# Lesson 2:

Introduction

The class will be introduced to Edublocks Python code by creating a sequence of code that will draw shapes and patterns on the screen.

## Learning Objectives

How to use edublocks

To understand how a sequence of code works.

How to use different loops to repeat a sequence.

## Key Vocabulary

Sequence, selection and iteration, modules.

## Preparation

#### Subject Knowledge

An understanding of creating code in a block based environment and understanding how key coding concepts are across all languages and can be illustrated using different languages.

Possible Misconceptions

That block languages are not as powerful as typed languages.

#### Pedagogy

Ideally the class will each have access to a computer and complete the tasks individually. The lesson can be completed with 1 computer per 2 children.

#### You will need

A computer running Windows / Mac or Linux or Chromebook

A web browser (Firefox, Edge, Google Chrome, Safari)

## Assessment Opportunities

Understand that coding concepts are not tied to just one language and that skills learnt in one language can be applied to others.

Basic orientation of the edublocks interface.

## Outline Plan

This is a quick overview of the accompanying slide deck.

| Introduction  (Slide 3 - 4)  5 Minutes | Turtle is a Python module ( a pre-written library of code that we can use in our code) that enables learners to draw images on the screen.  Turtle was created as part of the Logo language in 1967 as a means to learn how a sequence of code works. It used a robot connected to the computer to draw patterns on paper.   * With Turtle we can create a sequence of code in Python and see the output as an image. * Using different loops to repeat sections of the sequence. * Using maths and angles to draw shapes using simple equations. |
| --- | --- |
| (Slide 5)  5 Minutes | To start Edublocks, ask the class to open a web browser and type in  app.edublocks.org/editor  Select Python 3 as the mode, give the project a file name and click Create  Now is a good time to remind the class on how the edublocks interface works.  To the left are the blocks, organised by function.  Blocks are dragged from the left, into the centre of the screen and this is where we build our sequence.  In the top right is the RUN button which will run the code. |
| 5 Minutes  Slide (6) | To use Turtle we first import the Turtle module, found in the Turtle section of blocks.   * Drag the **import Turtle** block into the coding area. * Then we drag the **turtle = Turtle()** block and connect it to the previous block. This block enables the learner to quickly use the Turtle module without using the full syntax to call a class. This enables the class to get started much faster. * We then drag the **screen = Screen()** block and attach it to the previous. This tells Python that we wish to draw on a section of the screen. |
| 10 Minutes  (Slide 7) | Here the class is challenged to draw a square using Turtle.  Turtle has blocks to move around the screen.  **turtle.forward(90)** will move the turtle 90 pixels (steps) forward. This block has a drop down selection to move **left, right and backward; these** options can be used to rotate the turtle, or move backwards.  By default the turtle will face to the right of the screen. So moving forward will move the turtle 90 pixels to the right. If we use **turtle.left(90)** then the turtle will spin 90 degrees anti clockwise, pointing the turtle upwards.  The class need to understand the basic movement concepts, and use them to draw a square. Every learner should achieve this before moving on.  All of the blocks on the slide can make a square, the learner just need to duplicate a few to finish the task. |
| Group Activity  5 Minutes  (Slide 8) | Using a loop with a definite iteration, a for loop. We can iterate the loop four times to perform two steps that will move the turtle forwards, and rotate 90 degrees.  The class should understand that a definite iteration loop, a for loop, is the best option to repeat a sequence of code for a set number of times. |
| Group Activity  5 Minutes  (Slide 9) | Here we reuse the definite iteration loop from slide 8. But this time we use a calculation to determine the exterior angle required to draw a specific shape.  For example a square has four sides so to calculate the turn angle we use  360 / 4 = 90  For a triangle the class need to work out the angle by altering the calculation.  360 / 3 = 120  Replacing the ?? with 120 will draw a triangle on the screen.  Please note that Turtle uses the exterior angle to control the direction in which it faces. Some learners may expect to use interior angles. |
| 5 Minutes  (Slides 10-11) | Using two different loops.  **while True:** which runs the pattern code forever, and **for i in range()** which will draw the triangle shape. |
| Group Activity  10 Minutes  (Slides 12-14) | Slide 12 introduces changing the colour of the pen which is used to draw upon the screen.  Colours are a mix of Red, Green and Blue (RGB) and they are given values from 0 to 255. With 0 meaning none of that colour, and 255 meaning all of the colour.  So red would be 255,0,0.  The width of the pen can also be changed to create thick / thin lines.  Slide 13 is a challenge to the learners with criteria on which they can be assessed. All of the code to draw a pattern has been shown in the class.  Slide 14 is there as an example to the class. If some get stuck, or require clarification this slide can be used to illustrate the concept. |
| Group Activity  5 Minutes  (Slide 15) | Turtle has a special block, **turtle.circle()** which is a quick and easy way to draw a circle. The value **turtle.circle(50)** means that a circle with a radius of 50 pixels will be drawn on the screen.  Can the class use this to draw a repeating pattern? |
| Plenary  2 Minutes  (Slide 16) | Here we recap the learning from this lesson.   * How to control Turtle and write a sequence of code. * How we used different loops to perform certain actions (drawing triangles and repeating the code indefinitely) * We learnt how to calculate angles necessary to draw a certain shape. |
| Next Time  1 Minute  (Slide 17) | We will continue our learning with edublocks, and learn how to capture data from a user, and use it to control the code. |