available for students during the lesson.
scratch.mit.edu/ideas

☐ Make sure students sign into their Scratch accounts

Have students sign into their own Scratch accounts at scratch.mit.edu.

☐ Set up computers or laptops

Arrange computers so that students can work individually or in pairs.

☐ Set up a computer with projector or large monitor

You can use a projector to show examples and demonstrate how to get started.

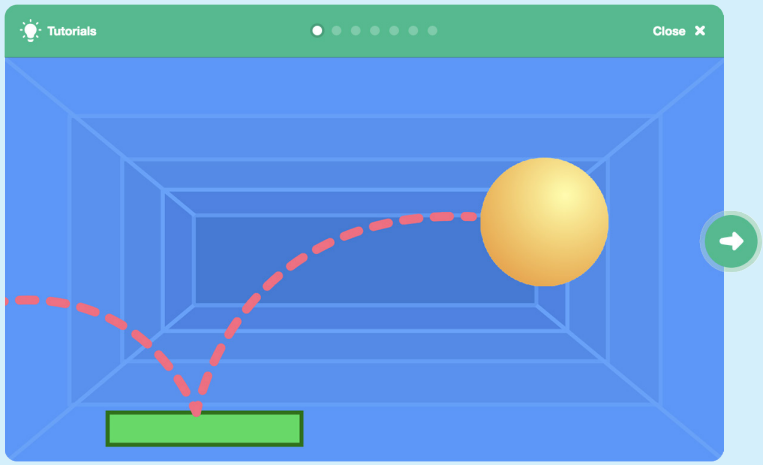
Imagine

IMAGINE

Begin by gathering the students to introduce the theme and spark ideas for projects.

Provide Ideas and Inspiration

Show the introductory video for the *Pong Game* tutorial. The video shows pong games with a variety of themes, including everything from soccer to a magic potion-themed Pong game.



View at scratch.mit.edu/pong

Warm-up Activity: Bouncing Ideas

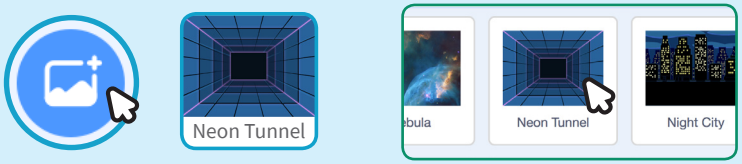
To get students thinking about a theme for their game, take turns calling out a theme, such as pizza pong or flower pong and brainstorming ideas for the type of images they could use to represent the theme.

Demonstrate the First Steps

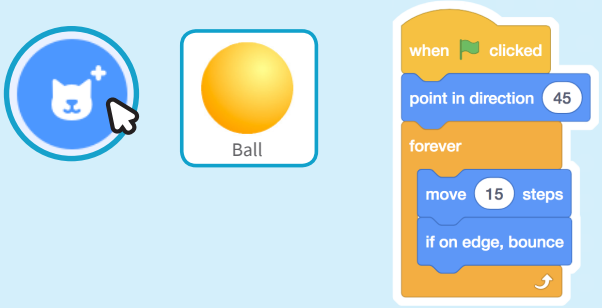
IMAGINE

Demonstrate the first few steps of the tutorial so students can see how to get started.

Go to the Scratch website. Click Create. Choose a new backdrop:



Choose a ball sprite and make it bounce around:



Add a paddle sprite and control it with the mouse:

Create

CREATE

Support students as they create pong games, on their own or in pairs.

Start with Prompts

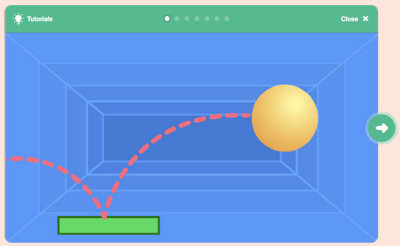
Ask students questions to get started

What background do you want for your game?

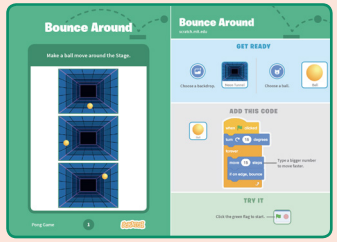
What color or type of ball?

Provide Resources

Offer options for getting started



Some particpants may want to follow the online tutorial: scratch.mit.edu/pong



Others may want to use the printed activity cards: scratch.mit.edu/ideas

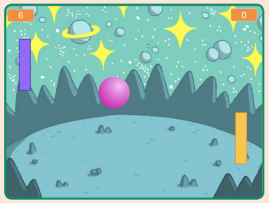
Suggest Ideas for Starting

- Choose a backdrop
- Choose or draw a ball sprite and make it bounce around
- Add a paddle sprite that you can control
- Make the ball bounce off the paddle

CREATE

More Things to Try

- Add sounds and color effects
- Keep score by adding a variable
- Add a way to win or lose the game
- Change the backdrop when you reach a certain number of points
- Duplicate the ball for an added challenge



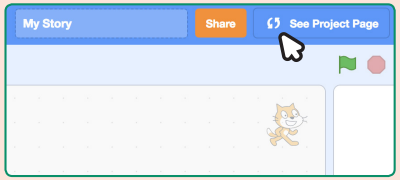
Offer strategies for problem solving

- Talk out what you're working on with someone
- Try out small bits of code at a time to figure out what's happening at each step
- Look closely at the blocks on the tutorial or activity cards to see if they are the same or different from the blocks you're using
- Look at the code for other pong games on the Scratch site

Prepare to Share

To add instructions and credits to a project, click the button: "See project page".

Then click the Share button if you want the project visible to others online.



Share

SHARE

Have participants share their projects with others in the room.

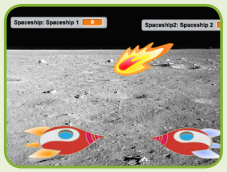
Ask questions to encourage reflection:

What did you notice about the games you tried?

What ideas might you add to your game?

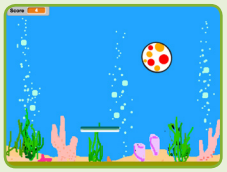
What's Next?

Here are a couple of other directions you could suggest:



Two-Player Game

For a more advanced project, try making a two-player game. To make a new version of your own project, click **File > Save as a Copy.**



Remix a Game

A different way to make a pong game is to remix someone else's project, adding images and ideas. Find a project to remix in the **Pong Game Studio**: scratch.mit.edu/studios/644508/ Click '**See inside**', then click the '**Remix**' button.

Scratch is a project of the Lifelong Kindergarten Group at the MIT Media Lab.

EDUCATOR GUIDE

Create a Story

With this guide, you can plan and lead a 55-minute lesson using Scratch. Students will create a story with settings, characters, and dialogue.

CREATE A STORY / EDUCATOR GUIDE

Lesson Outline

Objective: Students will create an animated story between at least two characters.

IMAGINE
10 minutes

First, gather as a group to introduce the theme and spark ideas.

CREATE
40 minutes

Next, help students as they create story projects, working at their own pace through the tutorial.

SHARE
5 minutes

At the end of the session, gather together to share and reflect.

Get Ready for the Lesson

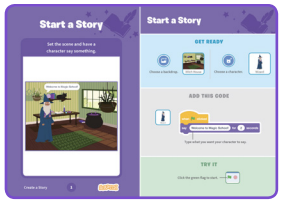
Use this checklist to prepare for the lesson.

☐ Preview the Tutorial

The *Create a Story* tutorial shows students how to create their own projects. Preview the tutorial before your lesson and try the first few steps: scratch.mit.edu/story

☐ Print the Coding Cards (optional)

Print a few sets of *Create a Story* cards to have available for students during the lesson. You can download from this page: scratch.mit.edu/ideas



☐ Make sure students sign into their Scratch accounts

Have students sign into their own Scratch accounts at scratch.mit.edu.

☐ Set up a studio for project sharing on Scratch

Set up a studio so students will be able to add their projects. Go to your ***My Stuff*** page, then click the ***+New Studio*** button. Type in a name for the studio.

☐ Set up computers or laptops

Arrange computers so that students can work individually or in pairs.

CREATE A STORY / EDUCATOR GUIDE

Imagine

IMAGINE

Begin by gathering the students to introduce the theme and spark ideas for projects.

Warm-up Activity: Story Starters in a Bag

Have students make up a brief story by giving them a bag with three objects in it, and asking them to include all of the items in the story. In each bag, you could include small objects, pictures of animals or characters, and/or words (people, places, or things). Divide students into groups of two or three, and have each pick a bag. Give them a few minutes to come up with a quick story.

Provide Ideas and Inspiration

You can show the Create a Story tutorial video to show students how they can start making stories in Scratch.

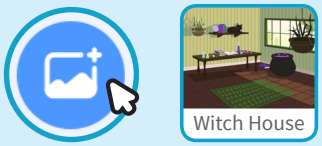
View the video at: scratch.mit.edu/story

Demonstrate the First Steps

IMAGINE

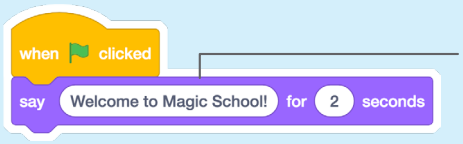
Demonstrate the first few steps of the tutorial so students can see how to get started.

In Scratch, click Create. Choose a backdrop.



Choose any character (in Scratch called a *sprite*).

Code your character to say something.



Type what you want your character to say.

Click the green flag to start.

Add another character.

Add code to the new character.

Use this block to have the second character wait before they say something.

Create

CREATE

Support students as they create Story projects, on their own or in pairs.

Start with Prompts

Ask students questions to get started

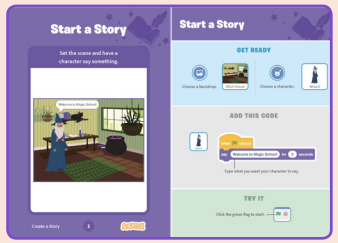
Where will your story take place?

What will happen first?

Provide Resources

Offer options for getting started

Some students may want to follow the online tutorial: scratch.mit.edu/story



Others may want to explore using the coding cards: scratch.mit.edu/ideas

Suggest Ideas for Starting

- Choose a backdrop.
- Choose a character.
- Make a character say something
- Make a character hide and show.

CREATE

More Things to Try

- Switch backdrops.
- Make your characters have a conversation.
- Move your characters.
- Change something when you click on it.

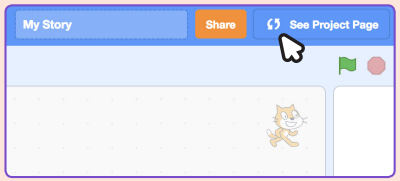
Support Tinkering

Scratch is designed to support creating by experimenting and tinkering. So, your students may want to start their stories without planning beforehand. As they create, one idea can spark another. Celebrate their sparks of creativity and the unexpected turns their stories may take.

Prepare to Share

To add instructions and credits to a project, click the button: "See project page".

Then click the Share button if you want the project visible to others online.



Share

SHARE

Help the students add their projects to a shared studio in Scratch. Give them a link to the studio. Then they can click 'Add Projects' at the bottom of the page.

Ask for volunteers to show their project to the group.

What's Next?

Students can use these ideas and concepts to create a variety of projects. Here are some variations on the story project you could suggest:

Retell a story

Start with a story you know and make it in Scratch. Imagine a new ending or a different setting.

Neighbourhood story

Take photos of your classroom, school, or neighborhood and use them as backdrops in your story.

Round-robin story

Give everyone 5 minutes to start a story. Then, have them switch to the next computer to add to the story. Repeat.

Created by the Scratch Team