



# Kindergarten CS Lessons

Since the standards drive the pedagogical component of teaching and serve as a guide to how students demonstrate understanding of the content, they must be incorporated as an integral part of the overall student expectations when assessing content understanding. By the end of Kindergarten, the computer science literate student will begin the practice of utilizing devices to perform basic computer operations, such as turning a computer on and off, and communicate basic hardware and software problems. Kindergarten students will be able to explain the importance of password protection and discuss how computer networks can connect people globally. With teacher assistance, kindergarten students will begin to develop an understanding of how to model and develop algorithms and programs, and collect data to make predictions. Students will be introduced to the impacts of computing, including how people lived and worked before and after the implementation of new technology, how to work responsibly online, and the importance of keeping login information private.

[Completing the Course in Code.org](https://www.code.org) will fulfill all of the CS standards!

## COMPUTING SYSTEMS

|   |   |
|---|---|
| <b>K.CS.D.1</b><br>Computing Systems<br><br>Devices | <b>With teacher guidance, select and operate an appropriate device to perform a task.</b><br>People use computing devices to perform a variety of tasks accurately and quickly. With teacher guidance, students should be able to select the appropriate device to use for tasks they are required to complete. For example, if students are asked to draw a picture, they should be able to open and use a drawing app/program to complete this task, or if they are asked to create a presentation, they should be able to open and use presentation software.<br><i>Practice(s): Fostering an inclusive Computing Culture: 1.2</i>   |
| You're already doing it!!                           | When guiding students to perform a task, indicate the program you want them to use then indicate why. Additionally, you can ask them which program would best complete the task at hand. Here are some examples: <ul style="list-style-type: none"> <li>● We're going to open a Google Doc because we use Google Docs to create documents.</li> <li>● We're going to open KidPix because it is a program used for drawing</li> <li>● Click on Google Chrome. Chrome is the browser we use to access the internet</li> <li>● We're going to make a presentation. Do you think we should use Google Slides or Google Docs (both are correct so ask them why they chose it)</li> </ul> |
| Extensions  | Have students decide which program would work best to complete a task   |

|   |  |
|---|--|
| <b>K.CS.HS.1</b><br>Computer Systems<br><br>Hardware & Software | <b>K.CS.HS.1 Use appropriate terminology in identifying and describing the function of common physical components of computing systems.</b><br>A computing system is composed of hardware and software. Hardware consists of physical components. Students should be able to identify and describe the function of external hardware, such as desktop computers, laptop computers, tablet devices, monitors, keyboards, mice, and printers. <i>Practice(s): Communicating about Computing: 7.2</i> |
| You're already  | The best way to guide students to mastering this standard is to use proper terminology   |



# Kindergarten CS Lessons

|            |   |
|------------|---|
| doing it!! | while talking to them about their computers. Include computer terms when they are learning letters (M - monitor, K - keyboard)  |
| Extensions | <a href="#">Computer parts lesson plan</a> (from Bloom Into EdTech blog)<br><a href="#">Computer parts flash cards</a><br><a href="#">Computer parts memory game</a> (this picture dictionary could be cut out to make a memory game!)<br><a href="#">Video about computer parts</a> (3:11)<br><a href="#">Parts of a computer slideshow and game</a> |

|  |   |
|--|---|
| <b>K.CS.T.1</b><br>Computer Systems<br>Troubleshooting | <b>Discuss basic hardware and software problems.</b><br>Problems with computing systems have different causes. Students should be able to communicate a hardware or software problem (e.g., when an app or program is not working as expected, a device will not turn on, the sound does not work, etc.). <i>Practice(s): Testing and Refining Computational Artifacts, Communicating About Computing: 6.2, 7.3</i>   |
| You're already doing it!!                              | When a student presents an issue "My ChromeBook isn't working", have them articulate the problem. Have them use descriptive words and appropriate terminology, then discuss possible solutions. For example:<br>Student: My screen is black<br>Teacher: Your monitor is black? Do you think the battery died? Should we plug it in?<br><br>Student: My game's not working!!<br>Teacher: You're not at the correct website. Maybe we should check to make sure you entered the URL correctly |
| Extensions   | As part of your rotating classroom jobs, include an IT Specialist(Information Technology) assignment. Have this student responsible for making sure the ChromeBooks are plugged in at the end of class. Allow them to be the "expert" that students go to first.<br><br>Identify experts within the classroom (in groups, at tables, etc.) that students should ask before coming to the teacher for help.  |

## NETWORKS & THE INTERNET

|   |   |
|---|---|
| <b>K.NI.C.1</b><br>Networks & Internet<br>Cybersecurity | <b>Explain that a password helps protect the privacy of information.</b><br>Connecting devices to a network or the Internet provides great benefit, care must be taken to use authentication measures, such as strong passwords, to protect devices and information from unauthorized access. This is an essential first step in learning about cybersecurity. Usernames and passwords, such as those on computing devices or Wi-Fi networks, provide a way of authenticating a user's identity. For example, students should enter a password independently and commit to keeping their password private. <i>Practice(s): Communicating About Computing: 7.2</i> |
|---|---|



# Kindergarten CS Lessons

|                           |   |
|---------------------------|---|
| You're already doing it!! | <p>Discuss with students why we don't share passwords. Login information should be kept private to protect the documents on your computer. Additionally, it protects you from having someone do something they shouldn't be doing under YOUR name!!</p> <p>Don't allow students to use another student's QR code</p>  |
| Extensions                | <p>As the year progresses, begin having students login by typing their username and password rather than using the QR Codes. Print their usernames and passwords on the back of their QR Codes so they can practice! Discuss that their passwords include letters and numbers to make it more difficult to steal!</p> <p><a href="#">Password and Internet Security video (7:50)</a> up until 1:45 is about passwords.</p> <p><a href="#">Color your password!</a></p> <p><a href="#">Password rap video (2:00)</a></p> |

|  |  |
|--|--|
| <p><b>K.NI.NCO.1</b><br/>Networks &amp; Internet</p> <p>Network Communication &amp; Organization</p> | <p><b>With teacher guidance, students define computer networks and how they can be used to connect people to other people, places, information, and ideas.</b></p> <p>Small, wireless devices, such as cell phones, communicate with one another through a series of intermediary connection points, such as cellular towers. This coordination among many computing devices allows a person to voice call a friend or video chat with a family member. For example, kindergarten students understand that they are part of non-computing networks such as family, class, school, etc. Kindergarten students should be able to explain that devices are connected, though details about connection points are not expected at this level. <i>Practice(s): Communicating About Computing: 7.3</i></p> |
| You're already doing it!!  | When connecting wirelessly to the projector, using the chromebooks with WiFi, etc. explain what you are doing! "I'm going to connect wirelessly to the projector!"   |
| Extensions   | <a href="#">What is the Internet (2:23)</a>  |

## DATA & ANALYSIS

|   |  |
|---|--|
| <p><b>K.DA.CVT.1</b><br/>Data &amp; Analysis</p> <p>Collection Visualization &amp; Transformation</p> | <p><b>With teacher guidance, collect and transform data using digital devices; Display data for communication in various visual formats.</b></p> <p>The collection and use of data about the world around them is a routine part of life and influences how people live. Many everyday objects, such as cell phones, digital toys, and cars, can contain tools (such as sensors) and computers to collect and display data from their surroundings. Students could collect data on the weather, such as sunny days versus rainy days, the temperature at the beginning of the school day and end of the school day, or the inches of rain over the course of a storm. Students could count the number of pieces of each color of candy in a bag of candy, such as Skittles or M&amp;Ms. Students could create surveys of things that interest them, such as favorite foods, pets, or TV shows, and collect answers to their surveys from their peers and others. The data collected could then be organized into two or more visualizations, such as a bar graph, pie chart, or pictograph. <i>Practice(s): Communicating About Computing, Developing and Using Abstractions: 7.3,</i></p> |
|---|--|



# Kindergarten CS Lessons

|                           |   |
|---------------------------|---|
|                           | 4.4   |
| You're already doing it!! | Counting, sorting and graphing in Math<br>Collecting data in Science<br>Recording number of students in the 100s club   |
| Extensions                | Students can <a href="#">create pictographs (abcya.com paint</a> - using the shape, stamp or sticker tools) Or with <a href="#">Google Drawings</a><br><br>Enter data in a Google Sheet and create a chart (put the sheet on the <a href="#">teacher directed website</a> and students can easily access it!) |

|   |   |
|---|---|
| <b>K.DA.S.1</b><br>Data & Analysis<br><br>Storage | <b>Recognize that data can be collected and stored on different computing devices over time and retrieved later.</b><br>All information stored and processed by a computing device is referred to as data. Data can be images, text documents, audio files, software programs or apps, video files, etc. It can be retrieved, copied, and stored in multiple places. As students use software to complete tasks on a computing device, they will be manipulating data. For example, students should be able to create and save a document. <i>Practice(s): Developing and Using Abstractions: 4.3</i> |
| You're already doing it!!                         | Any time students create, open and reopen documents they are learning that the files created were saved!  |
| Extensions  | Create documents in google Docs and open them again later! <a href="#">You can even create a template document and force a copy like this!!</a>   |

|   |  |
|---|--|
| <b>K.DA.IM.1</b><br>Data & Analysis<br><br>Inference & Models | <b>Discuss patterns in data to make inferences or predictions.</b><br>Data can be used to make inferences or predictions about the world. Students could analyze a Graph and pie chart of the colors in a bag of candy, identify which colors are most and least represented, and then make a prediction as to which colors will have most and least in a new bag of candy. For example, students preview a weather graph for one week in their city and make predictions about the weather for the following week. <i>Practice(s): Developing and Using Abstractions: 4.1</i> |
| You're already doing it!!                                     | Analyzing charts (behavior, reading, etc)<br>Graphs in Math / Science<br>Predicting the end of a story<br>Using context clues when reading   |
| Extensions  | Create a Google Sheet and have students log data such as number of sight words they can read, behavior, etc. <a href="#">Here's an example of a fluency log!</a><br><br>Transfer data to a Google sheet and create charts for students to analyze (together on the   |



# Kindergarten CS Lessons

|  |   |
|--|---|
|  | projector)<br><br>Predictions when rolling dice, spinning a wheel, etc. |
|--|---|

## ALGORITHMS & PROGRAMMING

|   |  |
|---|--|
| <b>K.AP.A.1</b><br><br>Algorithms & Programming<br><br>Algorithms | <b>With teacher assistance, model daily processes by following algorithms (sets of step-by-step instructions) to complete tasks.</b><br>Routines, such as morning meeting, clean-up time, and dismissal, are examples of algorithms that are common in many early elementary classrooms. Just as people use algorithms to complete daily routines, they can program computers to use algorithms to complete different tasks. Algorithms are commonly implemented using a precise language that computers can interpret. For example, students begin to recognize daily step-by-step processes, such as brushing teeth or following a morning procedure, as "algorithms" that lead to an end result. <i>Practice(s): Developing and Using Abstractions: 4.4</i> |
| You're already doing it!!   | Any step by step process that takes place in your classroom is an excellent opportunity to talk about algorithms and what happens if one step is missed or miscommunicated?<br><br>Clapping to get their attention! When I clap, you clap! This is an if/then command!<br>Math problems<br>Writing conventions (capital letters, periods)  |
| Extensions  | Create step by step instructions to complete a simple task<br><a href="#">Happy Maps - code.org unplugged</a><br><a href="#">Code.org Course A</a> (this has multiple activities that students can navigate)   |

|  |   |
|--|---|
| <b>K.AP.V.1</b><br><br>Algorithms & Programming<br><br>Variables | <b>With teacher assistance, model the way programs store and manipulate data by using numbers or other symbols to represent information.</b><br>Information in the real world can be represented in computer programs. Students could use thumbs up/down as representations of yes/no, use arrows when writing algorithms to represent direction, or encode and decode words using numbers, pictographs, or other symbols to represent letters or words. <i>Practice(s): Developing and Using Abstractions: 4.4</i> |
| You're already doing it!!  | Decoding words in language arts<br>Following arrows in directions<br>Colors to represent the noise level<br>Hand signals to identify different actions they want to take (bathroom, drink, sharpen pencil, etc.)  |
| Extensions   | <a href="#">Happy Maps - code.org unplugged</a><br><a href="#">Binary Bracelets - code.org unplugged</a><br>Navigate a maze using arrows - set a maze on the floor and hold up arrows for them to   |



# Kindergarten CS Lessons

|  |  |
|--|--|
|  | navigate<br><a href="#">The Big Event - code.org unplugged</a> |
|--|--|

|  |  |
|--|--|
| <b>K.AP.C.1</b><br><br>Algorithms & Programming<br><br>Control | <b>With teacher assistance, identify programs with sequences and simple loops, to express ideas or address a problem.</b><br>Programming is used as a tool to create products that reflect a wide range of interests. Control structures specify the order in which instructions are executed within a program. Computers follow instructions literally. Sequences are the order of instructions in a program. For example, sequences of instructions include steps for drawing a shape or moving a character across the screen. If the commands to program a robot are not in the correct order, the robot will not complete the task desired. Loops allow for the repetition of a sequence of code multiple times. For example, in a program to show the life cycle of a butterfly, a loop could be combined with move commands to allow continual but controlled movement of the character. For example, kindergarten students should be able to recognize loops and sequences in songs, rhymes, and games, such as the song B-I-N-G-O. <i>Practice(s): Creating Computational Artifacts: 5.1</i> |
| You're already doing it!!                                      | Any time you tell students to "continue until..." it is a loop!<br>Giving step by step instructions that must be done in order<br>Timelines in social studies<br>Experiments in Science  |
| Extensions   | <a href="#">Code.org Course A</a> (this has multiple activities that students can navigate)<br><a href="#">Happy Loops - code.org unplugged</a><br>Have students give commands to navigate a maze  |

|   |  |
|---|--|
| <b>K.AP.M.1</b><br><br>Algorithms & Programming<br><br>Modularity | <b>With teacher assistance, solve a problem by breaking it down into smaller parts.</b><br>Decomposition is the act of breaking down tasks into simpler tasks. For example, Students could break down the steps needed to make a peanut butter and jelly sandwich, to brush their teeth, to draw a shape, to move a character across the screen, or to solve a level of a coding app. <i>Practice(s): Recognizing and Defining Computational Problems: 3.1</i> |
| You're already doing it!!   | Decoding words by sound<br>Step by step instructions in anything<br>Completing a jigsaw or any other type of puzzle  |
| Extensions  | <a href="#">Code.org Course A</a> (this has multiple activities that students can navigate)<br><a href="#">Here's an explanation of modularity and CS</a> (not necessarily for students - maybe too advanced but a way to wrap your own head around it!)   |



# Kindergarten CS Lessons

|  |  |
|--|--|
| <p><b>K.AP.PD.1</b></p> <p>Algorithms &amp; Programming</p> <p>Programming Development</p> | <p><b>With teacher assistance, develop plans that describe a program’s sequence of events, goals, and expected outcomes.</b></p> <p>Programming is used as a tool to create products that reflect a wide range of interests, such as video games, interactive art projects, and digital stories. Students could create a planning document, such as a story map, a storyboard, or a sequential graphic organizer, to illustrate what an end product will do. Students at this stage may complete the planning process with help from their teachers. For example, kindergarten students could illustrate the beginning, middle, and end of a favorite story. <i>Practice(s): Creating Computational Artifacts, Communicating About Computing: 5.1, 7.2</i></p> |
| <p>You’re already doing it!!</p>   | <p><a href="#">Thinking maps</a></p> <p>Goal setting</p>   |
| <p>Extensions</p>  | <p><a href="#">Programming in code.org with angry birds!</a></p> <p><a href="#">Great computational thinking game that ties with Math!</a></p>   |

|  |   |
|--|---|
| <p><b>K.AP.PD.2</b></p> <p>Algorithms &amp; Programming</p> <p>Programming Development</p> | <p><b>With teacher assistance, identify attribution (credit) when using the ideas and creations of others while developing programs.</b></p> <p>Using computers comes with a level of responsibility. Students should credit artifacts that were created by others, such as pictures, music, and code. Credit could be given orally, if presenting their work to the class, or in writing or orally, if sharing work on a class blog or website. Proper attribution at this stage does not require a formal citation, such as in a bibliography or works cited document. <i>Practice(s): Communicating About Computing: 7.3</i></p> |
| <p>You’re already doing it!!</p>   | <p>Discussing the author of a book or a piece of art</p> <p>Students putting their name on their paper - gives THEM credit!</p> <p>Not stealing other people’s work</p> <p>Having students identify where they got information (which book, website, person, etc.)</p>  |
| <p>Extensions</p>  | <p><a href="#">Here’s a good article about copyright</a> and talking to kids about it.</p>  |

|  |  |
|--|--|
| <p><b>K.AP.PD.3</b></p> <p>Algorithms &amp; Programming</p> <p>Programming Development</p> | <p><b>With teacher assistance, debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops.</b></p> <p>Algorithms or programs may not always work correctly. Students should be able to use various strategies, such as changing the sequence of the steps, following the algorithm in a step-by-step manner, or trial and error to fix problems in algorithms and programs. For example, kindergarten students should be able to identify incorrect order in a series of events and place them in the correct order, such as getting ready for school or making a peanut butter sandwich. <i>Practice(s): Testing and Refining Computational Artifacts: 6.2</i></p> |
| <p>You’re already doing it!!</p>   | <p>Correcting work (math problems, language arts, etc.) / test corrections</p> <p>Alphabetize</p> <p>Numerical order</p>   |



# Kindergarten CS Lessons

|            |   |
|------------|---|
| Extensions | <a href="#">Unspotted Bugs code.org unplugged</a> |
|------------|---|

|   |   |
|---|---|
| <b>K.AP.PD.4</b><br>Algorithms & Programming<br>Programming Development | <b>With teacher assistance, using correct terminology, describe steps taken and choices made during program development.</b><br>At this stage, students should be able to talk or write about the goals and expected outcomes of the instructions they develop they create and the choices that they made when developing their instructions. This could be done using coding journals, discussions with a teacher, or teacher created classroom blogs. For example, kindergarten students could describe their thinking about a story map or set of instructions they develop.<br><i>Practice(s): Communicating About Computing: 7.2</i> |
| You're already doing it!!   | Solving and correcting math problems<br>Asking "how did you come up with that answer?"<br>Describing steps taken to complete something  |
| Extensions  | <a href="#">Coding in Scratch Jr.</a>   |

## IMPACTS OF COMPUTING

|  |   |
|--|---|
| <b>K.IC.C.1</b><br>Impacts of Computing<br>Culture | <b>Discuss how people lived and worked before and after the implementation or adoption of new computing technology.</b><br>Computing technology has positively and negatively changed the way people live and work. In the past, if students wanted to read about a topic, they needed access to a library to find a book about it. Today, students can view information on the Internet about a topic or they can download e-books about it directly to a device. <i>Practice(s): Communicating About Computing: 7.2</i> |
| You're already doing it!!                          | History and social studies<br>Changes in technology<br>Current events   |
| Extensions   | Show examples of the evolution of different technology - phones, computers, cameras, etc.<br><br>Discuss advantages and disadvantages of old and new technology   |



# Kindergarten CS Lessons

|   |   |
|---|---|
| <b>K.IC.SI.1</b><br>Impacts of Computing<br>Social Interactions | <b>Work respectfully and responsibly with others online.</b><br>Online communication facilitates positive interactions, such as sharing ideas with many people, but the public and anonymous nature of online communication also allows intimidating and inappropriate behavior in the form of cyberbullying. Teachers should facilitate a discussion in how to avoid sharing information that is inappropriate or that could personally identify them to others, and how to. Work in a kind and respectful manner. <i>Practice(s): Collaborating Around Computing: 2.1</i> |
| You're already doing it!!                                       | Behavior expectations<br>Anti-bullying discussions  |
| Extensions  | <a href="#">Know the Rule video</a> (1:53)  |

|   |   |
|---|---|
| <b>K.IC.SLE.1</b><br>Impacts of Computing<br>Safety, Law & Ethics | <b>Keep login information private, and log off of devices appropriately.</b><br>Using computers comes with a level of responsibility. Students should not share login information, keep passwords private, and log off when finished. <i>Practice(s): Communicating About Computing:7.2</i> |
| You're already doing it!!   | Inform students that they need to log on and off to prevent others from using the computer under their log in. Because it saves files, etc.<br><br>Reminding students the importance of keeping passwords private so others can't access your information online.                           |
| Extensions  | <a href="#">Password rap video</a> (2:00)   |

## TIPS and TRICKS

[AZ CS Standards Full document](#)

The core of Computer Science is really computational thinking! YOU do this with your students all day every day!! An excellent way of getting your students to think like computer scientists is to speak to them about how your everyday routines relate to computer science!

Notice that none of these standards refer to a specific program or app! They are written to develop computational thinkers! So, keep doing what you're doing but help them make the connections to computer science through your discussions!

Utilize the [teacher directed websites link](#) to make it easy for students to access documents and programs! You can add links to documents to share with students (google sheets, docs, slides, etc.) as well as websites you want them to use!!



# Kindergarten CS Lessons

Even Kindergarten students can begin using programs like Google Docs, Sheets and Slides! Let them change fonts, add pictures, enter data, etc. Let them type their sight words in a Google Doc or a Google Slide then change the fonts! Allow them to enter data into a Google Sheet and change the cell color or borders!

## USEFUL WEBSITES AND RESOURCES

[Internet Safety BrainPop video](#) (3:46)

[Internet Safety - Don't click on pop-ups video](#) (1:51)

[Keyboarding Practice on abcya.com](#)

[Links to many activities by grade level](#)

[Mouse skills practice](#)

[Mouse skills - Bees & Honey](#)

[Code.org Course A](#) (this has multiple activities that students can navigate over multiple class periods)

[Codeforfun](#) - TONS of grade appropriate lesson plans and materials to use sorted by topic

[Coding game for young students](#)

[Great game to play during recess!](#) - teaches if/then/else

[Another great get moving game!](#) - treasure hunt using directions

[Great short videos about Internet safety!](#) - excellent videos for beginning of the day when doing attendance!

[K-12 Digital Citizenship](#) from Common Sense Media