**Barefoot meets micro:bit**

Wildlife animations - Lesson 1

Recommended for

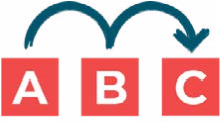
**ages 7-9**

Duration:

**1**

**hour**

Concepts and approaches covered



Sequence Outputs

# Overview

In this lesson, pupils familiarise themselves with the micro:bit and identify various components on the device. Pupils examine how to transfer programs to the device and undertake a related unplugged activity. Pupils create their own programs using the predefined images, then go on to create their own images by programming individual LEDs.

# Pupil objectives

■ I can explain what the micro:bit's components do

■ I can sequence code within the micro:bit MakeCode editor ■ I can design and code images using the micro:bit’s LEDs

# Before you start

Check the ‘Technical requirements’ section of the ‘Lesson overview’ document to ensure the micro:bit works successfully with your school.

Pupils can work individually or in pairs. micro:bit devices should be available from the beginning of the lesson for pupils to examine.

Ensure the images of the blank micro:bit, a micro:bit with the LEDs displayed as a smiley face, a computer showing the program, a smile.hex program file and USB cable are printed from the ‘LKS2 - Lesson 1 - Unplugged Handout’ file.

Ensure the blank micro:bit LEDs template is printed out for each pupil from the ‘LKS2 - Lessons 1 and 2 - LEDs - Blank’ file.

Check the ‘Health and Safety’ section of the ‘micro:bit lessons overview’ document to ensure the device can be used safely.

The videos included with these resources can be used as a guide for how to use the micro:bit. The videos can also be shown to pupils within the lessons, as indicated in the lesson plan below.

# Resources

■ Desktop or laptop computer with a web browser that can access the micro:bit MakeCode editor - <https://makecode.microbit.org/>and an available USB port, which can be used to transfer files to the micro:bit. Further details are outlined in the ‘Technical requirements’ section of the ‘Lesson overview’ document.

■ micro:bit devices

■ Images of the blank micro:bit, a micro:bit with the LEDs displayed as a smiley face, a computer showing the program, a program file and USB cable are printed from the LKS2 - Lesson 1 - Unplugged Handout file.

■ The blank micro:bit LEDs template printed out for each pupil from the LKS2 - Lessons 1 and 2 - LEDs - Blank file.

■ Pencil for completing the micro:bit LEDs template

■ The following support videos: [LKS2 Unplugged](https://vimeo.com/685909280/91f826c799), [LKS2 Intro](https://vimeo.com/683317156/5bcac42abd) and [LKS2 Image](https://vimeo.com/686699961/18413d48ef)

# Introduction 5 minutes

Explain to pupils they will be using the micro:bit device, which is a special type of computer they can program. Provide pupils with a micro:bit and ask them if they can identify the following parts of the device, which they will be using during these lessons - LEDs; USB connector (Slide 3).

Ask pupils to discuss whether the LEDs are inputs and outputs, and elicit that we say they are outputs, as they are displaying information from the computer.

(Please note - the LEDs are also used as an input to detect light, although this is beyond the scope of this lesson.) Ask pupils to also identify other parts of the device and discuss their functionality using the guidance within the ‘Lesson overview’ as required.

# Main activity 45 minutes

## Our first program - 5 mins

Explain to pupils that to program the micro:bit, we use a block based programming environment, similar to Scratch, called MakeCode. We use MakeCode on the computer to create our program, then transfer it to the micro:bit. Show pupils an example program (Slide 4) and ask them to discuss what they think the program does? Elicit the program displays a smiley face on the micro:bit LEDs when the program starts.

Explain the program needs to be copied from the computer to the micro:bit, which takes place using a USB cable. Show pupils the paper images of a blank micro:bit, a micro:bit with the LEDs displayed as a smiley face, a computer showing the program, a program file and USB cable from the ‘LKS2 - Lesson 1 - Unplugged’ file. Ask one pupil to hold the two micro:bit images, with three other pupils holding one other image each. Arrange the pupils as shown in slide 5.

Explain to children the program is currently on the computer and isn’t currently stored on the micro:bit. Point to the program and explain we are going to transfer this to the micro:bit.

Explain the program needs to be saved as a file to be transferred, so we have to download it first. Ask a pupil to press the ‘Download’ button on the computer image and explain this produces a file, which we have called ‘smile.hex’. Place the file image on top of the computer to show it is now stored on the device.

Explain to pupils we now need to transfer the program file, so we copy it on to the micro:bit using the USB cable. Ask a pupil to move the image of the file along the USB cable to the micro:bit (Slide 6). Once the file arrives at the micro:bit, replace the image of the blank micro:bit with the image of micro:bit with the LEDs displayed as a smiley face (Slide 7). Remind pupils the LEDs are known as outputs, as they are showing us information from the computer. Explain to pupils we need to do this process each time we want a new program transferred to the micro:bit.

### Creating a program - 25 mins

Show pupils the MakeCode programming environment by visiting [https://makecode.microbit.org/ a](https://makecode.microbit.org/)nd selecting ‘New project’, then giving the project a name, such as ‘Test’ (Slide 8).

Explain to pupils there are three main areas of the screen, which are labelled A, B and C (Slide 9) and have the following functionality:

■ A - Programming blocks - Blocks of code to use within your programs

■ B - Workspace - Where blocks of code can be placed to program the micro:bit

■ C - micro:bit simulator - an on screen version of the micro:bit, which can be used to see how a program functions

Demonstrate to pupils how to place instructions together to create the smiley face program from the introduction activity by dragging the ‘show icon’ block from ‘Basic’ into the ‘on start’ block in the programming area (Slide 10). Highlight how the output from the program (the LEDs displaying a smiley face) is shown on the micro:bit emulator on the screen, which enables us to check our program is working as expected.

Explain to pupils that the program is now going to be transferred from MakeCode on the computer to the micro:bit. Remind pupils of the earlier activity, where they saw the images of their program being transferred using the USB cable, and show them how to connect the USB cable to both the micro:bit and the computer.

Demonstrate to pupils how to download the file using MakeCode and find the file within the computer’s ‘Downloads’ folder by using the ‘Show in folder’ option (Slide 11). Demonstrate how to drag the highlighted file onto the micro:bit and show pupils how the flashing yellow light on the back of the micro:bit tells us the program is being transferred. Show pupils the program running on the device.

Explain to pupils it is important we use the micro:bit safely (Slide 12). Emphasise to pupils we should hold the micro:bit by the edges, as we want to avoid damaging any of the components. We should also avoid having any liquids near the micro:bit.

Ask pupils to make and transfer a program to display a smiley face on the micro:bit and explain that an overview of the required steps is displayed on slide 13. Once complete, ask pupils to attempt the other tasks on the screen, emphasising they only need to use ‘Basic’ blocks displayed on the slide. Support pupils as required and remind them each new program needs to be transferred to the micro:bit if required.

### Our own images - 15 mins

Explain to pupils we can also program individual LEDs to be on or off, which means we can produce our own image. Using the micro:bit LED template (Slide 14), show children how each LED can be coloured in to produce the smiley face image previously seen.

Explain to pupils they are going to be making images with the LEDs related to plants and nature. Ask pupils to discuss possible images that could be created related to this topic, such as a leaf, tree, or seed, and demonstrate by colouring the micro:bit LED template from the ‘LKS2 - Lessons 1 and 2 - LEDs - Blank’ file (Slide 15).

Show pupils the ‘show leds’ block (Slide 16). Demonstrate using MakeCode how to program the individual LEDs by clicking on the relevant squares on the ‘show leds’ block. Ask pupils to program the micro:bit to display their own image, followed by attempting the other tasks on slide 17. Support pupils as required and remind them each new program needs to be transferred to the micro:bit if required.

### Plenary 5 minutes

Ask pupils to share their program with someone near them, including explaining the blocks used within the program (Slide 18). Ask a number of pupils to share a program they particularly liked, including why they liked it.

# Differentiation

## Support

Pupils could be provided with the required blocks of code for each task, then asked to rearrange the blocks in order. Scaffolding learning in this manner is called a Parsons’ Problem.

## Stretch

More confident pupils could add additional blocks from ‘Basic’ to their programs, such as ‘Clear screen’ and ‘Forever’

# Assessment opportunities

Informal teacher assessment of understanding through lesson and assessment of pupils’ programs. Key understanding to assess:

■ Can pupils sequence instructions to produce the programs specified?

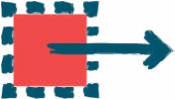
■ Can pupils explain the instructions they used to produce each program?

# Teaching notes

Concepts and approaches

## Sequence

Pupils create a sequence of instructions within their program to complete each of the tasks.

**Outputs**

The LEDs on the micro:bit display information from the device, so act as the output.

# Curriculum links

Please refer to the resource overview page on the website, to understand how the learning objectives covered in this lesson relate to the curriculum in your country.

**We love seeing our resources being used in the classroom.**

Please share your comments and pictures via our social channels:

  **@BarefootComp /barefootcomputing @barefootcomputing**

enquiries@barefootcomputing.org | barefootcomputing.org

Supported by With help from