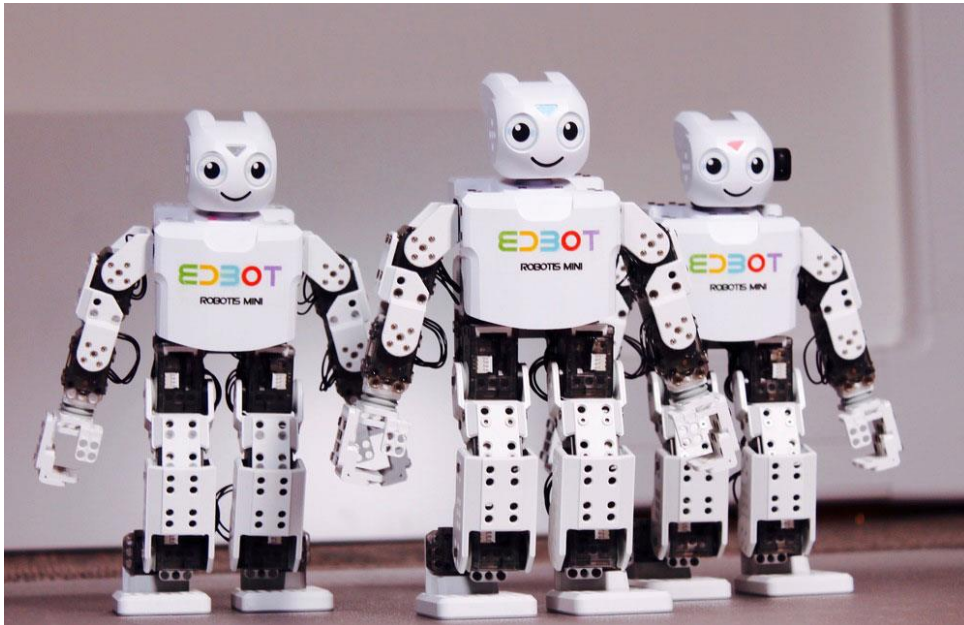


Edbot Advanced

Answer Book



In association with Nichola Wilkin Ltd

Table of Contents

Introduction	3
Lesson 1	4
Lesson 2	5
Lesson 3	6
Lesson 4	7
Lesson 5	8
Lesson 6	9



In association with Nichola Wilkin Ltd

Introduction

This answer booklet has been produced to give the teacher some help on the questions and problems which are included in the Edbot Advanced PowerPoint presentations and worksheets. The answers within this booklet only cover the questions on the PowerPoint presentations where there is a definite answer that is not included within the presentation itself.



In association with Nichola Wilkin Ltd

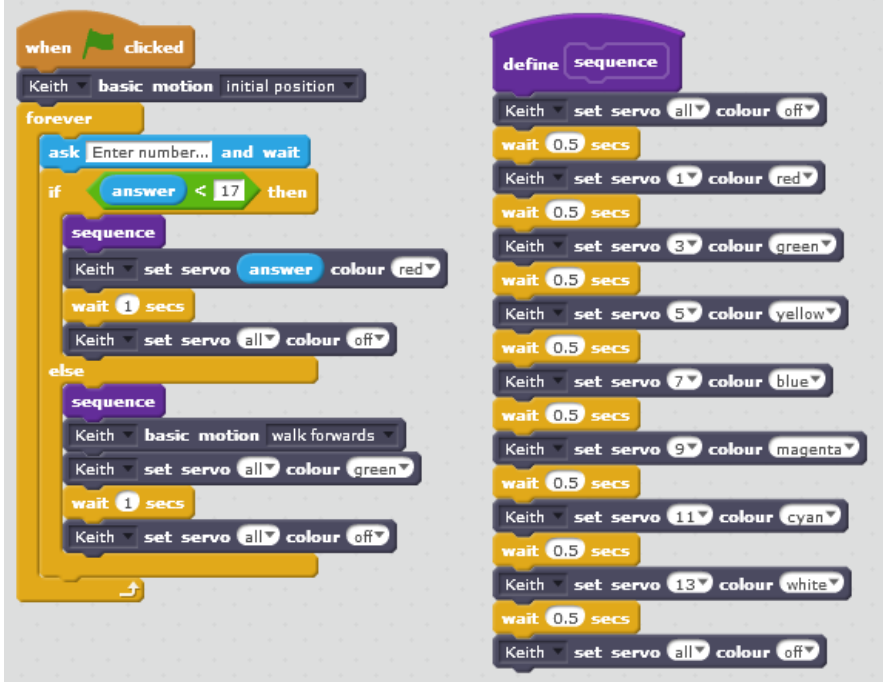
Lesson 1

Starter	<ol style="list-style-type: none">1. A robot that has similar characteristics to a human body, i.e. arms, legs and a head.2. More Blocks.3. Click on the green flag, press the correct key (if set up) or click on the sprite (if set up).4. A small motor that allows parts of Edbot to move.
----------------	---

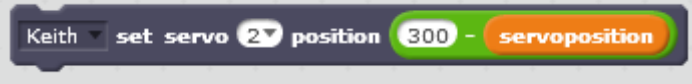


In association with Nichola Wilkin Ltd

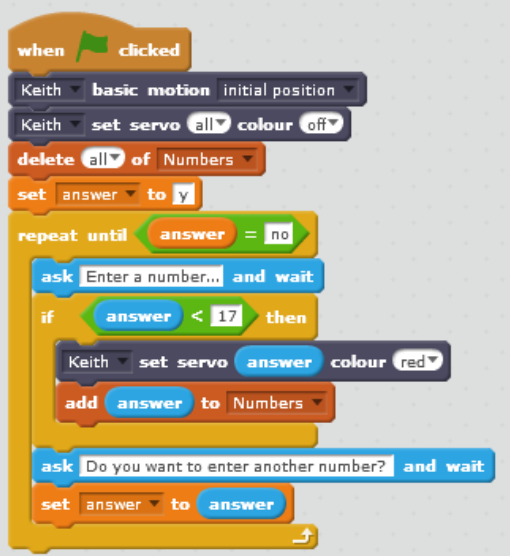
Lesson 2

<p>Starter</p>	<ol style="list-style-type: none"> 1. When the green flag is clicked it will make Edbot stand in the initial position. It will pick a random number between 1 and 10 and store that in a variable called num1. It will then pick a random number between 5 and 12 and store that in a variable called num2. It will then multiply those two numbers together and store that answer in a variable called total. It will then ask what is the total divided by num1. This will make sure the answer is a whole number (it will be num2). It will then check, if the answer that the user enters is equal to num2 and if it is Edbot will do a dance. If they get the answer wrong Edbot will do push-ups. This will then be repeated until 5 questions have been asked. 2. Add another variable called score that is set to 0 at the start of the program. Add 1 to the score each time they get the answer correct. At the end of the program tell the user what the score is.
<p>Slide 6</p>	<ol style="list-style-type: none"> 1. You would create your own block to define a set of instructions which you would need to repeat at different times in a program. By defining your block you can easily insert that code into your program in multiple locations without having to insert lots of lines. It is also easy to update that block, if necessary, as it is stored in one central location. 2. Click on More Blocks and select Make a Block. Enter a name and then create the code. To use the code in your script drag the purple name block into your main program.
<p>Slide 7</p>	<p>It will make Edbot walk forwards. It will check if there is an object in front of the distance sensor. If it detects something that is 20cm or closer, Edbot reverses and turns right a random number of times (between 1 and 3) before continuing to walk forwards. The idea behind the get_distance block is that sometimes sensors do not take accurate readings so it will take 5 samples over half a second and work out the average of those 5 as a more accurate distance.</p>
<p>Slide 8</p>	 <p>The screenshot shows two parts of Scratch code. On the left is a script starting with 'when green flag clicked', followed by 'basic motion initial position', a 'forever' loop containing an 'ask Enter number... and wait' block, an 'if answer < 17 then' block with a 'sequence' block inside, and an 'else' block with another 'sequence' block. On the right is a 'define sequence' block containing a series of 'set servo' and 'wait 0.5 secs' blocks for servos 1 through 13, each with a specific color.</p>

Lesson 3

Starter	<ol style="list-style-type: none">1. When the green flag is clicked it will make Edbot stand in the initial position. It will switch all the servo lights off and set a variable called light to 1. It will then change the lights on the servo that equals the variable to red and add 2 to the value. It will wait 2 seconds and will keep doing this until the variable called light is over 16 and then will stop. In effect this will change all the lights down one side of Edbot to red.2. A variable is a value you can change while the program is running.3. Light.4. In data, click on Make a variable, give it a name and click on OK.
Slide 4	When the green flag is clicked it will ask the user to enter 3 numbers and it will add them together. After the three numbers have been added it will ask for the total. If the user is correct Edbot will walk forwards and give them a message saying they are correct. If they are wrong Edbot will walk backwards and tell them they are wrong and what the total should be.
Slide 8	 <ol style="list-style-type: none">1.2. So it keeps repeating the code and doesn't stop after one loop.
Slide 10	<ol style="list-style-type: none">1. To allow you to insert repetitive lines of code.2. In More Blocks, click on Make a Block.3. Drag the purple block into your code.

Lesson 4

Slide 8	<ol style="list-style-type: none">1. A variable can only store a single value but a list can store multiple values.2. In Data, select Make a list.3. Add “thing” to Listname.4. Delete all of Listname.
Slide 9	<p>At the start of the program we added the line to clear the list.</p> <p>Extension Activity:</p>  <p>The image shows a Scratch script starting with a 'when green flag clicked' event. The code includes: 'Keith basic motion initial position', 'Keith set servo all colour off', 'delete all of Numbers', 'set answer to y', a 'repeat until answer = no' loop containing 'ask Enter a number... and wait', an 'if answer < 17 then' block with 'Keith set servo answer colour red' and 'add answer to Numbers', 'ask Do you want to enter another number? and wait', and 'set answer to answer'.</p>
Slide 10	<p>A completed version of the file is included in the Teacher’s folder called L4_DanceListANSWERS.sb2.</p>

Lesson 5

Slide 14

```
when space key pressed
  Keith basic motion initial position
  Keith set servo positions 1/290/2/10

when up arrow key pressed
  Keith basic motion crouch
  Keith set servo positions 1/290/2/10
  Keith set servo positions 5/100/6/200

when a key pressed
  Keith fight motion left hook
  Keith set servo positions 1/290

when 9 key pressed
  Keith basic motion initial position
  Keith set servo positions 1/50/2/50

when clicked
  Keith basic motion initial position
```


Lesson 6

Slide 5

Standard Answer:

```
when clicked
  delete all of answers
  Keith basic motion initial position
  repeat 5
    set num1 to pick random 10 to 20
    set num2 to pick random 10 to 20
    set total to num1 + num2
    ask join num1 join + join num2 = and wait
    if answer = total then
      Keith basic motion initial position
      set colour to green
      Lights
      add Correct to answers
    else
      Keith basic motion crouch
      set colour to red
      Lights
      add Wrong to answers
```

```
define Lights
  Keith set servo all colour colour
  wait 1 secs
  Keith set servo all colour off
```

The answer for the extension activity can be found on the next page.

Extension Activity:

Slide 5

```
when clicked
  set score to 0
  delete all of answers
  Keith basic motion initial position
  repeat 5
    set num1 to pick random 10 to 20
    set num2 to pick random 10 to 20
    set total to num1 + num2
    set question to join num1 join + join num2 =
    ask question and wait
    if answer = total then
      Keith basic motion initial position
      set colour to green
      Lights
      set data to join question join answer join Correct
      add data to answers
      change score by 1
    else
      say join The correct answer is total for 2 secs
      Keith basic motion crouch
      set colour to red
      Lights
      set data to join question join answer join Wrong
      add data to answers
  say join You scored score correct for 2 secs
```

The code defines a 'Lights' function that sets a servo's color to a specified color and then turns it off after a 1-second delay. The main script starts when a flag is clicked, initializes a score to 0 and an empty answers list. It then enters a loop that generates a math problem (two random numbers between 10 and 20 added together) and asks the user to solve it. Depending on whether the user's answer is correct, the servo is set to green (correct) or red (wrong), and the score is updated. The script concludes by displaying the final score.