



Functions

Computer Science Principles

What is a Function?

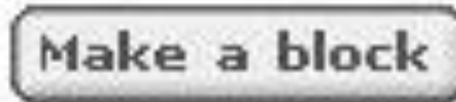
- In BYOB/SNAP, as well as other programming languages, we can take code blocks and put them in a new block that we can define.
- These new blocks are called **functions** or **procedures** because they perform a function for us, such as the ones that you have already been using in BYOB/SNAP.

What is a Function?

- We can create our script (or code block) once ~ and use it as many times as we want *without* having to re-write the code.
- We only need to “call” or “invoke” it.
- This is called *Procedural Abstraction*.

Functions in BYOB/SNAP

- To create a function in BYOB/SNAP, we are going to make our own block.
- Click on **make a block** at the bottom of the **Variables** tab.



- This will open up the **make a block** dialog box.

Make a Block in BYOB/SNAP

- You get to choose which palette the block will be in.
 - Choose the one that “makes sense”
 - Example
 - Our new block will be to draw a square, so we will choose Motion.

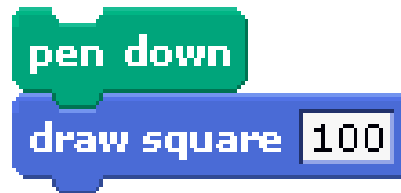


Types of Function Blocks

- There are three (3) types of function blocks you can create in BYOB/SNAP.
 1. Command
 - Creates a block that “makes something happen”
 - The same blocks that looks like puzzle pieces.
 2. Reporter
 - Creates a block that reports or returns a value.
 - These blocks look like some of your operator blocks, such as the $() + ()$ block. (rounded ends)
 3. Predicate
 - Creates a block that reports or returns either true or false.
 - These blocks look like some of your operator blocks, such as the $() < ()$ block. (pointed ends)

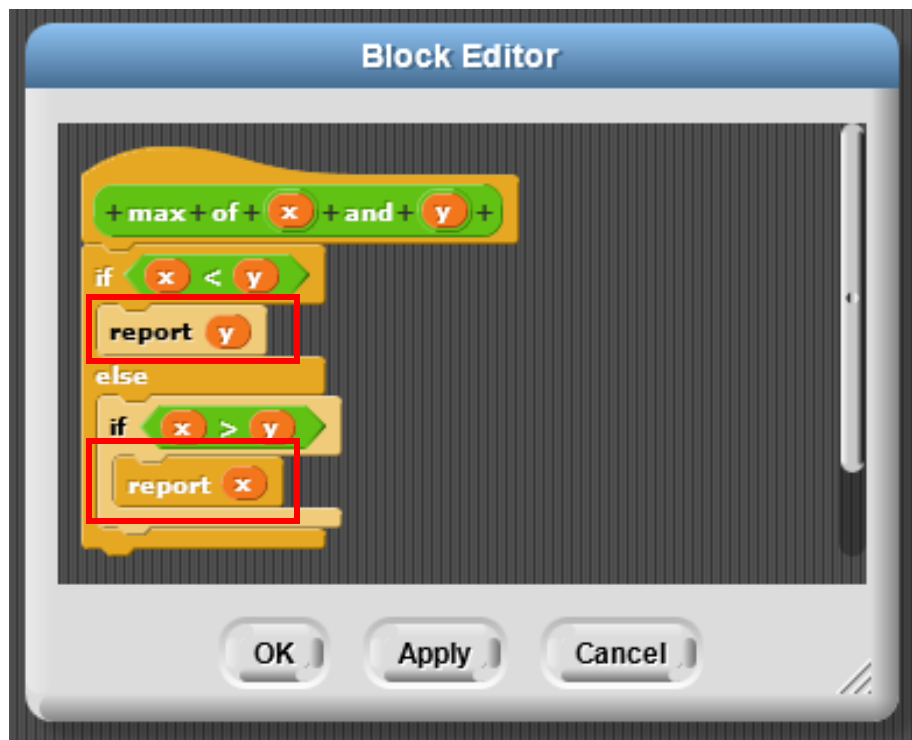
Command Function Blocks

- A command function block is created to have a task carried out without any value being returned.
 - Such as drawing a square.



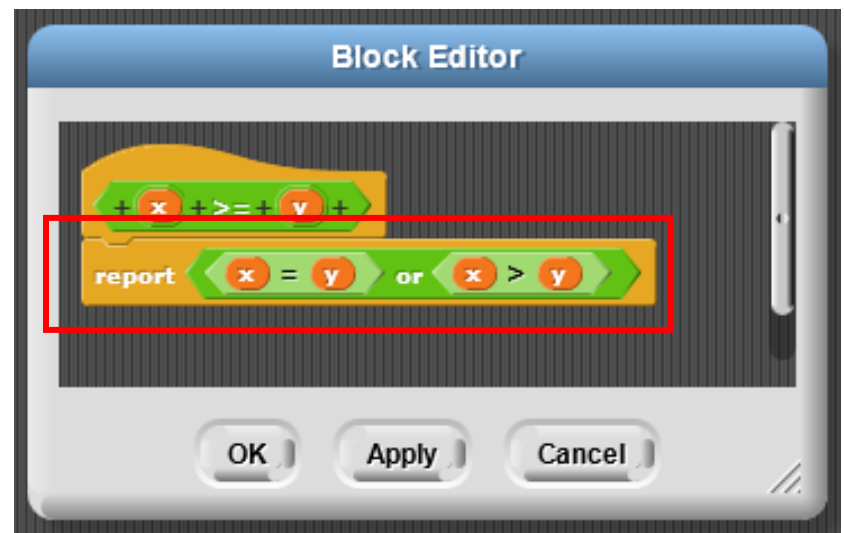
Reporter Function Blocks

- Reporter blocks can report a value.
- You will need to use the report () block to return or report the value.



Predicate Function Blocks

- A predicate block returns either true or false.
- You will need to report other blocks that evaluate to either true or false.
- The Block Editor will open with a report block.

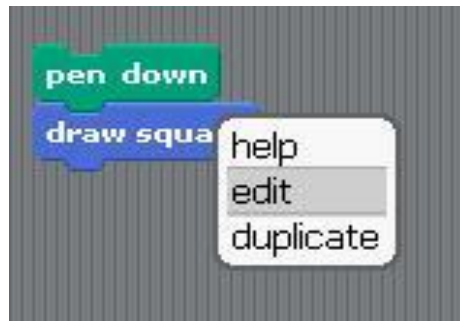


Arguments

- You have created a block that draws a square, but it only draws a square where each side is of length 100 steps.
- It would be great if we could specify how long we wanted each side to be.
- We will edit the block to accept an **argument** (or **input**), which tells it the length of the square it has to draw.

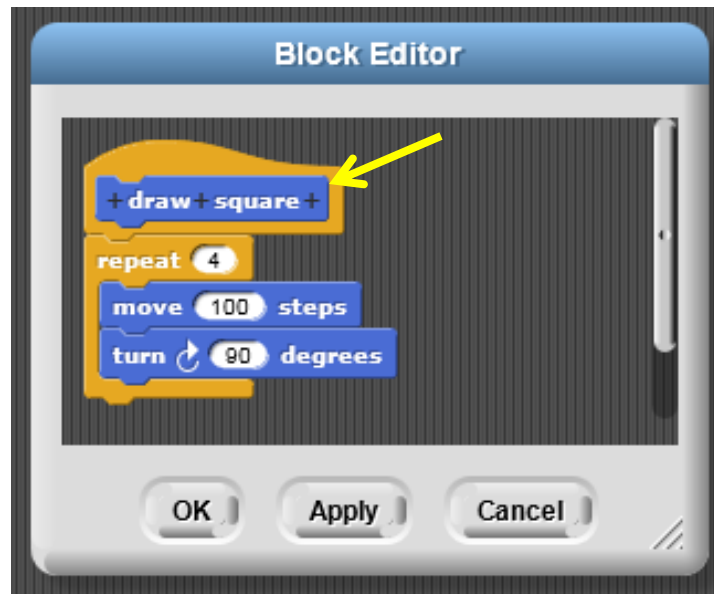
Arguments

- If you are modifying a block to add arguments, right-click on the new block and select **edit** to go back to the block editor.



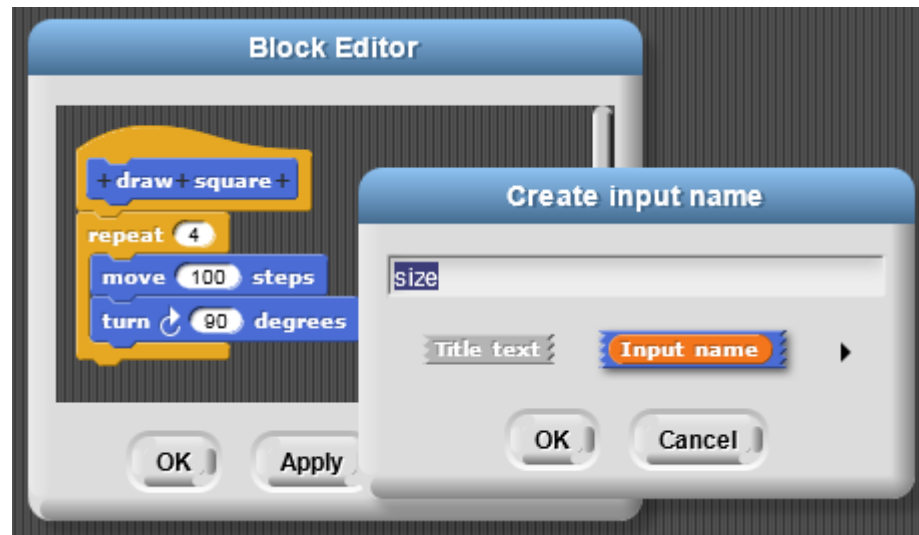
Arguments

- In the Block Editor, notice that when you move the mouse over the top row of the new block, some plus signs (+) show up.
- When you click on these plus signs, you can add more text or arguments.
- When you click on the text between the plus signs, you can delete or modify that text.
- Click on the plus sign at the far right as shown below:



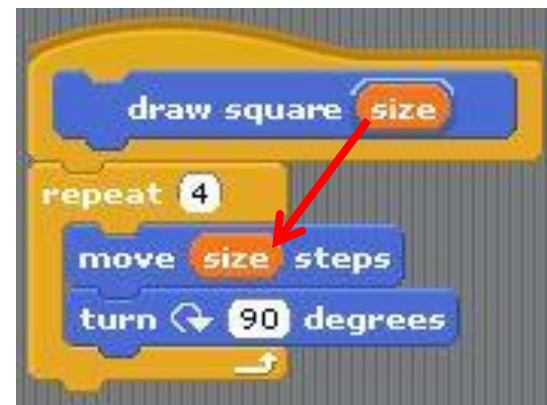
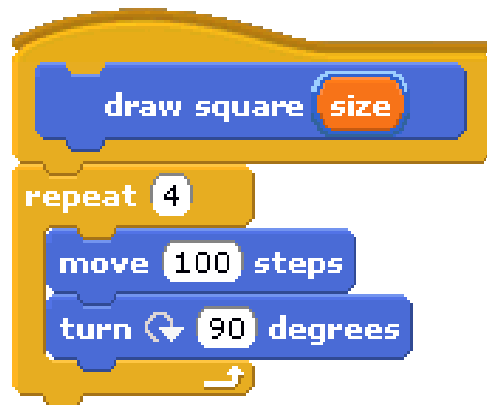
Arguments

- When you click on the plus sign on the far right, you should get the following dialog box.
- With this dialog box, we can select if we want to add input (orange) or more text (blue).
- We want to add the input size, so we type size, select Input Name and click OK.



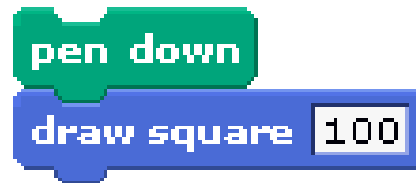
Arguments

- Now, we have a variable inside our block definition.
- Drag the variable size down into the move block.
- Whenever we need a new copy of a variable, we just grab the copy from that variable in the top row.



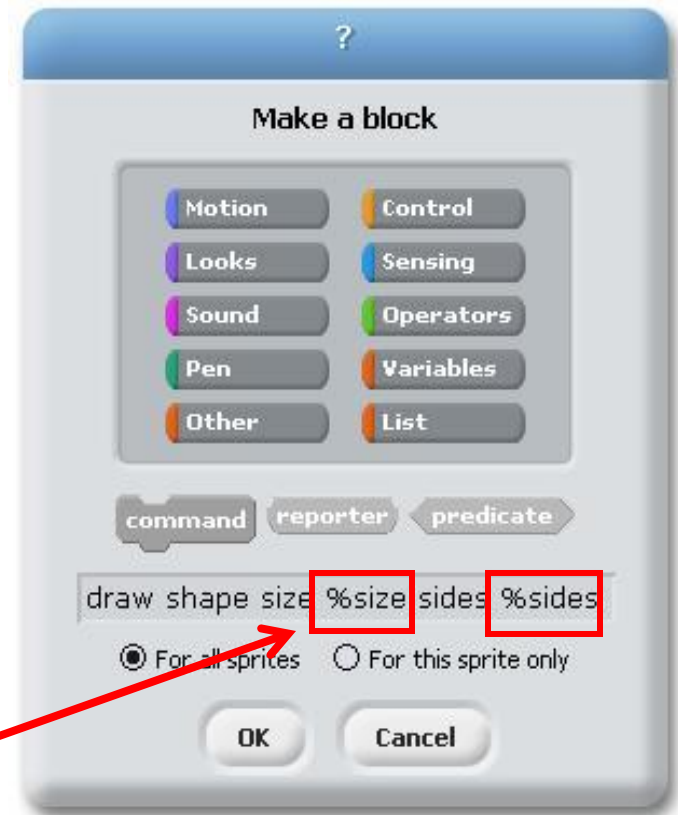
Arguments

- When we click OK, we will see that our draw square block now takes an argument.
- We can put different numbers in the blank and draw squares of different sizes!



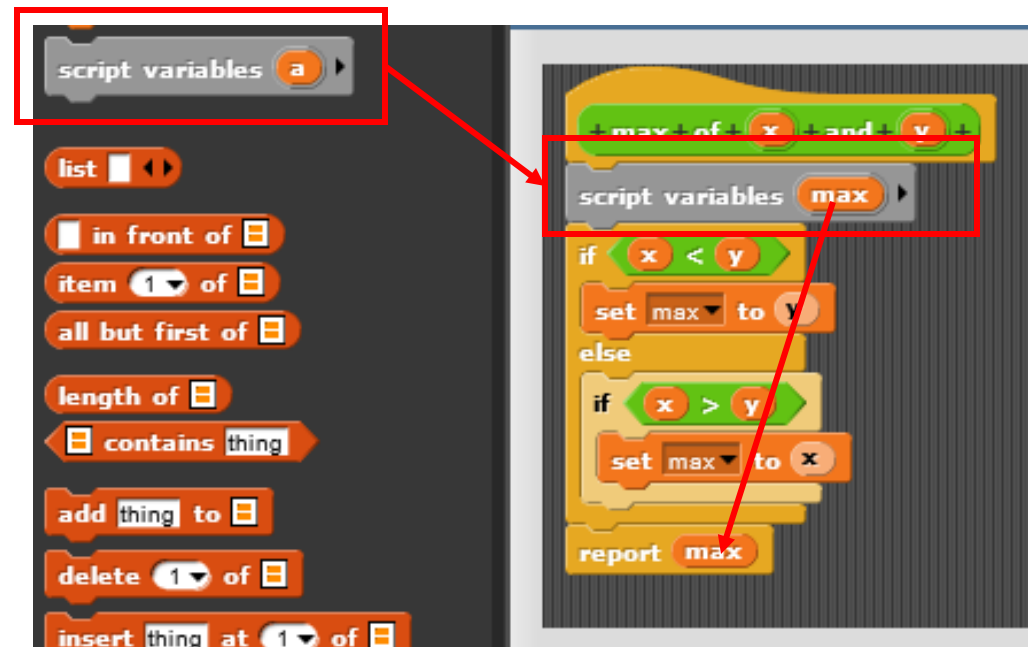
Arguments

- If you are just starting to create your function block, you can create the inputs to this block in exactly the same way as we did in the previous section, by clicking on the plus signs to add input.
- You can also type the names of the input as shown.
 - The percent signs (%) indicate that the word should be an input.



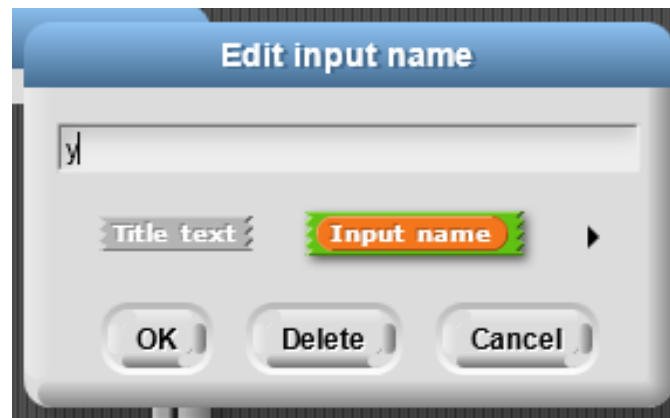
Script Variables

- A script variable is a variable that is only available inside that specific script.
- After you change the name, simply drag it into the Block Editor.
- You can then click and drag the variable's name from the script variables block into spots, such as the report box.



Input Types

- You can limit the types of values that can be entered into a block's argument.
- Open the Block Editor for the function block and click on the input's name.
 - The following window displays.



Input Types

- Then, click on the right arrow in the pop-up box.
- This will open the dialog box shown on the next slide.
 - This allows us to specify the shape of the slot.
- We want a numbers-only slot (as shown selected below).
- We can also specify that we want the variable to have a *default* value.
 - This is similar to blocks like move that always start out with the default value 10.

Input Types

The screenshot shows a dialog box titled "Edit input name". At the top is a text input field. Below it are two tabs: "Title text" and "Input name", with the latter being selected. The main area contains a grid of input type icons and labels:

- Object
- Number
- Command (inline)
- Command (C-shape)
- Text
- Any type** (highlighted with a red box)
- Reporter
- Any (unevaluated)
- List
- Boolean (T/F)
- Predicate
- Boolean (unevaluated)

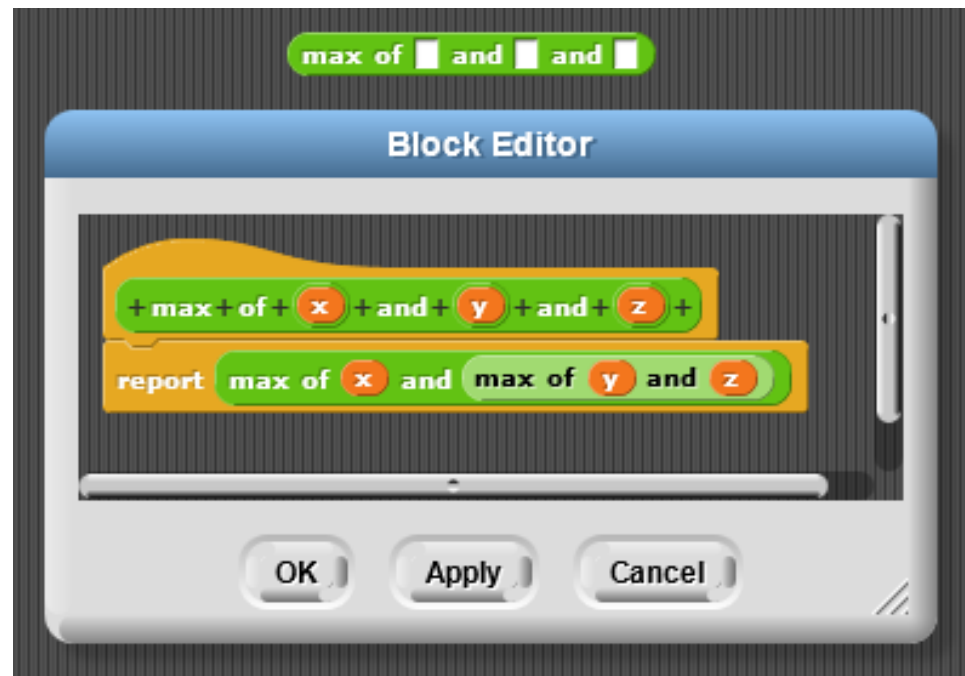
Below the grid, there are three radio button options:

- Single input** (highlighted with a red box). Default Value: [text input field]
- Multiple inputs (value is list of inputs)
- Upvar - make internal variable visible to caller

At the bottom are three buttons: "OK", "Delete", and "Cancel".

Composition of Functions

- Our custom-made blocks are blocks like any other, and we can use them in other block definitions.



Different Kinds of Variables

- Normal/Global Variables
- Sprite Specific Variables
- Arguments to a function
- Block Variables
- Script Variables

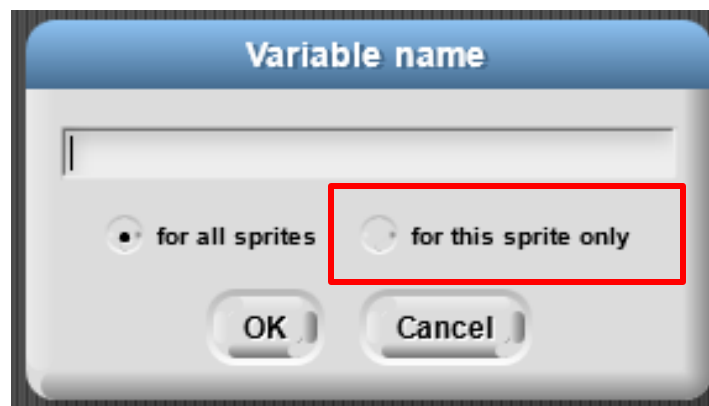
Normal/Global Variables

- These variables are made in the regular menu and can be used ANYWHERE!
- The variable "score" is an example.
- *These can be used by any sprite, in any block or in any script.*



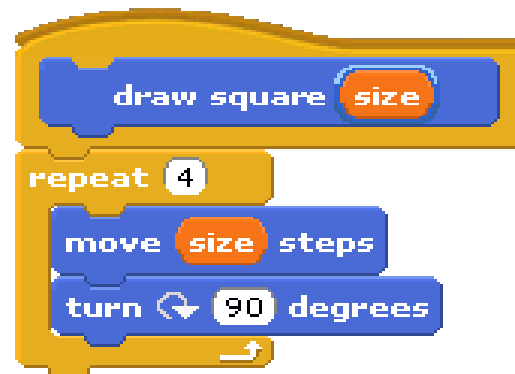
Sprite Specific Variables

- When you create a "normal/global" variable you can select that the variable is "For this sprite only".
- Then these variables will show up as variables listed below the line in the variables tab.
- We recommend *not* using these variables in blocks.



Arguments to a function

- A variable set by the person calling the function.
- We also refer to this as "**input**".
- *This can ONLY be used within the block editor.*



Script Variables

- The "script variable" block gives us a variable that we can use inside of this script.
- *These can only be used in that particular script.*
- *The script could be a block script (shown below) or a regular script.*

