

# Open:UK Ezine

00011011  
01110110  
1100010  
1111111  
110

Open  
Source  
is...



*...Doesn't make  
requirements  
of other code*



Issue

9

# A message from *the Ed*



Hi!

We've heard quite a bit about the Linux Foundation this course and in Lesson 9 we meet the CNCF - or Cloud Native Computing Foundation. You're probably beginning to understand the importance of Foundations and not-for-profit organisations like OpenUK in open source. Generally speaking, the Foundations are the home for code whereas OpenUK doesn't house or host code. Many of our OpenUK team of volunteers some of whom you have met in this Camp work with the CNCF.

OpenUK has a purpose of "Developing UK leadership and Global Collaboration in Open Technology". As you know, Open Technology is open source software, open hardware, and open data. We'll talk about open hardware another time, but you should have learned a lot about open source software and open data through this course.

OpenUK also works on 3 pillars, Community, Legal and Policy, and learning. You're of course doing lots of learning under our third pillar. And you have learned loads about community in this course. We bring our community together to influence laws and policies and hope to ensure that the UK becomes even more of a centre of excellence in open source software and of course Open Technologies.

*Amanda*  
CEO, OpenUK



# Ask Ashleigh



Ashleigh Monagle

## Invisible disability

When asked what your superpower would be, some people x reply saying that they would like the power of invisibility but for somepeople, unfortunately, this is not a choice. You have probably heard the phrase 'do not judge a book by its cover', but if not I will go into a little more detail and share with you how you can be a different type of superhero.

In the UK, 70% of disabilities are hidden, meaning that it is unlikely you would be able to tell they have a disability just by looking at someone. Some of these disabilities were mentioned in the neurodivergence column in Ezine [5] and others include Anxiety and Fibromyalgia (a condition that gives you pain all over the body). The lesson to be learned from this is that we should treat everyone fairly and not judge a book by its cover.

There are some people who need support and some who suffer from disabilities that prevent them from doing certain things. To help, we should all be open, respectful, and try not to judge people for their behaviours and choices. Always put your trusting foot forward and believe that people have good intentions.



Ashleigh

Digital Inclusion Lead, OpenUK

# Sustainability



## Energy

We previously discussed how burning fossil fuels such as natural gas, oil, and coal to power our towns and cities, our homes, cars, and devices is releasing carbon emissions that are causing climate change. We already know that countries and businesses have an important responsibility in transitioning to renewable energy coming from wind, solar, and waves to bring carbon emissions to zero and reverse climate change.

You may have heard companies and institutions use many terms for this - Net Zero, Carbon Negative, Carbon Neutral, Carbon Positive - it's a bit overwhelming and even the experts get confused! Just remember, the most important thing is to reduce emissions to zero, so carbon never reaches the atmosphere in the first place.

Carbon offsets allow organisations and people to compensate for their carbon emissions by giving money to initiatives that plant trees or capture carbon from the air. They can be a helpful baby step to reduce carbon, but that will not solve climate change. If organisations purchase carbon offsets but continue what they're doing anyway, it is considered another form of greenwashing. They are buying their way out of the responsibility to reduce carbon emissions, whilst still claiming they are sustainable.

There are simple things we and our families can do at home to save energy, money, and the environment:

- Switch energy suppliers to companies that use renewable energysources.
- Turn off the lights when you leave a room.
- Unplug devices when not in use.
- Lower your thermostat a bit and put on a jumper.
- Use LED Light bulbs, they're 90% more energy-efficient, last longer, and look more stylish.
- Take shorter showers. Play two of your favourite songs to keep time.
- Wash clothes in cold water.

So, while switching to renewable energy is important and a key step, wasting renewable energy isn't sustainable either!

## Keep Coding!

Christian

Chief Sustainability Officer, OpenUK



# Please Miss Boal



Pamela Boal

## Get into step!

As we focus on learning to become programmers, it can be easy to forget that the code is often only part of the solution. Physical computing involves systems that can sense and respond to the world around them, such as the micro:bit that is used in the step counter program.

Jonathan Sacks is quoted as saying “Science takes things apart to see how they work” but let’s not do that here and instead just discuss how this technology works!

Early step counters were entirely mechanical and they worked a bit like pendulum clocks as they used a swinging pendulum. This counts steps and displays the count with a pointer moving around a dial. You fixed them on your waist and, every time you took a step, the pendulum swung to one side then back again, causing a gear to advance one position and moving the hand around the dial.

Modern step counters work in a very similar way but are partly electronic. Inside there is a metal pendulum wired into an electronic counting circuit by a thin spring. As you take a step, the hammer swings across and touches a metal contact in the center, completing the circuit and allowing current to flow. The flow of current energizes the circuit and adds one to your step count. The pedometer shows a count of your steps on an LCD screen.

The challenge of this Lesson is to use the step counter to record as many steps as possible by walking rather than using the car or public transport.

## Keep Coding!

Pam Boal

Computer Science Lead, OpenUK



# Industry

## Cloud Native Computing Foundation

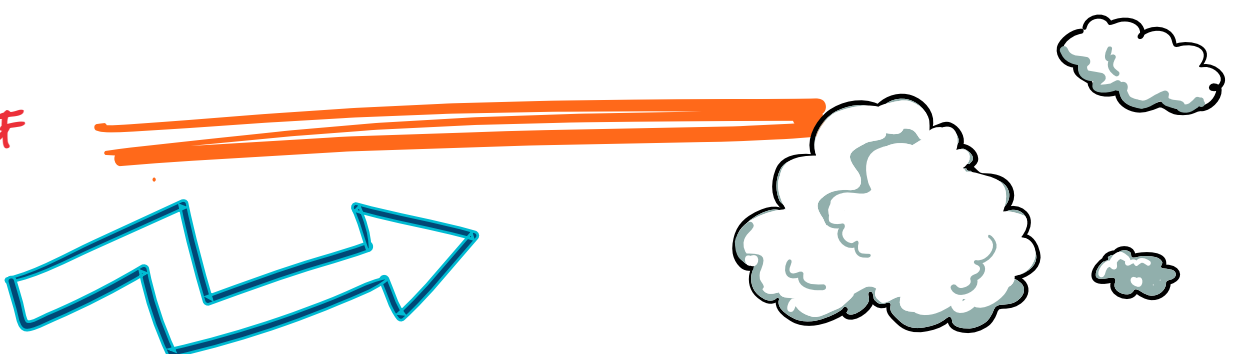


When you upload photos to the iCloud, listen to music on Spotify, or buy shoes online from Adidas, you are using cloud native technologies provided under the Cloud Native Computing Foundation (CNCF) umbrella. CNCF is a non-profit foundation that builds and sustains communities of people working on and with cloud technologies. It hosts over 100 software projects defining the future of cloud computing. Every project under CNCF is open source, and anyone can contribute to or use these for free. This also allows movement between cloud providers and prevents lock-in.

CNCF projects are used to build and run cloud infrastructure around the world. They enable companies to run thousands of applications on servers distributed around the world in a scalable and resilient manner. Further, cloud native technologies empower teams to make changes quickly with minimal effort, allowing them to rapidly improve the products and services we rely upon every day.

CNCF is at the center of the cloud native ecosystem and helps to drive these projects and paradigms forward. CNCF welcomes contributors from 177 countries around the world, and they work in a broad variety of areas, even beyond just software development. You can become a part of the community today and grow your cloud native career by visiting <https://contribute.cncf.io/>. You will see many participants in the ezines who are part of CNCF.

CNCF





# Open Data



## *Beyond coding – what are data skills?*

When we talk about ‘data skills’, you might think of data science skills like coding or programming. However, there are other data skills out there that are just as important – something we call ‘data literacy’.

Data literacy is being able to think critically about data – just like the way that English literacy teaches us how to think critically about books or poems. Data literacy skills include:

- Examining the ways data can be collected, and deciding on the best approach;
- Understanding what data is relevant for a particular task;
- Interpreting data in graphs or infographics;
- Analysing the accuracy of claims based on data. For example, identifying whether data has been misunderstood, or if the data was biased (is unfair in some way or favours certain viewpoints); and
- Using good practices when collecting, using, and sharing data.

These skills are just as important as coding or programming. Without them, we might miss out on all the benefits that data can bring – and we may even make mistakes that hurt people.

For example, imagine someone is building a tool that uses bus timetable data to help people plan bus journeys across a city. If they use out-of-date bus timetable data that they found online (but didn’t think critically about), the tool won’t be handy. However, if they can spot that the data isn’t fit-for-purpose, they could then find accurate bus timetable data and build a helpful tool!



# Learn with Lowena



Lowena Hull

## What makes an effective team?

Teamwork is an essential skill in everything we do today. Very rarely will you be working on something completely by yourself. You'll often be in a team of people with different skill sets who hopefully complement each other. It's important to know how to work effectively with other people and how to collaborate. This is critical for open source but also in all walks of life.

It's a lot easier to say what makes a bad team than to pinpoint what makes a team excel, but here are a few things I've noticed. A good team understands what its members' strengths are and properly utilises them, playing to their strengths. This means knowing what each member is good at and finding tasks that need doing that suit them. Strengths and desires are not always the same.

A good team collectively discusses issues and seeks solutions rather than trying to place blame on any team member or members. If something goes wrong, it's far more constructive to go over what went wrong, to understand why that was so, and to work out if it can be prevented from happening again.

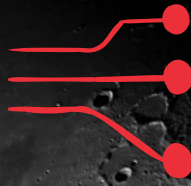
Finally, a good team communicates often. There's plenty of other things good teams do (and things bad teams do that you want to avoid!) but these are some of the most important things I've noticed.

Lowena

Student, Cambridge University



# Nasa



Chris Mattaman

## Making it fun

Make open source software development a fun activity and not only work or a day job or something that you risk falling out of love with. To do this, keep a fresh perspective on how you organise your code. Keep a fresh perspective and even introduce some humor into the way that you create tickets to register and fix bugs and update and make progress on your software. Introducing some humor, or in addition, gamifying contributions to your project allows others to enjoy what they do and creates incentives for people to keep contributing.

Making it fun could range from handing out awards for most contributions in a week; hardest bug squashed, competitions for PRs (Pull Requests) merged with smallest modifications and other games and prizes and awards that you can encourage the lifeblood of your project - its people - to be interested in continuing to contribute and be the community.

Also, organise your code in a way that others can work on it in parallel and enjoy without dealing constantly with synchronous approvals and check-ins using Git. Save those moments for when they are absolutely necessary and other than that keep it fun and enable parallel and shared ownership and development.

Chris

Chief Technology and Innovation  
Officer, NASA Jet Propulsion  
Laboratory



# Entrepreneur *in residence*

## *Why I wanted open source in my business*



Matt Barker

Open source development is founded on values such as community, freedom, ethical practice, collaboration, and continually strives to be relevant and modern. These values match my own view of how businesses should be run, so it's no surprise that we chose to place open source at the heart of our company.

When many companies begin to contribute and invest in the same core open source technology it becomes an "ecosystem". In Jetstack's case, the ecosystem we contribute to is Kubernetes. Over the last 5 years, Kubernetes has become very popular and is changing the nature of cloud computing. Interestingly, our business has grown and evolved to match the needs of the Kubernetes ecosystem in the types of skills we hire, the services we sell, and the culture we develop.

But open source strategy also makes good business sense, and Jetstack uses open source as a strategy to grow opportunity and improve our standing in the Kubernetes ecosystem. We do this by improving and evolving our own open source project - called "cert-manager". Since this project has become extremely popular in the ecosystem, this means we spend a lot of effort ensuring our community of "cert-manager" users benefits directly from our development work.

It also means Jetstack is well regarded in the ecosystem; it builds our credibility which means we attract both customers and partners who want to work with us. These partners include Google, Amazon, Red Hat, and Microsoft. Having now established a very popular project a lot of the hard work has already been done, as our software is in the hands of millions of users thanks to the power of open source. This now gives us a tremendous opportunity to monetise by offering products to those users.

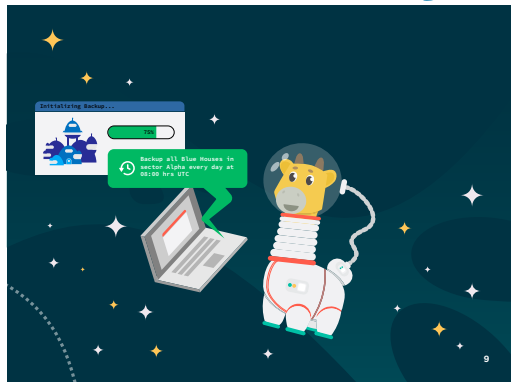
*Matt*

President & Co-Founder, Jetstack



# Kubernetes

## An illustrated guide



“There can also be any number of accidents in space, and we should also be prepared for malicious destruction. Our current systems only allow us to bring back the house, they don’t bring back any contents inside it.”

Phippy pauses for a second and replies, “Sounds like what we need is an insurance policy (backup and disaster recovery platform) to restore homes to a known good state instantly—and this has to include all the home’s custom data, not just the structure. It also has to be flexible enough to account for extreme diversity in home design and outpost infrastructure.” “Exactly!” exclaims Captain Kube.

“Additionally,” Phippy continues, “depending on the backup policy requirements, we can use our backup robots to create restore points—small compressed packages containing all records for each house. But to be more efficient, they should only capture changes to the house architecture and library content since the last backup.”

“They can ship these packages to a local warehouse for fast recovery. Then the warehouses will clone the packages and send them to a space station (offsite backup), so if a catastrophe takes out the warehouse, there’s still a copy available.” “I like this!” muses Captain Kube. “Of course, there’s also the issue of integrating these solutions with our current ecosystem...”



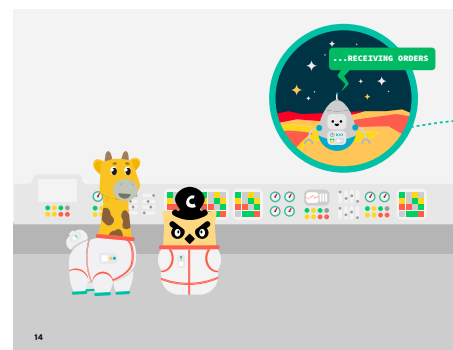
“Infrastructure built for Earth (VM-based backup) does not work on the outpost,” says Captain Kube. “We depend heavily on automation (CI/CD) and standard monitoring (Prometheus), and we provide every citizen with secure (RBAC) self-service abilities (CRD-based APIs).”

“Fortunately,” Phippy explains, “our Backup Ruby robots are purpose-built for outposts precisely because they integrate exactly into the operations you just described. They allow citizens (devs) to customize backup and recovery to their preferences, while natively integrating

into the outpost administrators’ (ops) existing monitoring, automation, and security systems. Earth-based solutions are definitely too unwieldy for development in this new world.”

“Given our accelerated pace of development, how will restores of old backups work with rapid outpost upgrades (Kubernetes API changes) or things in the backup that aren’t up to code (expired SSL certificates or secrets)?” asks Captain Kube. “The backup platform needs an intelligent transformation engine,” replies Phippy.

“It will recognize these challenges and apply changes to enable the house to work with the latest outpost infrastructure (update to latest Kubernetes APIs) and bring the house up to code by regenerating (requesting a new SSL certificate) any obsolete architectural components.” Phippy explains. “This will allow fast restores without requiring any manual work from outpost citizens.”



**To be continued....**

# Word game

C K S S Q U G N N S F P W S U C K C X P  
 C U S Q E N M O F E R M T K L U K P T G  
 C V F Q I U I O B R Z G G I B F O S P N  
 X Z D K K T L K Z I D V M E S F I U D A  
 J U L S C C P A K A G A R A K L X R A E  
 Z A J N O L X M V L T N X D X P E H Q L  
 W Y U V O O Q F X E E E R S P X C E N O  
 A F H O H U Y T L T K E Y W O R D A P O  
 E M O G Q D E E E E C L G C W L T B R B  
 S E R V E R A S U U G W N U V C M I L S  
 C R K F K D I X H H W A I W Z W N R N D  
 Q L O P E A S T E K C A R B D N U O R O  
 A P I R N G C Z O R H F O E I C T D W G  
 A C C E L E R A T I O N T N V F C R F P  
 D C K D N S C W Q F V Y S P H A B E J L  
 T O D I U T Y G G F N E S Z X R G F M P  
 B P T S R P K U Q G C H X T A M S D A V  
 A D W Y G I K S S M U N B T D I K N C I  
 B R E T E M O R E L E C C A J B C T P O  
 E V I T A N D U O L C R S A Z J D Q W Y

Acceleration  
 Accelerometer  
 Average  
 Boolean  
 Client  
 Climate Leader  
 Cloud  
 Cloud Native  
 CNCF  
 COP

Function  
 Keyword  
 Kubernetes  
 Round Brackets  
 Serial  
 Server  
 Sleep  
 Storing  
 Values  
 Walking

GAMES



courtesy of [puzzlemaker.discoveryeducation.com](https://puzzlemaker.discoveryeducation.com)

# Thanks for reading!

## Contributors

### Editorial:

Amanda Brock - Editor @amandabrockUK

Georgia Cooke - Creative Director [nuwcreative.com](http://nuwcreative.com)

### Columnists:

Ashleigh Monagle - Ask Ashleigh

Pam Boal - Please Miss Boal

Cristian Parrino - Sustainability

CNCF - Industry

Chris Mattaman - NASA

Lowena Hull - Learn with Lowena

Matt Barker - Entrepreneur in Residence

Open Data - ODI <https://theodi.org/>

CNCF - Kubernetes An Illustrated Guide <https://www.cncf.io/phippy/>

The characters Phippy, Captain Kube, Goldie, and Zee and the two books are owned by The Linux Foundation, on behalf of the Cloud Native Computing Foundation, and licensed under the Creative Commons Attribution License (CC-BY), which means that you can remix, transform, and build upon the material for any purpose, even commercially. If you use the characters, please include the text “phippy.io” to provide attribution <https://phippy.io>

All content is contributed by the author and the opinions of the author, and may not represent the opinion of OpenUK. ©OpenUK and licensed in accordance with: <https://creativecommons.org/licenses/by/4.0/>



The OpenUK glove kit giveaway and Ezine are made possible thanks to the generous support of

Course sponsored by



In kind glove sponsor



OpenUK glove inspired by



Giveaway sponsored by

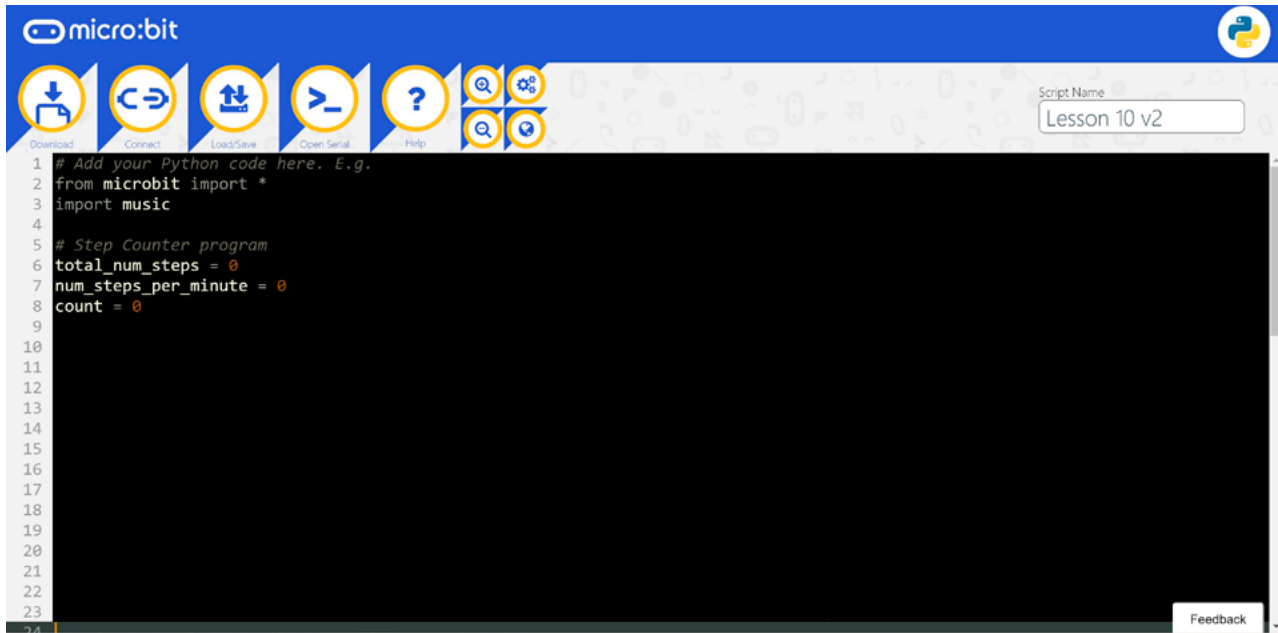


NOMINET

© OpenUK 2020. OpenUK is a not-for-profit company limited by guarantee, company number 11209475, registered at 8 Coldbath Square, London, EC1R 5HL. Contact [hello@openuk.uk](mailto:hello@openuk.uk) [openuk.uk](http://openuk.uk) @openuk\_uk



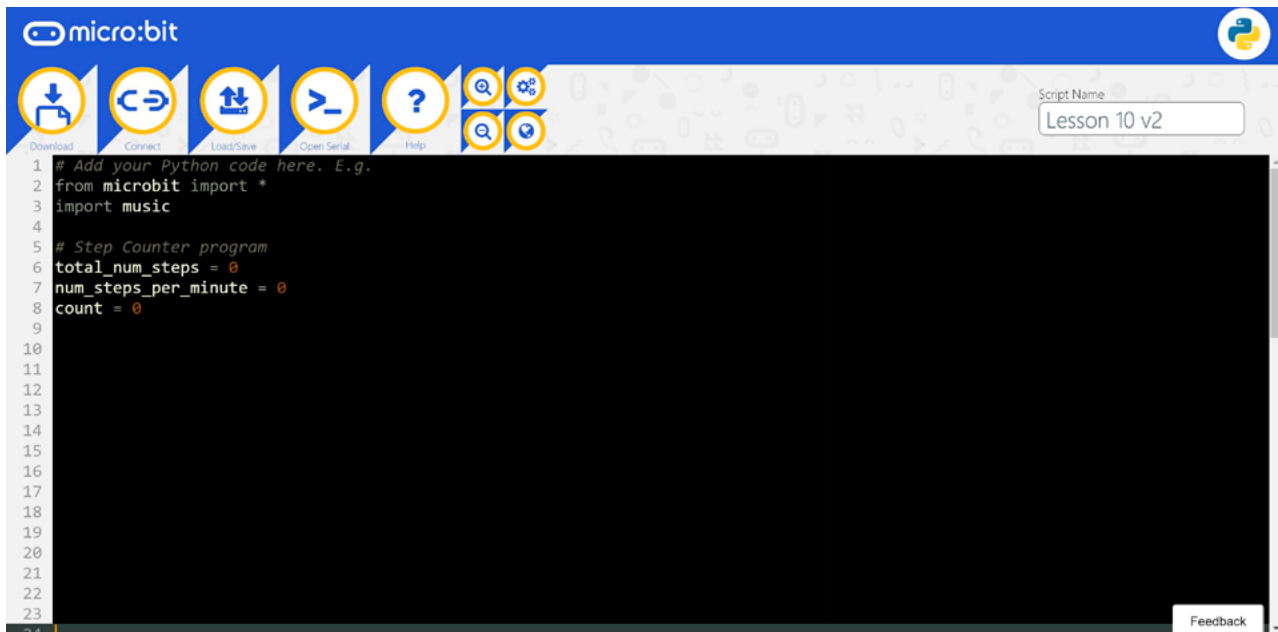
# Instructions Lesson 9



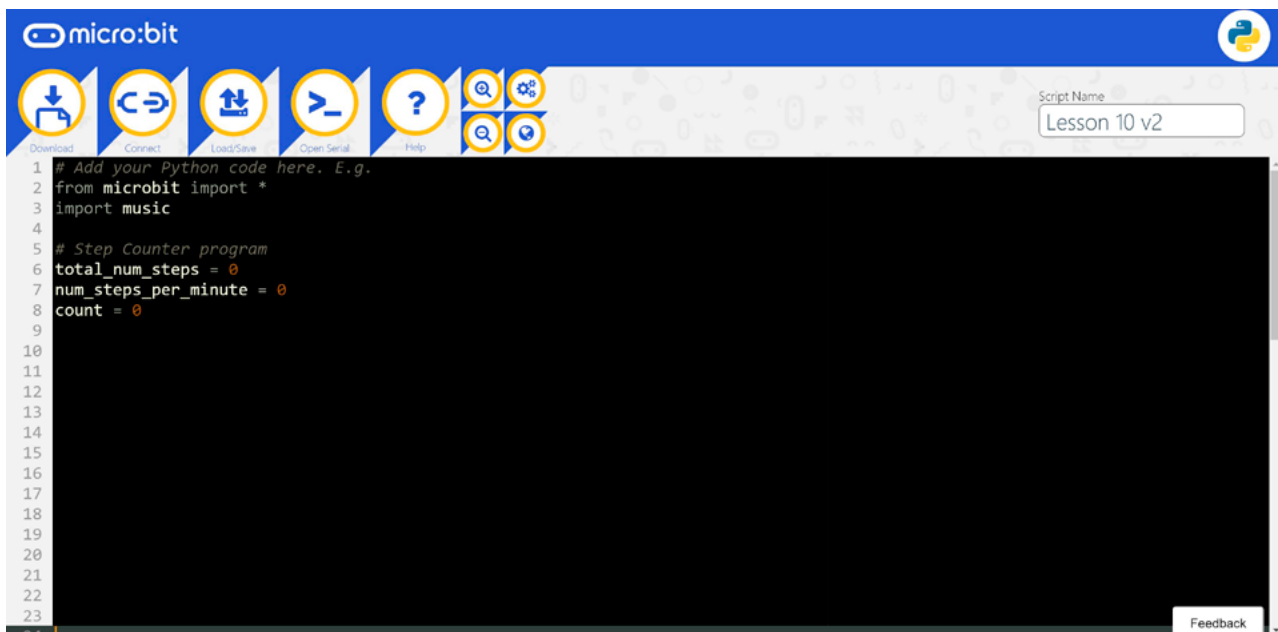
The screenshot shows the micro:bit Python editor interface. The top bar is blue with the 'micro:bit' logo on the left and the Python logo on the right. Below the bar is a toolbar with icons for Download, Connect, Load/Save, Open Serial, Help, and a 2x2 grid of icons (Search, Run, Stop, and another). The main area is a black code editor with white text. The code is as follows:

```
1 # Add your Python code here. E.g.
2 from microbit import *
3 import music
4
5 # Step Counter program
6 total_num_steps = 0
7 num_steps_per_minute = 0
8 count = 0
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
```

On the right side, there is a 'Script Name' input field containing 'Lesson 10 v2' and a 'Feedback' button at the bottom right.



This screenshot is identical to the one above, showing the micro:bit Python editor interface with the same code and layout.



This screenshot is also identical to the ones above, showing the micro:bit Python editor interface with the same code and layout.