# Lesson plan - “Slider mu plotter

## Using the MonkMakes Slider for micro:bit component

## Introduction

The particular focus of the lesson is collections of data and specifically tuples. The program is relatively straightforward to understand, meaning that there will be more time given to either the modify or make sections of PRIMM to enable the students to build their engagement and consolidation of the concepts learned during the scheme.

Learning objectives

* To successfully set up a micro:bit/slider plotter circuit and flash code to the micro:bit using Mu
* Apply knowledge of tuples
* To be able to read and interpret a program containing a while loop
* To be able to read and interpret a program containing the time module
* To be able to successfully modify a program containing multiple programming constructs

## Keywords

plot, index, set, dictionary, list, tuple, while, for, range, variable, function, in-built, user-defined.

## Preparation

**Subject knowledge:**

This lesson is suitable for a class who already have a basic knowledge of programming using python. The coding for the micro:bit is written in a reduced version\* of python called micropython but for practical purposes the syntax will appear the same to the user. In addition the students should have already used the micro:bit, practising simple set up and smaller micro:bit specific programs so that they are used to building and flashing programs. Initially, building the circuit for the “slider mu plotter” is a question of copying a diagram so a practical demonstration of handling the components gently and clipping with alligator clips would suffice.

**Pedagogical approach:**

The lesson is planned using the PRIMM pedagogy which stands for:

P - Predict

R - Run

I - Investigate

M - Modify

M - Make

For this reason it is important that the class have quick and easy access to the coding of the program which can be accessed via: https://tinyurl.com/4a4422ub

The very first step is for the students to view the code and attempt to predict what the code does. We use a tool called a **code conversation** which provides the teacher with a conversation style **talkthrough** of the micropython code for you to support your students’ developing knowledge.

**Practical set-up and development environment:**

For this lesson you need to be working in the Mu environment, to take advantage of the plotter fetaure Mu (https://codewith.mu).

**Resources that you need:**

* Presentation
* Starter worksheet questions
* Starter worksheet answers
* Code conversation full solution
* Code conversation template (level 1 - basic)
* Code conversation template (level 2 - intermediate)
* Plenary worksheet
* Plenary worksheet answers
* Hardware Per pair - 1 micro:bit, 1 USB/micro USB cable, 3 alligator clip leads, 1 slider
* Access to Mu development environment

## How the students’ progress is assessed

**Slider mu plotter**

The presentation itself begins with a starter testing whether the students know the different functions of the four main collections (tuple, list, set and dictionary). It is interspersed with pair or whole class discussion opportunities. As this is predominantly a practical lesson observing the students’ success in flashing the program, compiling the circuit and making small successful modifications to the program will also enable you to assess their progress. The learners also complete a worksheet at the end of the lesson which demonstrates their understanding of key principles conveyed during the lesson.

## Plan (with approximate timings)

|  |  |
| --- | --- |
| Starter activity 5 mins | **Collections revision**  The students are given a worksheet containing questions about the features of different collections: dictionaries, sets, lists and tuples. The students answer the worksheets individually then feedback as a group to ensure that they understand the feature of the different collections. |
| Focus tasks 5 mins | **Tuple consolidation and explanation of slider mu plotter**  In this section the students are given some explicit teaching about the use of the mu plotter. As there is a one item tuple in the program this links to some explicit teaching about the syntax of a one item tuple. |
| PRIMM 8 mins | **Predict: students view code and attempt overview code conversation**  The students are shown the code and are asked to predict what the whole program does i.e. an overview in a couple of sentences.  For example: This program uses the python editor program Mu. It has to be used with Mu because it uses Mu’s plotter facility to plot values sent out from the micro:bit to your computer over USB. When you move the slider about a reading “data” is sent to the Mu plotter and you can see the values change on the Mu plotter on your computer screen. |
| PRIMM 8 mins | **Run: students compile the circuit, flash and run the code**  Share the slide with instructions on the board to enable the students to build the circuit, flash and run the code. Ask them if the circuit behaves as they would have expected from the **predict** phase of the lesson. Encourage the students to move the slider up and down and see how the mu plotter responds. |
| PRIMM 13 mins | **Investigate: teacher and students use correct terminology to identify the syntax and features of the program**  Using the A4 worksheet, the learners attempt to complete the code conversation matching the correct explanation to each line of code.  There are three levels of the code conversation:  Level 1 - basic (some less challenging code lines to complete), Level 2 - intermediate (all code lines to complete)  Using the slides talk through the different steps in the program. |
| PRIMM 11 mins | **Modify: students modify the code to create a new program**  Students are invited to modify the program by first declaring what their new program should do, then by making the modification, saving the new program and testing it by flashing it to the micro:bit.  Suggestions for achievable alterations:-  Perhaps you could try putting other values into the tuple to produce a graph?  Could you reverse the output of the slider, the further to the right lower the number? |
| **Plenary**  5 mins | **Slider mu plotter plenary worksheet**  Students complete the plenary worksheet testing knowledge of lists. |
| **Homework** | **Consider what other modifications you could make to the program, still using the slider component** |

## The Author

This lesson plan and all its parts were created by Dr. Paula Beer of Beer Academic Consultancy in collaboration with Monk Makes Ltd.



Dr Paula Beer has taught Computer Science and IT education to new and established teachers since 2007. Her own research has focused on the use of play and collaboration in computer science. She enjoys supporting practicing teachers by designing accessible lesson planning materials to get students engaged in computer science through play and collaboration. Paula has also produced educational materials for The Raspberry Pi Foundation, been a secondary school teacher, written a successful book (Hello App Inventor!) and has previously worked in IT project management for a media company and for the NHS.