Micro:fit - Your DIY Step Counter

Overview

Ever wonder how your smartwatch knows exactly how many steps you've taken? In this exciting project, we'll demystify fitness tracking technology by creating our own step counter using a micro:bit! Unlike expensive commercial devices, our DIY fitness tracker will help you understand the basic principles of motion detection and how simple sensors can count your daily steps.

Using the micro:bit's built-in accelerometer, we'll program it to detect movement patterns that match walking or running motions. You'll learn how real fitness trackers work while building your own wearable device that displays your step count right on the LED matrix. Best of all, you'll gain hands-on experience with physical computing, data processing, and real-world sensor applications.

Whether you're interested in wearable technology, fitness tracking, or just love creating cool gadgets, this project offers a perfect blend of computer science and physical activity. By the end, you'll have your very own working step counter and a deeper understanding of the technology we use every day!

What you will Learn

How to:

- [ ] Create and use a variable.
- [ ] The micro:bit shake function.
- [ ] Use a while True loop.
- [ ] Display a number on the LED matrix.
- [ ] Use the button function.

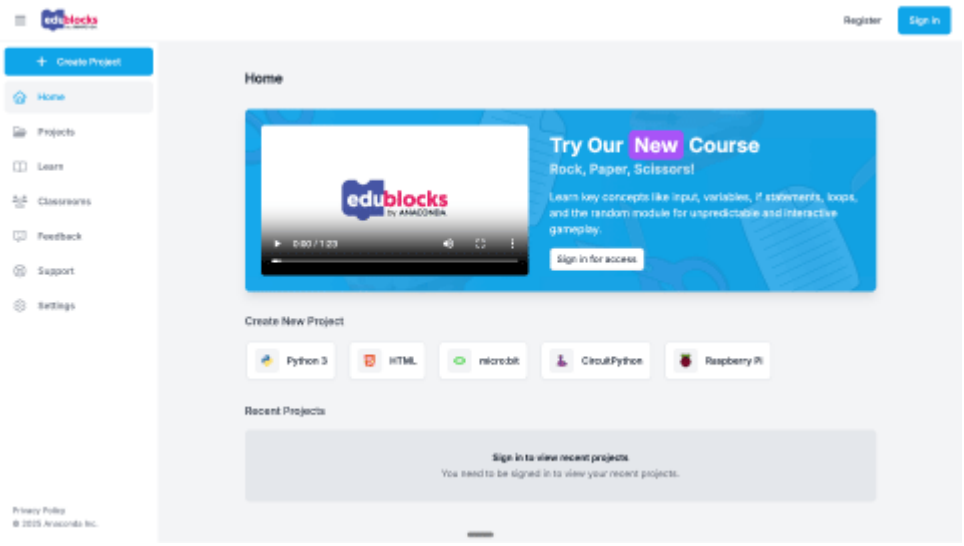
What you will Need

- 1 x micro:bit
- 1 x micro USB cable
- 1 x battery pack for the micro:bit (optional)

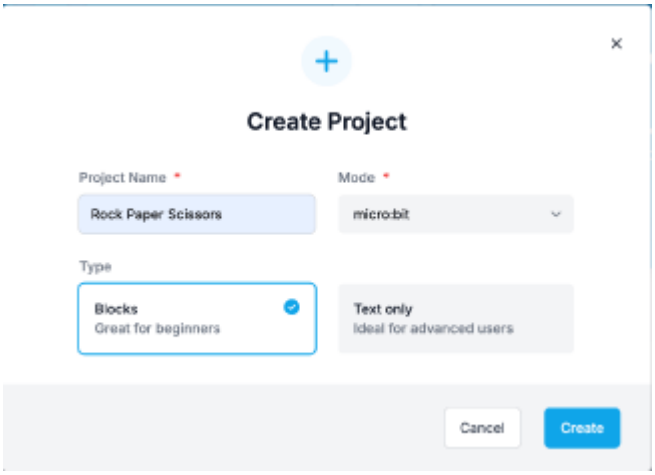
Lets move on to opening the EduBlocks editor.

Navigating to EduBlocks

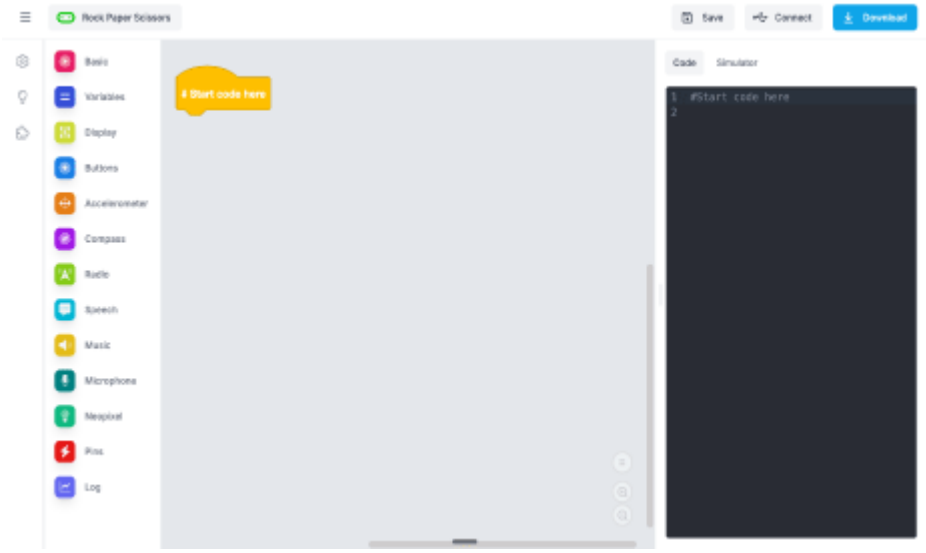
1. Open your favourite browser (we recommend Google Chrome).
2. Within the address bar of the browser type app.edublocks.org or on a tablet or phone press create code.



3. Select micro:bit under **Create New Project** to open the micro:bit coding editor. Name your project **FitBit Clone** and make sure you have **Blocks** selected under **Type**. See the image below.



4. Select **Create** to open the micro:bit editor.

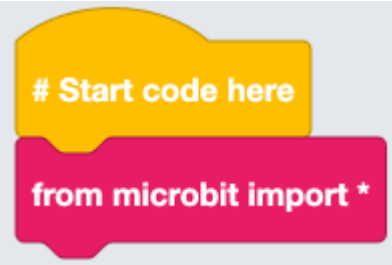


We are now ready to start coding!

Coding

Importing Python Libraries

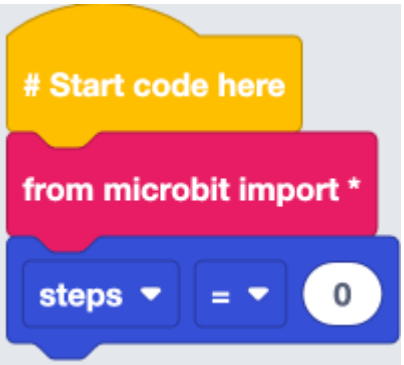
From the Imports menu which is found under the Basic menu, select and drag a `from microbit import *` block to the code area and attach it to the `# start code here` block.



Creating Steps Variable

> WHAT IS A VARIABLE
> Think of a variable as a box that stores information that can be used throughout our program. We give variables a descriptive name so we and others can understand what is going on within our program.

1. From the Variables menu, select **New variable...** . Give it the name **steps** press Enter.
2. From the Variables menu, select and drag a `steps = 0` block to the code area and attach it under the `from microbit import *`. Select **r** and pick **steps.

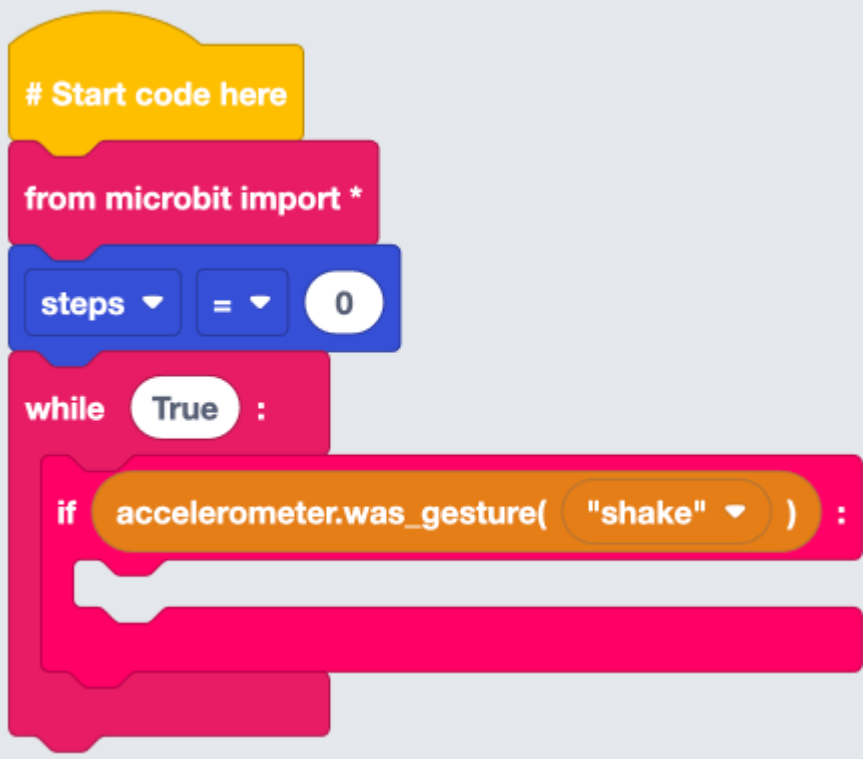


Creating a While Loop

From the Loops menu which can be found under the Basic menu, select and drag a `while True:` block to the code area and attach it under the `steps = 0` block.

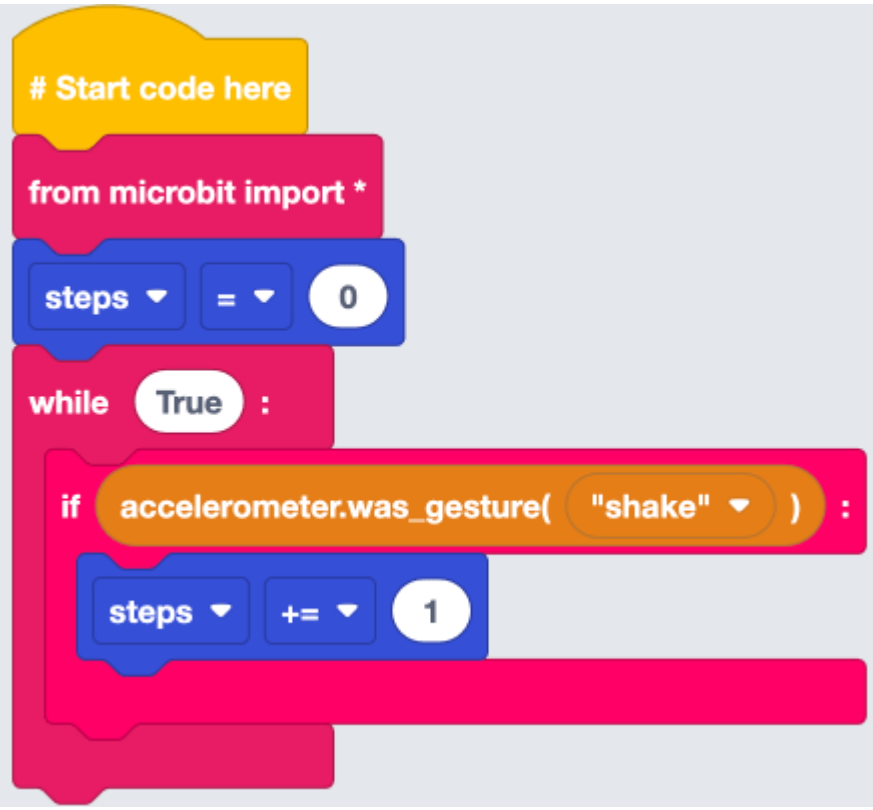
Creating the Shake Function

1. Select Logic from the Basic menu, Select and drag an `if True:` block to the code area and attach it within the `while True:` block.
2. From the Accelerometer menu, select and drag an `accelerometer.was_gesture('shake')` block and attach it within the **True** of the `if` block.



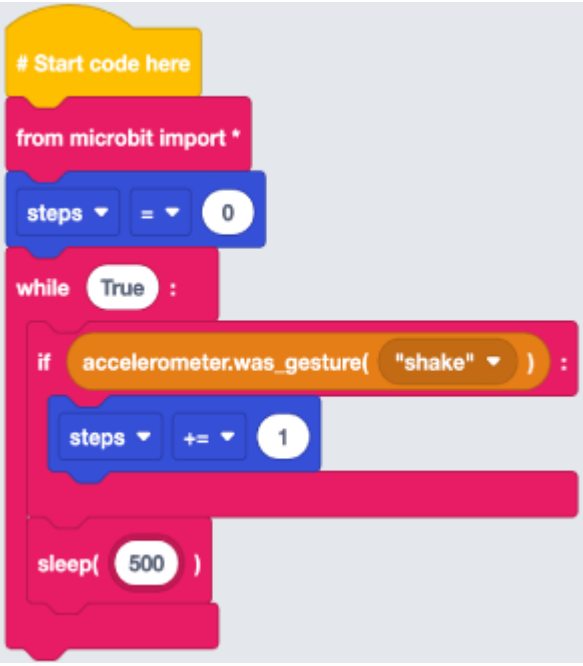
Increasing Steps

From Variables, select and drag a `steps = 0` block to the code area and attach it within the `if accelerometer.was_gesture('shake')` block. Select **r** and choose **steps** . Select the **=** and choose **+=**. Change the **0** to **1**.



Pausing the Code for 500ms

From Statements within the Basic menu, select and drag a `sleep(1000)` block to the code area and attach it below the `if accelerometer.was_gesture('shake')` block. Change **1000** to **500**. This will pause your code for half a second.



Resetting Steps to 0

1. From Logic within Basic, select and drag an `if True:` block to the code area and attach it under the `sleep(500)` block.

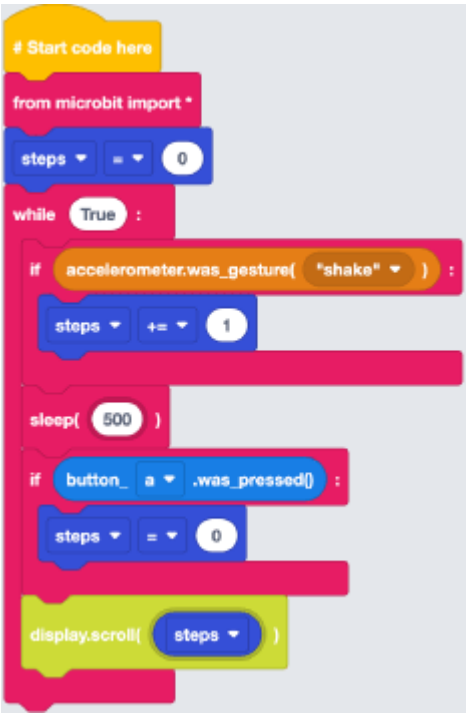
2. From Buttons, select and drag a `button_a.was_pressed()` block to the code area and attach it within the **True** of the `if` block.

3. From Variables, select and drag a `steps = 0` block to the code area and attach it within the `if button_a.was_pressed()` block.

Displaying Steps on the LED Matrix

1. From Display, select and drag a `display.scroll ("Hello World")` block to the code area and attach it under the `if button_a.was_pressed()` block.

2. From Variables, select and drag a `steps` block to the code area and attach it within the **"Hello World"** of the `display.scroll ()` block.



We have now completed our code. Lets move on to downloading our code to the micro:bit.

Downloading the code to the micro:bit

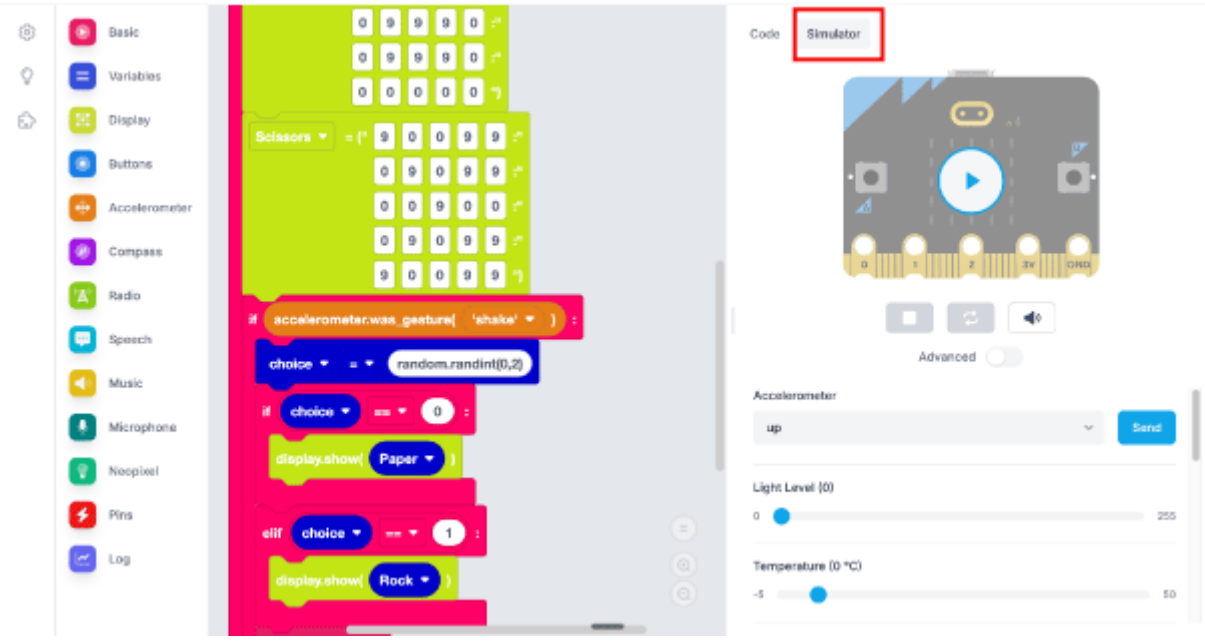
1. Take the micro USB cable and connect the micro:bit to the computer.
2. Select **Connect** and follow the pop-ups on screen to pair the micro:bit to the web browser.
3. Select the **Download** button to download your code to the micro:bit.

How to Play

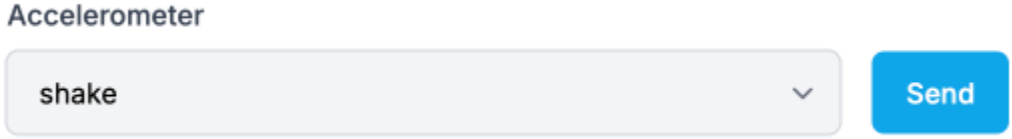
Web Browser

The EduBlocks Editor has a built-in micro:bit simulator so you can use this if you don't have a micro:bit handy.

1. On the right-hand side of your screen select **simulator**.



2. From the **Accelerometer** menu below the simulator select **shake**.
3. Select **send** this will now simulate a shake of the micro:bit and display a number on the micro:bit simulator.



Using the micro:bit

Once you have downloaded the code to your micro:bit you can shake the micro:bit and see your steps add up.

Find a way to attach the micro:bit to you and go for a walk or run and see how many steps you can get.