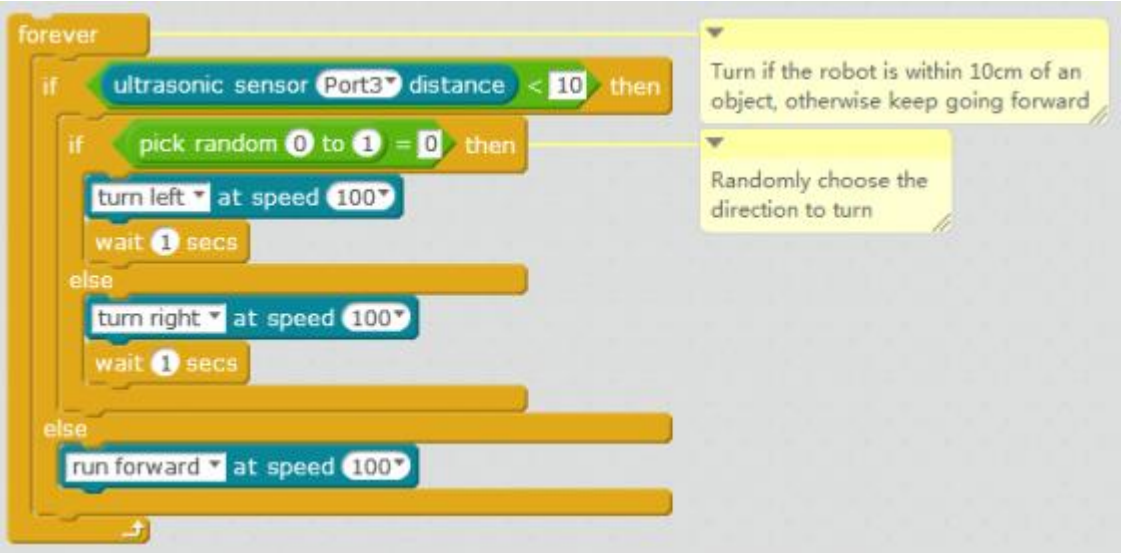
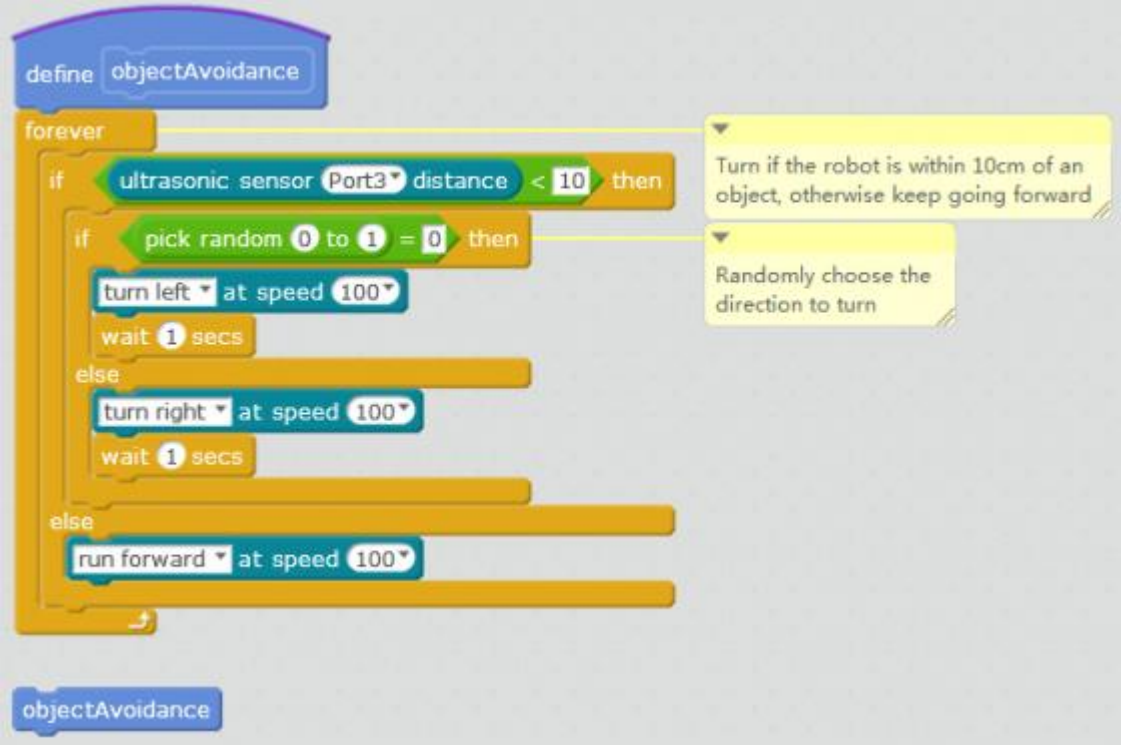
Object Avoidance with Functions and Parameters

Functions are a useful feature to make your code easy to read and well-organized. The real power of functions comes with parameters though. In this tutorial, we will explore that power by re-writing the object avoidance program using functions and parameters.

This is the original object avoidance code:



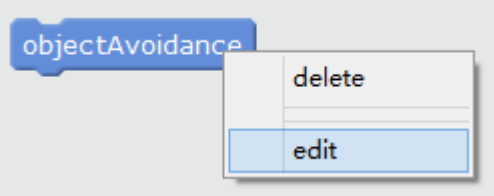
I am going to now put that code in a new function (New Block):



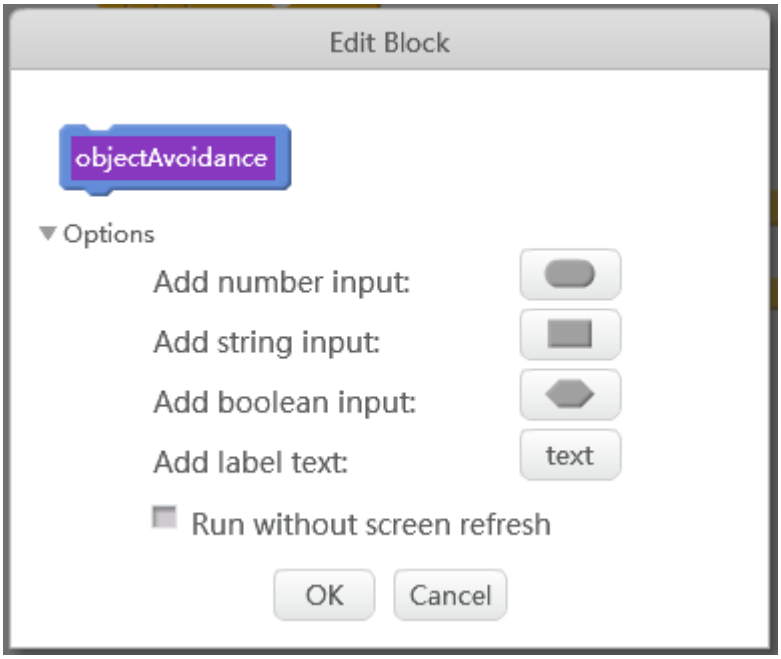
I can call the function with the objectAvoidance block at the bottom of the picture above. Now what if I wanted to run this code, but instead of a motor speed of 100, I want 200? Or

150? Or 255? This is where parameters come in. A parameter allows a value to be passed to the function.

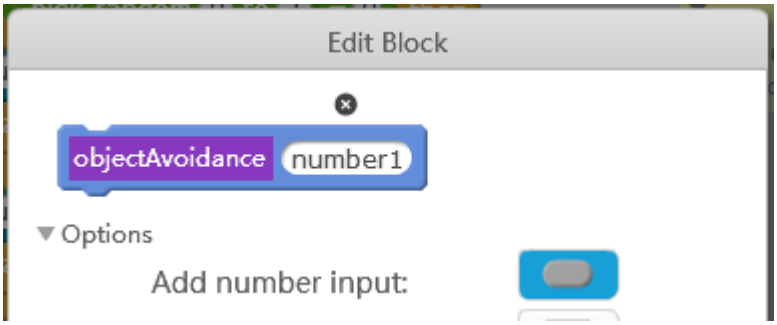
To pass a parameter in mBlock, you need to either create them when you create the function or after you've created your function, right-click on any of the function blocks and select 'edit':



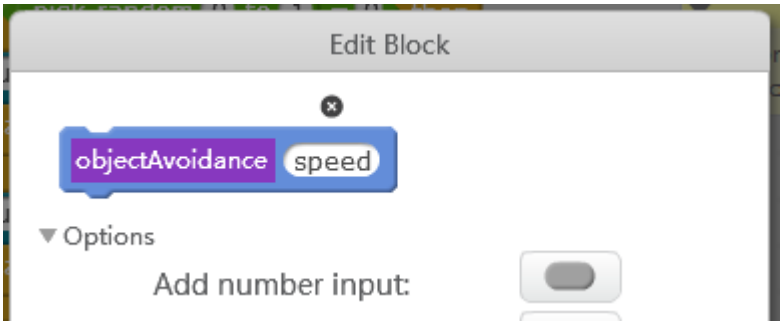
Then go into 'Options':



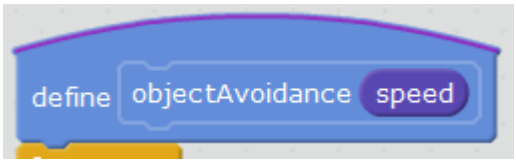
This is where you can add parameters. When you add a parameter, you need to specify the type of parameter: number, string or Boolean. A number is just a number, a string can be numbers or letters and Boolean is either true or false. In my example, I want my function to run the program with different motor speeds. As motor speeds are a number, I will choose to 'Add number input'.



As you can see this creates a new input (parameter) in the block at the top. It names it “number1”, but as this will represent the speed in my program, I am going to change the name to “speed”:

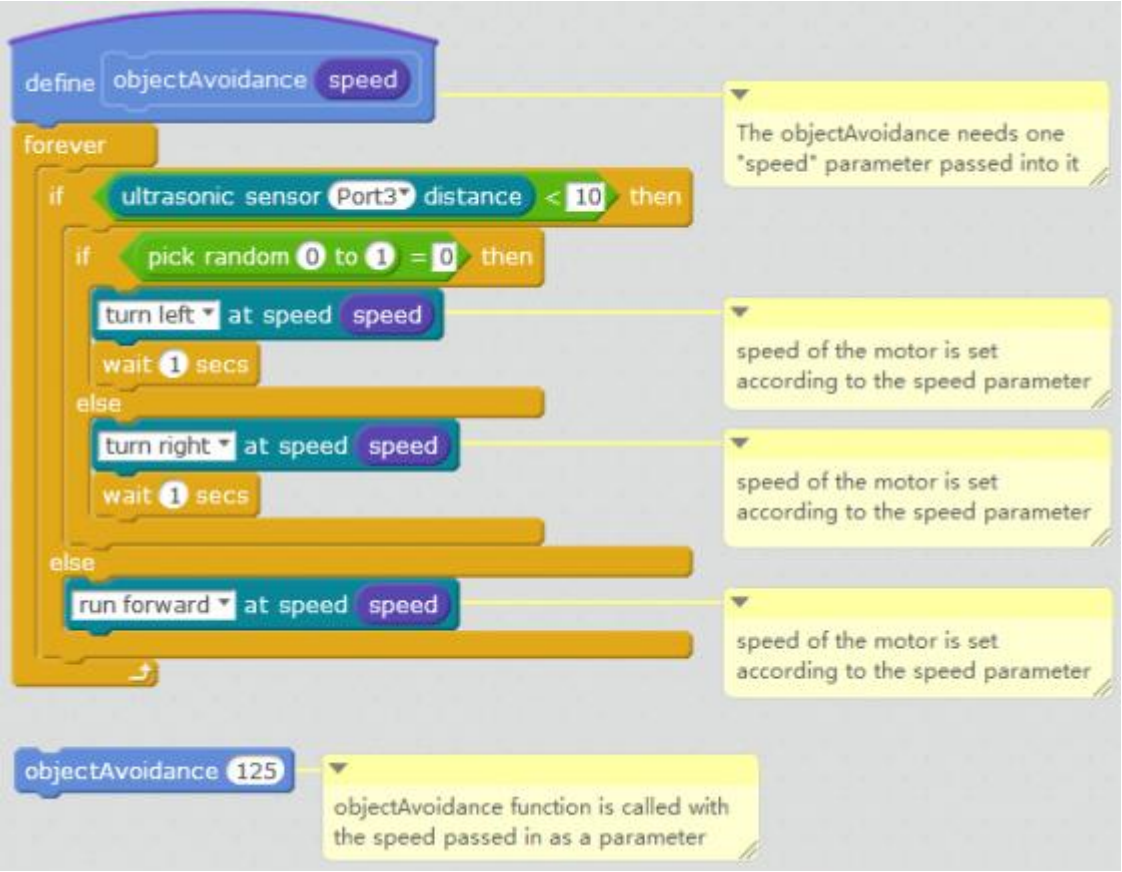


Now, when I press ok, the function has the parameter in its definition block:



And its call block:

The only thing that remains for us to do is use the speed parameter when we set the speed of the motors. We can do this by dragging and dropping the speed parameter from the objectAvoidance definition block into the places where we set the speed. The final function looks like this:



At the bottom you can see the objectAvoidance function is called and it passes in a speed of 125.

Challenges:

1. Pass in different values of speeds and fun this function.
2. Add another parameter that controls the direction of the turn – 0 for left, 1 for right, 2 for random.
3. Write another program that uses functions and parameters.