ESCHER X NENDO

BETWEEN TWO WORLDS

2 DEC 2018 – 7 APR 2019

DIGITAL CREATIVES: TESSELLATE BY CODE

WORKSHOP INSTRUCTIONS

ONLY IN MELBOURNE

AT NATIONAL GALLERY OF VICTORIA

DIGITAL CREATIVES: TESSELLATE BY CODE WORKSHOP INSTRUCTIONS

The NGV Digital Creatives Programs enable students to use digital technologies to explore, interpret, create and share works of art.

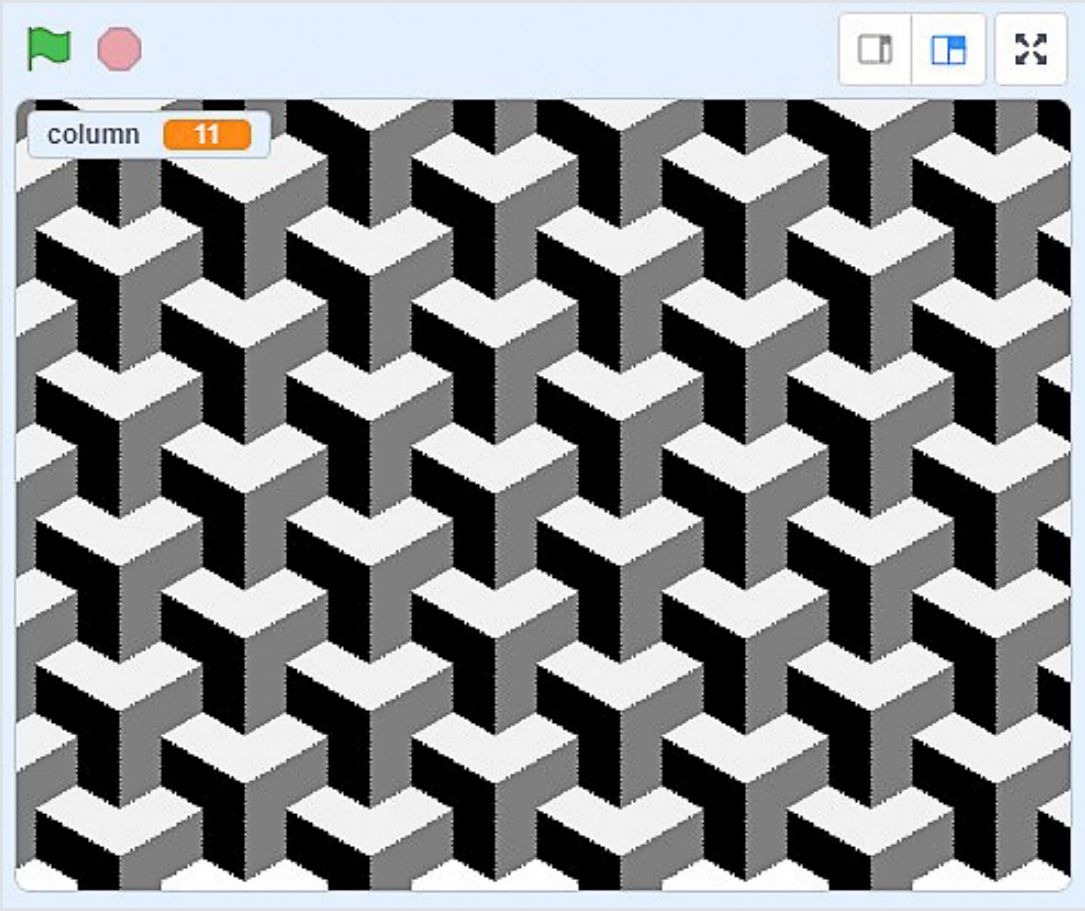
SUPPORTED BY

AIM

In this workshop, we will use computer code to generate a tessellation. We will use Scratch – a block-based visual programming language – to make a monochromatic rotation/ translation tessellation that looks like stacks of neatly repeating cubes. The pattern is made up of one shape rotated around a fixed point that repeats across the surface. The final pattern will look like this:

OUTCOME

The finished project can be viewed **here**.



M. C. Escher
Regular division of the plane no. 123 (Fish)
April 1964 (detail)
pen and ink, pencil and watercolour
Escher Collection, Gemeentemuseum
Den Haag, The Hague, the Netherlands
© The M. C. Escher Company, the Netherlands. All rights reserved

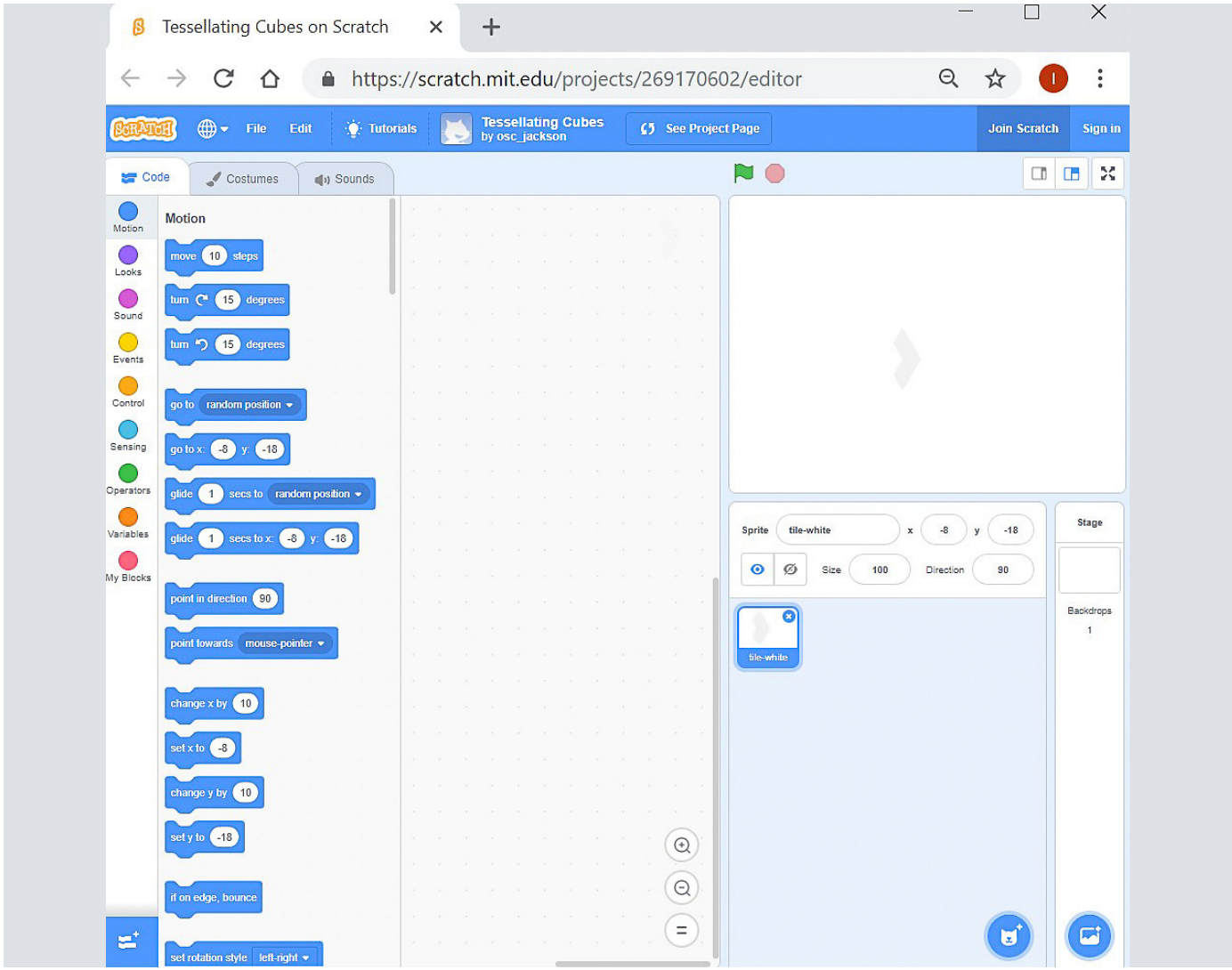
STEP BY STEP INSTRUCTIONS

STEP 1

Open the Tessellating cubes project in Scratch. **(https://scratch.mit.edu/projects/269170602/#editor)**

You'll notice a single sprite that looks a bit like an arrow, called *tile-white*. If you click on the **Costumes** tab, you'll see the same sprite with two more costumes: *tile-grey* and *tile-black*.

Return to the Code tab. Click and drag the tile-white sprite onto the stage.

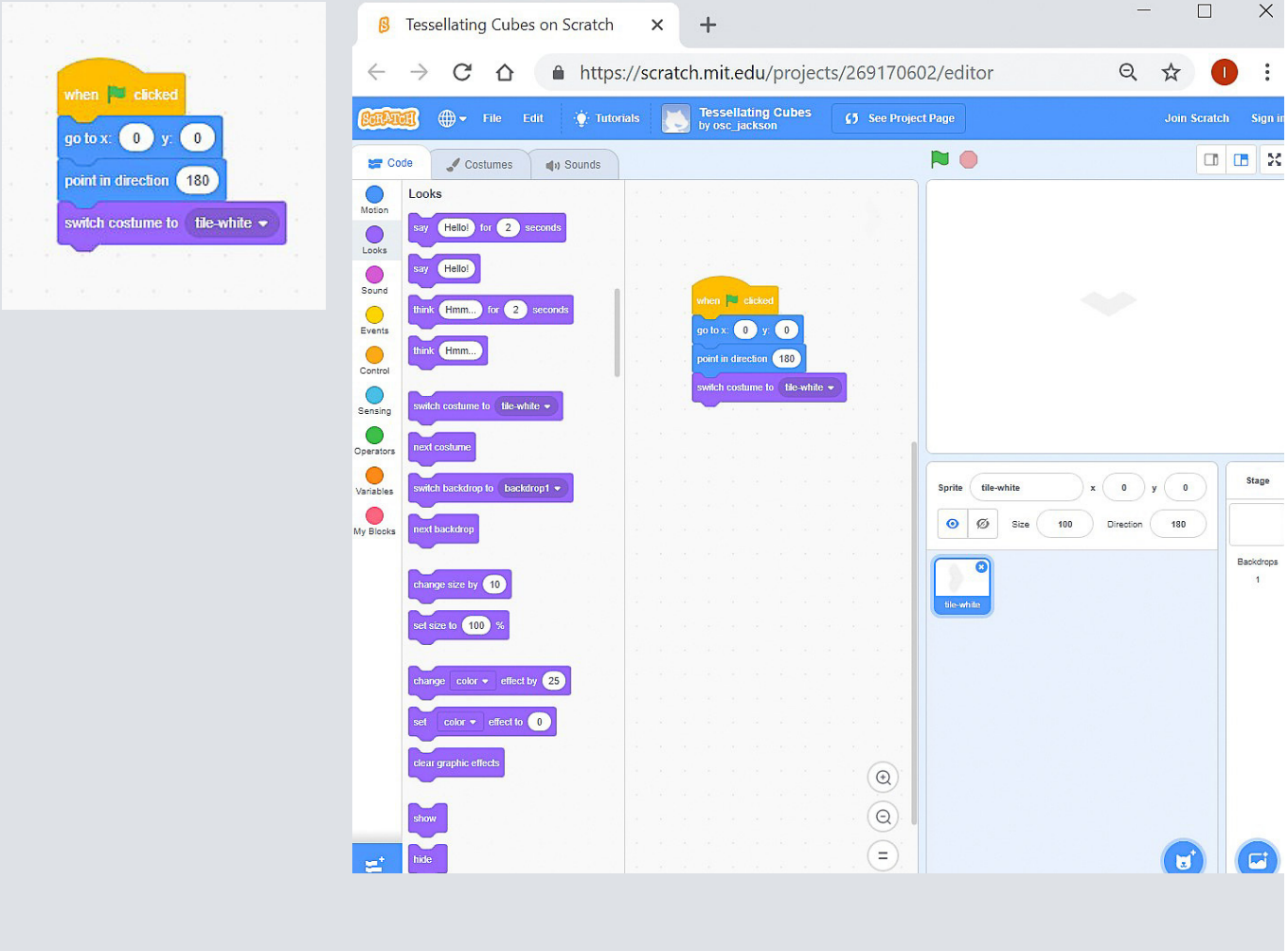


STEP 2

We want the *tile-white* sprite to start in the middle of the stage and to point downwards.

To do this, you need to make a *script* (this tells your sprite what to do). Begin by dragging the **event** block, **when (flag) clicked**, to the script area. Then, in the **Motion** blocks, click and drag the **go to x_ y_ block** so that it attaches to the **event** block.

Change the numbers to x: 0 and y: 0. This will make your sprite move to the centre of the stage. Next, drag the **point in direction block** to the script area and change the number to 180 (90 is the default setting). From the **Looks** block category, add the **switch costume to *tile-white*** block to your script.



STEP 3

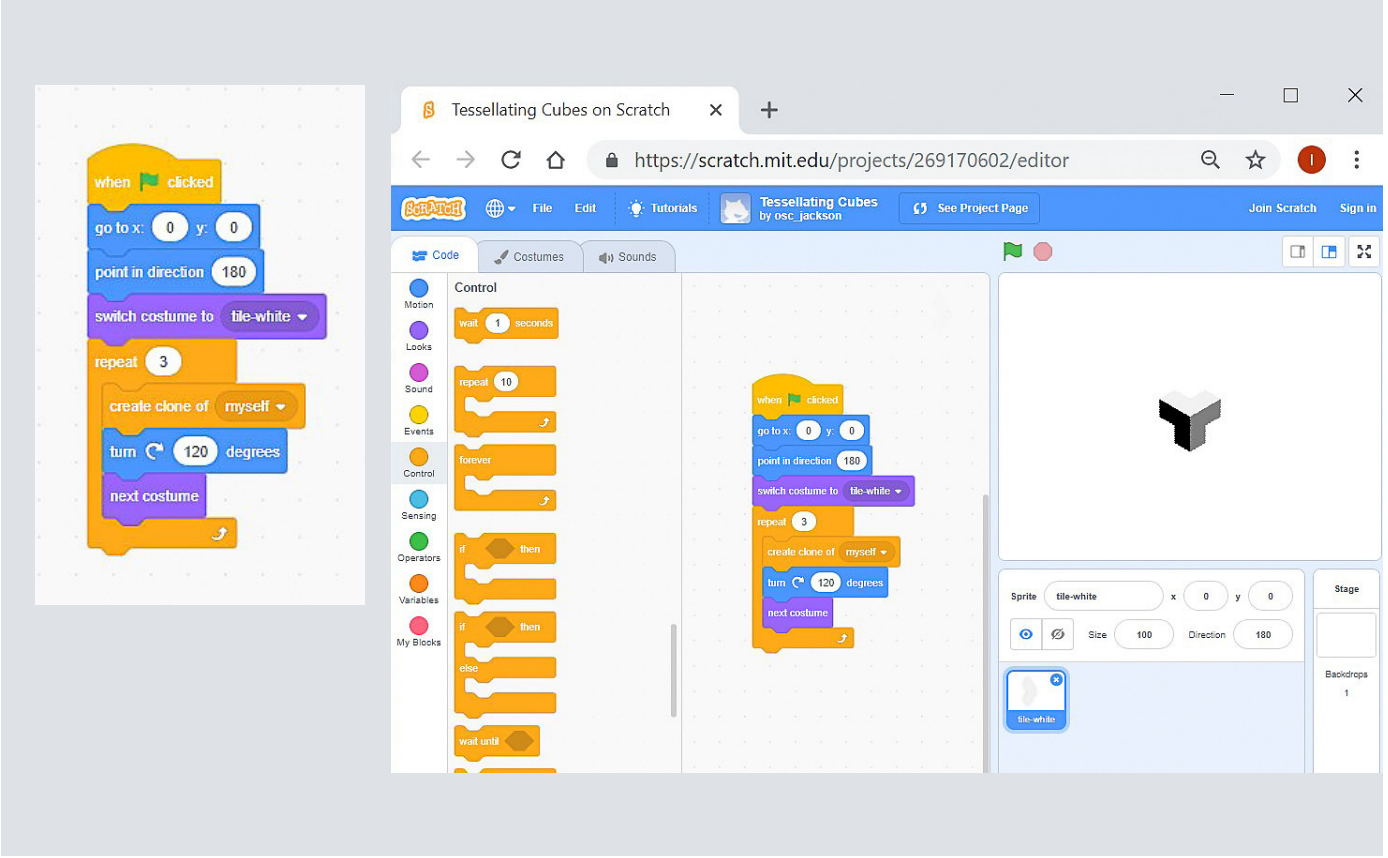
To repeat the shape we can make a clone. In the Control block category, click and drag create clone of myself to the script area and add this to your line of block codes.

Next, drag the motion block turn (clockwise) _ degrees and change the number to 120.

We want the clone to be a different colour, so click and drag the next costume block from the Looks category.

We want to repeat this set of instructions three times. In the Control block category, click and drag the repeat block to sit underneath the switch costume to tile-white block, as shown in the image below. Now change the number to 3 (to repeat 3 times).

We have created the shape for our tessellating pattern.



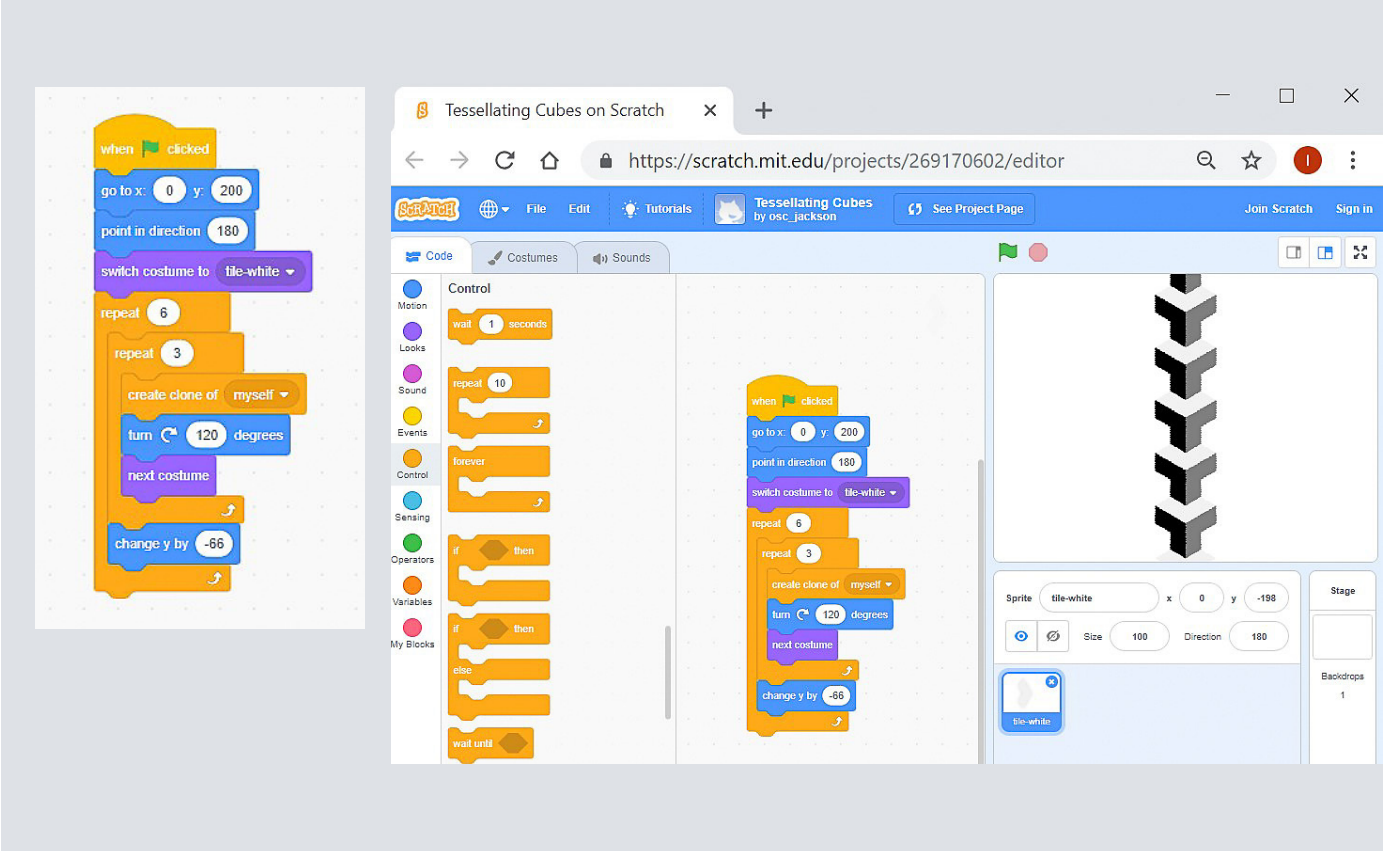
STEP 4

We now want to make the shape repeat in a column by changing the vertical axis, which is the y axis.

Let's start by moving our shape to the top of the stage. Change the initial y coordinate to y: 200 in the **go to x:0 y:0 block** (this should be the second code block in your script). Next, click and drag the **motion** block, **change y by** to the end of your script. Set the number to -66.

Wrap another repeat block around the **repeat 3** and **change y** blocks, as shown. Change the repeat number to 6.

Click the green flag to see your shape repeat 6 times!



STEP 5

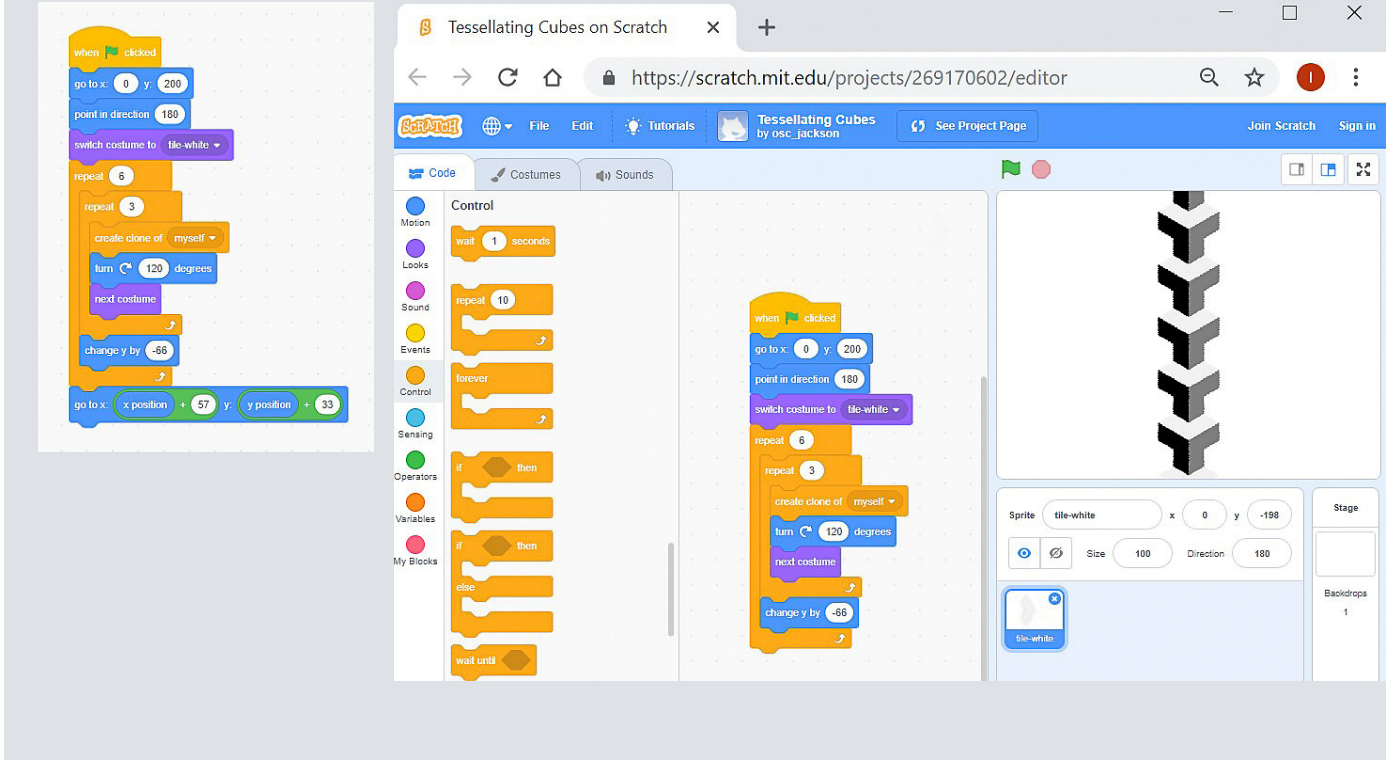
Operator

Now we are going to make the next column, but this time we are starting from the bottom and going up. First we will reset the starting point using an **operator** block (Operator blocks script maths operations).

Begin by adding the **motion** block: **go to x: y:** . In the **operator** block category, drag this block to sit inside both the x and y brackets. Next, add the **motion** blocks **x position** and **y position**. Modify the equations to read (x position + 57) and (y position + 33)

Your script should look like this:

To make your shape repeat vertically (going up), copy the series of blocks created in Step 4 and add them to your script. This time, **change y by** 66. Your shape should tessellate up and down in 2 columns, like so:



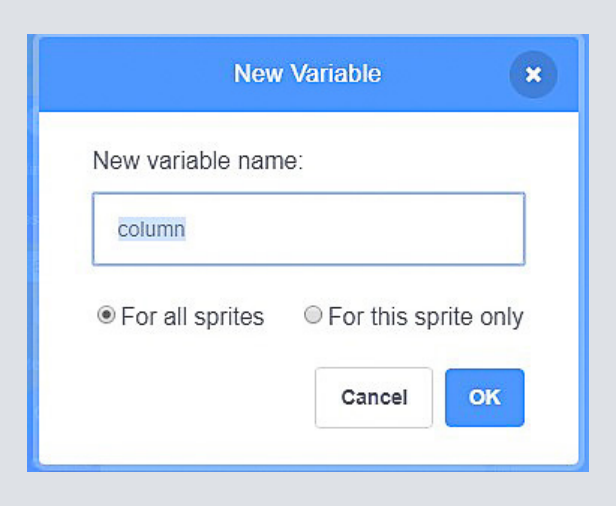
STEP 6

Variables

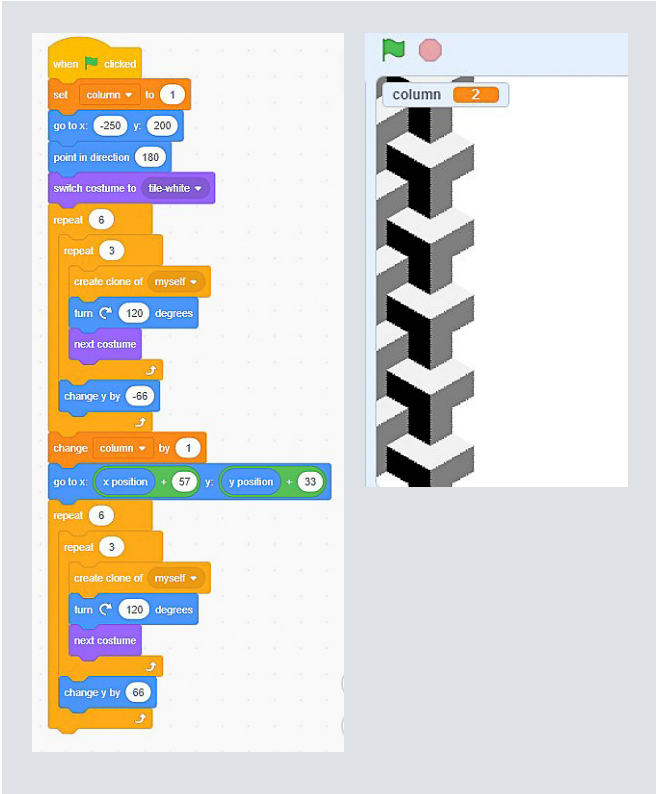
We want to fill our stage with this tessellating pattern, so now let's reset our starting point to the edge of the stage, like so:

To continue the pattern, you could repeat the same steps, but there would be a lot of blocks of code.

Instead, we can make the program recognise odd and even numbered columns and change the action based on that. Go to the **Variables** category of blocks. Click **Make a Variable** and give this any name. Be sure to click the option: **For all sprites**.



You will be using the **Set columns to 0** and **Change columns by 0** blocks from the **Variables** category. Set the number to 1 for each block. Click and drag each block to your script, as shown:



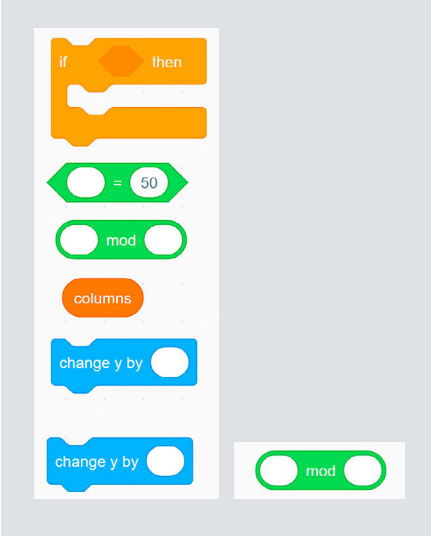
STEP 7

We want odd numbered columns to form downwards, and the even ones, upwards.

We can use an **if/else Control** block to differentiate between odd and even columns. If our column variable is odd, the y value is minus (–) and if it's even, y is positive (+).

You will need to create a series of blocks to tell the program: 'if the column is an odd number, place the next clones down by 66 each repetition, otherwise, place them up by 66'.

To achieve this, you will need to use these blocks:

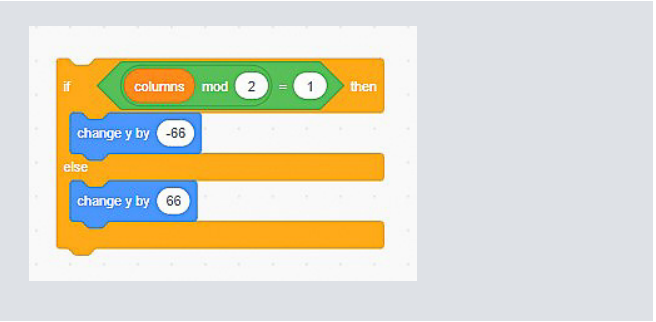


(mod means modulo, which is a way to find every second column (1, 3, 5 etc.))

HANDY TIP

It may help to find these blocks and place them in your script area separately.

Arrange the blocks and add the numbers, as shown:



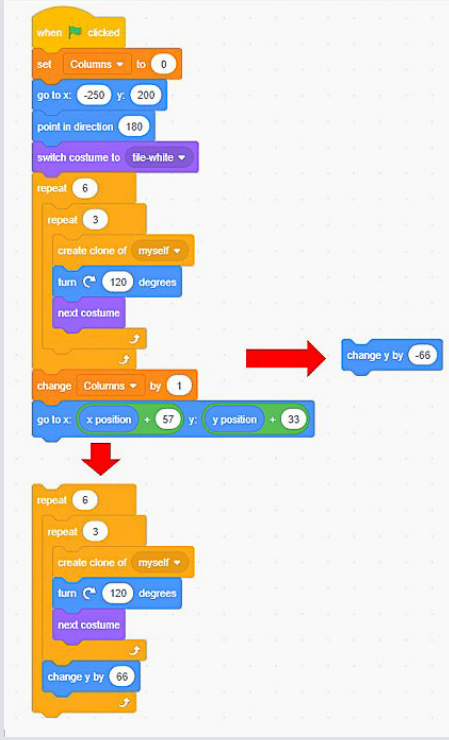
STEP BY STEP INSTRUCTIONS

NOW USE WHAT YOU HAVE LEARNT TO MAKE YOUR OWN TESSELLATION!

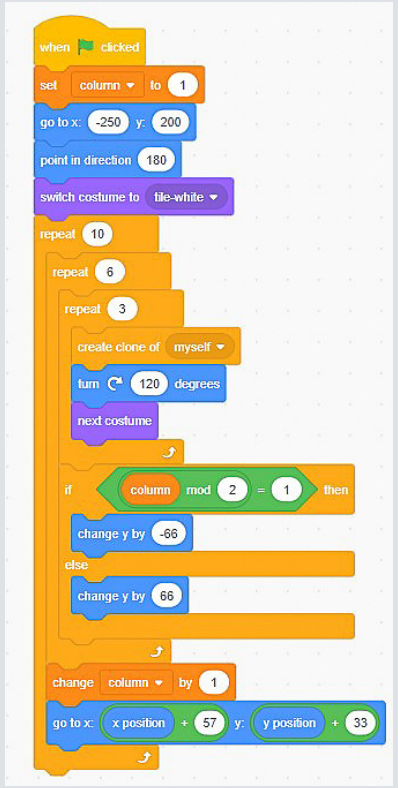
STEP 8

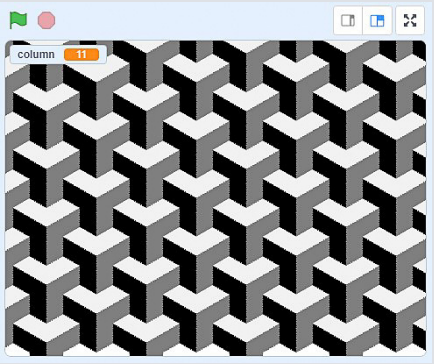
Now, we are going to replace some of the code blocks used earlier. This will simplify the code and make your tessellation come to life!

Remove and delete the blocks as shown below.



Add your **if/else** series of block codes, from step 8, and the repeat 10 block, like so:





Test your code to see if it works.