# Lesson plan - “your biggest fan”

## Using the fan component

## Introduction

This lesson gives you time to introduce the fan component to the learners. They will begin to understand the usefulness of functions and variables in programs. It also enables the students to understand how to program the buttons and vary the power input to the micro:bit’s components. It is important to note that the relay forms part of the circuit and is necessary as it allows an output of the micro:bit to turn things on and off. A micro:bit can turn an LED on and off directly, but anything more powerful requires something like a relay or a transistor.

## Learning objectives

* To successfully set up a micro:bit/electronics kit circuit and flash code to the micro:bit
* To be able to read and interpret a program containing functions
* To identify the presence of in-built and user-defined functions
* To be able to successfully modify a program containing functions

## Keywords

Function, relay, switch, button, variable, in-built, user-defined, subroutine

## 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 fan is a question of copying a photograph 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/yxud3gtt

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:**

We recommend either using the on-line Python editor (https://python.microbit.org) or Mu (https://codewith.mu).

**Resources that you need:**

* Presentation
* Starter worksheet
* Starter worksheet answers
* Code conversation solution
* Code conversation template (level 1 - basic)
* Code conversation template (level 2 - intermediate)
* Code conversation template (level 3 - difficult)
* Plenary worksheet
* Plenary worksheet answers
* Hardware Per pair - 1 micro:bit, 1 USB/micro USB cable, 5 alligator clip leads, 1 relay, 1 battery holder with battery.
* Fan speed controller program in micropython accessible to students
* Access to appropriate development environment (the on-line Python editor or Mu

## How the students’ progress is assessed

The first activity will enable you to determine if the students are familiar with the names of the components in the fan program. The presentation itself 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 | **Fan circuit worksheet**  Students are given a worksheet featuring photographs of the different fan circuit components. They are also given a list of the names of different components and are asked to match the photograph with the name and predict the purpose of the component. |
| Activity 1 5 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:  The program starts with power speed as 0 then when button a is pressed the speed of the fan increases until it reaches maximum speed. In addition the speed can be decreased via button b. |
| **Activity 2**  15 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. |
| Activity 3 20 mins | **Investigate: teacher and students use correct terminology to identify the syntax and features of the program**  Using 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 (more challenging code lines to completed), Level 3 (whole program to interpret). |
| Activity 4 10 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:-  Reduce the number of power steps that the program makes from 9 to 3.  Reduce the number of power steps and change output to letters i.e. Low (L), Medium (M), High (H)  Reduce the number of power steps and change output to scrolling words i.e. Low, Medium, High. |
| **Plenary**  5 mins | **Fan plenary worksheet**  Students complete the plenary worksheet testing knowledge of components and micropython. |
| **Homework** | **Consider what other modifications you could make to the program, still using the fan and relay components** |

\* For more information about the difference between python and micropython (used here) access - https://github.com/micropython/micropython/wiki/Differences

## 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.

