



**Topic:** STEM

**Badge:** Coding For Good 1: Coding Basics

**Suggested Supplies:** Computer or phone with internet access

When we use computers to do schoolwork, watch a movie, or shop, we don't usually think about what makes them work. Without computer programmers and the code they write for computers, we wouldn't be able to do any of those things. In this badge, you'll learn about what programmers do and become one yourself.

Steps:

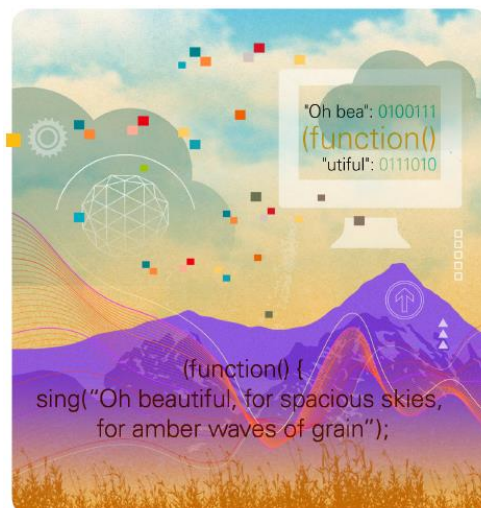
### **Step 1: Learn about functions through song lyric**

Computers can fly airplanes, restart stopped hearts, and give you directions to your favorite restaurant! Programmers write the algorithms computers use to accomplish tasks. Algorithms are the step-by-step instructions a computer follows. The algorithms are made up of different parts like functions, arguments, and variables.

In coding, a function tells a computer to do something. It's similar to a verb, like sing or whistle.

You can make a function like `sing()` more specific by including an argument. An argument tells the computer what words to sing. For example, `sing("Oh beautiful, for spacious skies, for amber waves of grain")`.

You can use the same function, `sing()`, with different arguments or variables to get your computer to sing other verses or an entirely different song.



# WORDS TO KNOW

**Algorithm** a series of specific instructions. By creating a sequence of instructions that can be applied to many circumstances, you're creating an algorithm.

**Argument** a part of code that makes a function more specific and reusable in a number of different ways: it adds details to the function that are changeable. In many programming languages, arguments are represented as a list separated by commas inside the parentheses.

**Choreographer** a person who arranges dance movements for dancers to perform.

**Code** a series of instructions that make up a program directing a computer to do something.

**Computer** an electronic machine that can store and process data. A computer has hardware, which is the machine itself, and software, which is a set of instructions.

**DRY** a concept in programming that means "Don't Repeat Yourself."

**Function** one of the basic building blocks of a program. It's a type of instruction similar to a verb: a function does something. In JavaScript, as in most programming languages, it has a special form, which is the name of the function followed by '()'. For example, `turnLeft()` and `whistle()` are two examples of functions. The `()` tells the

computer to "do" the named function. "Doing" a function is typically described as "calling" a function or a "function call."

**Function declaration** the process of creating a new function. This allows a coder to write reusable code.

**Iteration** when you repeat a process. For example, each time a program loops, the code inside the curly brackets runs.

**JavaScript** a computer programming language.

**Loop** a function that repeats a piece of code a number of times. For example, a loop using pseudocode syntax might look like this:

```
for 1...3 {  
  The repeating code  
  goes here  
}
```

This piece of code loops 3 times, starting at 1. Each time the program goes through an iteration, which means one looping, the code inside the braces runs.

**Programmer** a person who writes algorithms to create programs or code for computers.

**Pseudocode** a way to plan a computer program using human-friendly language. It's not actual programming, but a written description of the key

elements of an algorithm or program. It's used as a quick way of thinking about a program without completely writing it out in code.

**Sequence** the order in which the computer performs the steps the programmer writes.

**Software** the end product of written computer code.

**Syntax** rules for how a program is written. These rules have a purpose similar to written grammar: it's a standard format that the computer understands. In programming, the syntax is more specific and detailed, and needs to be exactly correct for a computer to know what to do. For that reason, programmers often use pseudocode to help them flesh out ideas without the burden of being too exact.

**Variable** in coding, a variable holds or "stores" information. This makes it easy to reuse information that might be used many times.

**Whiteboarding** the process of writing or reading code on a whiteboard or in other unplugged ways. It's used often as a way of understanding code without the assistance of a computer.

## Step 2: Learn about loops through song patterns

**Computers were invented to do repetitive tasks, and they're really good at it!** Loops in a program tell the computer to repeat an action. For example, a computer programmer might use a loop to tell a computer to repeat a math calculation using different sets of data.

If you were writing a song in computer code, lyrics that repeat could be written in a loop. The chorus in a song is a loop, because you repeat it. Verses aren't loops because they don't repeat, and the words for verses are different.

Programmers like to follow the concept of DRY—Don't Repeat Yourself! Loops make code more efficient. They let the computer do the repeating, instead of the programmers!

## Step 3: Write an algorithm duet

Two heads are better than one. When you're first planning a new project with others, you work together to put all of your ideas down on paper. They might not be the best ideas or the ones that you use in your final project, but they're a good starting point.

Programmers write code in much the same way. They often work in teams to write programs. Programmers often work in teams to write programs. Any program will be revised many times.

Collaboration also allows programmers to share ideas and inspire each other. Programmers build on their teammates' ideas, often creating better code than if they wrote it alone. Sometimes songwriters and musicians collaborate, too.

What are the benefits of working together on a creative project? Can you think of any challenges to collaborating or drawbacks? What makes a good collaborative team?

## Step 4: Code a performance routine

**Create a dance, step-by-step!** Just like programmers give specific step-by-step instructions to a computer, a choreographer creates specific step-by-step instructions for dancers.

The similarity between coding and choreography doesn't end there. When a choreographer creates a dance, she

- uses a special language that dancers understand (Many names of dance moves are in French. Some dance moves have unusual names like sugar or shimmy.)
- writes down the steps in sequence
- uses a special syntax to indicate repeating steps or variation
- might write the dance out in words, like pseudocode, while she's figuring it out

When she's finished, she might put it into a special type of dance notation that looks like a flow chart or drawing to document her creation so it can be shared with others.

## Step 5: Share your coded routine with others

**What’s the big dance craze today?** In the 1920s, everybody was doing the Charleston. In the 1970s, it was the Hustle. Some famous dances have been handed down for hundreds of years. Some are taught from one dancer to another, but others have been written down.

Imagine a dancer hasn’t seen a dance and was trying to recreate it based only on a set of written notes from a choreographer—it would be a lot like a computer running a program. The dancer could only do what was written down. The notes would have to be very detailed and specific for the dancer to recreate the dance accurately.

Since a computer can’t guess or make assumptions about code, programmers share their code with their colleagues to make sure it’s clear and understandable. One way programmers gather feedback is by using a process called whiteboarding where they write their code on a whiteboard and ask others to look for places that are confusing.

Having other people look at your code to find and fix errors, improve the user experience, or clarify instructions is an important step in writing good code.

### Computer-Assisted Design

People use computers to design all kinds of things: cars, houses, clothes, toys. Programmers write the code computers use in design. You could have a career creating computer design programs or using them.

**Raye Montague** did both while working for the Navy! Around 1970, she was assigned a project by her boss: create a computer program to design warships. Ms. Montague programmed a computer to do just that. Before she created this program, it took about two years to design a ship. Her program made it possible to design a ship in less than 24 hours.

Her program revolutionized ship design for the Navy and became central to ship and submarine design. She received the Navy’s Meritorious Civilian Service award in 1972 in honor of her work. Ms. Montague worked for the Navy for 33 years.

### Computer Pioneer: KAREN SPARCK JONES

Karen Sparck Jones was a self-taught programmer who taught computers to understand human language. Ms. Jones combined linguistics, or the study of language, and statistics to create algorithms that allowed computers to recognize relationships between words. Her work created the foundation for search engines like Google. Ms. Jones was also an advocate for women in the field of computers and mentored a generation of programmers.



“Computing is too important to be left to men.”

SEARCH

**\*Once completed, this badge can be purchased at the following direct link:**

<https://www.girlscoutshop.com/ambassador-coding-basics-badge>

**Please visit the following link to complete a quick survey on this virtual badge:**

<https://gswestok.wufoo.com/forms/virtual-badge-completion-survey/>