

Lesson 16: Functions with Return Values



Adapted from Code.org curriculum

Objectives



- ↻ Use the return command to design functions.
- ↻ Identify instances when a function with a return value can be used to contain frequently used computations within a program.
- ↻ Design functions that return values to perform frequently needed computations within a program.

Getting Started



- 🌀 Unplugged Activity: Return Values with Go Fish
- 🌀 Today we are going to look at how to write our own functions with return values. We are going to explore this idea by playing the classic card game Go Fish
- 🌀 We will use the “Return Values with Go Fish” Activity Guide
 - 🌀 Groups of 4 with a set of cards
 - 🌀 Play “Go Fish”
 - 🌀 Finish the worksheet

Getting Started



☞ Why do we need to return some information from the Responder to the Asker?

Getting Started



- ❧ Why do we need to return some information from the Responder to the Asker?
 - ❧ Once the asker has gained the information, he uses it to continue computing information.
 - ❧ The asker can not easily gain the information without the help of the responder, as he doesn't have access to the cards.

One possible algorithm:



- ↻ var gaveCard = false
- ↻ for each card in responder's hand
 - ↻ if card is equal to desiredCard
 - ↻ gaveCard = true
 - ↻ RETURN card
- ↻ if gaveCard is false
 - ↻ RETURN "Go Fish"

Return Values



- ↻ A process in which a question is asked, something computes an answer and gives the answer back to the asking location

Activity:



- As we saw playing Go Fish, we often need to ask for information and receive an answer to be able to make decisions.
- We have seen a few different functions that do something like this, such as `randomNumber`, `getText`, and `includes`.
- Up until now, though, we have never been able to create our own functions that return information. Today we are going to learn how to write functions with return values.

Code Studio



Complete Code Studio Stage 16

Introduced Code:



return

Wrap-up:



- Return values are a useful way to move useful information generated inside of a function to the rest of your program.
- There is another way we can do this. If we write our function so that it stores its output within a global variable, then that information will be accessible when the function terminates, since the global variable will stay in scope.

Prompt:



“Why would we prefer to write a function that returns a value to using the strategy shown above? How might return values make our function more generally useful? How might they make our code easier to reason about?”

Prompt



- ❧ A function that saves output in a global variable must specifically reference that variable by name. Since that name is “hard-coded,” your function can only ever save information in that variable. If we wish for our functions to be generally useful, we should be able to decide how to use the output a function generates.
- ❧ A global variable is accessible by all functions. As a result, it can be difficult to determine every location in your program that modifies this variable. Reasoning about how its value will change over the course of the program is much harder, as is debugging unexpected behavior. Using a return value limits the effects of a function to the local variables of the function and the place where the function was called.

The ability to return values is closely tied to the concept of scope. All variables declared within a function are in local scope and so will be removed once the end of the function is reached. As a result any useful information generated during that function will be lost. One solution to this problem is storing the value in a global variable, but this is generally considered bad programming practice. Global variables can be accessed by many functions and so reasoning about their logic requires considering the logic of all of those functions. Return values are a way to move information out of the local scope of a function without using a global variable. As a result a function call can be treated as if it were the type of data that a function returns, and it is up to the programmer to determine if or how it will be used.

Vocabulary:



Return Value - A value sent back by a function to the place in the code where the function was called from - typically asking for value (e.g. `getText(id)`) or the result of a calculation or computation of some kind. Most programming languages have many built-in functions that return values, but you can also write your own.