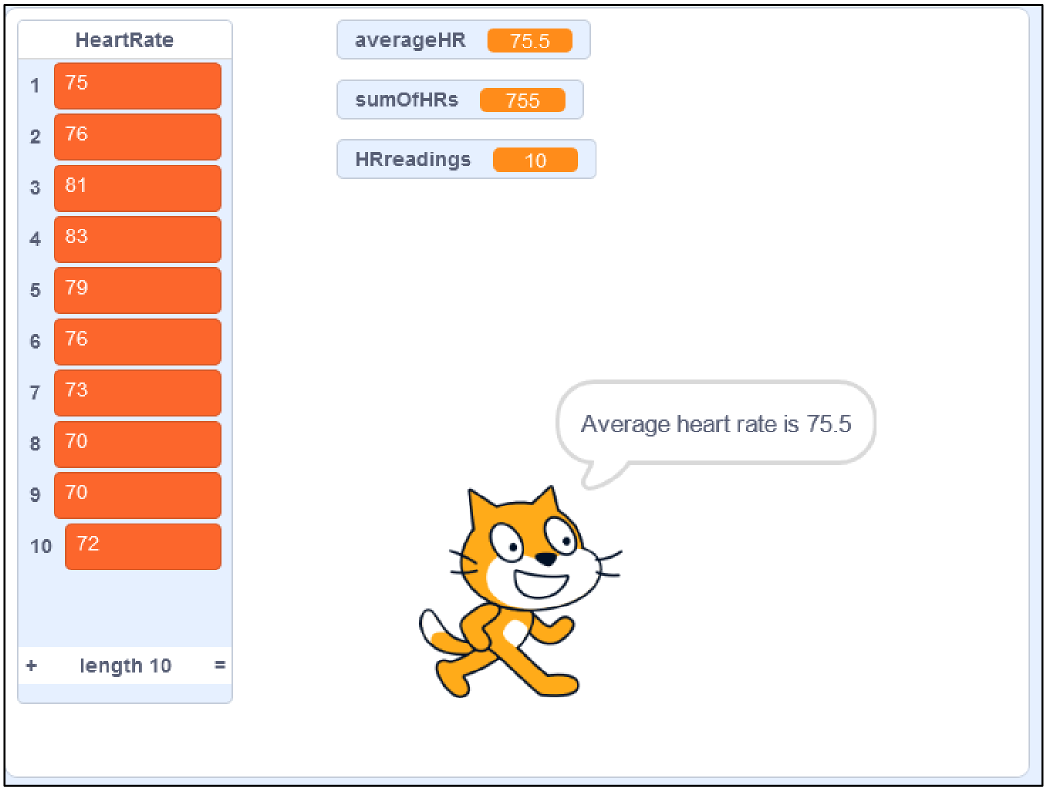
STORING AND PROCESSING HEART RATE DATA

In this activity you will create a program that will:

- store data related to a person's heart rate; and
- calculate a person's mean heart rate.

When the program is complete, the following will be displayed on the screen after we execute the program:



	HeartRate
1	75
2	76
3	81
4	83
5	79
6	76
7	73
8	70
9	70
10	72

+ length 10 =

averageHR 75.5

sumOfHRs 755

HRreadings 10

Average heart rate is 75.5

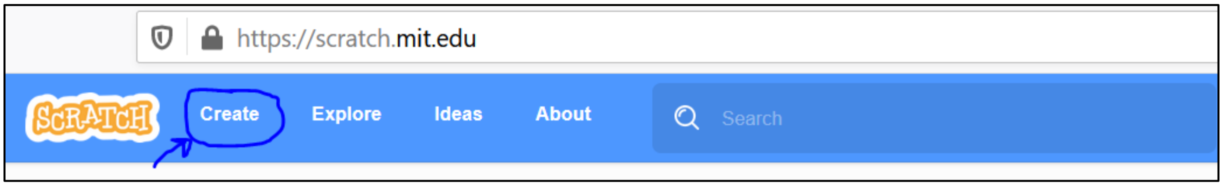
The guided steps below will take you through the process of creating the program.

Read the steps carefully, and try to understand how each step is important to the program.

If you make a mistake and your program doesn't run properly… don't panic.

This is just a normal part of writing and debugging a program.

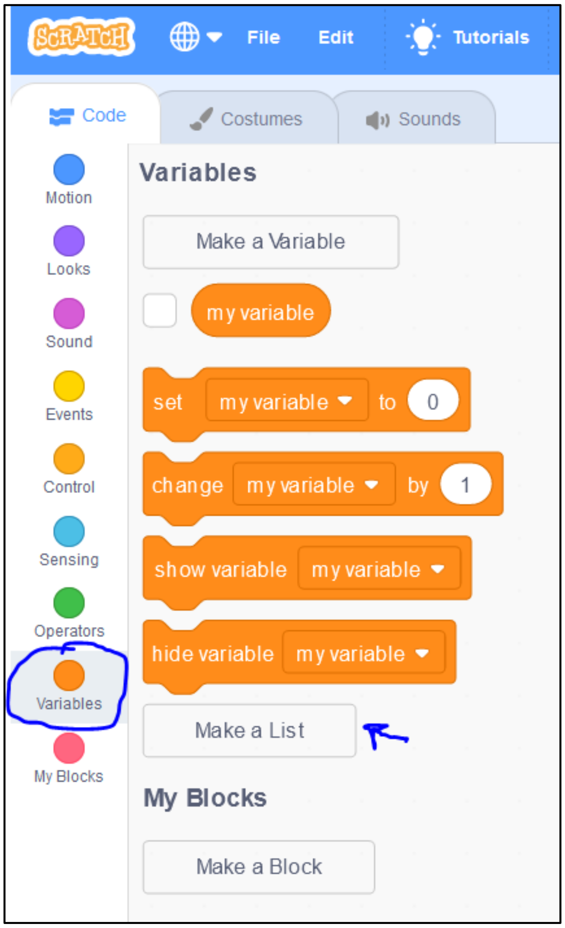
1) Go to the Scratch website and create a new project:

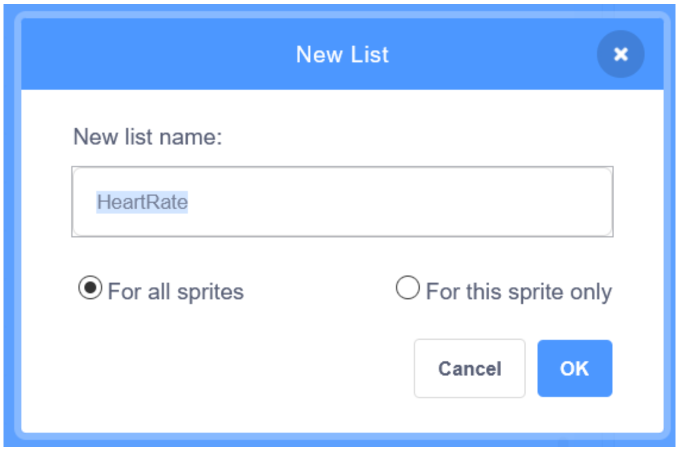


https://scratch.mit.edu

S. & L. Floyd

2) Our first step is to create a list in our program that will store our data.
For this project, our data will include ten different readings of our heart rate.
You can create a list by clicking **Variables -> Make a list**
Name your list **HeartRate** and click **OK**



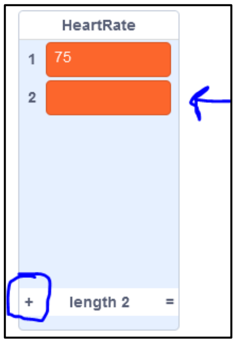


New list name: HeartRate

● For all sprites ○ For this sprite only

3) Now let's add our heart rate values into our list. You might have recorded your own heart rate measurements over the last day or two but if not, you can use the values that have been entered here.

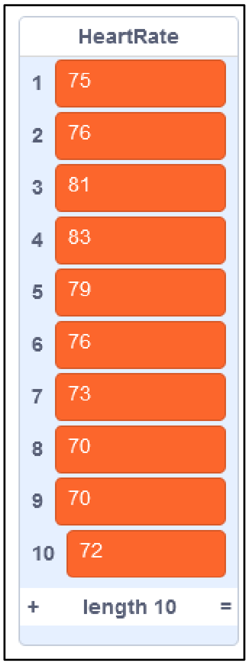
To enter additional values click the "**+**" symbol on the list, and then type in each of the values.



HeartRate

1	75
2	

Do this ten times, so that your list looks like the one on the right. You might have to click and drag on the border of the list, so you can resize it and see all 10 entries.



HeartRate

1	75
2	76
3	81
4	83
5	79
6	76
7	73
8	70
9	70
10	72

S. & L. Floyd

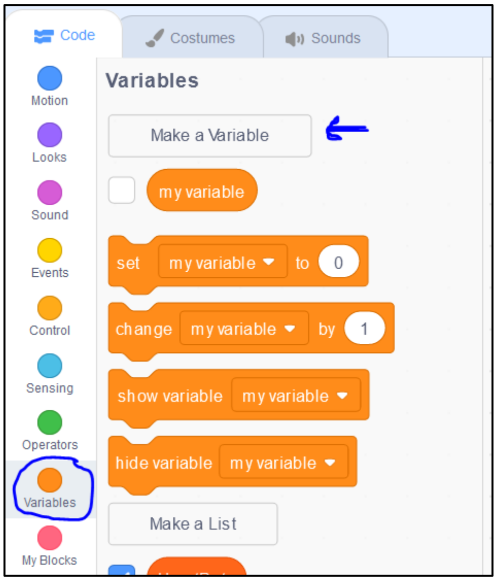
4) In order to write a program that will calculate the mean, we are going to keep track of a few more things in our program:

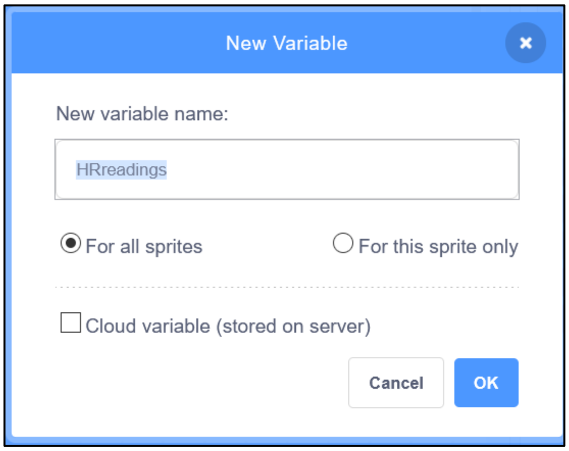
- The number of heart rate readings
- The sum of the heart rate readings
- The average heart rate

We will create a variable for each of these.

> *A variable is like a storage box in your program, that has a name. The variable can store one thing, and anytime you need to refer to the thing stored in the variable, you just use the name of the variable.*

Click **Variables -> Make a Variable** and then name your first variable **HRreadings** and click **OK.**




New Variable

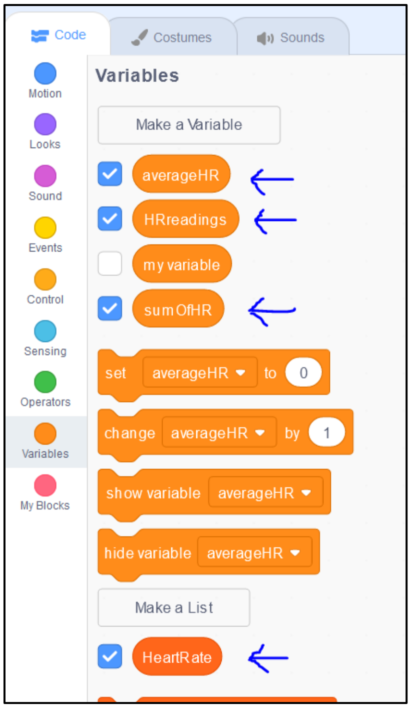
New variable name: HRreadings

● For all sprites ○ For this sprite only

□ Cloud variable (stored on server)

Now make two more variables:

- one called **sumOfHR**
- one called **averageHR.**



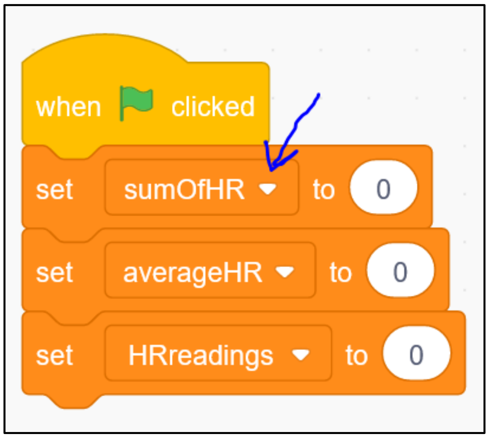
You should now have one list, and three variables that have been created.

These three variables and the list will hold all of the data that our program needs. Now it's time to write the code.

5) This program will run when we click the green flag, so we need to drag that block of code out first. Then we need to set up our three variables to make sure that they are set to 0 at the start of the program.

> *Initializing variables means giving variables their initial, or starting values, at the very beginning of the program. The values inside the variables might change when the program runs, but at the start, they often have to have a specific value.*

Drag out the appropriate blocks, so that your program resembles the following (you can change the variable that is being initialized to zero by clicking on the drop down arrow and finding the appropriate variable name):

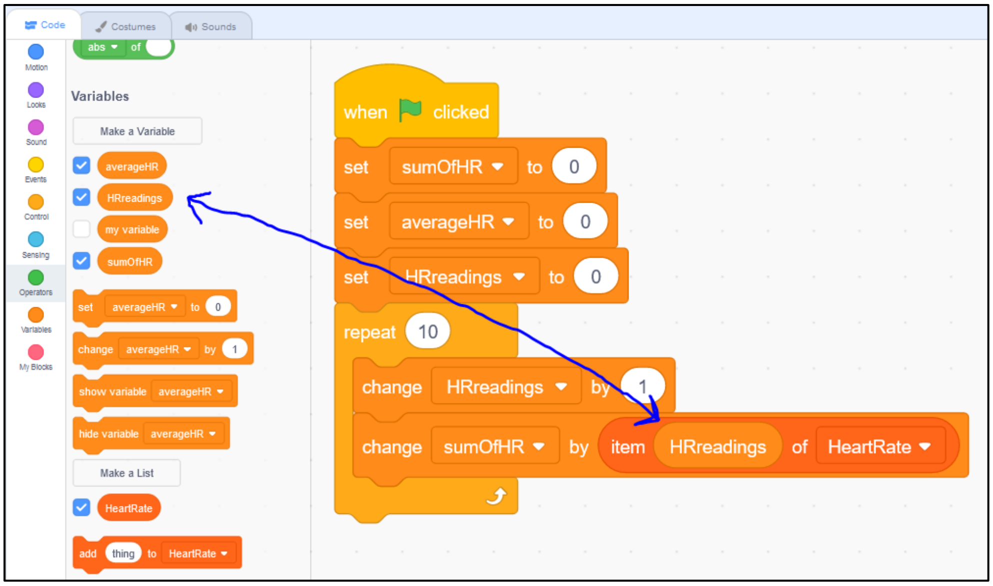


```
when ⚑ clicked
set sumOfHR ▾ to 0
set averageHR ▾ to 0
set HRreadings ▾ to 0
```

6) Now we have to write the code that will calculate the mean of the heart rate readings.

The algorithm we are going to use, to find the average is as follows:
-add up all of the heart rate values
-divide the sum of the heart rate values by the number of readings

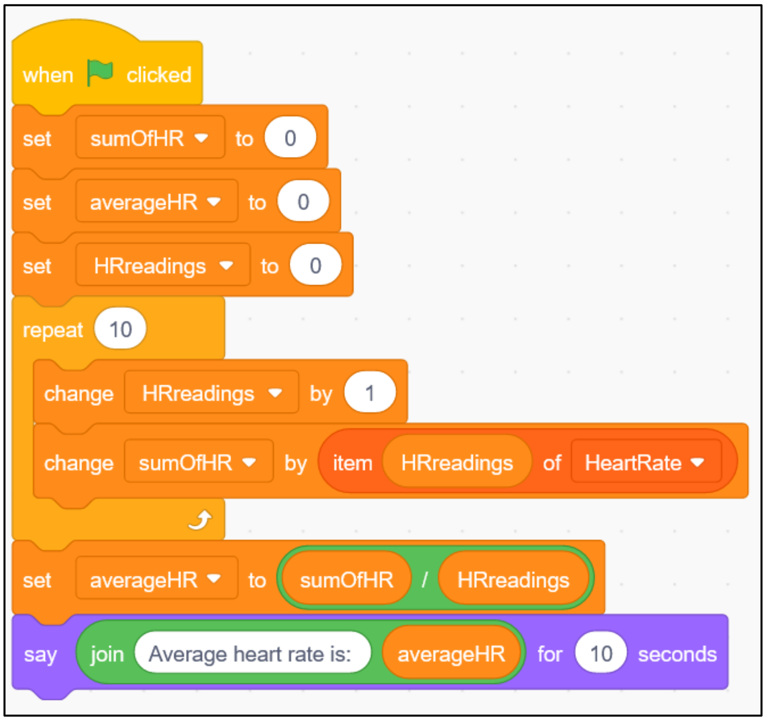
We can code this algorithm by first using a loop to add up all of HR entries. Drag out the appropriate blocks, so that your program resembles the following (you will also drag out the HRreadings variable name, to place it into the "item…" block of code):



```
when ⚑ clicked
set sumOfHR ▾ to 0
set averageHR ▾ to 0
set HRreadings ▾ to 0
repeat 10
    change HRreadings ▾ by 1
    change sumOfHR ▾ by item HRreadings of HeartRate ▾
```

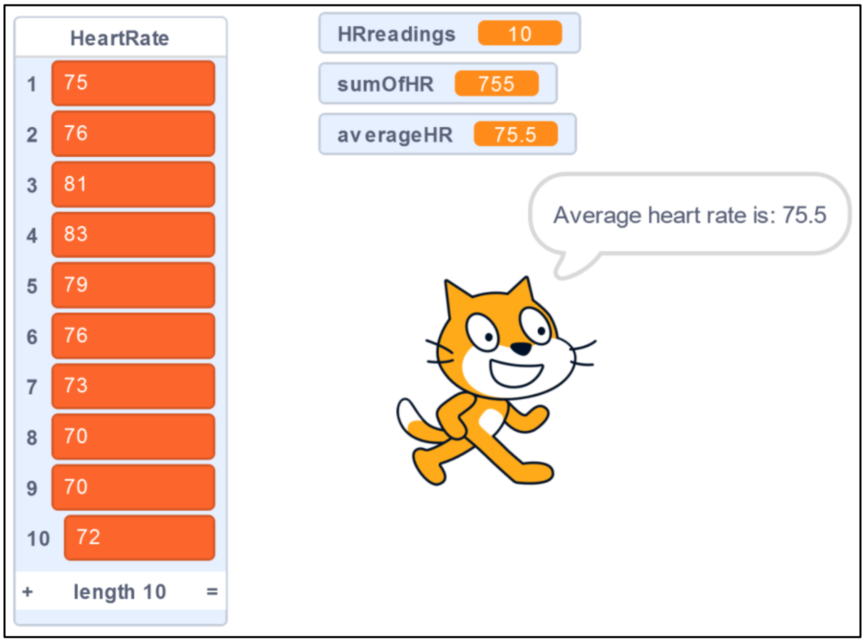
S. & L. Floyd

7) Now that we have added up all of the heart rate reading values, all that is left to do is to divide the sum by the number of readings, and then output that value to the screen.
Drag out the appropriate blocks, so that your program resembles the following:



```
when 🏳 clicked
set sumOfHR ▾ to 0
set averageHR ▾ to 0
set HRreadings ▾ to 0
repeat 10
    change HRreadings ▾ by 1
    change sumOfHR ▾ by item HRreadings of HeartRate ▾
↻
set averageHR ▾ to sumOfHR / HRreadings
say join Average heart rate is: averageHR for 10 seconds
```

8) Now try executing your program by clicking on the green flag.
The program should output the following:



```
HeartRate
1   75
2   76
3   81
4   83
5   79
6   76
7   73
8   70
9   70
10  72
+   length 10   =

HRreadings  10
sumOfHR  755
averageHR  75.5

Average heart rate is: 75.5
```

If the program doesn't execute exactly as shown, then you will have to go back to debug your program. This is a normal step in the programming process, so don't worry. Just compare your code to the code above and see what is different. Watching the video tutorial might also help.

CONGRATULATIONS!
You have written a program to calculate the mean heart rate based on ten heart rate readings.

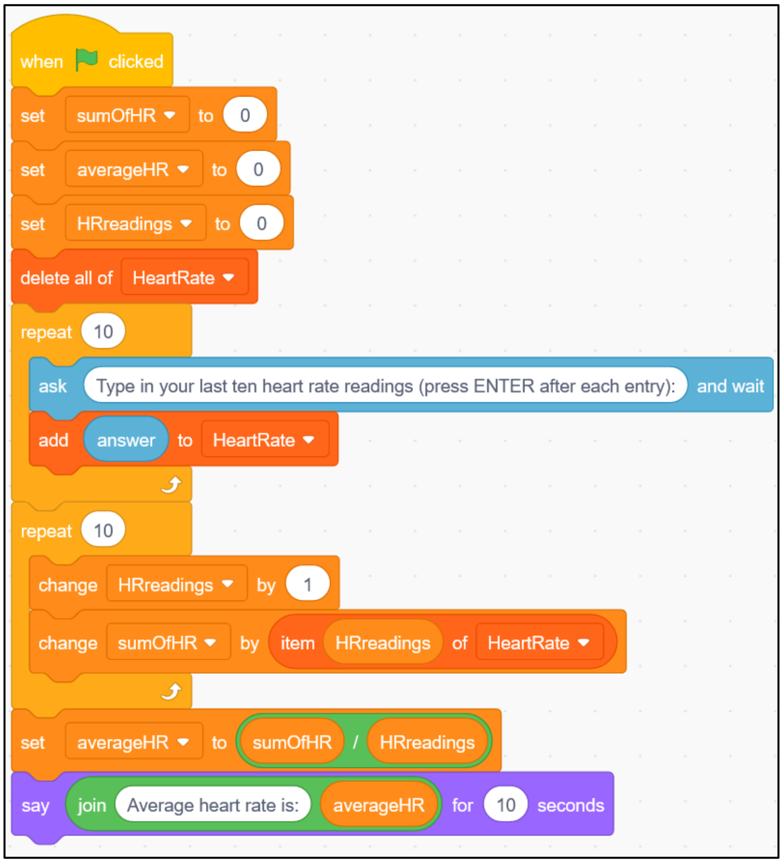
But wait… there's more!

A) See if you can alter this program to calculate the mean of 15 heart rate readings.
Hint:
- You will have to add five more heart values to the list by clicking the "+" symbol and entering in five more values.
- You will have to change the loop, so that it doesn't loop 10 times, but instead loops 15 times.

B) See if you can alter this program to calculate the mean of 20 heart rate readings.

C) See if you can start a new project and create a program that calculates the mean of some other data. You could use data from sports (like the points or goals per game), or from art (the price of recent paintings that have sold), or the weather (the daily high temperature from the last few days).

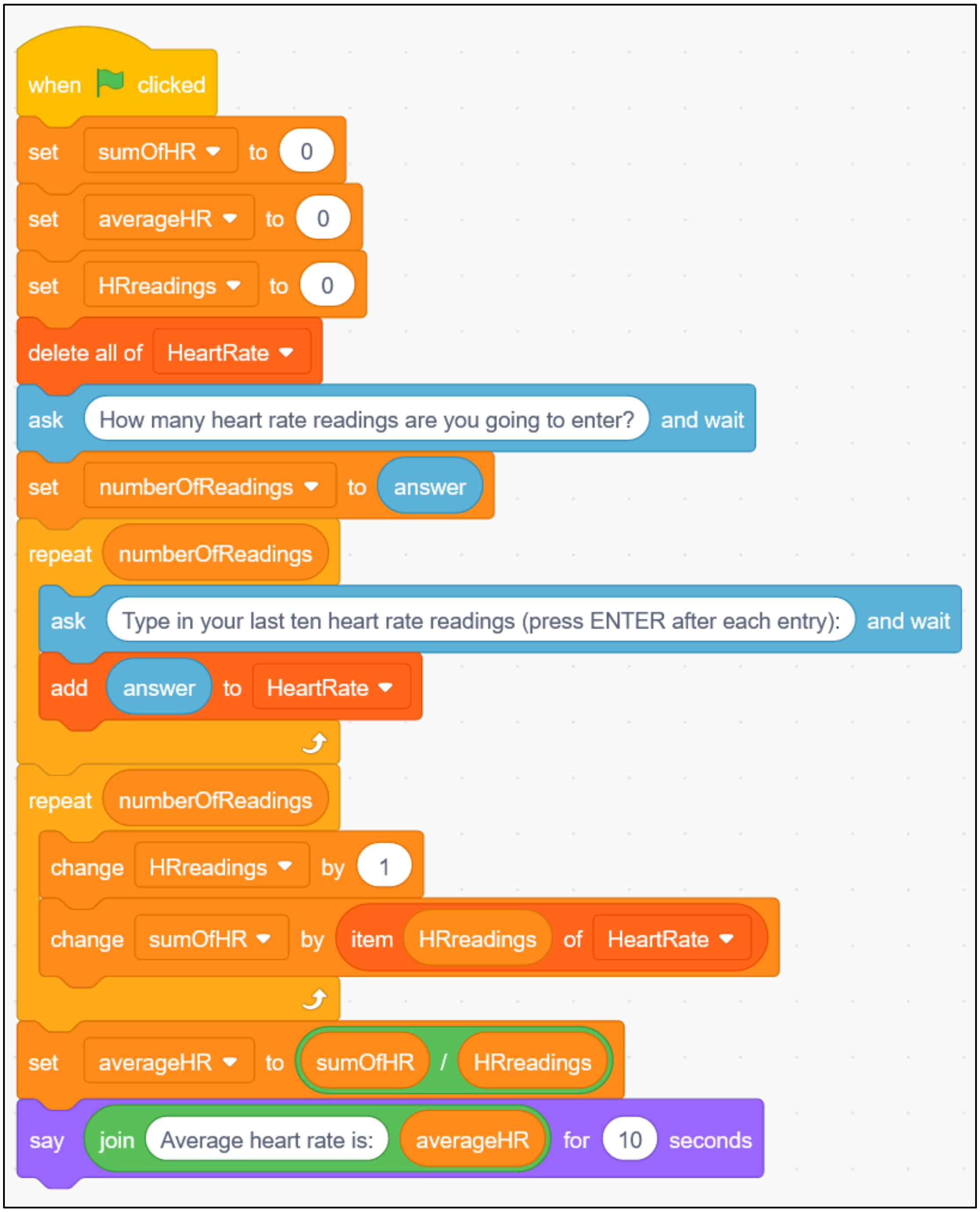
D) See if you can start a new project and get the code below to work. With this code, the user types in the ten values for the heart rate, instead of the programmer setting them up at the start, like we did in the previous programs. This program will still require a list, and the same three variables:



```
when ⚑ clicked
set sumOfHR ▾ to 0
set averageHR ▾ to 0
set HRreadings ▾ to 0
delete all of HeartRate ▾
repeat 10
    ask Type in your last ten heart rate readings (press ENTER after each entry): and wait
    add answer to HeartRate ▾
repeat 10
    change HRreadings ▾ by 1
    change sumOfHR ▾ by item HRreadings of HeartRate ▾
set averageHR ▾ to sumOfHR / HRreadings
say join Average heart rate is: averageHR for 10 seconds
```

S. & L. Floyd

E) See if you can alter your project so that instead of always typing in 10 heart rate readings, the user of the program can first enter in how many heart rate readings they are going to enter. The code below will help you. Notice now that we have a new variable called numberOfReadings and we ask the user for this first value right away, at the start of the program. This value then becomes the counter for the two loops (repeat blocks). COOL!



```
when 🏳 clicked
set sumOfHR ▾ to 0
set averageHR ▾ to 0
set HRreadings ▾ to 0
delete all of HeartRate ▾
ask How many heart rate readings are you going to enter? and wait
set numberOfReadings ▾ to answer
repeat numberOfReadings
    ask Type in your last ten heart rate readings (press ENTER after each entry): and wait
    add answer to HeartRate ▾
repeat numberOfReadings
    change HRreadings ▾ by 1
    change sumOfHR ▾ by item HRreadings of HeartRate ▾
set averageHR ▾ to sumOfHR / HRreadings
say join Average heart rate is: averageHR for 10 seconds
```

F) See if you can alter some of the programs above so that instead of heart rate data, the program works with other data that you are interested in.

S. & L. Floyd