Functions Activity

Scratch allows you to create custom functions. These can be used to simplify your code and exploit repeating patterns. In this activity, we’re going to create our own functions to solve simple problems. For example, the program below calls a function that draws a hexagon of a given size.

A screenshot of a game

Description automatically generated

In particular, notice that:

* For every **function** we create in scratch, we need to define the input and the output
* A **function** helps us hide details from the caller (abstraction)
* A **function** by itself doesn’t do anything, we need a **program** to call the function as well

Assignment:

For each of the following problems, create a function/program to solve it. Unlike with programs, **we are not going to use the backpack for functions**. Instead, leave your functions in the main coding space.

* 1. Create a **function** named “triangle” that takes a size as an input and then creates a triangle of that size. *After* you create the function, **create** a program that uses the new block to make sure it works.
  2. Create a **function** named “square” that takes a size as an input and then creates a pentagon of that size. *After* you create the function, **create** a program that uses the new block to make sure it works.
  3. Create a **function** named “pentagon” that takes a size as an input and then creates a pentagon of that size. *After* you create the function, **create** a program that uses the new block to make sure it works.
  4. Create a **function** named “polygon” that takes a size and a number of sides as inputs and then creates a polygon of that size and shape (hint: look at how the number of sides relate to the degree turn in prior problems). *After* you create the function, **create** a program that uses the new block to make sure it works.
  5. Create a **program** that uses a loop, a variable, and the function from the previous problem to create the following shape:

A blue line drawing of a house

Description automatically generated

* 1. Make a **function** named “house” that takes a size as an input and then draws a simple house of that size. *After* you create the function, **create** a program that uses the new block to make sure it works. For example:

A drawing of a house

Description automatically generated

* 1. Create a **program** that draws a triangle AND at each corner of the triangle, draws a house (using the function from the previous problem).
  2. Make three (or more) **functions** that make a sprite “dance” in different ways (e.g., jump, twirl, slide left). Create a program that combines these “dance move” functions to create a dance routine set to music.
  3. Create a **function** named “square\_number” that takes a number as an input and then prints out the square of that number. For example, an input of 5 should result in the number 25 being printed. *After* you create the function, **create** a program that uses the new block to make sure it works.
  4. Create a **program** that uses a loop, a variable, and the function from previous problem to print all of the squares of the numbers between one and ten.
  5. (Bonus) Create a **function** named “prime” that takes a number as an input and then prints out “YYY is a Prime”.
  6. (Bonus) Create a **program** that uses a loop, a variable, and the function from the previous problem to print all of the numbers that are primes between one and one hundred.