

First program: Hello!

Programming for beginners with the BBC micro:bit

John Davies

Bearsden and Milngavie u3a

2026 May 13

Introduction

We'll start with a simple program, which displays Hello! on the micro:bit's LEDs.

A **program** is a set of instructions to the computer, written in a particular language.

This is often obscure text, such as:

```
from microbit import *  
while True:  
    display.scroll('Hello!')
```

but we shall use a straightforward, graphical approach with **blocks**:



What does the program do?

The focus of this task is on learning to construct and download the program, but here is a summary of its function.



- ▶ The **forever** block is a type of **loop**: the computer executes the code within it repeatedly. Here it repeats 'forever' (until the computer is reset).
- ▶ A **string** is a (short) piece of text – a sequence of characters – contained within quotation marks. Here the string (with its quotation marks) is "Hello! ".
- ▶ The **show string** block scrolls the string (text) across the display of the micro:bit, which is only large enough to show a single character at a time.

First steps

Two main stages are required before the excitement of seeing the micro:bit execute your first program.

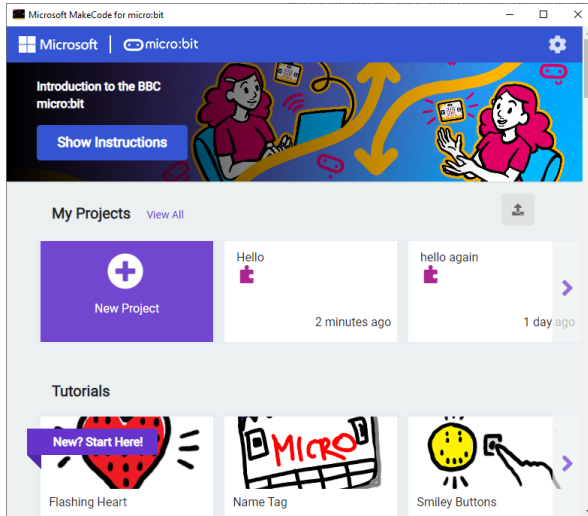
1. **Construct** the program and test it in the simulator. This can be done within an app or a web browser.
2. **Download** the program to the micro:bit. This can be done over Bluetooth or USB.

If you have programmed before, the second step may puzzle you. Why can't you just click a Run button?

On a traditional computer (perhaps a BBC Micro!) the program is constructed and executed on the *same* hardware. Here we want to run the program on the micro:bit but it isn't powerful enough to run the software required to construct the program. We therefore need a more powerful computer (your laptop or tablet) for construction, which requires the further step of downloading the finished program to the micro:bit. *Unfortunately this can be a troublesome step.*

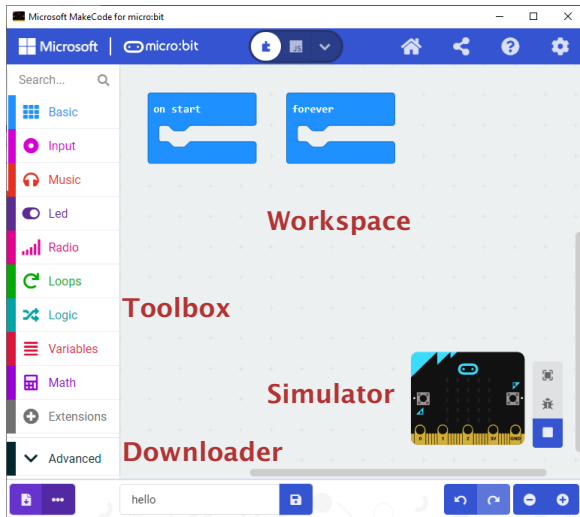
The MakeCode home screen

Run the app or open the [web page](#) and log in. You should see a screen like this.



The MakeCode editor

Click on New Project and give it a name. You should see a screen like this.



Build the program

Follow these steps to construct the program and watch it on the Simulator.

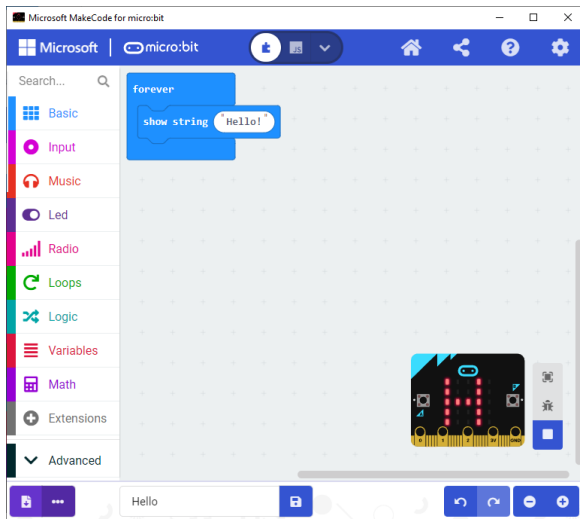
1. We don't need the on start block so drag it back into the Toolbox, which turns into a rubbish bin.
2. Add a show string block and locate it within the notch of forever.
 - ▶ Click on Basic in the Toolbox. This shows a set of blocks, including the one that we need (you may need to scroll).
 - ▶ Click on show string, which drops a copy into the Workspace. It has a greyish colour to show that it needs to be placed correctly. It contains Hello! already.
 - ▶ Hold down the mouse on show string, whose outline turns yellow. Drag it on the screen over the forever block, whose notch enlarges and turns yellow when it is ready to accept show string. Let go!
3. Click somewhere else in the Workspace to show that you have finished moving the blocks. This alerts the Simulator, which starts to run your program.

That's it! You should see Hello! scroll across the LEDs of the simulator.

Congratulations! You have written your first program.

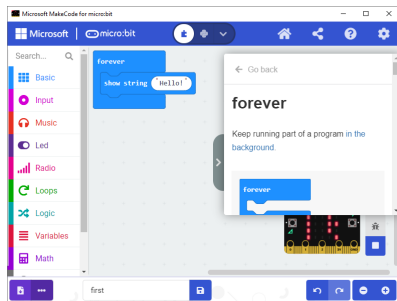
The complete program

The program should look something like this.



Help!

MakeCode contains several sources of help.



- ▶ Right-click on a block and choose Help to get specific information and examples.
- ▶ The question mark icon (top right) offers a range of topics including a Tour of MakeCode, Reference, from which you can get information on any block, and a link to the [Help & Support](#) web pages (with great detail).

Download the program

The final stage is to download the program into the processor on the micro:bit so that the program runs on the real hardware. The details depend on the computer that you use.

1. USB with the app or a web browser based on Chrome (including Edge)
2. USB with a different web browser (including Safari and Firefox)
3. Bluetooth with the app

The micro:bit must be paired with the computer when you use Bluetooth for the first time. I recommend that you use USB if available (not possible with an iPad). Downloading is also called ‘flashing’ because the program is downloaded into flash memory on the micro:bit.

USB with the app or a web browser based on Chrome

Follow these steps.

1. Connect the micro:bit to your computer with a USB cable, possibly with an adapter. Don't connect the battery pack because the USB cable provides power. The micro:bit springs to life and runs the program that was stored in it previously, probably 'meet the micro:bit'.
2. Click the Download button in the app or web page. The micro:bit stops running its existing program, the yellow LED on the back flashes as your program is downloaded, and after a few seconds the micro:bit starts to run your program.

Congratulations! Your first program is now running on the micro:bit.

The program continues to run until you disconnect the micro:bit or press the Reset button on the back, which restarts the program.

USB with a web browser *not* based on Chrome

This requires additional steps.

1. Connect the micro:bit to your computer with a USB cable.
The micro:bit springs to life and runs the program that was stored in it.
2. Check that the micro:bit appears as a disk drive called MICROBIT on your computer and open a window for it.
3. Click the Download button in the app or web page.
You will see an alert that Your code is being downloaded as a .hex file. You can drag this file to your micro:bit using your computer's file explorer.
4. Follow this instruction: drag the file (with extension .hex) from the Downloads folder to the window for the MICROBIT disk.
The micro:bit stops running its existing program, the yellow LED on the back flashes as your program is downloaded, and after a few seconds the micro:bit starts to run your program.

Congratulations! Your first program is now running on the micro:bit.

Bluetooth: getting prepared

Using Bluetooth can be tricky and you must be familiar with the **Reset** button on the back of the board, between the USB and power sockets.



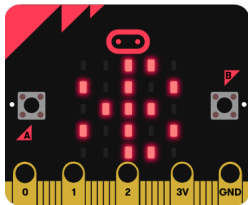
Try the web page [Transfer code to the micro:bit](#) if the following instructions fail.

Bluetooth: download and pairing I

Most steps are guided by the app itself. Here is how to get started.

1. Connect the battery pack to the micro:bit, taking care to insert the fiddly connector correctly.
The micro:bit springs to life and runs the program that was stored in it.
2. Click the Download button in the app (bottom left).
3. The app instructs you to Reset micro:bit to Bluetooth mode by pressing Reset three times rapidly.

All LEDs should light, followed by the Bluetooth symbol, which is in turn followed by a unique pattern for your micro:bit.



Bluetooth: download and pairing II

4. Press Continue in the app. Adjust the pattern shown in the app to match that on the LEDs of the micro:bit.
5. Press Next to pair the devices. You may be asked to agree to allow access. You may also be asked for a code but this has never happened to me.
6. At last, your code is downloaded to the micro:bit and starts to run.

Congratulations! Your first program is now running on the micro:bit.

If step 3 above doesn't work, try this (Show another way in the app).

- ▶ Hold down buttons A and B on the front, while pressing and releasing the Reset button on the back (awkward).
- ▶ Release buttons A and B once your unique pattern is shown.

If that still doesn't work, 'power cycle' the micro:bit by removing and replacing the power connector and try step 3 again.

For subsequent programs, you always have to reset the micro:bit to Bluetooth mode (steps 2 and 3) but the program should then download without further ado.

Extend the program

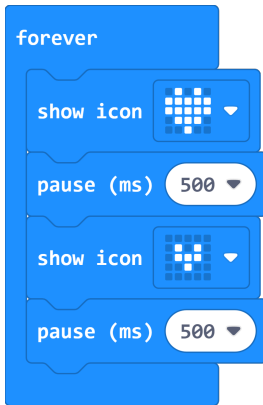
Now that you have a program running, extend it by personalising the message.

1. Click on the text within the `show string` block and edit it to something more personal. Don't worry about the quotation marks and final space character: MakeCode adds them for you.
2. Click the workspace outside the block. The simulator starts running your new message.
3. Download the new version to the micro:bit and see your name in lights!



Second program: beating heart

Now construct a completely new program to display a beating heart on the LEDs.
Again it is based on a forever loop but this time the loop contains four blocks.



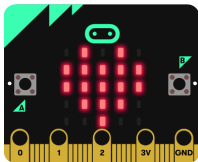
Build the program

The main steps are the same as before.

1. Return to the home screen of MakeCode, create and name a New Project.
2. Remove the on start block and pick the other blocks from the Basic section of the Toolbox. Just place the blocks for now: we'll adjust their contents later.
3. The first show icon block slots into forever as before.
4. Drag the second block, pause (ms), over the bottom of the forever block, which highlights and opens a gap below show icon. Let go and pause snaps into place.
5. Repeat with the second show icon block and the second pause (ms) block.
6. The show icon block has a dropdown menu for choosing the particular icon. Most are emojis and have names that appear if you hover over them. Conveniently, the large heart is first and the small version second.
7. The duration of the pause is given in milliseconds or ms, which are thousandths of a second. Select 500 because 500 ms is half a second. This gives one second for the complete heartbeat: about right.

Simulate and download the program

Click outside the blocks when you have constructed the program and the simulator shows a beating heart. Then download the program to the micro:bit.



Why not adjust the pauses or change the icons? Perhaps add sound, such as the play block in the Music category?

Now combine this program with the previous one to display on the micro:bit:

I ♥ u3a

You will need the show string and show icon blocks, perhaps more (show leds and clear screen might come in handy).