

Scratching Deeper

Methods and Parameter Passing

Produced Mairead Meagher
by: Dr. Siobhán Drohan



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
<http://www.wit.ie/>

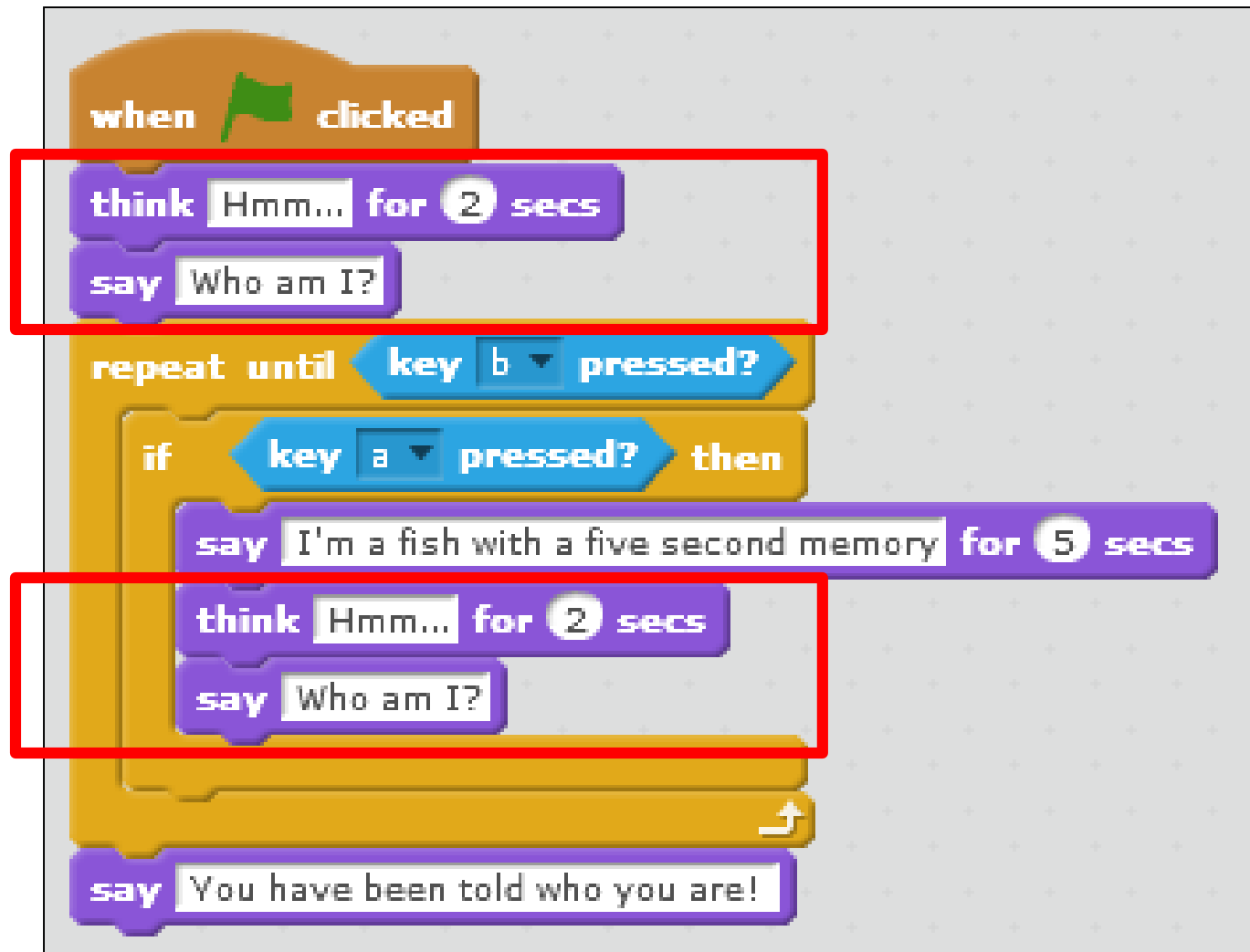
Topics list

- **Recap: Methods**
- Passing Parameters into Methods.
- An Example from Scratch website.

Recap: Methods

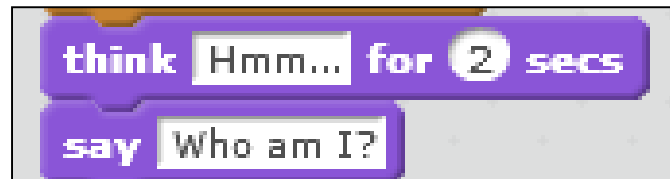
- Methods are “a collection of statements that are grouped together to perform an operation”.
- When you call a method, the program transfers control to the method and its statements are run.
- When a method is finished running, control is returned back to where the method was called from.

Recap: We have code duplication



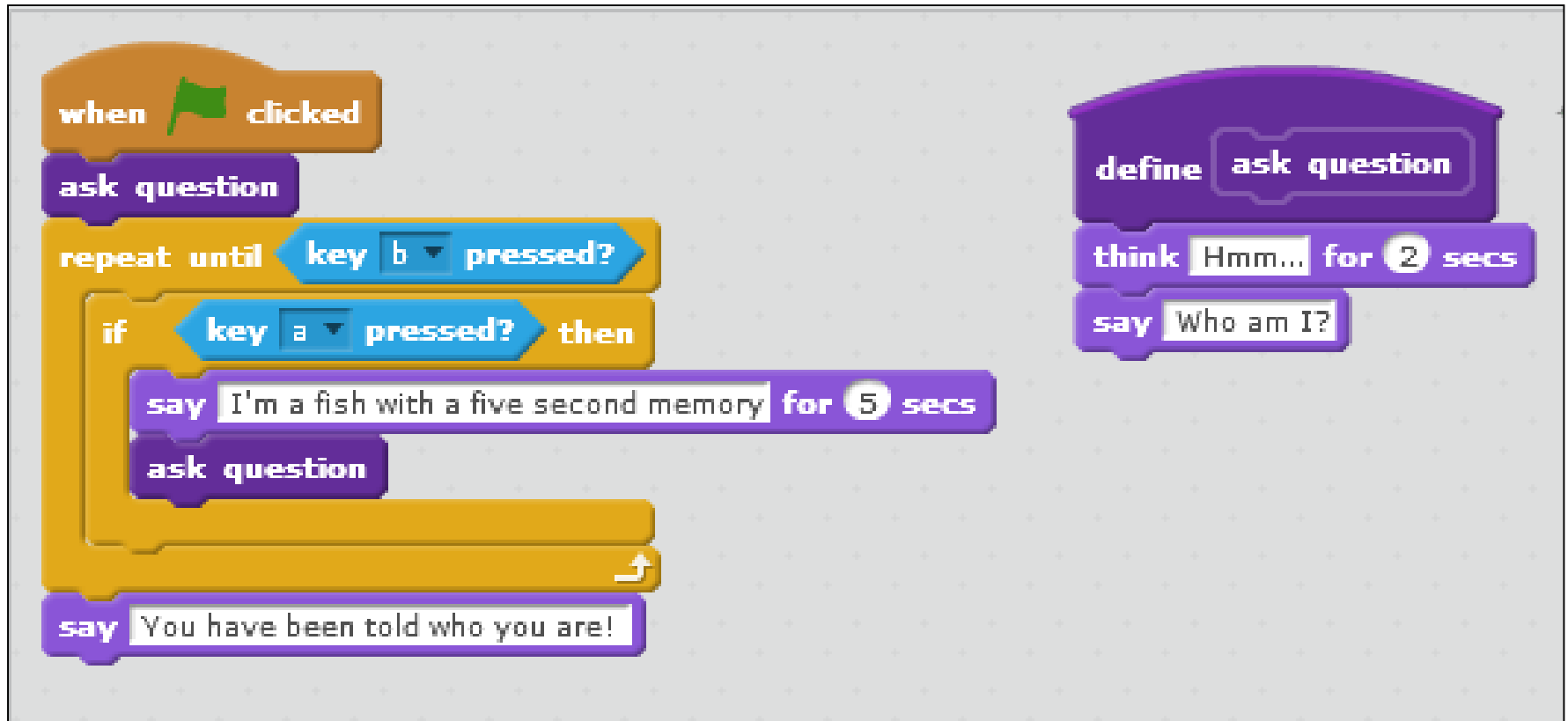
Methods and Code Duplication

- We can use methods to cut down on code duplication.
- In our example, we will move this duplicated code into one method:



- And where the code used to be, we will call the method instead.

Recap: SomethingFishy6

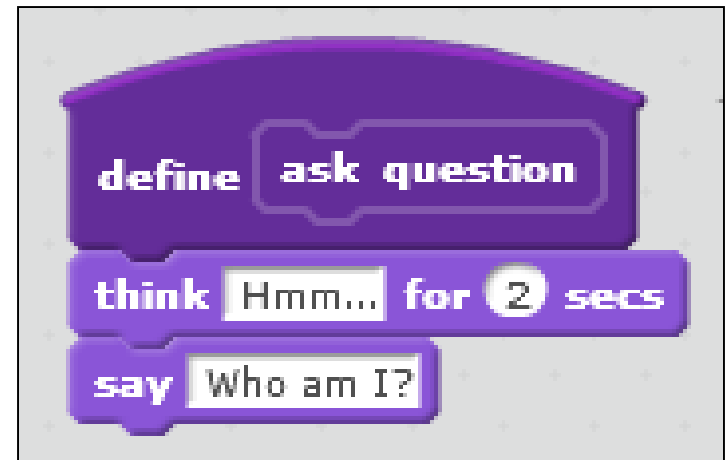


Topics list

- Recap: Methods
- Passing Parameters into Methods.
- An Example from Scratch website.

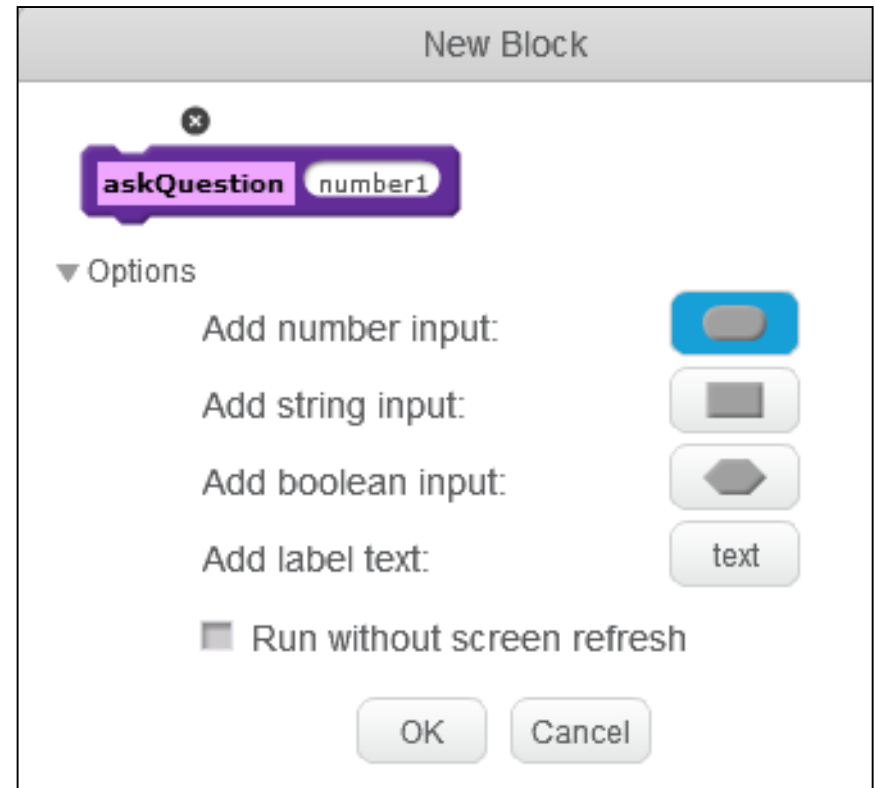
Passing information into a method

- When called, our method will execute its two statements.
- The method always *“thinks”* for 2 seconds.
- But what if we wanted to *“think”* for a variable amount of time?



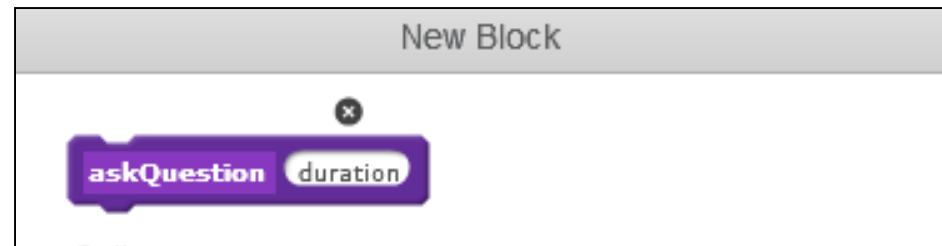
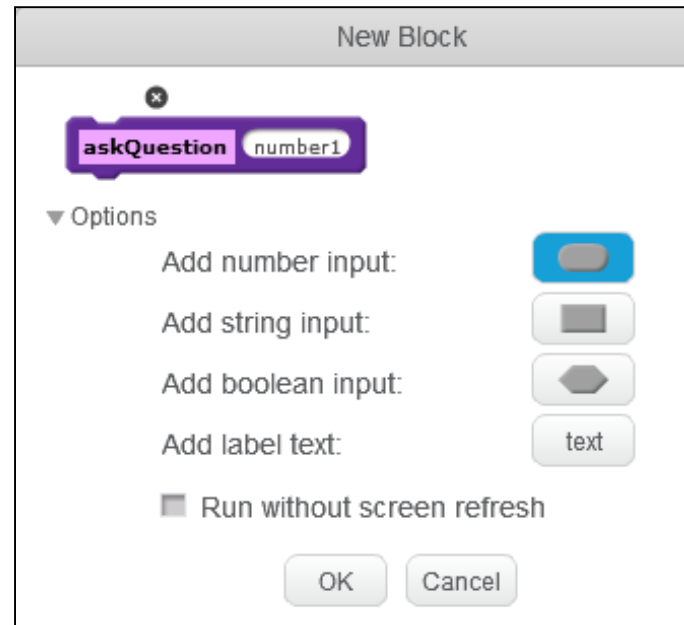
Passing information into a method

- We can pass information, of different types, into a method.
- When creating a **New Block** in Scratch, click on the **Add number input** option.



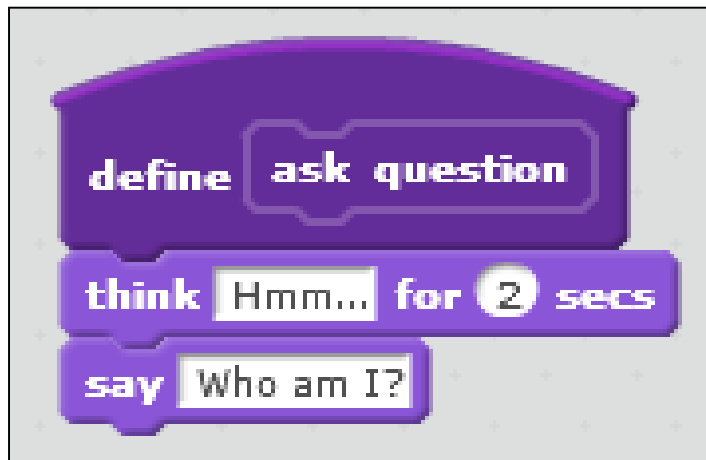
Passing information into a method

- A variable called **number1** is added to the method name.
- We can rename this to any name we wish e.g. **duration**.



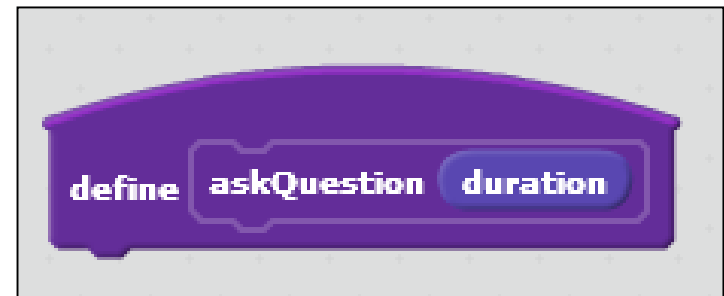
Passing information into a method

Old version of the method



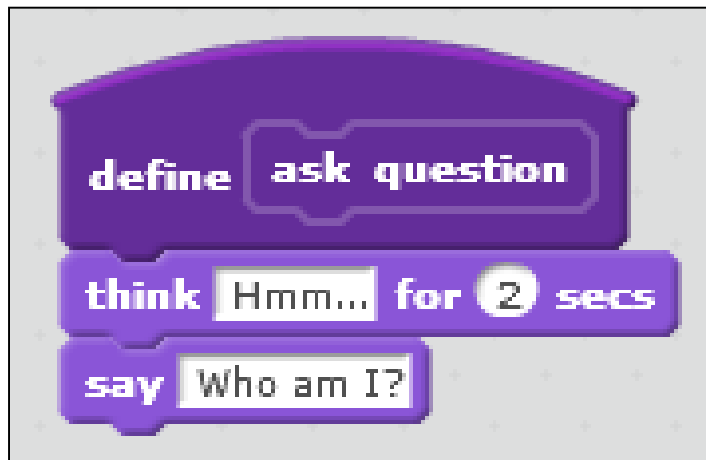
New version of the method (so far)

A **duration** is passed into the method.

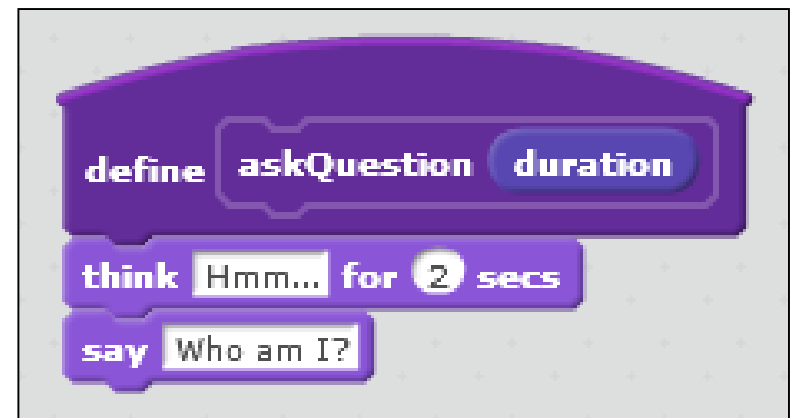


Passing information into a method

Old version of the method

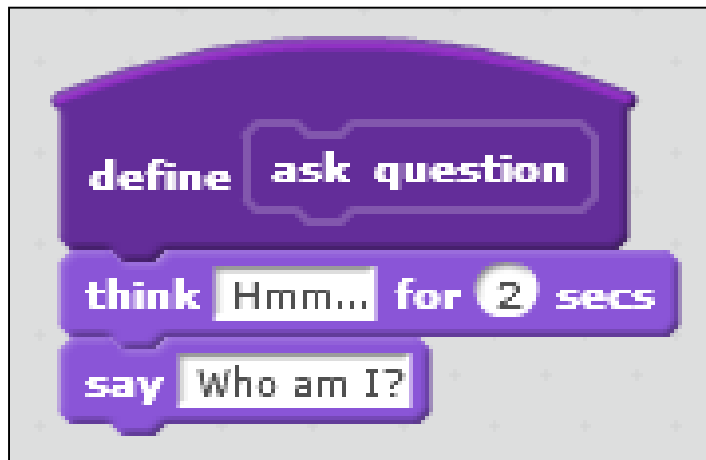


Adding the two statements to the new version of the method, we have:

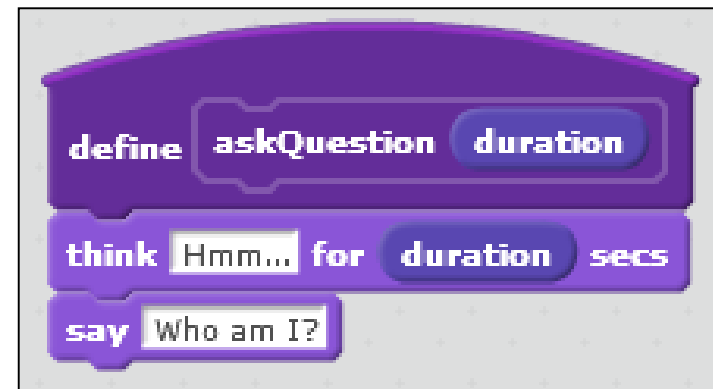


Passing information into a method

Old version of the method

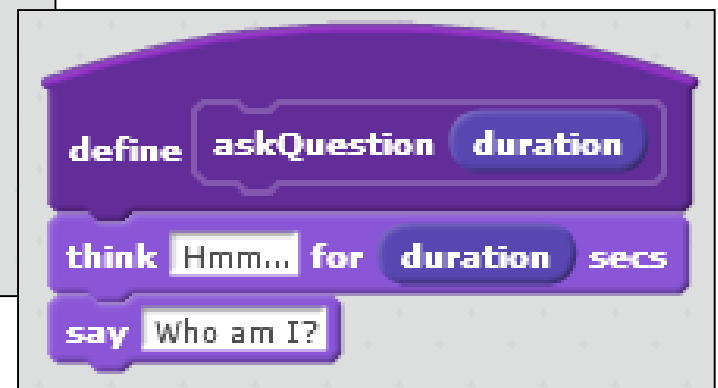
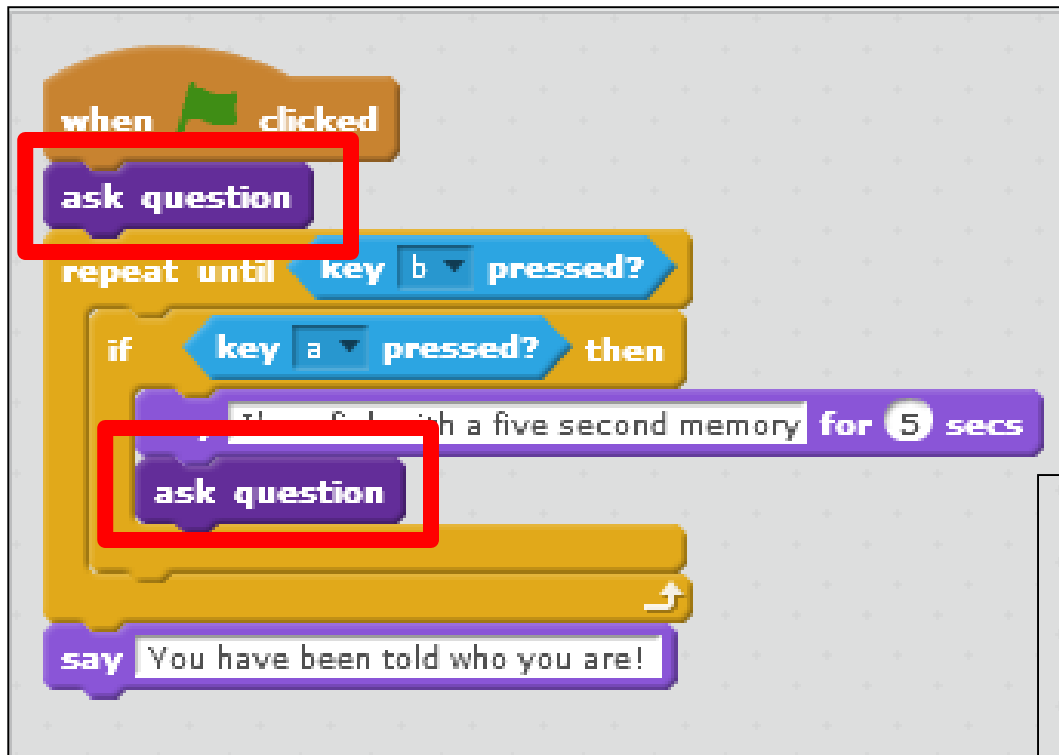


Instead of “thinking” for 2 seconds, our fish will now think for the **duration**.

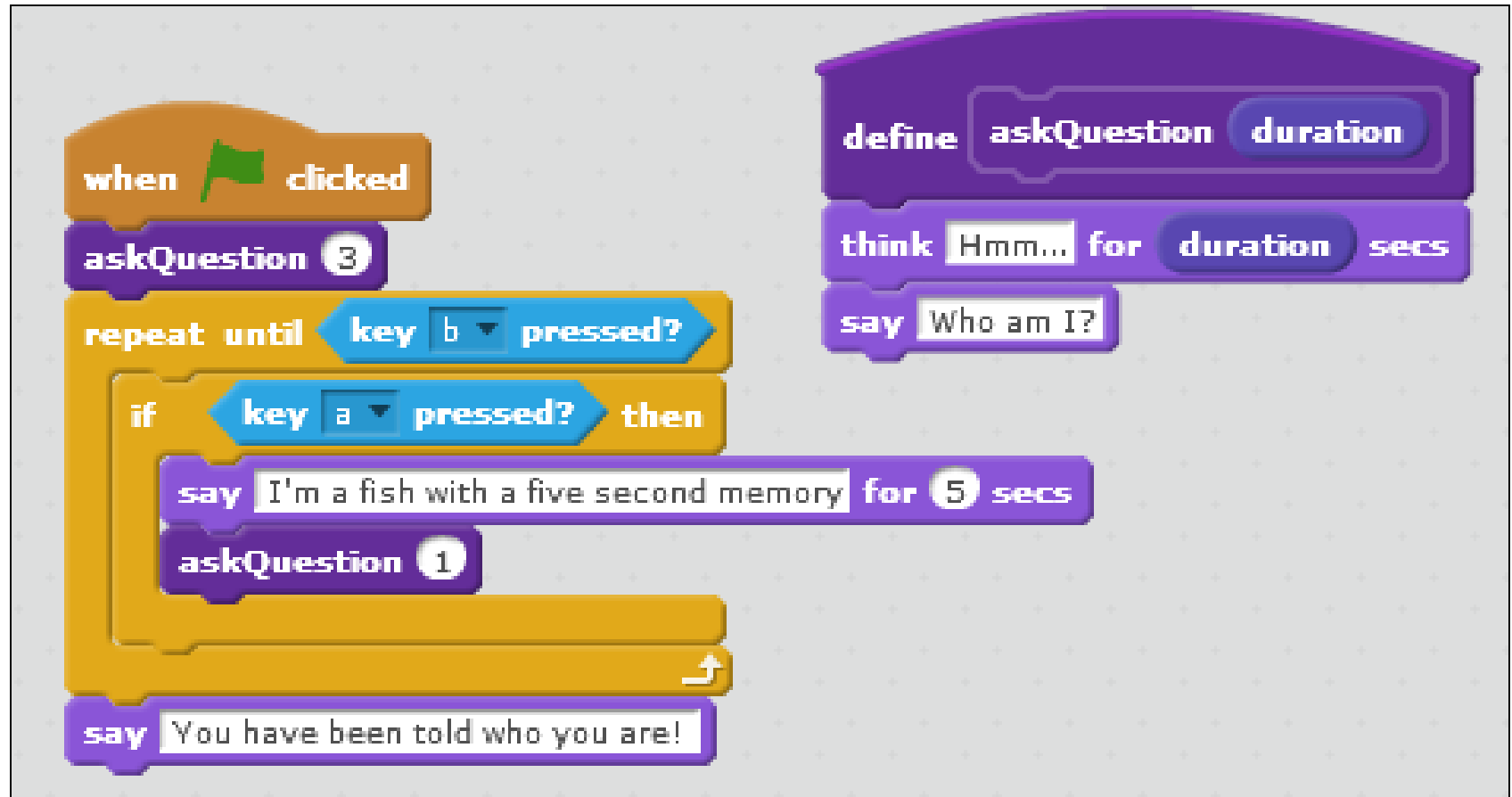


Passing information into a method

- Now that our method is rewritten, we now have to refactor our code to use it:



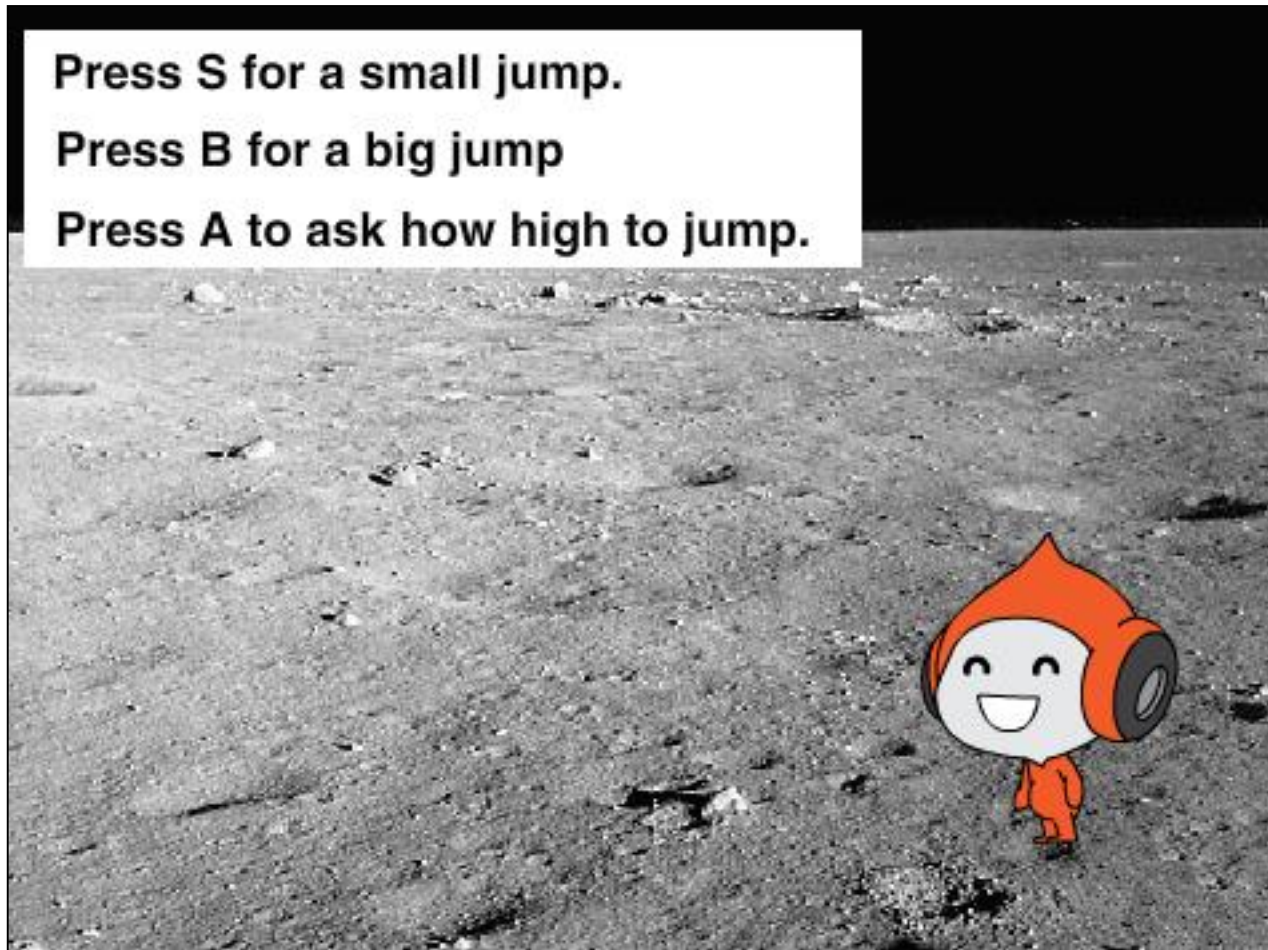
SomethingFishy7



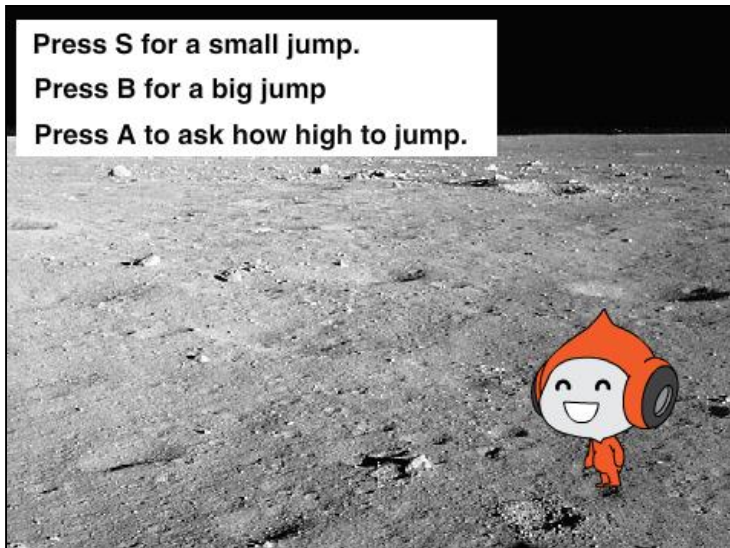
Topics list

- Recap: Methods
- Passing Parameters into Methods.
- An Example from Scratch website.

PicoExample



PicoExample

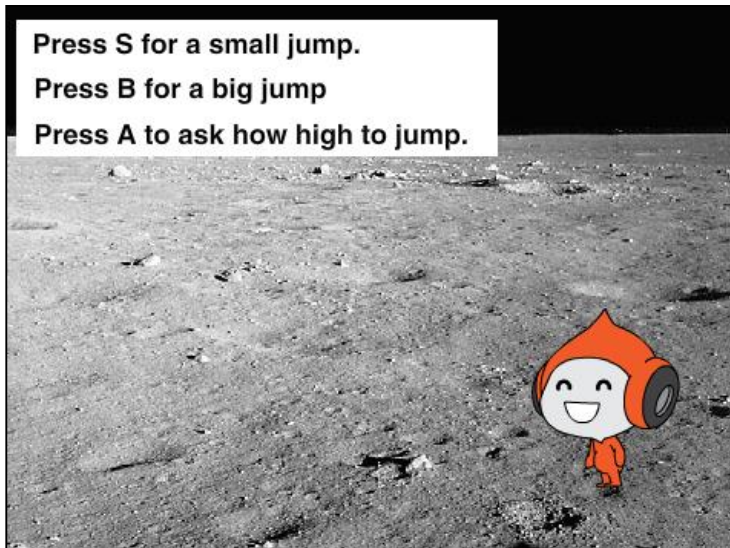


Regardless of which key is pressed (S, B or A), Pico will jump up a given amount of pixels and return to his starting point.

We could write this part of the program as a method.

The method could be called **jump** and the number of pixels to jump could be passed as a parameter.

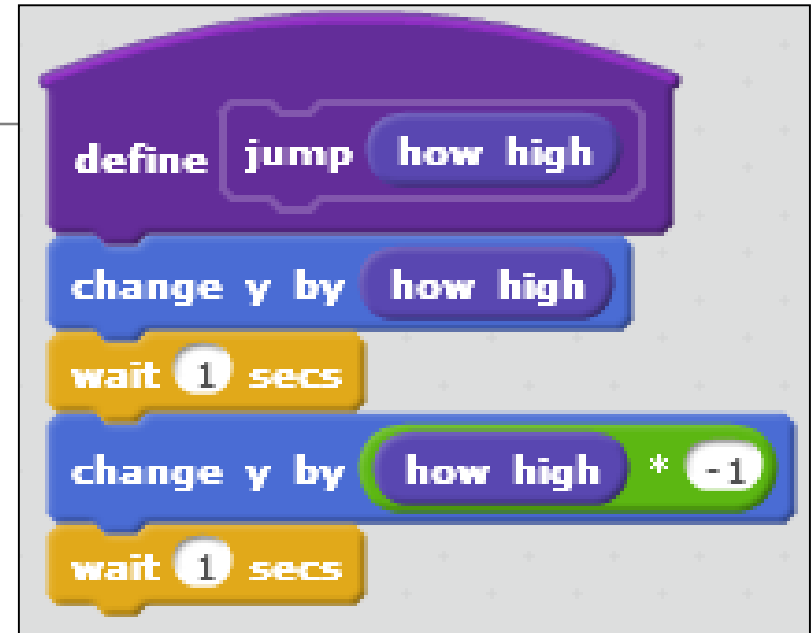
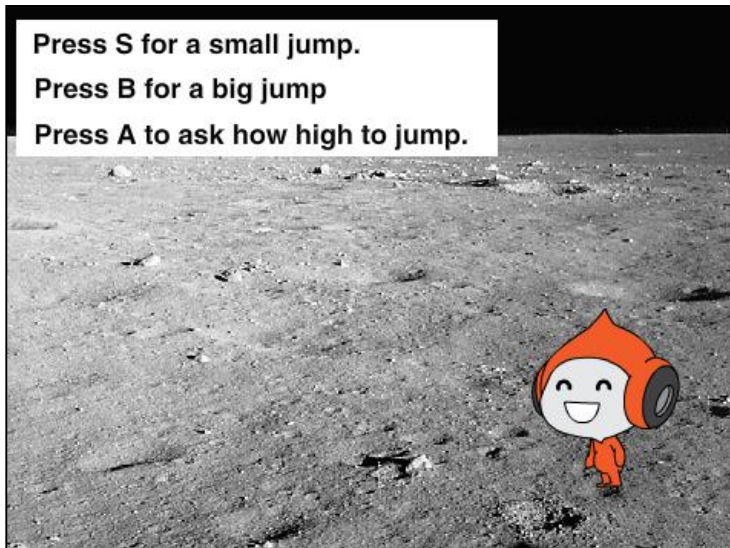
PicoExample



The processing in the method should:

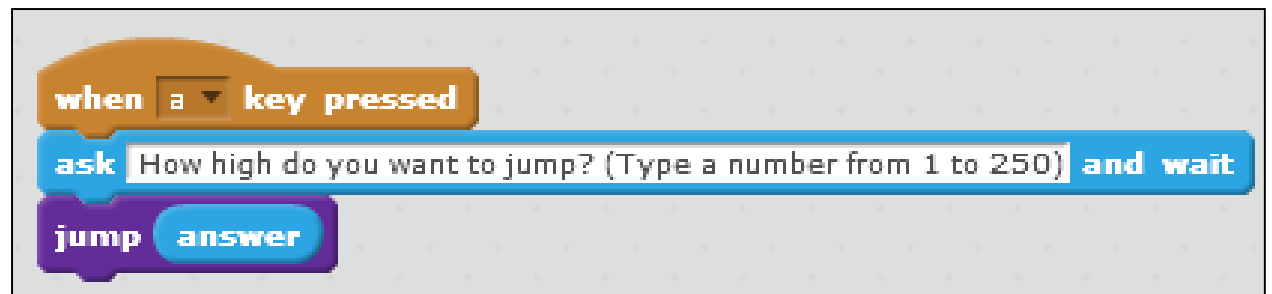
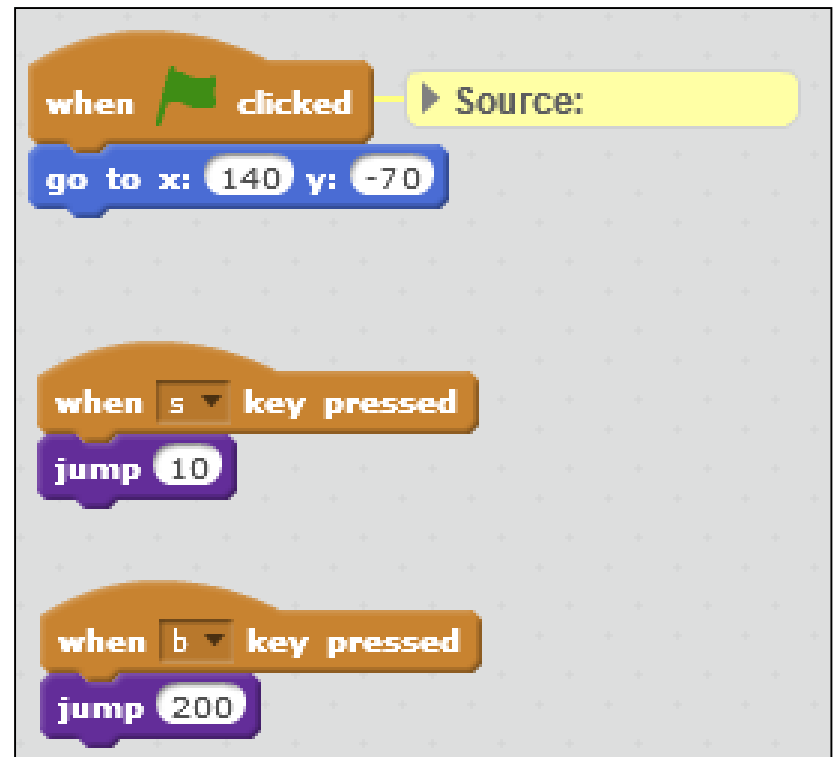
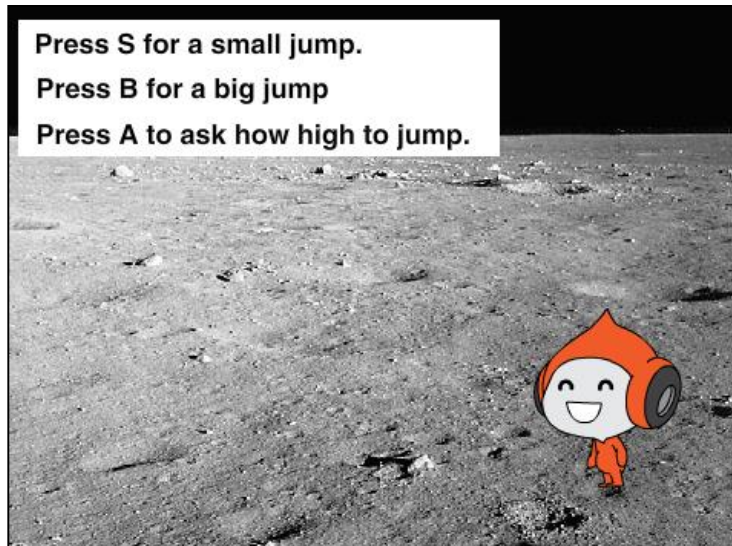
1. Change the y value for Pico by the value passed in as a parameter i.e. **how high**.
2. Change the y value for Pico by the negative of the value passed in as a parameter i.e. return to his starting point.

PicoExample



Now that the jump method is written, the next step is to call it based on the key pressed.

PicoExample



PicoExample – Complete Code

The image displays a Scratch script for a game character named Pico. The script is organized into two main sections: a main execution area on the left and a function definition area on the right.

Main Execution Area (Left):

- When clicked:** A "when green flag clicked" event block followed by a "go to x: 140 y: -70" block.
- When s key pressed:** A "when key pressed" event block for the 's' key followed by a "jump 10" block.
- When b key pressed:** A "when key pressed" event block for the 'b' key followed by a "jump 200" block.
- When a key pressed:** A "when key pressed" event block for the 'a' key followed by an "ask" block with the text "How high do you want to jump? (Type a number from 1 to 250)" and a "wait" block, followed by a "jump answer" block.

Function Definition Area (Right):

- define jump how high:** A function definition block for "jump" with the parameter "how high".
- change y by how high:** A "change y by" block with the value "how high".
- wait 1 secs:** A "wait" block for 1 second.
- change y by how high * -1:** A "change y by" block with the value "how high * -1".
- wait 1 secs:** A "wait" block for 1 second.

Questions?

