



# Variables



## Introduction

Scratch is a visual programming language suitable for children and teens to create their own interactive stories, games and animations. Projects can also be uploaded and shared with the Scratch community via the Scratch website for others to enjoy.

In this exercise we'll learn about variables. Variables are a means to record values throughout a program and can be changed and saved when required. These values can then be used to control areas and functionality within our project, a high score leaderboard is the perfect example.

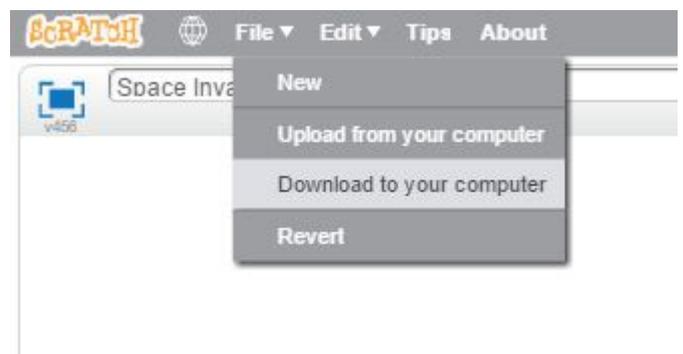
For this lesson we're going to build a simple dice game, throwing two dice and adding them together. Then we'll take the leader board idea and build a simple means to record the highest roll.

Let's get started by first loading Scratch. You can either install Scratch on your computer or load it via your web browser by navigating to <https://scratch.mit.edu/>

### Top Tip

Don't forget to save your work regularly. It's good practice to get in the habit of saving any progress you've made. So when you're feeling great after solving a coding problem, save it, save it, save it!

The save options may differ slightly between Scratch versions but head on up to the File option in the top left and you'll find it.

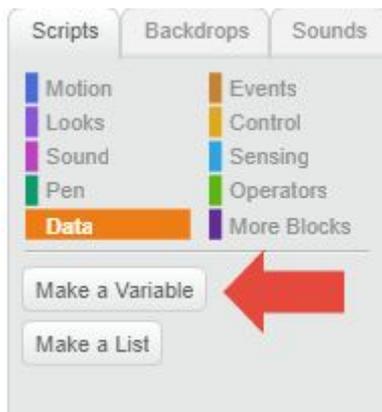


## Let's get started

## Step 1: Creating the Variables

We first need to create the variables that we'll be assigning values to. We can create as many variables as our program needs and even create more along the way. For our dice game however only two variables are needed to start with: one for our first dice and the another one for our second dice.

They can easily be created in Scratch by making use of the 'Make a Variable' button found in the Data section of the Scripts tab.

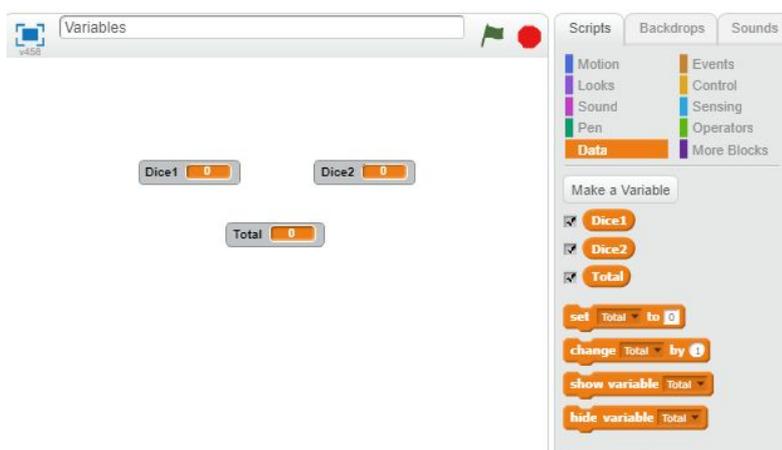


Create two separate variables and name them as shown below:



We will also need a third variable to record the total on our two dice rolls, create this in the same way as the others and call it Total.

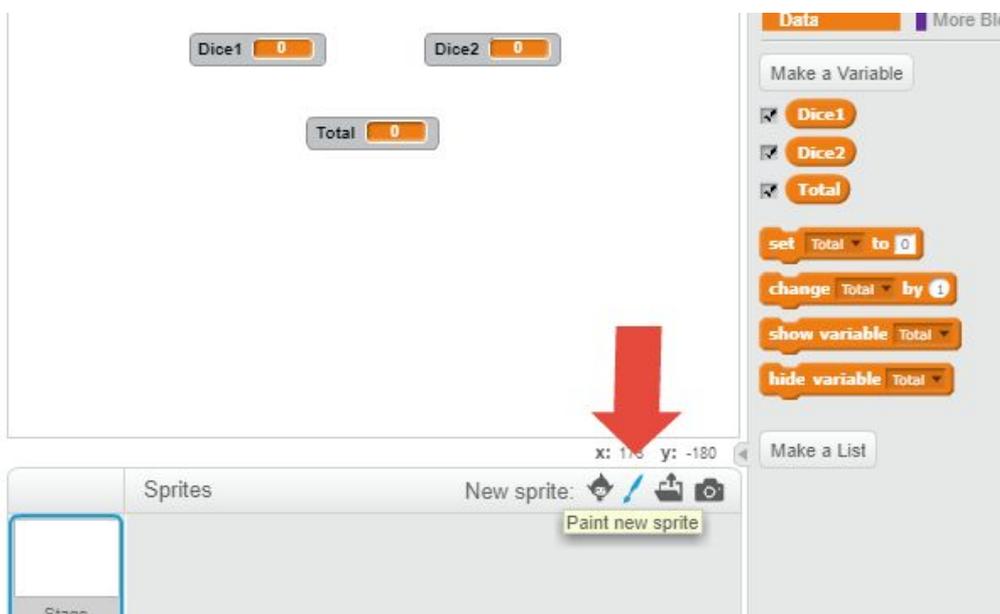
Whenever a new variable is created within Scratch it will appear in our main work area on the left. We can then rearrange our variable boxes by dragging the with the mouse into any position we like. Rearrange them into a similar position below:



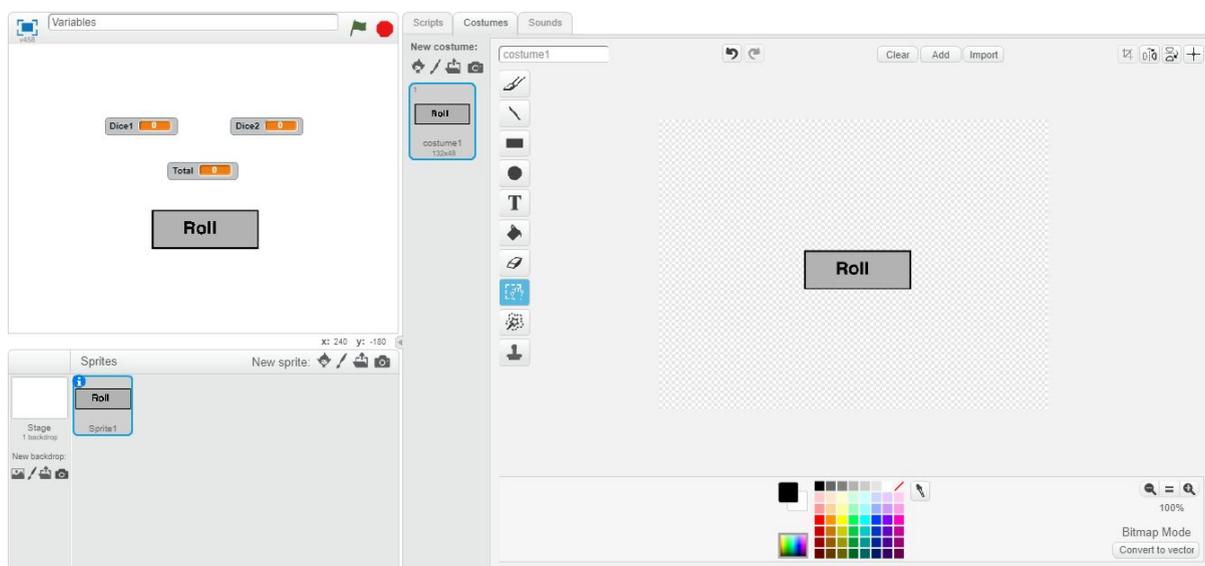
As explained earlier, variables are a means to record values throughout our program and it's these boxes that will display the current values assigned to them as the program runs.

## Step 2: Button Creation

We now need to create a button for the user to press. The pressing of the button will then tell our program that we want it to roll the dice to start the game. Our button is created by first selecting the 'New Sprite' button shown below:



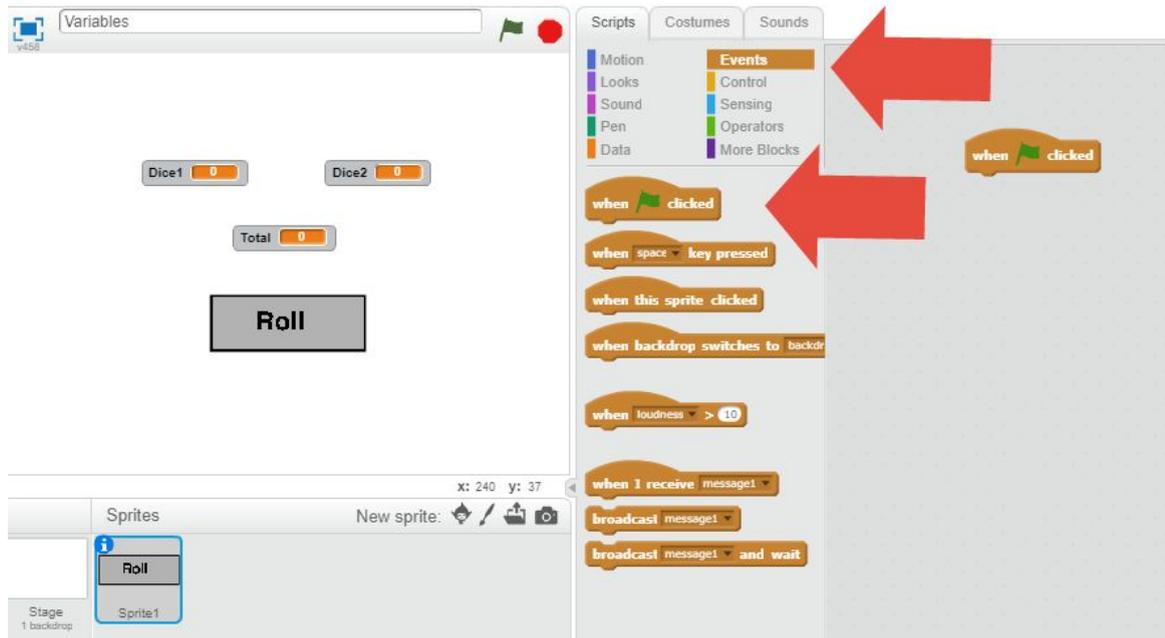
We can draw whatever we want our button to look like and this is all achieved by using the editor on the right. For this exercise I've just kept it nice and simple, sprites can always be redrawn anytime we wish so don't worry about getting it looking right first time.



## Step 3: Programming

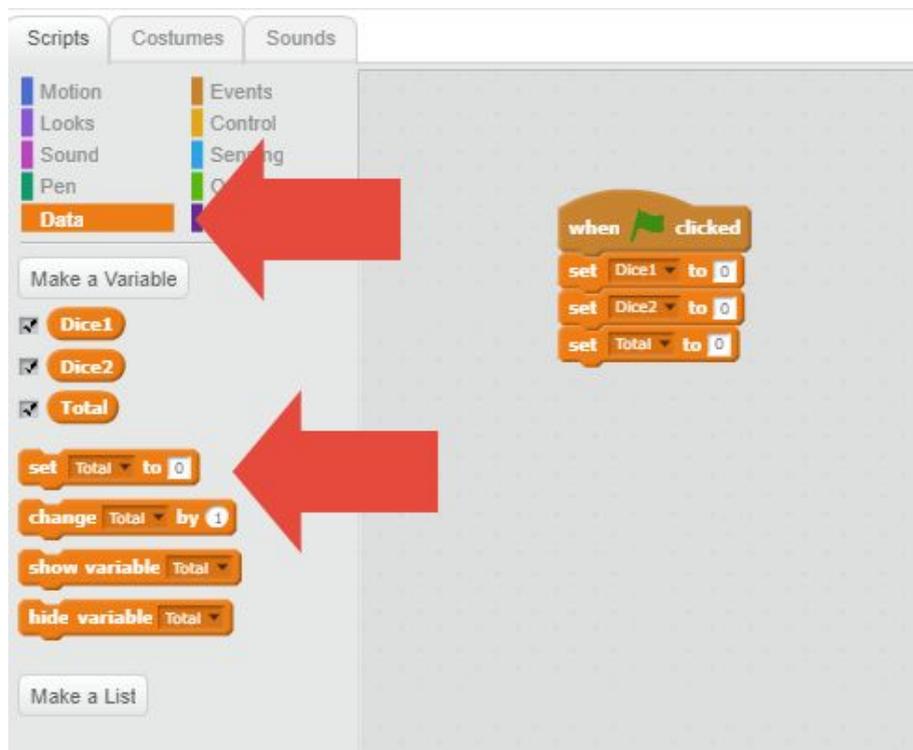
We now have all the elements in place to start building our dice game so let's get programming.

Making sure our button that we created in the previous section is selected, head on over to the Scripts section and drag into our work area the Green Flag block that's located in the Events section as shown:

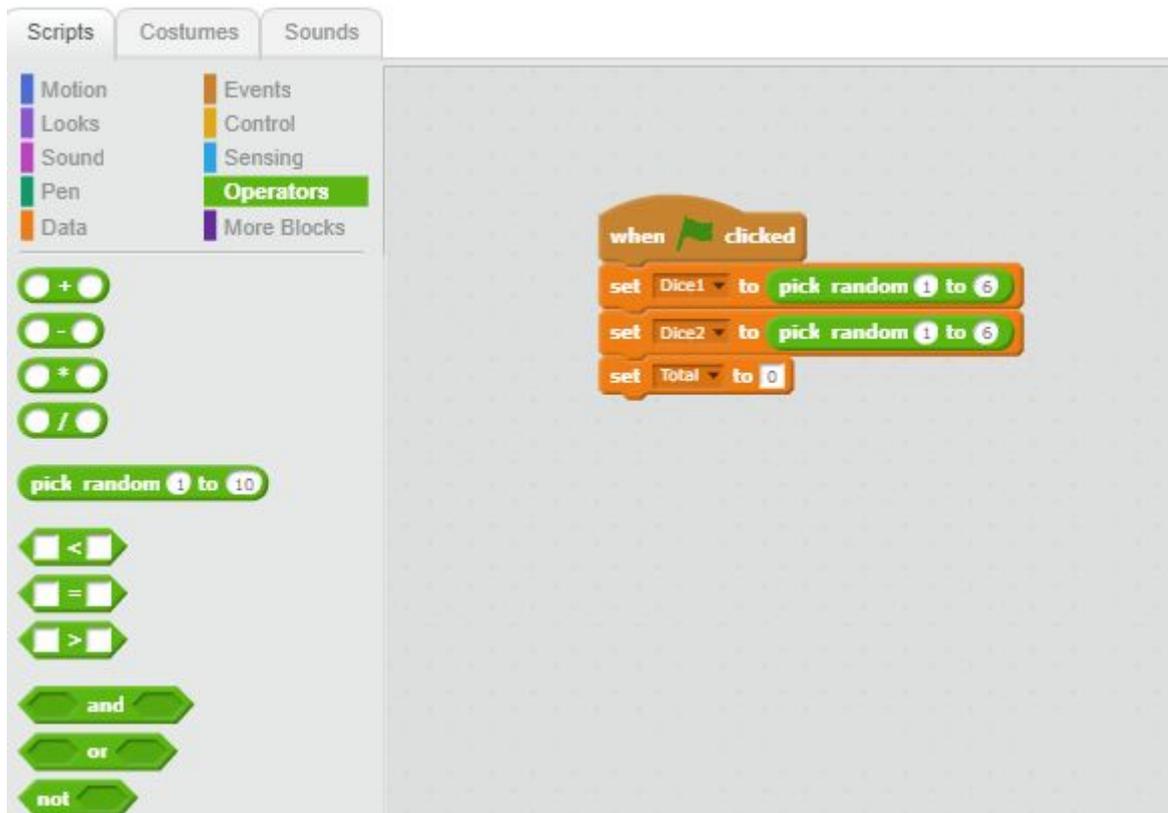


Every program written in Scratch starts with a click of the green flag. Whatever commands we now place under the Green Flag block will be carried out when the flag is clicked on. In the case of our dice game, we will first ensure that our program is rolling and calculating the dice correctly when the flag is click on.

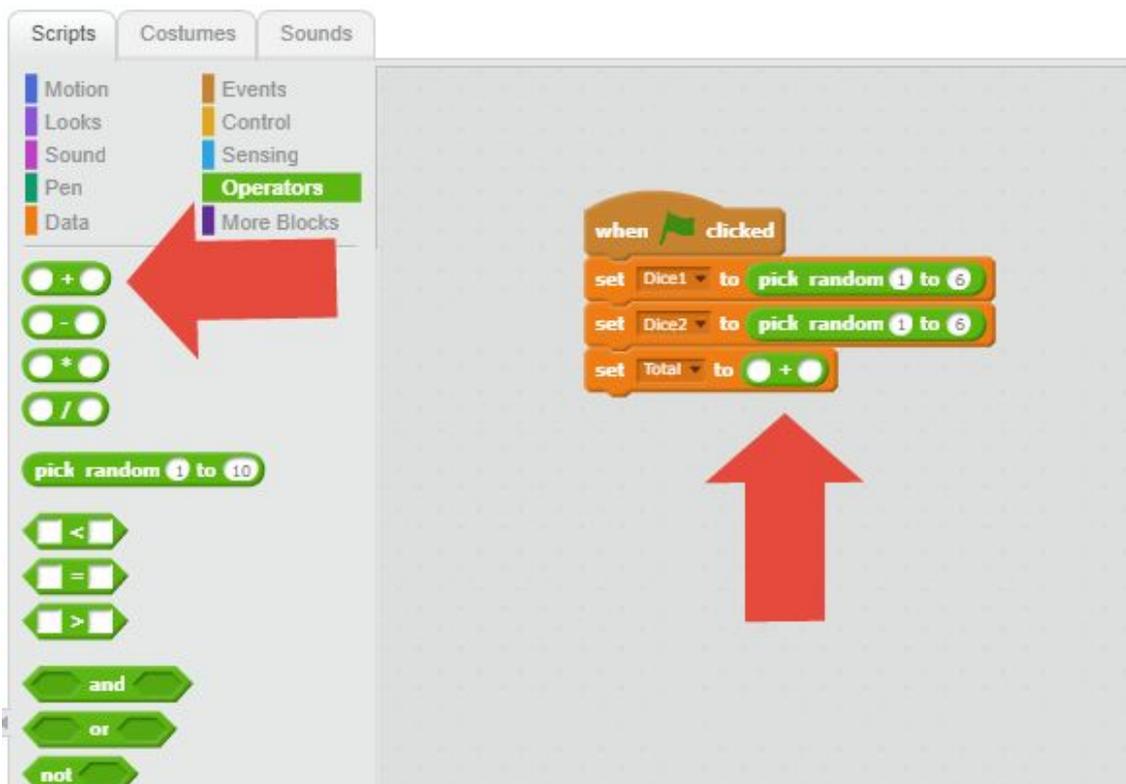
We do this by using the Set block found in the Data section as shown. Drag three of these blocks over and place them under the Green Flag block, selecting the correct variable from the dropdown in the block each time. It should look something like this:



In the Operators section you will see that there is a 'Pick Random' block. The block automatically picks a random number between any two values specified. Drag over this block and add it to both Dice1 and Dice2 as shown. As there are six values on a dice, instruct the block to only select random numbers between 1 and 6:

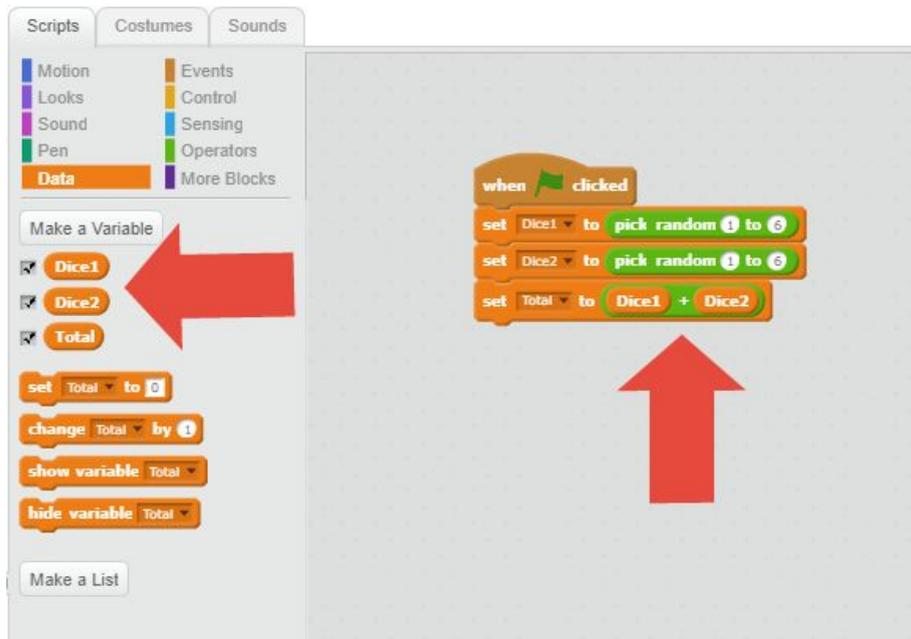


As we have random values for Dice1 and Dice2, our third variable called Total will be the sum of each dice. Drag in an Addition block as shown and place it in the 'Set Total' block:



As Scratch reads a program from top to bottom, the first two steps have already assigned values to our dice variables so whenever we need to refer to our Dice1 and Dice2 values all we need to do is reference the variables by name only.

We do this by looking in the Data section where we can see a list of all the variable we've created. Pick the variable(s) we need and drag them over into the Addition block as shown:



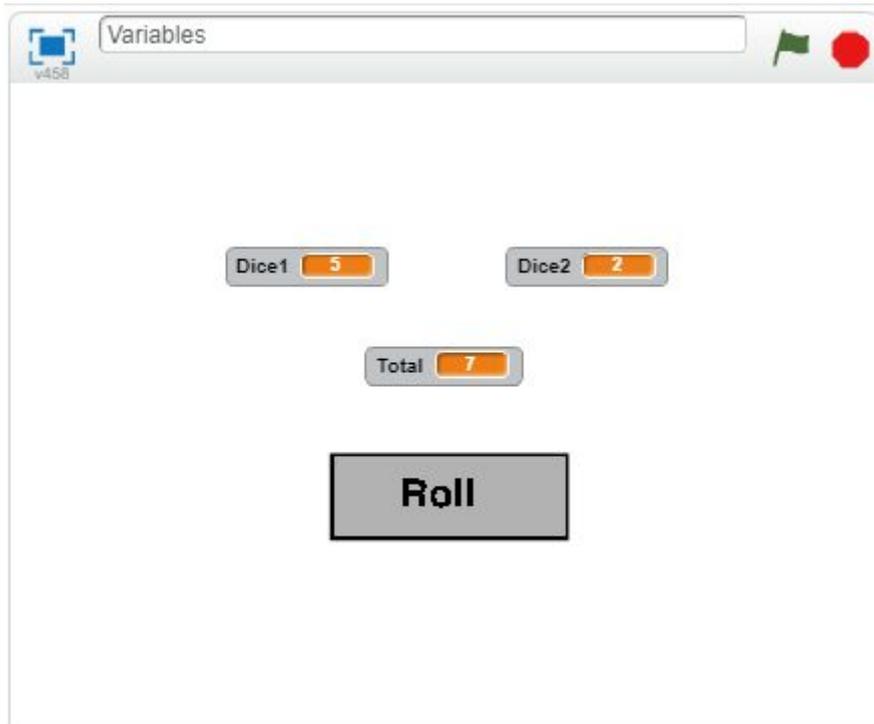
We now have our first program involving variables. You can give it a test by clicking on the green flag. Each time the green flag is clicked you should see random dice values being assigned to Dice1 and Dice2 as well as the Total keeping a record of their combined value.

#### Step 4: Involving the Button

We now want to involve the Roll button we created. First swap out the Green Flag block and replace with the 'When Sprite Clicked' button as shown below. Now our Roll button comes into play.

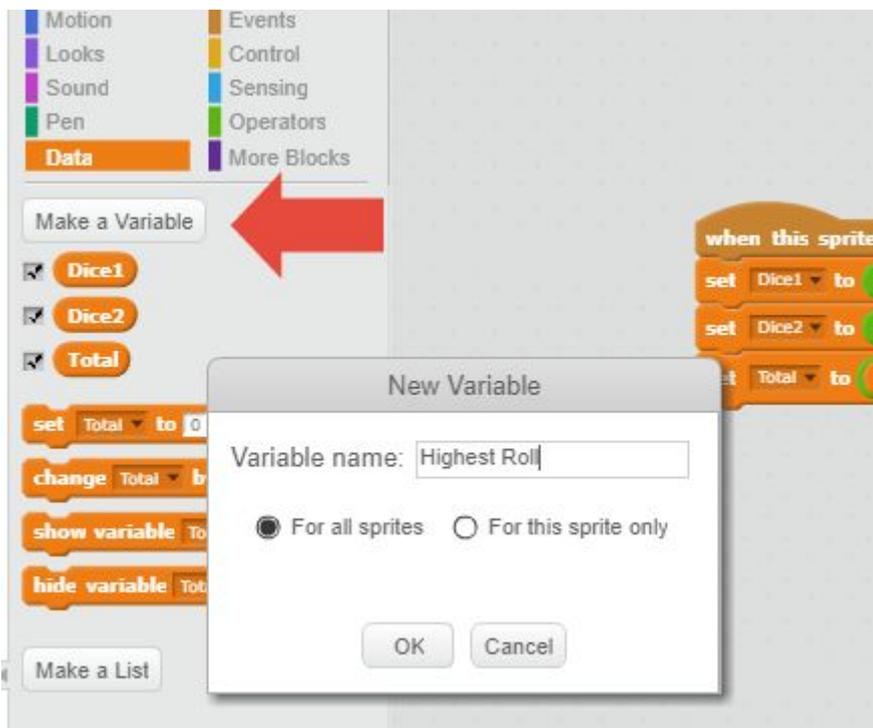


You can test the program at anytime by clicking on the Roll button in the work area.

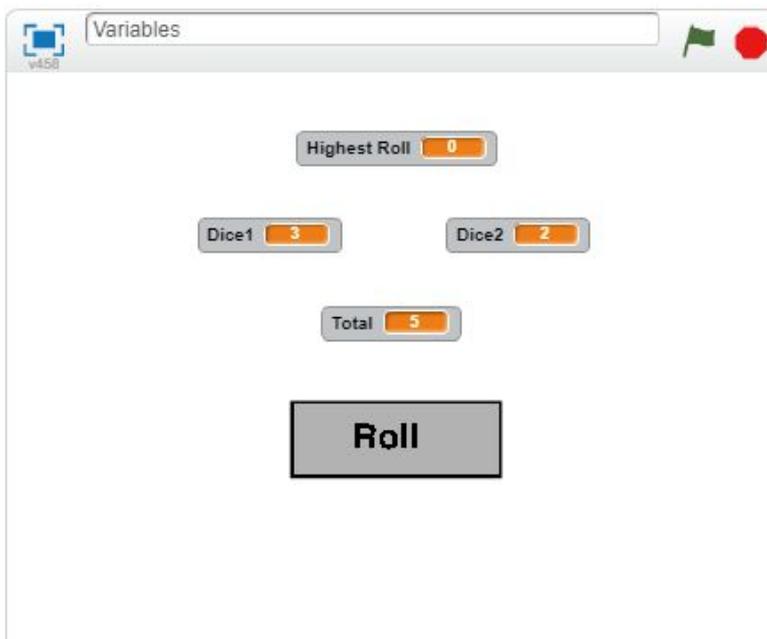


### Step 5: Recording the Highest Total

Now let's keep a record of the highest total rolled. To be able to do this we first need to create another variable that will record the highest total achieved. Create another variable just like before by heading over to the Data section. Call it Highest Roll.

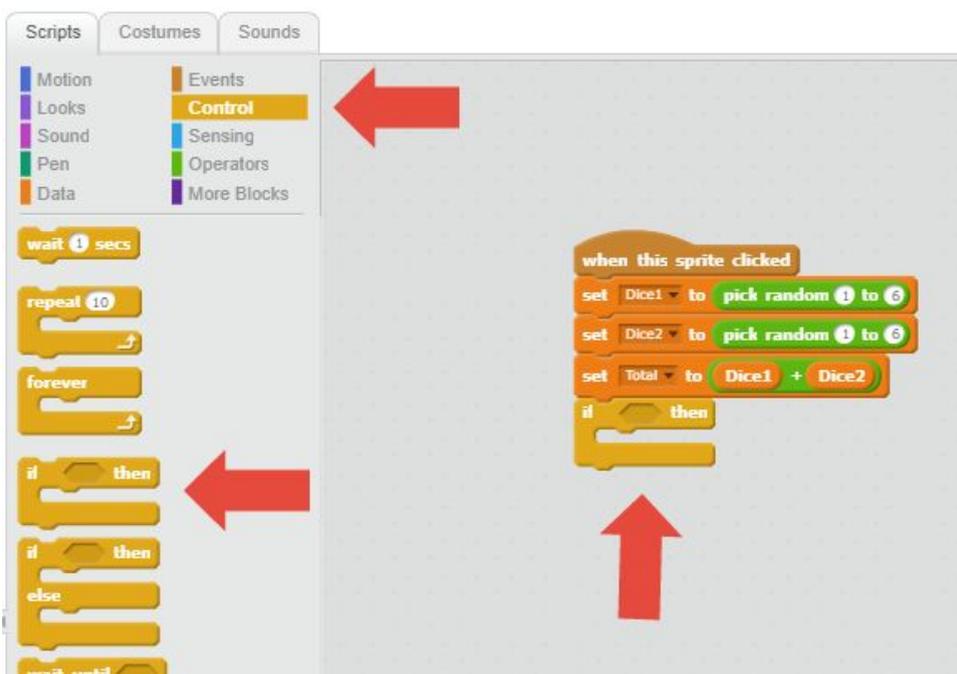


Position it so that it's not getting in the way of anything else in our work area.

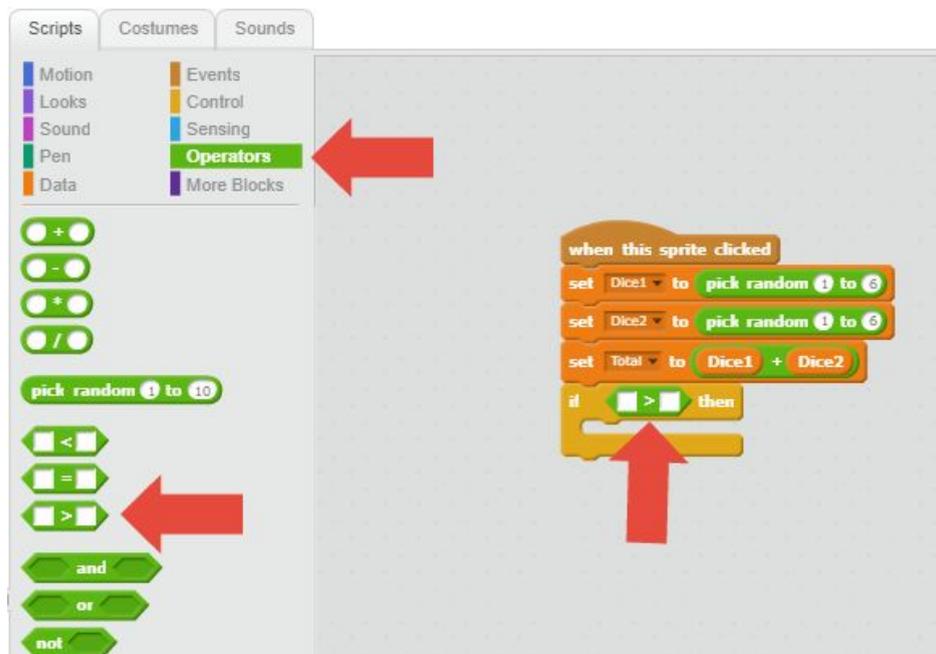


To work out whether or not a roll of the dice is the highest value we need to add to program we've already built in the previous steps. Within the Control section there is the IF block, this block allows us to run certain parts of a program ONLY IF a condition has been met, in this case "Has the most recent roll of the dice beaten the previous highest total?"

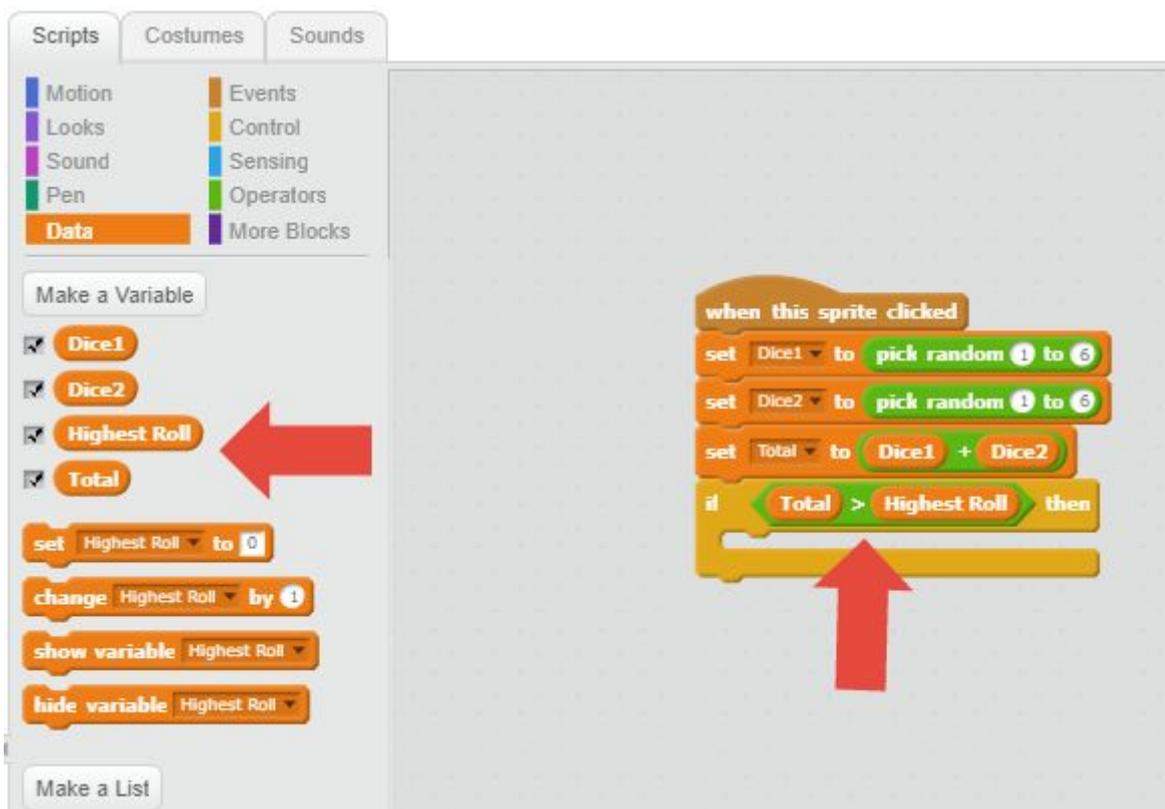
Drag the IF block over and position it under our other blocks as shown:



In the Operators section there is a Greater Than block and it is this that we'll use to work out whether our condition is true or not, have we achieved the highest roll? Drag it into the IF block as shown:

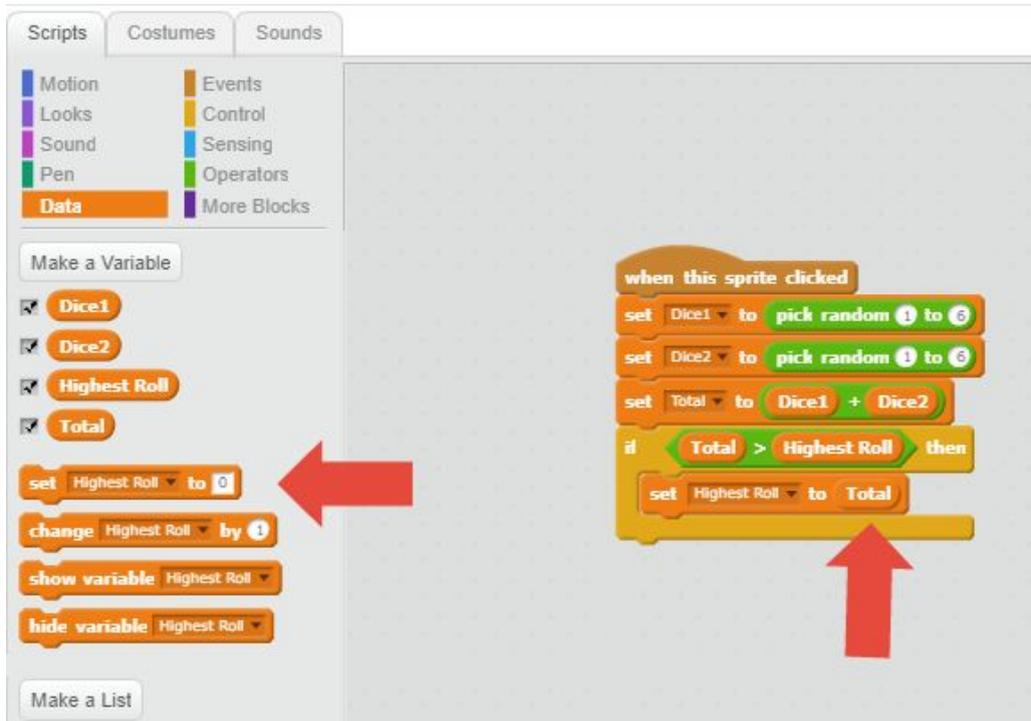


We will be comparing two variable values to determine whether our condition is true. Find the list of our variables in the Data section and drag over the Total variable and Highest Roll and position them within the 'Greater Than' block as shown. We are basically asking is our current total greater than the highest total?

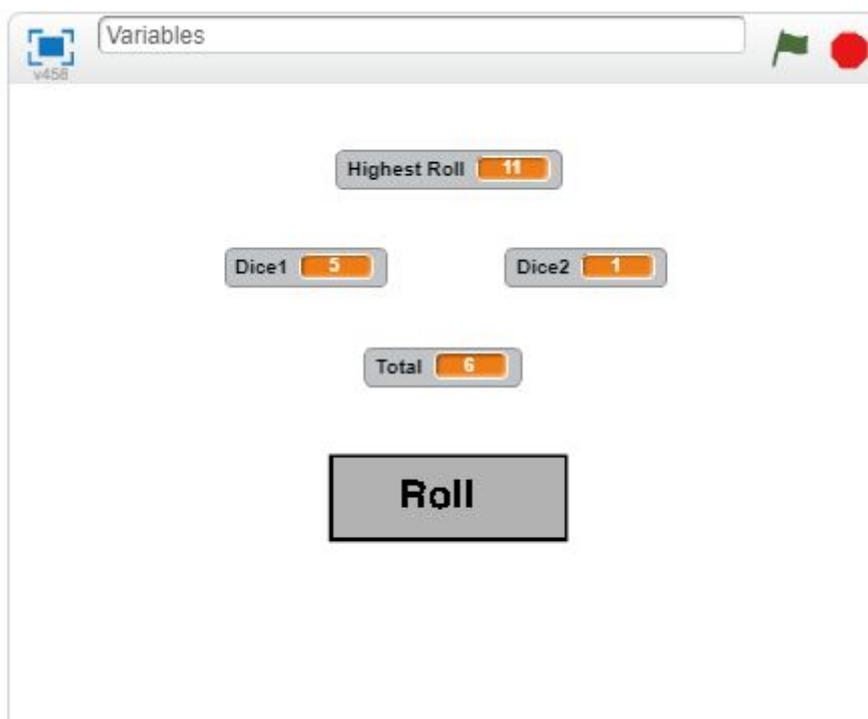


If we have been lucky enough to roll the highest total then whatever blocks we place within the IF block will ONLY run if our condition has been met. If we haven't rolled the highest total then the blocks within the IF block is ignored.

Upon rolling the highest total we need to update the Highest Roll variable value with the current Total value. This is achieved by using a Set block that we've previously used. In this case we're setting the Highest Roll value to Total as shown below:



Now when we roll a high number our Highest Roll will record it and only change if we manage to roll a higher value. If it's a lower value then it will ignore it.



## Things to think about.

How do we reset the Highest Score variable to zero along with the other variable values from the previous game? Clue: Remember how we started with the When Green Flag Clicked block? Maybe we could bring that back into play.



How would you add a visual indicator to alert us that we've achieved the highest roll?

---