MODULE 5: INVESTIGATION 1

Polygon Fireworks, Night Skyline

This investigation focuses on developing and building pupils' understanding of variable through the creation of polygon firework patterns. The initial investigation recalls the polygon patterns from Year 5 placing a greater focus on the use of an unknown through "ask and answer" to vary different attributes of the polygon patterns.

From a concept development perspective, the **answer** block is distanced from its companion **ask** block within the script by prompting the pupil to vary aspects of the polygon. Variable development continues as pupils realise the limitations of "ask and answer" and give a name to a value by defining a user variable for the skyline towers.

- **Activity 5.1.1** – Ask and Answer
- **Activity 5.1.2** – Unplugged: Polygon Predictions
- **Activity 5.1.3** – Naming Values
- **Activity 5.1.4** – The Sky at Night

Scratch starter project

5-Polygon Firework
5-Polygon Firework INT1
5-Polygon Firework INT2
5-Polygon Firework FINAL

LINKS TO PRIMARY NATIONAL CURRICULUM

CURRICULUM OBJECTIVES	LINK WITH SCRATCHMATHS
Mathematics Solve problems, including missing number problems, using multiplication and division Use simple formulae Pupils calculate the perimeter of rectangles and related composite shapes Find all factor pairs of a number Draw 2D shapes using given dimensions and angles Distinguish between regular and irregular polygons (KS3) Work with experiments that involve random numbers Describe positions on the full coordinate grid	▶ Pupils are required to discuss and build simple formula which incorporates multiple variables and involves multiplication and division to create their polygon fireworks. ▶ [Extension] As an extension pupils are asked to build a script for their sprite to calculate and say the perimeter of the polygon it has drawn. ▶ Pupils are prompted to recall factor pairs of 360°. ▶ Pupils are required to use variables to specify the side length and angle of a generalised polygon. ▶ Pupils are asked to discuss what is the same and what is different between regular and irregular polygons. ▶ [Extension] Pupils build scripts that randomly position polygon fireworks and towers of squares within a set area. ▶ Pupils are required to use their knowledge of the full coordinate grid to position their polygons and towers.

Module 5 • Investigation 1 • Activity 5.1.1
Ask and Answer

Learning Objectives

Explore how to use the ask and answer blocks to draw different types of regular polygons.
Explain what is the same and what is different between regular and irregular polygons.

Activity Instructions

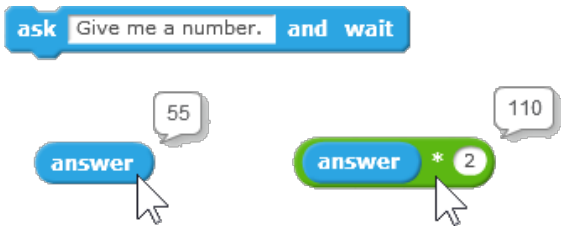
Pupils open project **5-Polygon Firework**, **Save as a copy** (online) or **Save as** (offline) and rename. The final version of this project at the end of Activity 5.1.1 will be **5-Polygon Firework INT1**.

1. Pupils explore the **ask** and **answer** blocks: they keep them isolated, click the **ask** block and type in the answer. Where is the text of the answer (the value) stored? Pupils click the **answer** block to find out. They also click the **check box** next to the **answer** block to see its small **monitor window** in the stage.
2. Pupils build a script: When the Beetle is clicked, it will ask *What is your name?* When they answer and press Enter, the Beetle will greet them by the name, using the **say** block with the **answer**. They explore the **join** *word1 word2* block to build a sentence for nicer greeting.

3. Pupils modify the script: When the Beetle is clicked, it asks what **pen size** it should use, then sets it and draws a line, a square, a regular polygon…
4. Pupils modify the script: When clicked, the Beetle asks what size the side of the square (a polygon) should be.

continues on the next page

Mathematics Connections



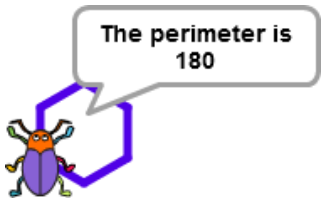
Note that through the **answer** block we are making the next step **towards the concept of a variable**. If we ask for a value and type in e.g. 55, we can then use the **answer** block as a variable in an algebraic expression, see above on the right.

Define regular polygon [all sides equal, all angles equal]. Show example of an irregular polygon, e.g. a house shape. *What is this called?* [an irregular pentagon] Use examples asking: *Is this irregular, is this regular?* Support pupils to define the regular polygon concept by asking: *What it is?* and *What it is not?*

The Beetle turns through the exterior angle of the polygon, it can be helpful to draw as a diagram on the board. Note the connection between the interior and the exterior angle [interior angle + exterior angle = 180°]

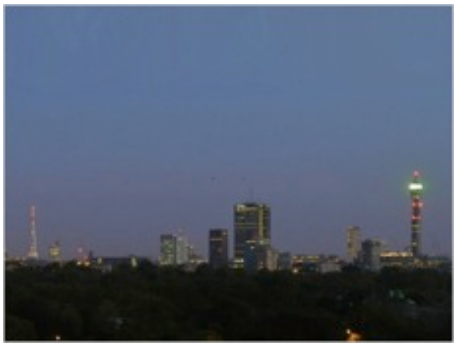
...continued

5. **[Extension]** After drawing the polygon, the Beetle will say what the perimeter of that square (polygon) is.



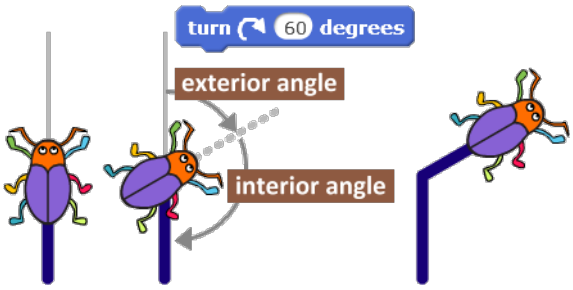
6. Pupils continue modifying their script: The Beetle first asks for the **pen size** and sets it, then asks what length the side of the square (polygon) should be.

7. The Beetle asks how many sides the polygon should have, then draws such polygon with a fixed side length, e.g. 30.



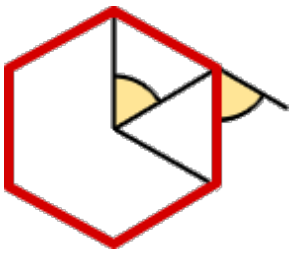
8. Pupils switch the backdrop to the *night skyline* and generalize the previous script: the Beetle asks for the number of sides, then draws many small polygons of that type scattered around the sky – by jumping to random positions. They may use the **set random pen shade** block or other random blocks. They may run the same script several times – giving different answers to the **ask** block.

9. Encourage pupils to simplify their scripts (making them more readable) by making a new block **polygon** with the **answer** block in it and use it in their scripts as a shortcut.



Ask pupils to draw a regular hexagon with a perimeter of 180: *What is the length of the side?* Draw a regular pentagon with a perimeter of 95: *What is the length of the side*? Draw an equilateral triangle with a perimeter of 100. Write a simple formula which connects perimeter of hexagon and side length [e.g. perimeter of hexagon = 6 × side length]

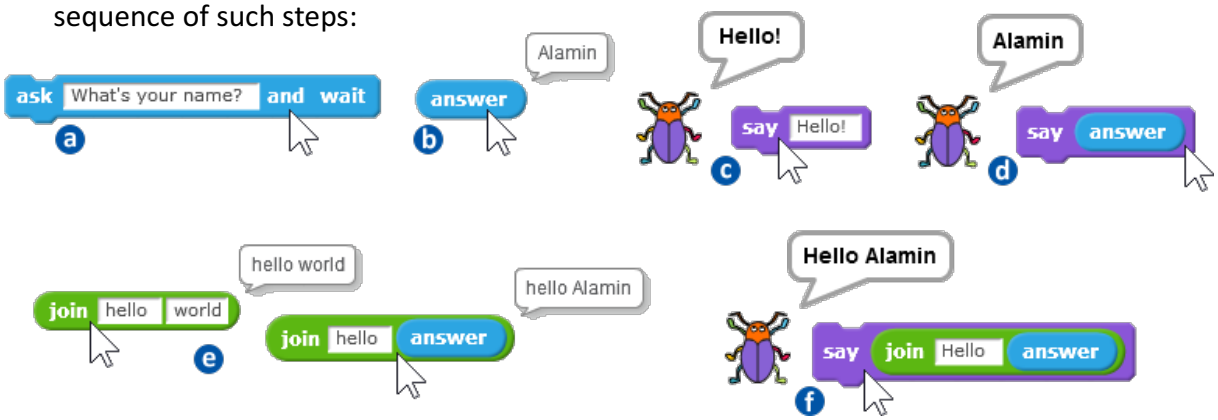
Recall factor pairs of 360° from year 5, e.g. 90° and 4 (square); 60° and 6 (hexagon); 72° and 5 (pentagon); 120° and 3 (equilateral triangle). *Why is 360° important?* [The Beetle turns through 360° as it moves around the polygon.] An irregular polygon will also turn through 360° degrees. This fact is useful when solving geometry problems where an angle is unknown.
Another way of seeing this: For any regular polygon the exterior angle of a regular polygon is the same as the angle that a circle is divided into.



CONNECTIONS TO Y5 SCRATCHMATHS

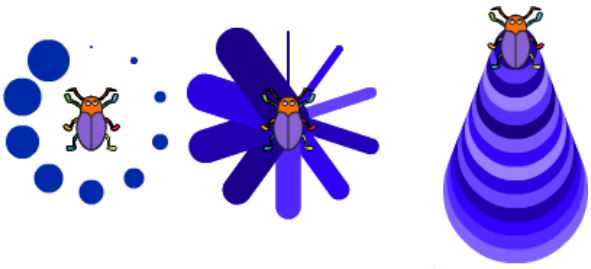
*Please note the **blue** numbers on the left link to the numbered steps in the **activity instructions***

1 The approach we apply here has been introduced in Module 1 and used throughout all Y5 modules: Always build scripts "from inside out", i.e. make sure you understand what each 'bit' does, only then start combining them. The following picture is an example sequence of such steps:



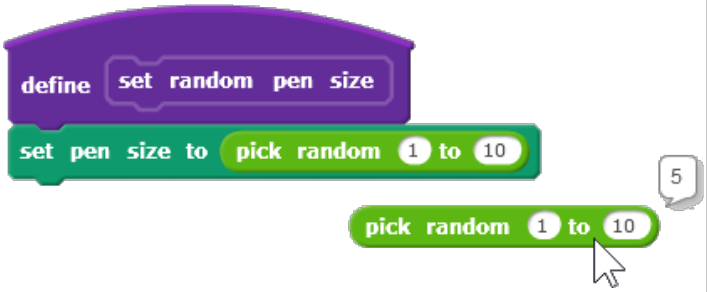
3 In Module 2, Activity 2.1.1 we started using a pen tool of a sprite, with some of its attributes, namely pen colour and pen size. We started using the following **Pen** blocks:

Pupils learned how to use the colour picker of the **set pen color to …** or alternatively to use the **set pen color to** *number_of_colour* block. In the additional materials for Module 2 there is a poster with 40 colours and their number codes. Also there are several other posters and sheets with challenges, one of them exploring the pen size, how to set it, use it and change it.



Note that in **Challenge 3: Explore the pen size** of the extension materials for Module 2 pupils are encouraged to use a pair of blocks **set pen size to …** and **change pen size by …**, which enables us to **set a certain value** and then **change it**. This is exactly what we will do later in this module with variables.

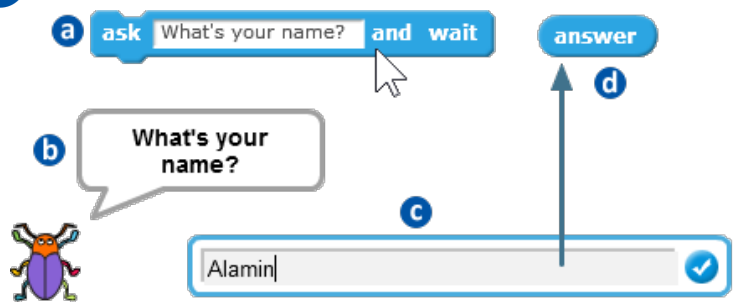
8 In Module 2, Activity 2.3.4 we applied another strategy: pupils were provided with several new **set random …** blocks, used the blocks in their scripts, and only then explored their definitions by decomposing and modifying. Thus they become familiar with the **pick random … to …** block.



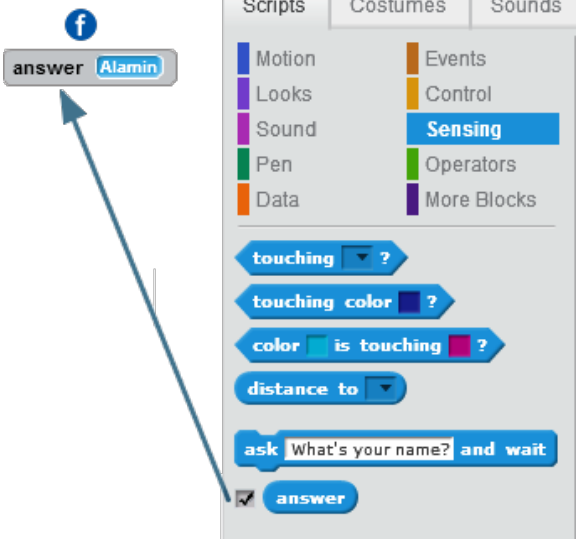
ADDITIONAL SUPPORT

*Please note the **blue** numbers on the left link to the numbered steps in the **activity instructions***

1



If we click the ask block (1a), the sprite asks the question (1b) and an edit line (1c) appears in the stage. We type in our response and press Enter or click the check mark. The answer is then "stored" in the answer report block (1d) and can be used in our script(s). Click the isolated answer block to see the value (1e).

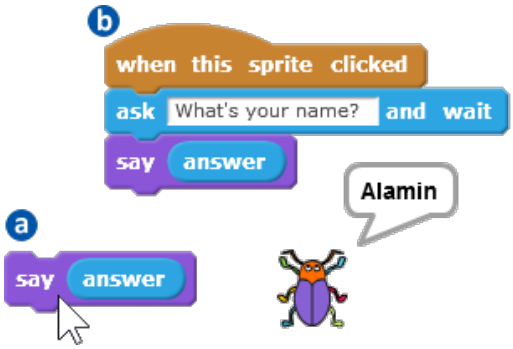


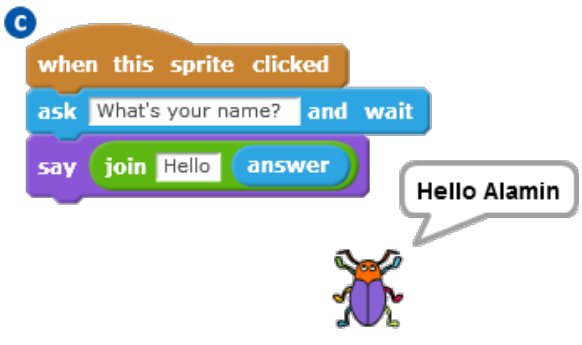
To view the value in the **monitor** of answer, we can also click the checkbox next to the answer block in the Scripts tab (1f).
The key difference between the answer block and the monitor is that **the block can be used in another block as its input** (see 2 below) while the monitor is just visual information for us to read.

The answer reporter block can be used as e.g. an input for the say block, see (2a).
So when the Beetle is clicked it will ask the question, then use the answer in the say block to greet us, see (2b).

Explore the join block (in Operators) to join together *Hello* and the value of the answer. Note that we added a space at the end of *Hello* so that the two words are separated by a space, see (2c).

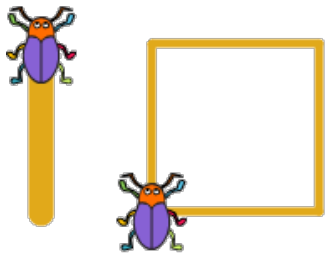
2





ADDITIONAL SUPPORT CONTINUED

3



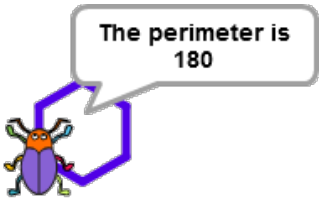
```
when this sprite clicked
ask [What pen size?] and wait
set pen size to (answer)
move (100) steps
```

```
when this sprite clicked
ask [What pen size?] and wait
set pen size to (answer)
repeat (4)
    move (100) steps
    turn (90) degrees
```

4

```
when this sprite clicked
ask [What side length?] and wait
set pen size to (5)
repeat (4)
    move (answer) steps
    turn (90) degrees
```

```
when this sprite clicked
ask [What side length?] and wait
set pen size to (5)
repeat (6)
    move (answer) steps
    turn (60) degrees
```



5

```
when this sprite clicked
ask [What side length?] and wait
set pen size to (5)
repeat (6)
    move (answer) steps
    turn (60) degrees
say (join [The perimeter is] (6 * answer))
```

6

```
when this sprite clicked
ask [What pen size?] and wait
set pen size to (answer)
ask [What side length?] and wait
set pen size to (8)
repeat (3)
    move (answer) steps
    turn (120) degrees
```

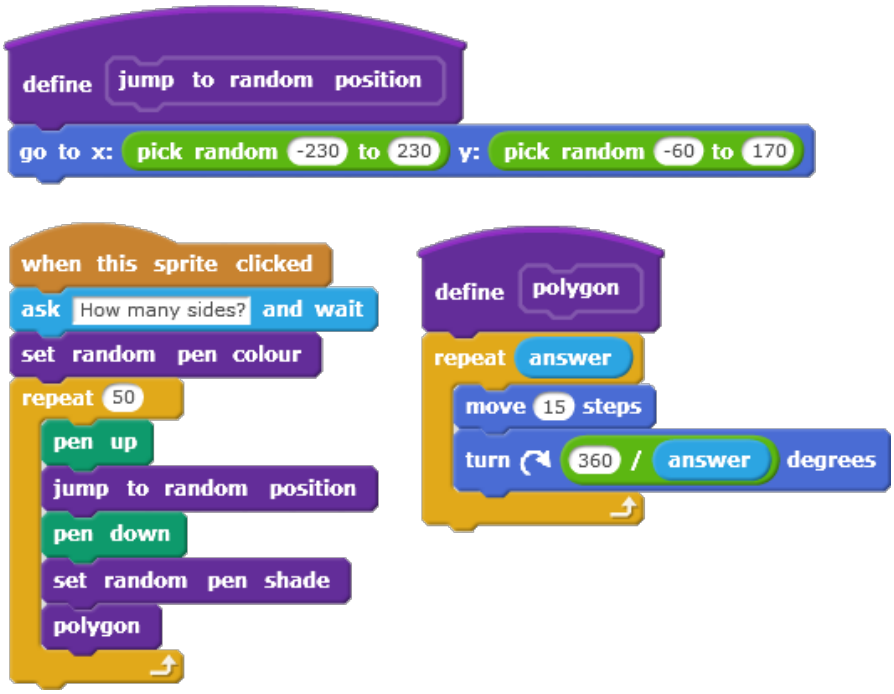
7

```
when this sprite clicked
ask [How many sides?] and wait
repeat (answer)
    move (30) steps
    turn (360 / answer) degrees
```

ADDITIONAL SUPPORT CONTINUED

8

Pupils will make their own block **jump to random position**, thinking about appropriate values for the **pick random … to …** It might be reasonable not to use numbers -240 and 240 but reduce them a bit so that the Beetle does not hit the edge when drawing a polygon.



MODULE 5 • INVESTIGATION 1 • ACTIVITY 5.1.2

Unplugged: Polygon Predictions

LEARNING OBJECTIVES

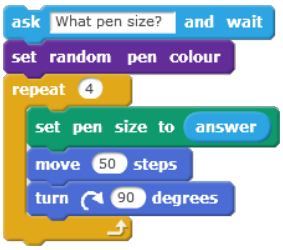
Envisage the behaviour of a script which uses the **ask** and **answer** blocks in different ways.
Explain how the corresponding outcome drawing was changed by the **answer**.

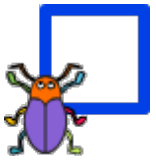
ACTIVITY INSTRUCTIONS

Print and distribute the pupil worksheet 5.1.2 or do the activity as a class.
Ask the pupils to explain how the **ask** and **answer** blocks are being used in the scripts, what the scripts will produce and whether the scripts can be simplified or improved.

SOLUTION

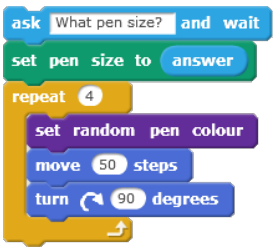
1





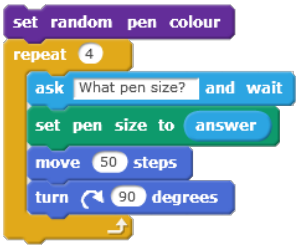
The Beetle asks for the pen size, selects a random colour and draws a square using the **answer** as the pen size. The pen size, however, is unnecessarily set four times – inside the **repeat** block. It only needs to be set once before the **repeat** block.

2



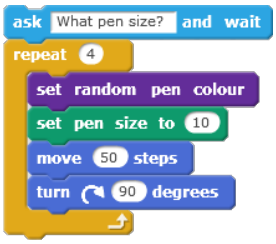
The Beetle asks for the pen size and uses the **answer** in the **set pen size …** block. The Beetle then draws a square setting a random colour for each side.

3



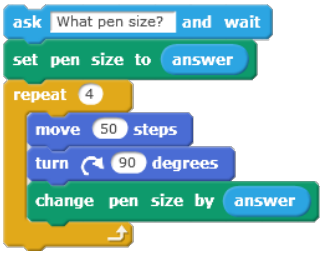
The Beetle sets a random pen colour then draws a square. For each side it asks for the pen size and uses it to draw that side.

4



The Beetle asks for the pen size but does not use the **answer** anywhere. It draws a square using random pen colour for each side, setting pen size to 10 inside **repeat** again and again, instead of setting it just once at the beginning.

5



The Beetle asks for the pen size and uses the **answer** in the **set pen size …** block. The Beetle then draws a square and increases the pen size by the **answer** repeatedly after drawing each side.

PRINT

INVESTIGATION 1
Activity 5.1.2

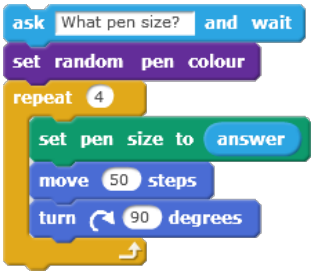
NAME

WHAT TO DO

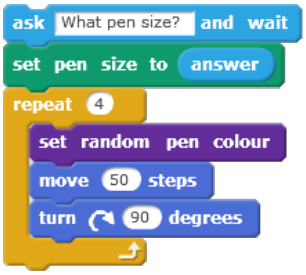
Read the scripts below. For each of them draw the picture it will create and explain in words what each script will do in the box on the right.

ASK ANSWER SCRIPTS

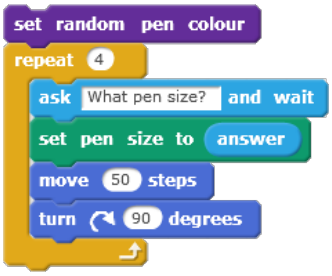
1



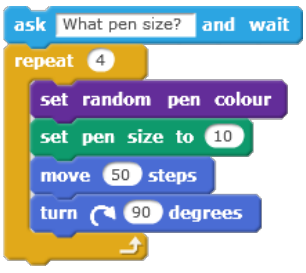
2



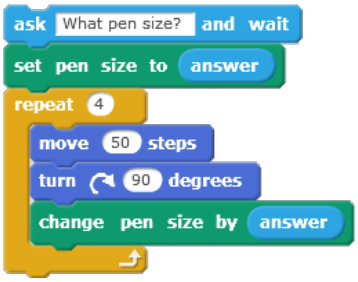
3



4



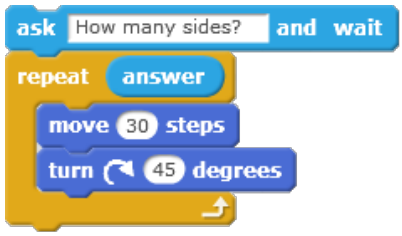
5



EXTENSION ACTIVITY INSTRUCTIONS

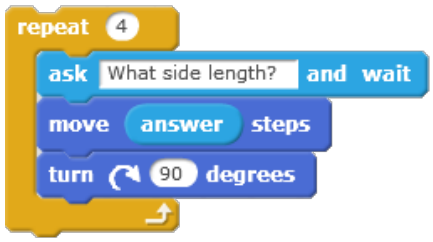
Do the following as a class: Each of the scripts below was intended to draw a regular polygon. However, in each script there is a bug. Envisage the original intention, explain the bug and suggest a fix.

1

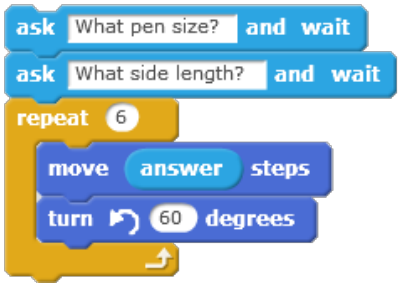


2

3

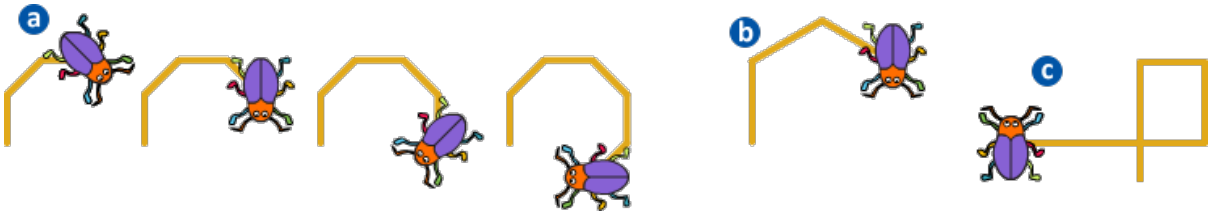


4



ADDITIONAL SUPPORT

1. In this script the **answer** is not used in the **turn** block at all. Therefore instead of drawing a polygon of the **answer** sides, the Beetle draws only **answer** sides of an octagon of the fixed size, see (a) below.
2. In this script the angle to turn by is wrong, the Beetle must turn by **360 / answer**, that is twice as much as it turns now, see (b).
3. In this script the question is asked four times, as the **ask** block is inside **repeat**. It means that if we do not answer the same value each time, the Beetle will not draw a regular polygon, see (c).
4. In this script two questions are asked but the **answer** to the first one is never used for anything but overwritten by the second **answer** immediately.



MODULE 5 • INVESTIGATION 1 • ACTIVITY 5.1.3
Naming Values

LEARNING OBJECTIVES

Explore how to use variables within a script to store different values at the same time.
Explain why we now need variables to draw multiple regular polygons of different sizes.

ACTIVITY INSTRUCTIONS

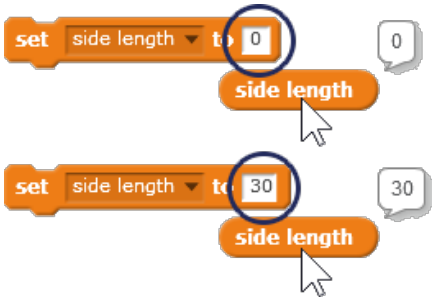
Pupils continue in their own version of project **5-Polygon Firework**, or open the **5-Polygon Firework INT1**, **Save as a copy** (online) or **Save as** (offline) and rename. The final version of this project at the end of Activity 5.1.3 will be **5-Polygon Firework INT2**.

1. Pupils combine two questions in their Beetle script: the Beetle should first **ask** about the side length of the polygon to be drawn, then about the number of its sides. However this is not possible using only the tools we have already used. Observing the monitor of the **answer** block, go through the script step by step so that pupils discover this problem themselves.
2. To **remember** the answer of the question asked, we have to **give that value a name** – **to store the value in a variable**. Pupils make a variable named **side length**. They drag two isolated blocks in the scripts area: **set side length to …** and the reporter block **side length**, keep them isolated and explore, observing also the small reporter window. They set different values to the variable. Similarly, they create the second variable **number of sides**.
3. Pupils snap two blocks: a question *What side length?* in the **ask** block and **set side length to …** the **answer**, run it and explore the value of the **side length** variable in its small monitor.
4. Pupils build the whole script from step 1 again, asking two questions and setting each variable to the corresponding **answer**. Then they modify the **polygon** block so that it uses these two variables instead of the **answer** block.
5. Pupils make the third variable **number of polygons** and add another question in the script: *How many polygons?* When clicked, the Beetle will ask three questions and draw that many polygons of the size and type as answered by the pupils.

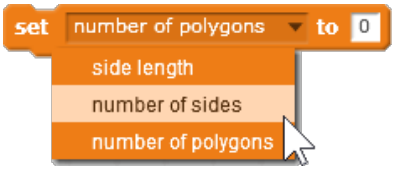
   Note that there is only one **set** *variable* **to …** block with a drop down list of all the variables.

MATHEMATICS CONNECTIONS

You may prefer to do most of this activity (up to point 4, including) using the empty *plain* backdrop.



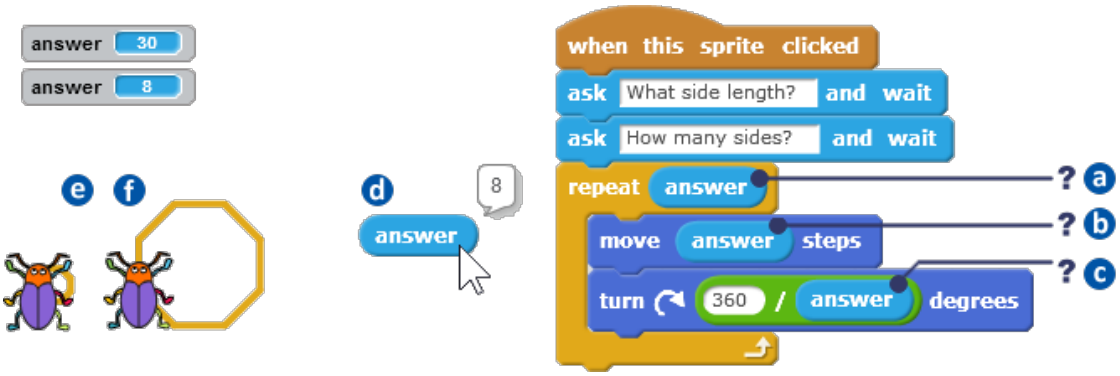
Note that the actual setting of a variable happens only after you run the block – by clicking or running a script containing that block.



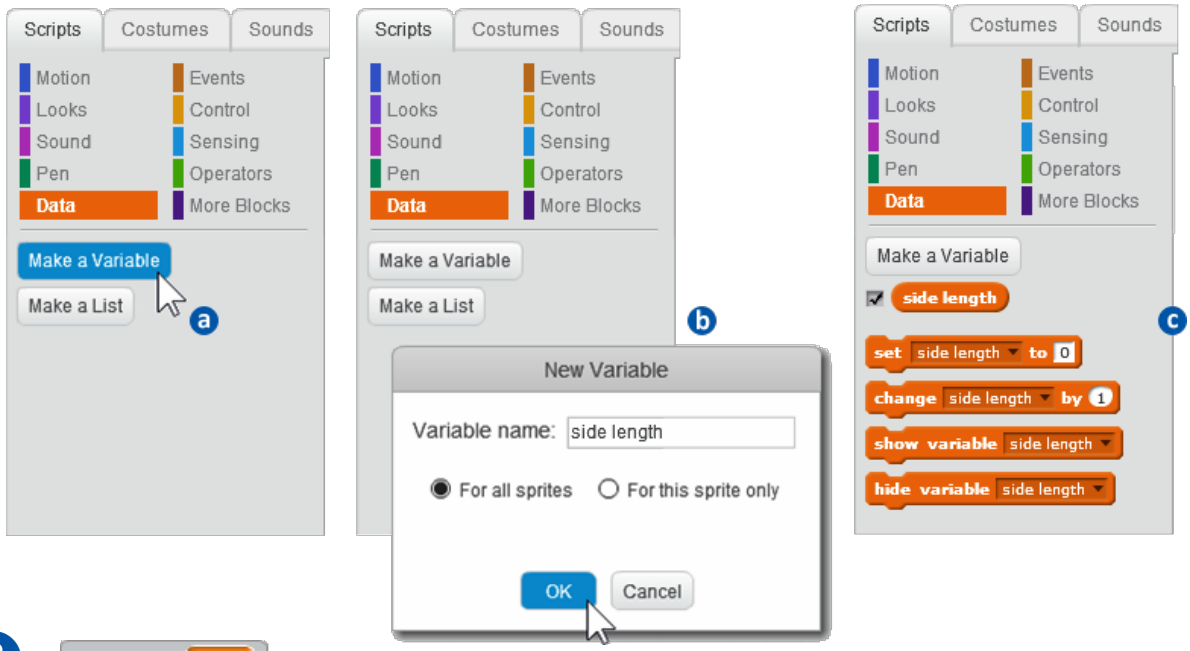
ADDITIONAL SUPPORT

*Please note the **blue** numbers on the left link to the numbered steps in the **activity instructions***

1 Here is an attempt to solve the task but **it does not work properly**. The answer block appears three times in the script, (1a) and (1c) refer to the second answer and (1b) refers to the first answer. However, as soon as we answer the second question, the first value of answer **is lost** and replaced by the second answer, see (1d). That is why the Beetle uses value 8 for (1a), (1b), and (1c) and draws (1e) instead of intended (1f).



2 To make a variable we go to the **Data** group and click the **Make a Variable** button (see 2a). After we type in the name of the new variable and click OK button (1b), several new blocks appear in the Data group. In this activity we use the reporter block side length and the set side length block.

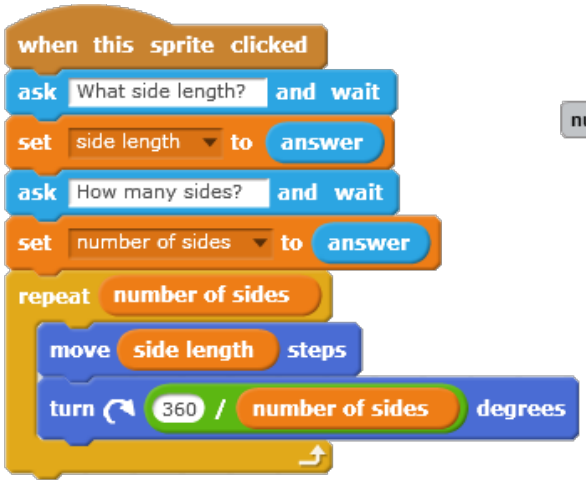


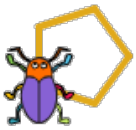
3

ADDITIONAL SUPPORT CONTINUED

4

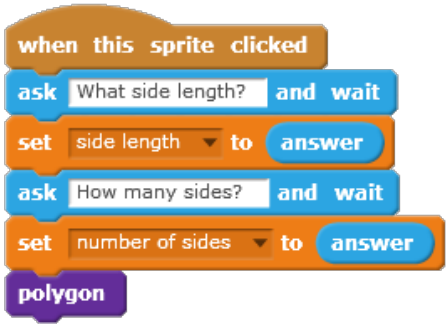
a

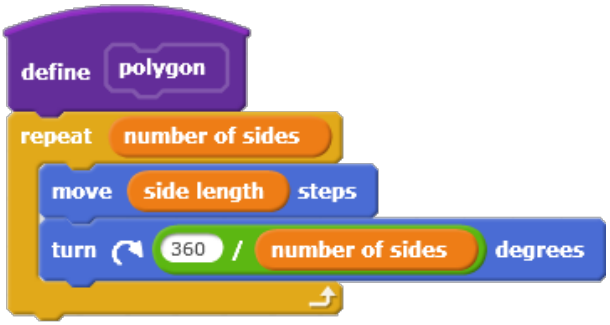




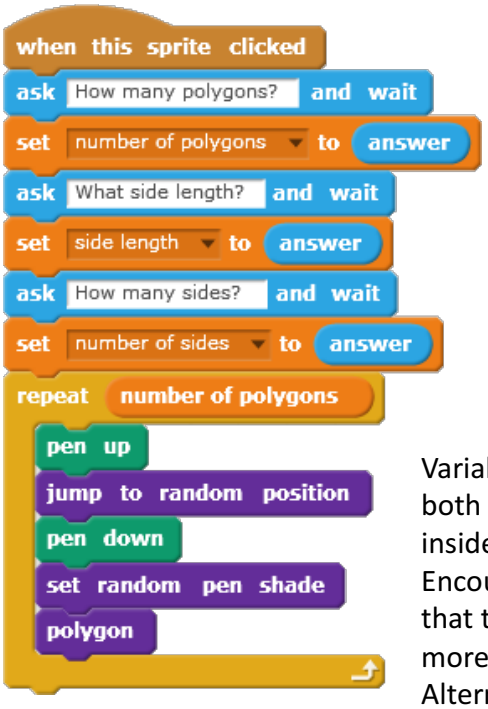
The Beetle asks two questions and keeps the answers in variables **side length** and **number of sides**. Both variables are then used to draw a polygon, **number of sides** is used twice. (4b) is an alternative solution using our own block **polygon**.

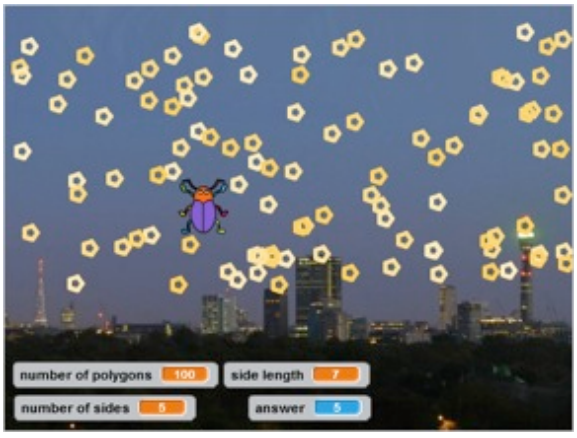
b





5



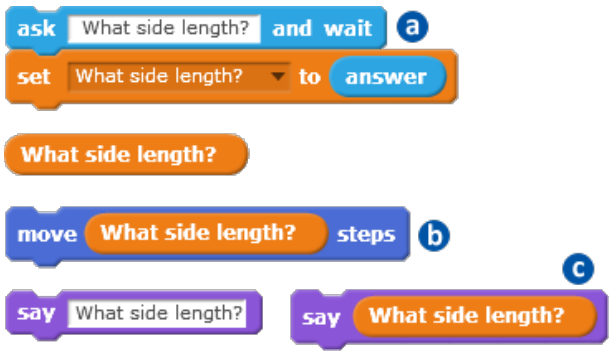


Variable **number of polygons** is used as the **repeat** value, both **side length** and **number of sides** variables are used inside the **polygon** block definition.
Encourage pupils to make and use the **polygon** block so that the **when this sprite clicked** script is shorter and more comprehensible.
Alternatively, both **pen up** and **pen down** blocks might be moved inside the **jump to random position** definition.

ADDITIONAL SUPPORT CONTINUED

Choosing names for variables

Although pupils are encouraged – and supported by Scratch – to give any name to variables as they wish, a name can easily become confusing, instead of helpful. To the right, you see a real example from a school: a pupil used the text of the question as the name of a variable, the value. The confusion may occur when the variable is then used in other blocks, see (b) and (c).



The name of a variable should reflect what the 'answer' represents. In this case it could be e.g. **length** or **side length**...

EXTENSION IDEAS

Explore the following **Surprising polygons:**

Star polygons are drawn by connecting one vertex of a regular polygon to another (non-adjacent one) and repeating until you return to the start (the first one in the row above). To demonstrate what is happening, try walking around a five-pointed star, paying careful attention to your turning. You will see the four walls of the room twice, not once as you would for a regular polygon. You have turned a total of 360° twice, or 720°. All of the star polygons here are found by using multiples of 360°.

Module 5 • Investigation 1 • Activity 5.1.4

The Sky at Night

Learning Objectives

Explore how to draw towers of squares of different heights and in random positions.
bridgE to mathematical quantities and formulas to calculate side length or height of a tower.

Activity Instructions

Pupils continue in their own version of project **5-Polygon Firework**, or open the **5-Polygon Firework INT2**, **Save as a copy** (online) or **Save as** (offline) and rename. The final version of this project at the end of Activity 5.1.4 will be **5-Polygon Firework FINAL**.

1. Pupils make their own block **square** using the **side length** variable to draw it. They build a script: when the Beetle is clicked, it will ask *What side length?* then draw a tower of 10 small identical squares atop each other.
2. It is not necessary to have only 10 floor towers. Pupils make a new variable **number of floors** and build a more powerful block **tower** which will draw a tower of identical squares – defined by the **number of floors** variable.
3. Pupils modify their script for the Beetle to first ask for the number of floors and save the answer in variable **number of floors**. Then it will ask for the side length and save the answer in variable **side length** and draw a corresponding tower.
4. **[Extension]** Pupils generalise their solution so that the script draws a night skyline of many towers of different numbers of floors and different side lengths. The script will repeat the **tower** part, asking each time for the input value – or, alternatively, setting them at random with an appropriate minimum and maximum. All towers will be scattered at random.
5. **[Extension]** Pupils create a sky full of polygon fireworks, then a skyline of towers, combining all previous steps. Note that for firework part and for the skyline part it might be useful to have two different **jump to random position** blocks, so that the whole scene could be created in one click.

Mathematics Connections

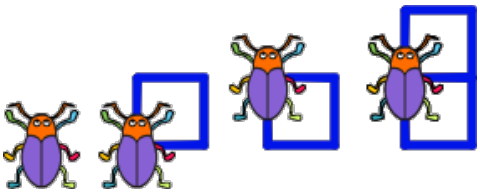
Draw out the structure of the towers on the white board, indicate the starting and ending point of the Beetle drawing it. *Where do we need to move to start the next floor? What is the algorithm?* [first draw a square then move upwards the side length]

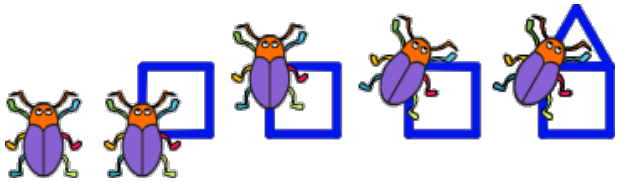
Connect the Beetle output with mathematical quantities and formula. E.g. *How tall is the tower?* Write as a simple formulae. [height of tower = side length * number of floors]
Pose questions: *If a tower is 120 tall, and side length of the square is 15, how many floors does it have?*

CONNECTIONS TO Y5 SCRATCHMATHS

*Please note the **blue** numbers on the left link to the numbered steps in the **activity instructions***

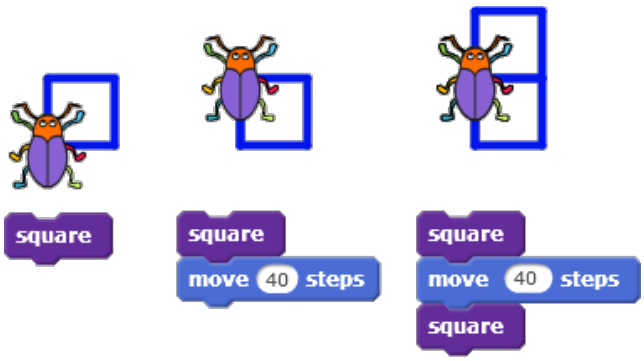
1 In Module 2, Activity 2.2.1 pupils drew a square and a equilateral triangle. In Activity 2.2.3 they were encouraged to give a name to their square script, making their own **square** block. In Activity 2.2.4 they made another new block – **triangle** and were asked to use these new blocks to draw a tower of two squares and also a house.



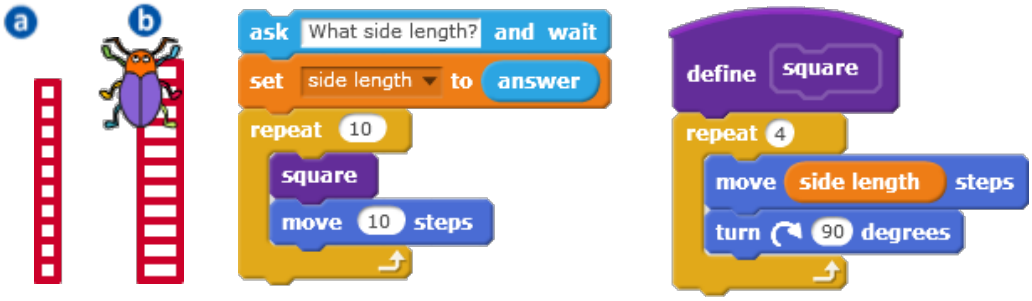


In the additional support of that activity we suggested to encourage pupils to build a script and run it step by step thinking about the questions below:

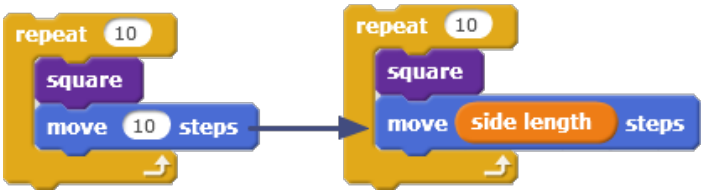
- Where will my Beetle finish after drawing the first **square**?
- Which direction will it point in?
- Where exactly do I want it to draw the second **square**? Which block will make the Beetle get there?
- Will it then point in the correct direction?
- Where will it finish after drawing the second **square**?



Now in our new **square** block from Activity 5.1.4 we make use of a variable **side length** which is set by using the **ask** and **answer** blocks. Some pupils may come with the following solution based on a generalization of the Y5 task:



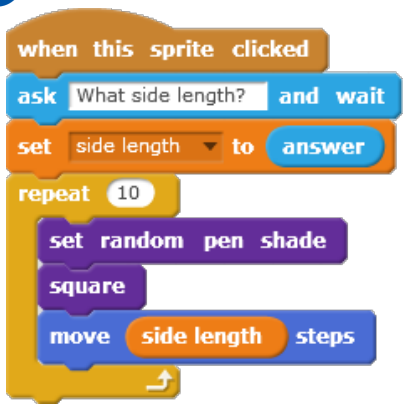
While it looks like a correct solution for the situation when we set the value of **side length** to 10 (example (a) above), it is easy to demonstrate the problem if we set the value of **side length** to be e.g. 20 (example (b) above). The Beetle needs **move** exactly by **side length** from one square to the next one, whatever value it is.

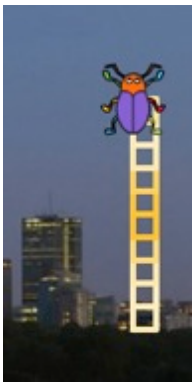


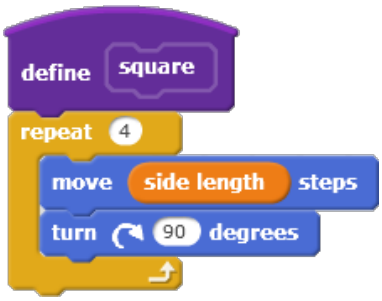
ADDITIONAL SUPPORT

*Please note the **blue** numbers on the left link to the numbered steps in the **activity instructions***

1



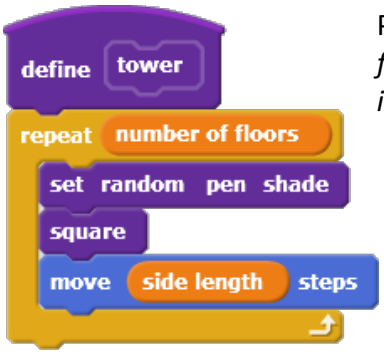




In the definition of the **square** block we use the **side length** variable, the value of which will be set in the **when this sprite clicked** script by **ask** and **answer**.

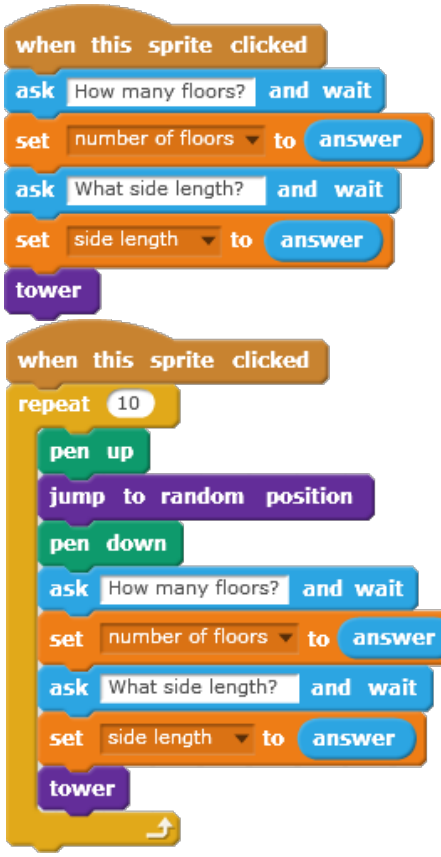
In the script for the Beetle each square can have the same colour or may have different pen shades or different pen colours, pupils can choose.

2

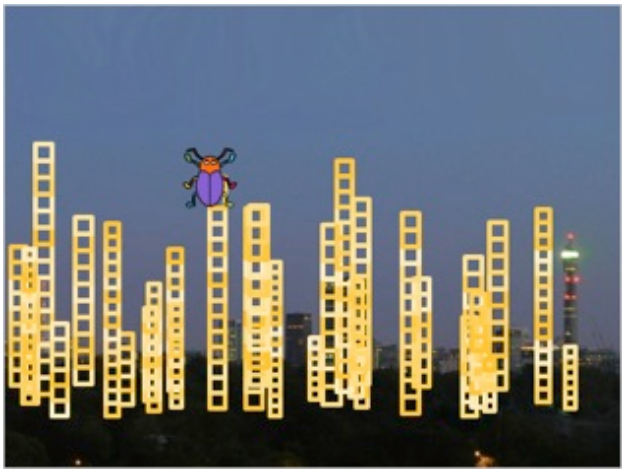


Pupils should start using the vocabulary: *The sprite asks for … then saves or keeps the answer (or answered value) in a variable …*

3

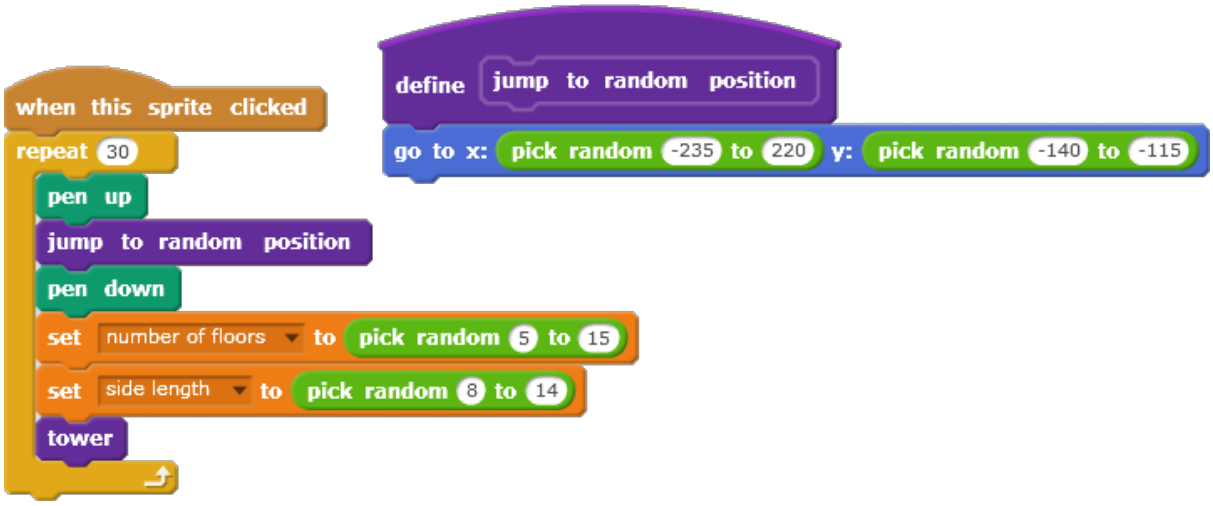


4



ADDITIONAL SUPPORT CONTINUED

Alternative solution with **number of floors** and **side length** set at random, without any asking: Carefully choose an appropriate minimum and maximum for each value, including the ranges for **x position** and **y position** for random jumping.



5