

LESSON 4

Lesson No. 4. Appendix No.1. Creating the calorie calculator.

Using a visual programming language to create a calorie calculator kalorskaičiuok-

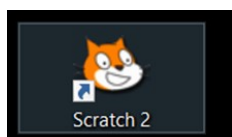
Task:

- To create a calorie calculator using the “Scratch” programming language.

Following the provided instructions, you are going to create a simple programme which allows you to calculate how many calories you use during a day when performing various activities.

In the chart activities defined by level of intensity (in METS) are presented. These activities will be needed for calculating the used calories. (ⓘ *useful information*):

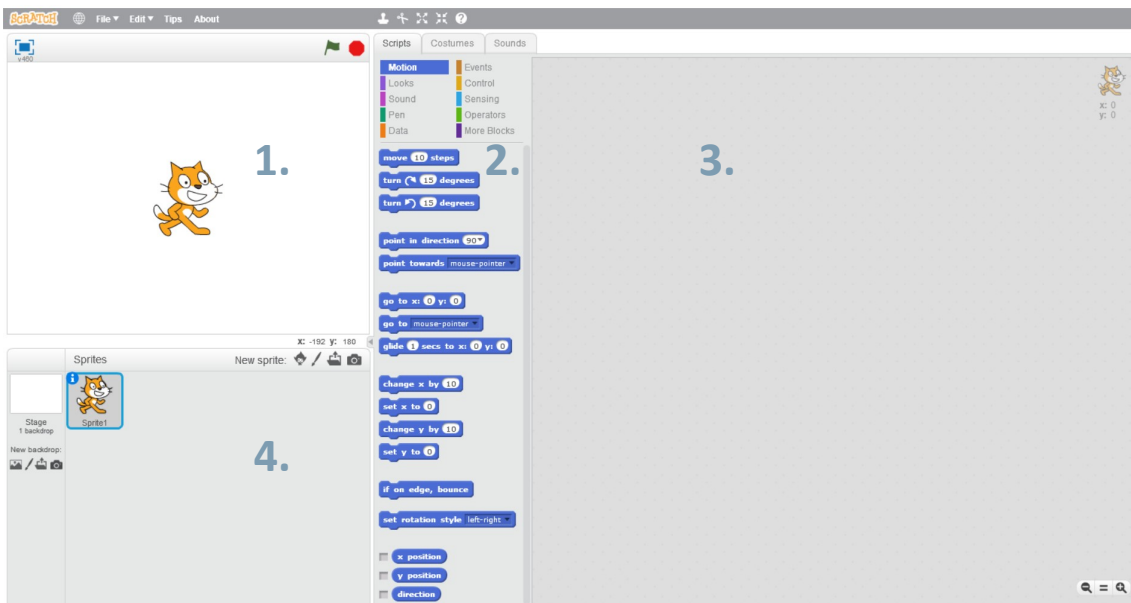
Activity	Additional information	MET ratio
Sleeping	-	0.93
Lying down	-	1.1
Sitting	-	1.43
Walking	Slow ≈4km/h	3.0
Walking	Vigorous effort ≈6.5km/h	4.0
Reading	-	1.5
School activities	-	1.56
Ball games	-	3.57
Football	-	7.1
Bicycling	Slow (≈16km/h)	4.0
Bicycling	Moderate (≈22km/h)	9.0
Bicycling	Vigorous effort (25-30km/h)	14.0
Dancing	Slow dancing, aerobics	6.0
Dancing	Vigorous effort	7.0
Jogging	Slow ≈8.3km/h	9.0
Jogging	Moderate ≈11km/h	11.5
Jogging	Vigorous effort ≈16km/h	16.0
Swimming	Slow	6.0
Swimming	Vigorous effort	10.5



LESSON 4

1.1. This is how the program should look like:

The area of the “Scratch” program

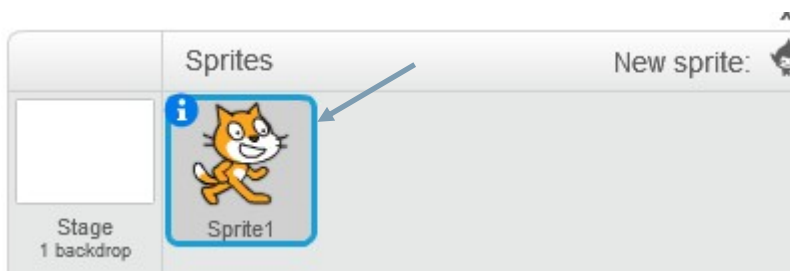


1. Results area where the created graphs are visible;
2. Field of all commands;
3. Script area;
4. Object area.

1.2. „Scratch“ might be programmed in different languages. Press the globe icon on the upper left side and choose the language from the list:




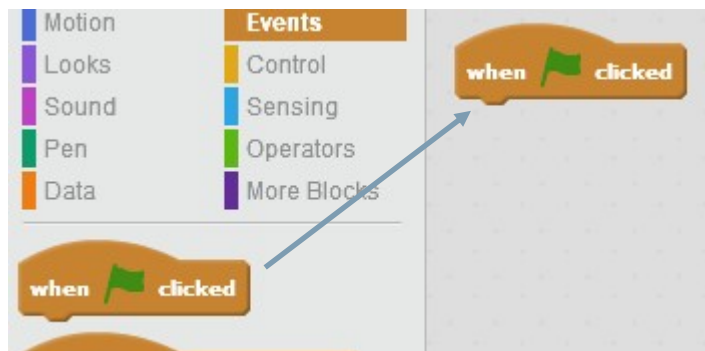
1.3. Prepare for programming the first **object** – the cat.



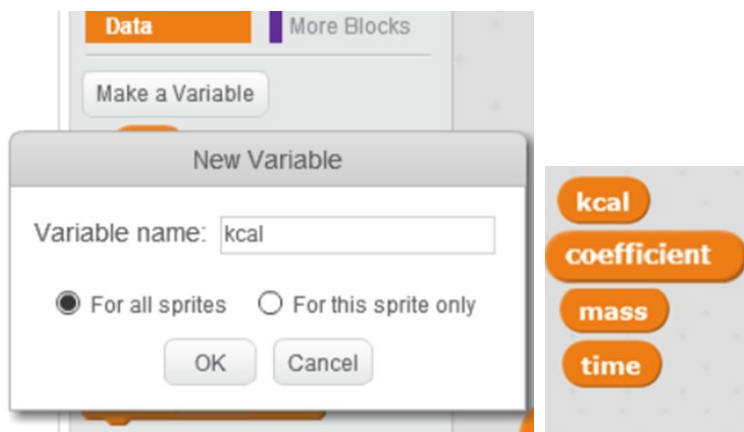
An object is a part of the program which has its data and functions.

LESSON 4

1.4. Start the program pressing the green “flag” („when  clicked“). Choose the first program of the commands field „Events“ and add it into the programming field:

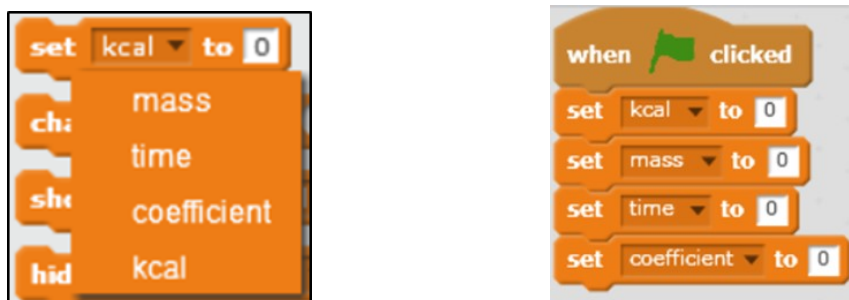


1.5. Now create 4 **variables** in which the main information, including number of calories, human body weight (mass), activity time and MET ratio (coefficient), will be saved. Press “Make a Variable“:



A variable is data (numbers, letters, etc.) which may change depending on instructions or determined conditions.

1.6. Then add the commands for variables „set ... to ...“ (the same command with different variables) in order for them to be “0” in the beginning (the commands might be found in the “data” field):

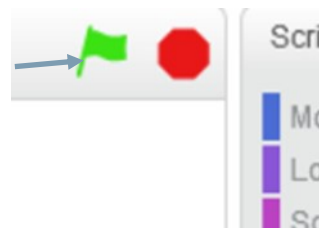


LESSON 4

1.7. Now program your calculator “the cat” to greet you. Use the command „say ... for .. secs“ which might be found in the commands field “looks” . You can write the wanted text and how long the text should be seen.

In the command “say” write the wanted greeting. The finished program will look like this:

2. It is very **important** to test the created program in a programming process. Test the program and check if it works. Press on the flag and make sure that the cat is greeting you and the variables are equal to 0.



3. Now start creating the main algorithm which will help you to count the used calories.

The idea of the algorithm:

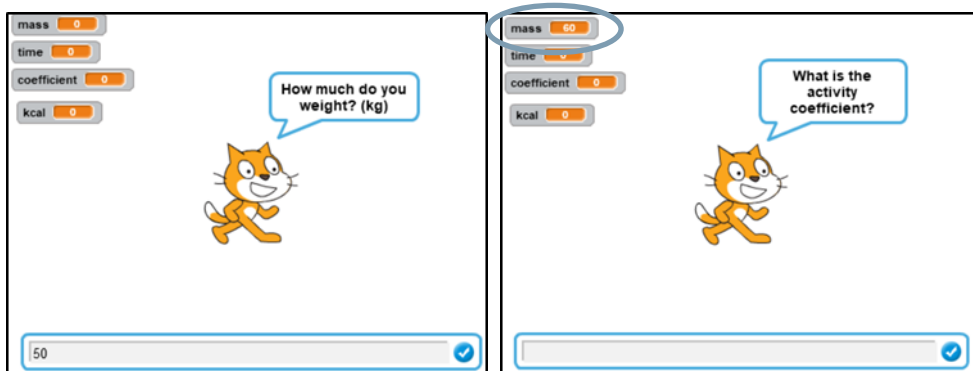
- Ask the user what the coefficient of the activity is and how long the activity took place. Then we count the used calories. The number of the calories is shown in the variable.

3.1. You will have to know the values of variables— duration of an activity, mass, coefficients of activities . First of all, ask how much a body weighs. Save the answer (the needed commands “ask ... and wait” and “answer” might be found in the „sensing“ field):

3.2. Then create questions about the coefficients and duration of activities. Use the “ask ... and wait” command and the command of variables “set ... to ...”:

LESSON 4

4. Test the program once again. Now the programmed cat should not only greet you but also ask some questions and save the answers. For example, after adding the weight, it should be saved in the variable of mass and a question about the activity's coefficient should appear:

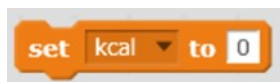


5. Finally, create the last algorithm – calculating based on the presented formula:

Used energy (kcal) = 0.0175 x MET (taken from the chart) x mass (kg) x time (min)

You can create the algorithm on your own or use the provided example.

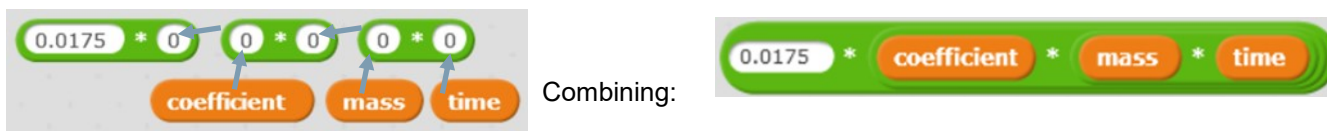
Set the answer to be multiplication can be



saved in the calories variable (commands of the sum or found in the “maths” field):

5.1. In the formula you see that in total 4 numbers will appear: time, mass, MET coefficient and the number 0.0175.

Create four more fields for multiplication where the variables will be put:



When you combine the entire formula, you are going to get this algorithm:



The program is finished. Now you can test the program if there are no mistakes. If it was created properly, it should calculate like this: e.g. if the typed mass is 50 kg, the coefficient is 9 and the time is 60 min., the received answer should be 472.5 kcal. (0.0175*9*50*60=472.5)

Check if the program works properly.

LESSON 4

Lesson 4. Appendix No. 2. Challen-

Using a visual programming language to create a calorie calculator kalorskaičiuok-

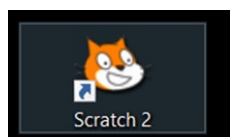
Task:

- To create a calorie calculator using the “Scratch” programming language.

Following the provided instructions, you are going to create a simple programme which allows you to calculate how many calories you use during a day when performing various activities.

In the chart activities defined by level of intensity (in METS) are presented. These activities will be needed for calculating the used calories. (① *useful information*):

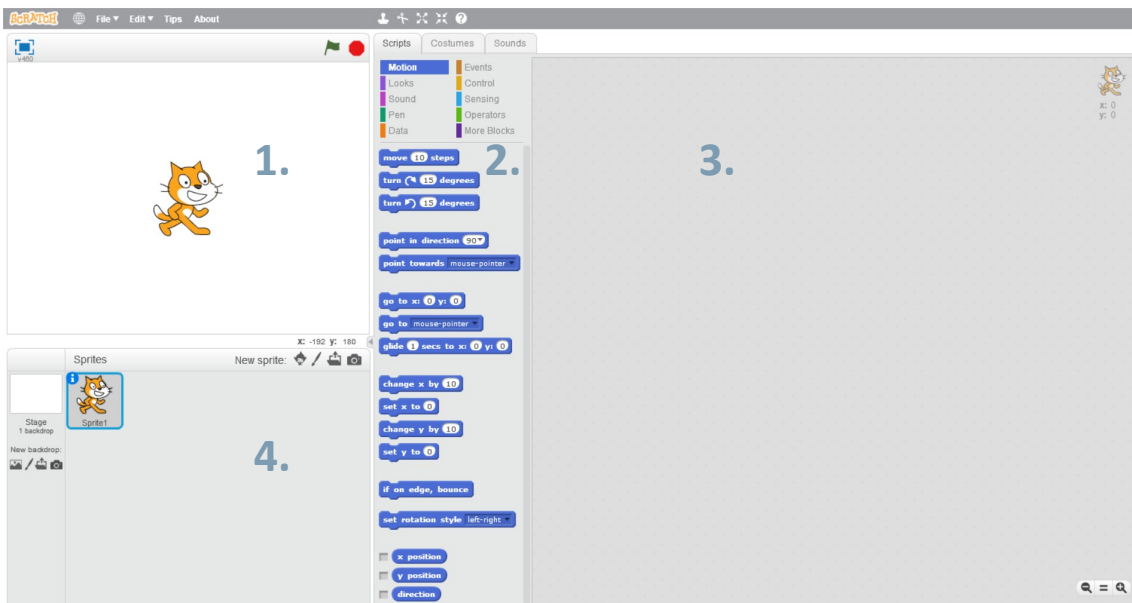
Activity	Additional information	MET ratio
Sleeping	-	0.93
Lying down	-	1.1
Sitting	-	1.43
Walking	Slow ≈4km/h	3.0
Walking	Vigorous effort ≈6.5km/h	4.0
Reading	-	1.5
School activities	-	1.56
Ball games	-	3.57
Football	-	7.1
Bicycling	Slow (≈16km/h)	4.0
Bicycling	Moderate (≈22km/h)	9.0
Bicycling	Vigorous effort (25-30km/h)	14.0
Dancing	Slow dancing, aerobics	6.0
Dancing	Vigorous effort	7.0
Jogging	Slow ≈8.3km/h	9.0
Jogging	Moderate ≈11km/h	11.5
Jogging	Vigorous effort ≈16km/h	16.0
Swimming	Slow	6.0
Swimming	Vigorous effort	10.5



LESSON 4

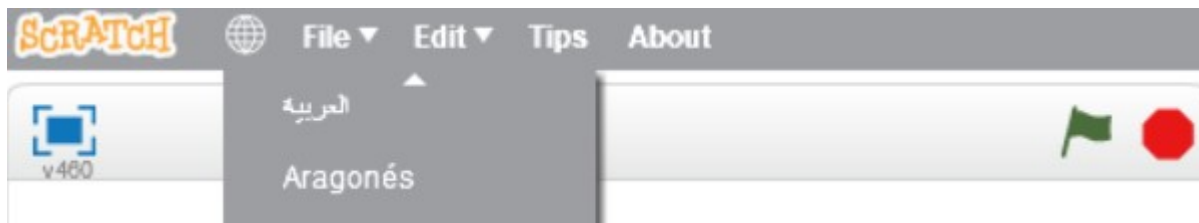
1.1. This is how the program should look like:

The area of the “Scratch” program

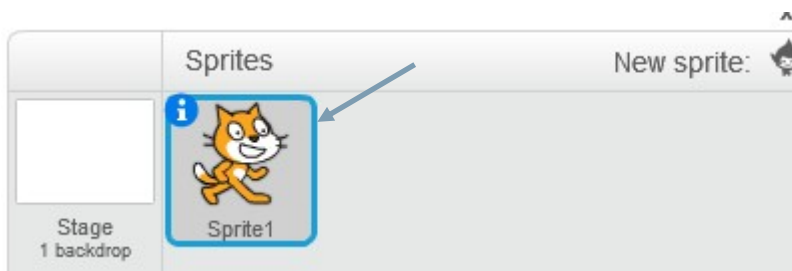


1. Results area where the created graphs are visible;
2. Field of all commands;
3. Script area;
4. Object area.

1.2. „Scratch“ might be programmed in different languages. Press the globe icon on the upper left side and choose the language from the list:




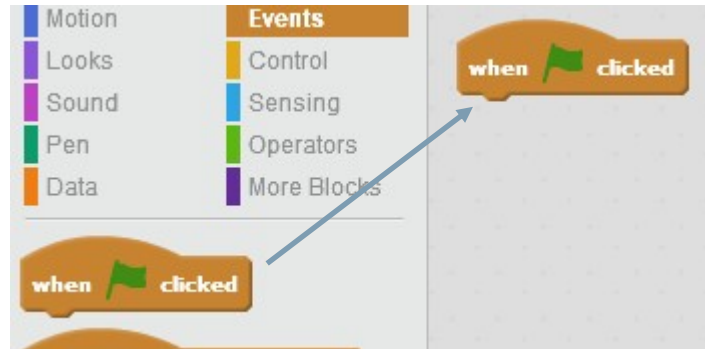
1.3. Prepare for programming the first **object** – the cat.



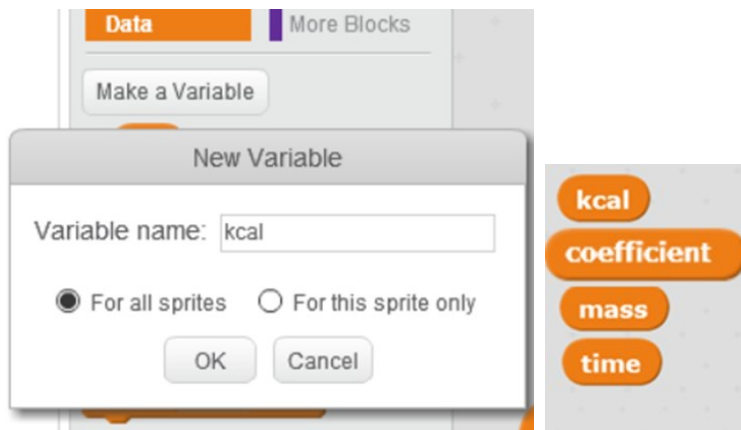
An object is a part of the program which has its data and functions.

LESSON 4

1.4. Start the program pressing the green “flag” („when  clicked“). Choose the first program of the commands field „Events“ and add it into the programming field:

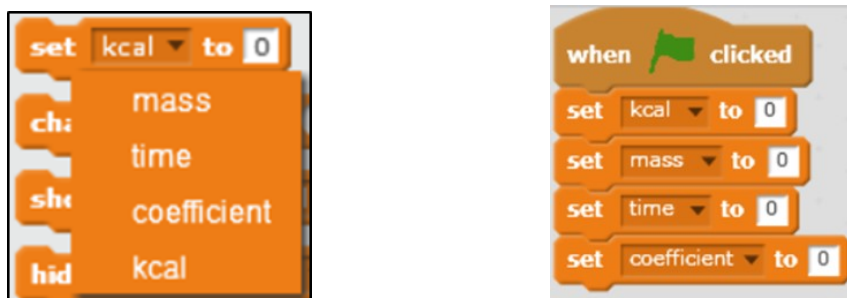


1.5. Now create 4 **variables** in which the main information, including number of calories, human body weight (mass), activity time and MET ratio (coefficient), will be saved. Press “Make a Variable“:



A variable is data (numbers, letters, etc.) which may change depending on instructions or determined conditions.

1.6. Then add the commands for variables „set ... to ...“ (the same command with different variables) in order for them to be “0” in the beginning (the commands might be found in the “data” field):



LESSON 4

1.7. Now program your calculator “the cat” to greet you. Use the command „say ... for .. secs“ which might be found in the commands field “looks” . You can write the wanted text and how long the text should be seen.

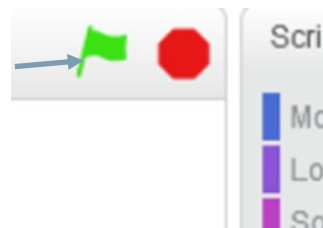
```
say Hello! for 2 secs
```

In the command “say” write

the wanted greeting. The finished program will look like this:

```
when green flag clicked
  set kcal to 0
  set mass to 0
  set time to 0
  set coefficient to 0
  say Hello, I am a calorie counting mashine! for 2 secs
  say I will ask you a few questions to calculate how many calories have you burned. for 5 secs
```

2. It is very **important** to test the created program in a programming process. Test the program and check if it works. Press on the flag and make sure that the cat is greeting you and the variables are equal to 0.



3. Now start creating the main algorithm which will help you to count the used calories.

The idea of the algorithm:

- Ask the user what the coefficient of the activity is and how long the activity took place. Then we count the used calories. The number of the calories is shown in the variable.

3.1. You will have to know the values of variables – duration of activity, body mass, coefficients of activities. However, for example, the value of mass will remain the same independently from how many activities you will include. First of all, ask „how much do you weigh“. Save the answer (you will find the needed commands „ask ... and wait“ and „answer“ in the „sensing“ field) and then perform the **cycle** „forever“ you will find it in the „Controls“ field.

Cycle is the repetition of certain commands.

```
say I will ask you a few questions to calculate how many calories have you burned
ask How much do you weight? (kg) and wait
set mass to answer
set mass to answer
forever
  ask Are there any more activities? (answer: yes or no) and wait
```

LESSON 4

1.7. Then insert **the conditional sentence** in the cycle „forever“ (you will find it in the „Controls“ field and you might find the command „=“ in the „operators“ field). Using it you will check if there is a need to continue calculating the calories used in another activity or to finish the program (you might find below how to join commands):

```

forever
  ask Are there any more activities? (answer: yes or no) and wait
  if no = answer then
    [ ]
  else
    [ ]
  
```

A conditional sentence is a command which decides if there is a need to carry on other indicated commands.

Put certain commands in the conditional sentence. There are two possible options: 1) you will finish the program or 2) you will continue asking the user about the duration of activity and coefficient. You can program individually or using this example (you will find the command „stop this script“ in the „controls“ field):

```

set mass to answer
forever
  ask Are there any more activities? (answer: yes or no) and wait
  if no = answer then
    say Thank you. Your burnt calories are written in the calorie variable. for 5 secs
    stop this script
  else
    ask What is the activity coefficient? and wait
    set coefficient to answer
    ask How much time did the activity last? (in minutes) and wait
    set time to answer
  
```

4. Test the program once again. Now the programmed cat should not only greet you but also ask several questions. Test the program twice – when you answer „no“ and when the answer is „yes“ ! For example:

If you write „yes“:



the program will



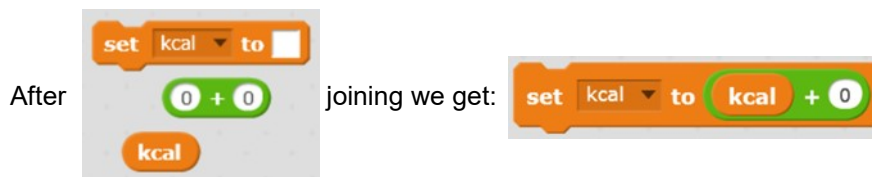
continue:

LESSON 4

5. Create the last algorithm – calculating based on the presented formula:

$$\text{Used energy (kcal)} = 0.0175 * \text{ET (taken from the chart)} * \text{mass (kg)} * \text{time (min)}$$

You can create the algorithm on your own or use the provided example. Firstly, take into consideration that you will need to add the calculated calories of another activity to the already calculated number of calories. So create this algorithm (you will find the commands of sum and multiplication in the „maths“ field):



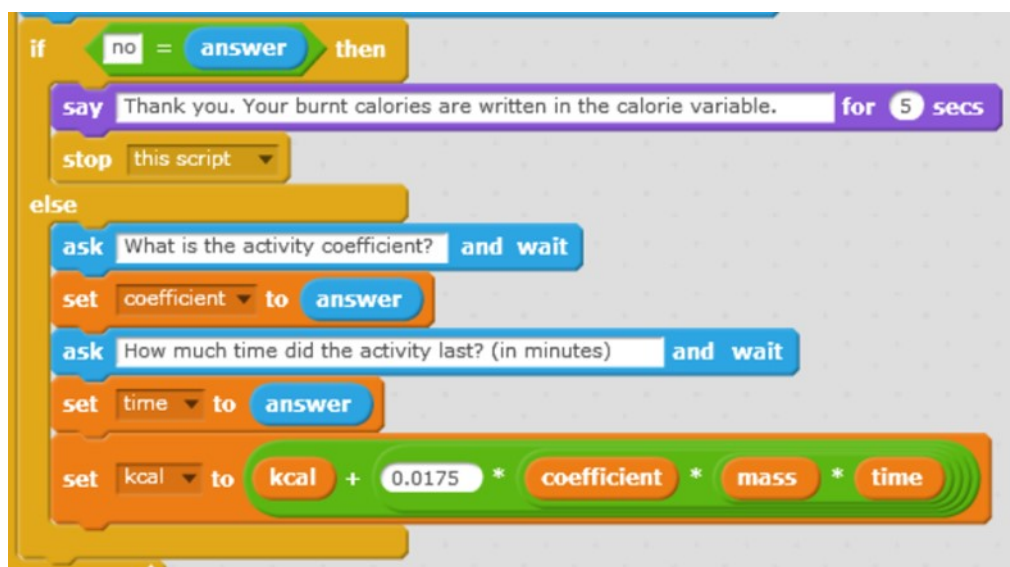
5.1. We see in the formula that there are going to be 4 numbers: duration of activity, body mass, MET coefficient and the number 0.0175. Create four more fields of multiplication where you will write variables:



Joining:



After joining the entire formula you will get this algorithm:



6. The program is finished. Now you can test it if there are no mistakes. If you created the program correctly, it should calculate like this: for example, when typing in the mass 50 kg, coefficient 9 and time 60 min., the answer should be 472.5 kcal. (0.0175*9*50*60=472.5).

Test if the program works correctly.