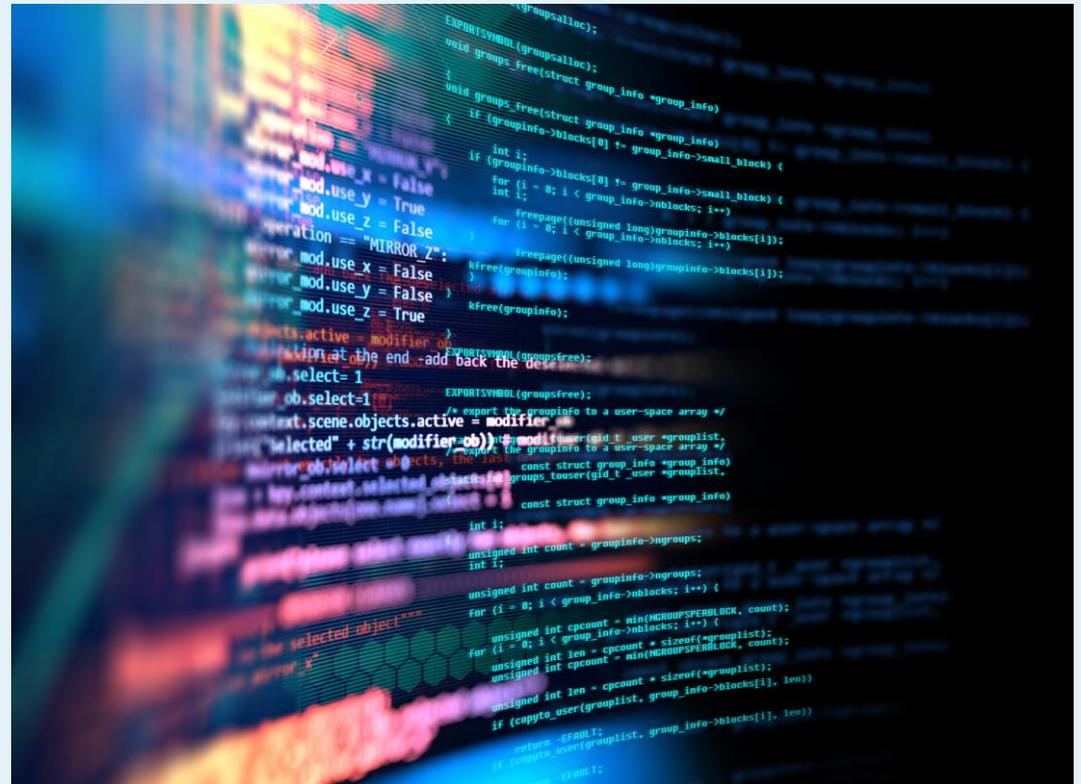


Student workbook

Introduction to EduBlocks

Code Playground



Introduction to EduBlocks

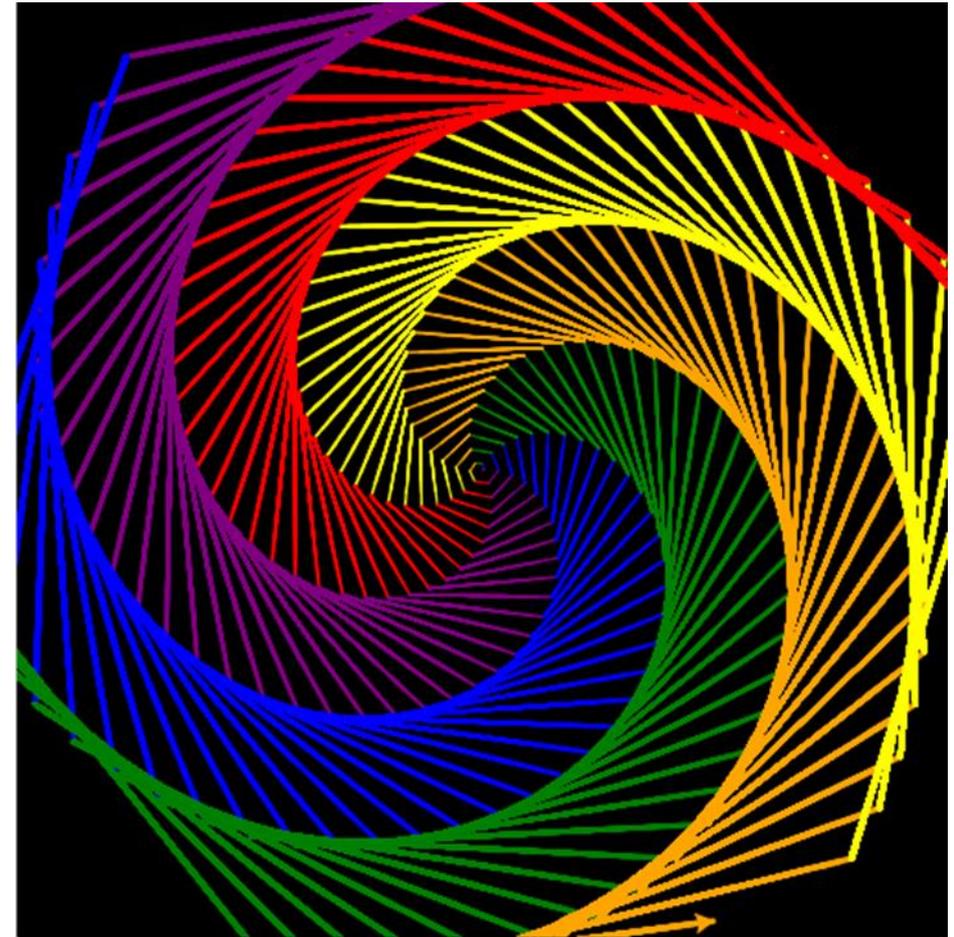
EduBlocks project

EduBlocks is a free tool that helps anyone learn how to code with text-based languages like Python, using familiar block coding as used in Scratch.

This session will allow you to use blocks to create a colourful spiral and be able to see what block coding looks like in Python coding language.

This project is designed for EduBlocks 4.0 and will show you the basics of what a turtle is and how you can control it.

Hope you enjoy the project!



Introduction to EduBlocks

EduBlocks project

Step 1

“Turtle” is a Python feature like a drawing board, which lets us command a pen to draw and move around the board using set functions.

To get started, we need to build the background and create a list of colours for our pen, or turtle as we will now call it. Import the turtle, then we need to identify the turtle and screen settings.

The numbers (255, 0, 0) in the screen colour block are the Red, Green, Blue (RGB) colour values, so:

255, 0, 0 is Red

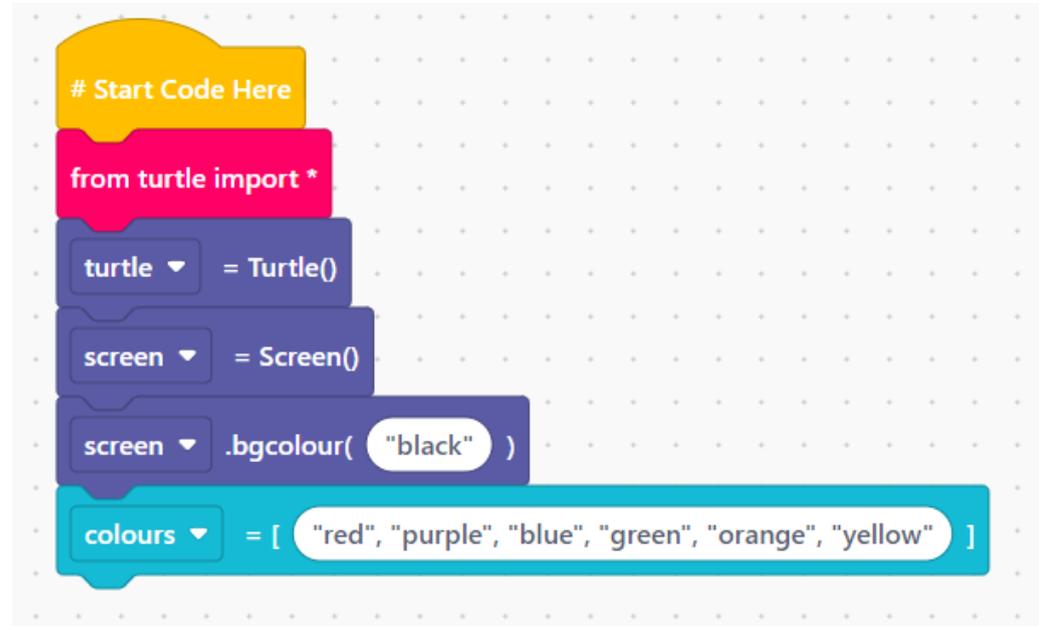
0, 255, 0 is Green

0, 0, 255 is Blue

We have changed the screen background colour to “black.”

You can experiment with the different RGB colour values to make your own, or alternatively add the name of the colour in quotation marks. Quotation symbols are used to create a string object in Python which assign it to a variable. In this code, “black” is assigned to the screen background and the list of colours to the colours variable.

Block code



Python code

```
1 # Start Code Here
2 from turtle import *
3 turtle = Turtle()
4 screen = Screen()
5 screen.bgcolor("black")
6 colours = ["red", "purple", "blue", "green", "orange", "yellow"]
```

Introduction to EduBlocks

EduBlocks project

Step 2

Now we have coded the background and defined the turtle, we need to add code to tell the turtle where to move to.

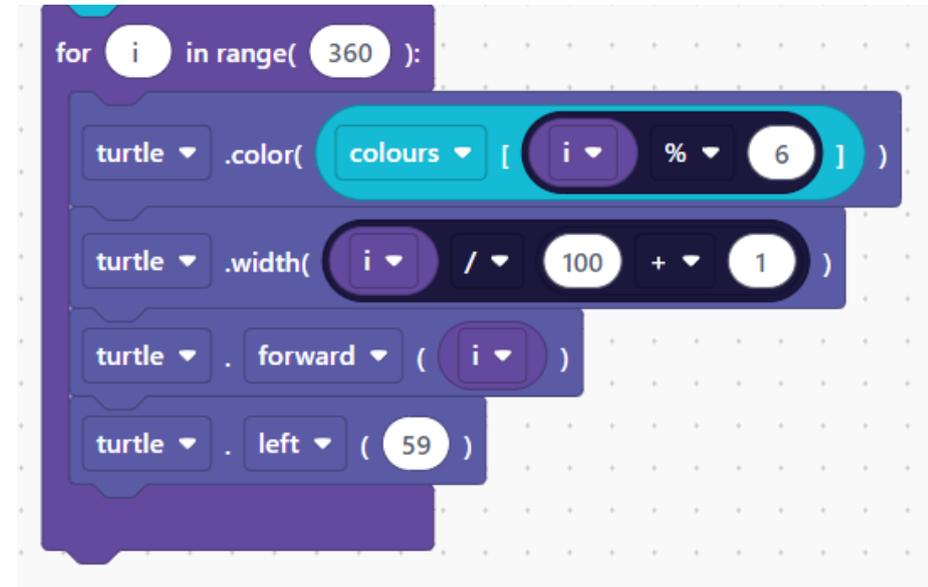
We start by adding a loop, which will run the code inside it many times. In this code, we are asking the turtle to draw 360 lines.

Set the turtle colour to use the colour list variable, then we can use a Math block with a % operator (or modulus) to assign values to variables. The `i` block is used as a temporary variable and 6 is the number of colours in our colour list.

Now we need to use another Math block to set the line width to get wider each loop.

The remaining turtle blocks set the direction and angle of the turtle to create the spiral.

Block code

A screenshot of the EduBlocks programming environment showing a loop of four turtle blocks. The first block is a 'for' loop block with 'i' in a variable input and '360' in a range input. The second block is a 'turtle .color()' block with a 'colours' dropdown menu, an 'i' variable input, a '%' operator, and a '6' constant input. The third block is a 'turtle .width()' block with an 'i' variable input, a '/' operator, a '100' constant input, a '+' operator, and a '1' constant input. The fourth block is a 'turtle .forward()' block with an 'i' variable input. The fifth block is a 'turtle .left()' block with a '59' constant input.

```
for i in range( 360 ):
    turtle .color( colours [ i % 6 ] )
    turtle .width( i / 100 + 1 )
    turtle .forward( i )
    turtle .left( 59 )
```

Python code

```
7 for i in range(360):
8     turtle.color(colours[i % 6])
9     turtle.width(i / 100 + 1)
10    turtle.forward(i)
11    turtle.left(59)
```

Introduction to EduBlocks

EduBlocks project

Step 3

That's it! You have coded the complete spiral! Now run your code to see what it looks like. Change the colours, angle or width of the Turtle to create new shapes. Can you find how to make the Turtle move faster?

Python code

```
1 # Start Code Here
2 from turtle import *
3 turtle = Turtle()
4 screen = Screen()
5 screen.bgcolor("black")
6 colours = ["red", "purple", "blue", "green", "orange", "yellow"]
7 for i in range(360):
8     turtle.color(colours[i % 6])
9     turtle.width(i / 100 + 1)
10    turtle.forward(i)
11    turtle.left(59)
```

Block code

The block code is as follows:

```
# Start Code Here
from turtle import *
turtle = Turtle()
screen = Screen()
screen.bgcolour("black")
colours = ["red", "purple", "blue", "green", "orange", "yellow"]
for i in range(360):
    turtle.color(colours[i % 6])
    turtle.width(i / 100 + 1)
    turtle.forward(i)
    turtle.left(59)
```

Notes

Code Playground

Barclays Bank UK PLC is authorised by the Prudential Regulation Authority and regulated by the Financial Conduct Authority and the Prudential Regulation Authority (Financial Services Register No. 759676). Registered in England, Registered No. 9740322. Registered Office 1 Churchill Place, London E14 5HP.