THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

DEPARTMENT OF
COMPUTER SCIENCE
Au Reikura

2025

Contributors

J. Turner

J. Kasmara

Au Reikura
School of Computing and Mathematical Sciences

BOT MARLEY TUNES

Last week we introduced you to Bot Marley and the basic movements available. Today, you will have the opportunity to experiment with adding tunes to his movements in preparation for The Big Performance Night event in the next session.

Figure 1: Our Roomba robot vacuum, Bot Marley

Song list

If you're stuck thinking of songs, check the list below to help you create your playlist. We know these songs aren't the hippest ones around. This list is just to get you started!

🎸 Rock/Alternative 🎸	🎤 Hip-Hop/R&B 🎤
• Heat Waves – Glass Animals • fake friend – nothing.nowhere • Colors – Black Pumas • All About You – the Knocks • Is it True – Tame Impala • Oh Yeah! – Green Day • Ew - Joji • Level of Concern – 21 Pilots • Bang! – AJR	• Outta Time – Bryson Tiller • So Done – The Kid LAROI • Buss It (Radio Edit) – Erica Banks • Streets – Doja Cat • Death bed – powfu • Take What You Want – Post Malone • Old Town Road – Lil Nas X, Billy Ray Cyrus • Butterfly Effect – Travis Scott • These Days – Rudimental

- OK OK – HOKO
- Somebody That I Used to Know – Gotye
- Remembering Sunday - All Time Low
- Upside Down – Mondo Cozmo
- Do I Wanna Know - Arctic Monkeys
- No Roots – Alice Merton
- The Man – The Killers
- Feel It Still – Portugal, the Man
- Something to Believe In – Young the Giant
- Girlfriend – Anderson East
- I Bet My Life – Imagine Dragons
- Trouble – Cage the Elephant
- Ocean Avenue – Yellowcard
- The Middle – Jimmy Eat World
- Good Grief – Bastille
- Brand New City – Mitski
- Diary of Jane – Breaking Benjamin
- Welcome to Your Life – Grouplove
- HandClap – Fitz and the Tantrums
- Take It All Back – Judah and the Lion
- You & I – Colony House
- Shake It Out – Florence + the Machine
- The Closest That I am to Living Life on the Edge – Valencia
- Cecilia and the Satellite – Andrew Macmahon and the Wilderness
- I Will Wait – Mumford and Sons
- Death of a Bachelor – Panic! at the Disco
- Lonely Boy – The Black Keys
- Grand Theft Autumn – Fall Out Boy
- Roll Up – Fitz and the Tantrums
- Magic – B.O.B
- Hotline Bling – Drake
- Hey Mama – Kanye West
- Now or Never – Kendrick Lamar
- All We Got – Chance the Rapper
- Crazy in Love – Beyonce + Jay-Z
- Pretty Girl Rock – Keri Hilson
- Happy – Pharrell Williams
- Dynamite – Taio Cruz
- Kick, Push – Lupe Fiasco
- Hold On, We're Going Home – Drake
- Survivor – Destiny's Child

Once you have selected your playlist, set it aside for now and let's continue with today's lesson by picking up where we left off with the Bot Marley Session 1 Exercises.

Robot Dance Recital

One of the main features of the Create 3 robot, Bot Marley, is the ability to move precisely on flat surfaces. In the last session, you hopefully had enough practice moving your robot forwards and backwards, as well as turning left and right. Once you're comfortable driving your robot around, you've earned the title of dance choreographer because of your programming skills commanding your robot to move and turn to perform a basic dance.

Here's a question for you:

What dance recital do you think we had Bot Marley perform in the last session?

If you guessed that we were programming the robot to dance along to the song "Hokey Pokey", you are correct! Give yourself a pat on the back. We created our first dance choreography for Bot Marley, inspired by a version of the Hokey Pokey by The Learning Station. Check it out here: https://youtu.be/iZinb6rVozc?si=KbXxs8jB9KNlcItx (start watching from 0:16).

To recap

So far, we learnt that using the command:

```
await robot.move(16)
```

This drives Bot Marley forward by 16 cm. By editing the parameter, you can adjust both the distance travelled and the robot's direction. For example, a positive value of 16 will move Bot Marley forward 16 cm, while a negative value of 16 will move Bot Marley backward 16cm.

```
await robot.turn_left(90) await robot.turn_right(90)
```

These commands rotate Bot Marley 90 degrees to the left (counter-clockwise) and to the right (clockwise), respectively. Editing the parameters allows you to adjust the number of degrees turned.

Lyrics	Robot Actions
You put your right foot in	Drive 30cm forward
You put your right foot out	Drive 30cm backward
You put your right foot in	Drive 30cm forward
And you shake it all about	Turn 5° to the right Turn 5° to the left Turn 5° to the right Turn 5° to the left
You do the Hokey Pokey and you turn yourself around	Turn 360° to the right
That's what it's all about!	Drive 5cm forward Drive 5cm backward Turn 5° to the right Turn 5° to the left

Figure 2: Sample Dance Recital for Bot Marley

Your shake-about motion may differ from the sample dance recital shown in Figure 2, but the idea is that when Bot Marley completes a full circle, he is dancing to the tunes.

```
You do the Hokey Pokey and you turn yourself around
```

Revisit your code at https://python.irobot.com/ and pause this video to make the necessary adjustments for Bot Marley to perform the right actions to the `Hokey Pokey lyrics`.

Figure 3: Shake-about motion of Bot Marley

Sing off – music with science

Once you have finished making adjustments, we want Bot Marley to sing along while performing the programmable dance moves. Follow these steps:

- Search for a beginner-level piano music sheet of the song – "`Hokey Pokey`".
- Download or take a screenshot of the music sheet, then convert it into **Python Note Frequencies**.

A tip: Look for a music sheet that includes lyrics written beneath the music notes.

Here, we are combining music with science, based on the song "`Hokey Pokey`".

Example:

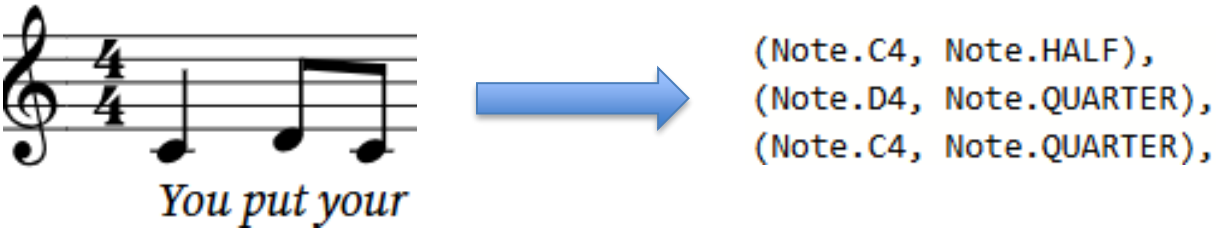


Figure 4: Music notes conversion

Pause the video and see if you can convert the notes for the next set of lyrics:

```
right foot in; You put your
```

I think you are getting the hang of it, nice work! Here's what you should have so far:

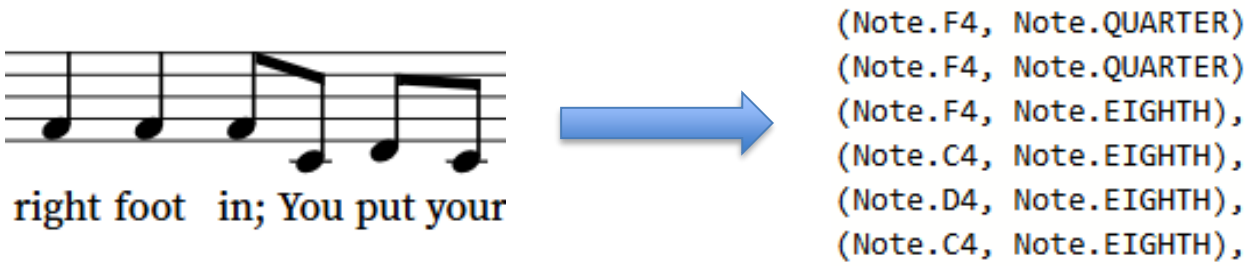


Figure 5: 2nd set of music note conversion

What patterns do you see in these two examples?

The answer is:

```
(Note.C4, Note.HALF),
(Note.D4, Note.QUARTER),
(Note.C4, Note.QUARTER),
```

```
(Note.F4, Note.QUARTER)
(Note.F4, Note.QUARTER)
(Note.F4, Note.EIGHTH),
(Note.C4, Note.EIGHTH),
(Note.D4, Note.EIGHTH),
(Note.C4, Note.EIGHTH),
```

Here is the pattern we see:

1. **`Note.`** indicates playing a particular note.
2. **`C, D, F`** these are the musical notes.
3. **`4`** suggests that the fourth octave should be played.
4. **`HALF, QUARTER, EIGHTH`** stores the value of the beat.

We know that with a standard piano/keyboard, there are 7 octaves in total, comprising 88 keys. The first note on an 88-key is an A (A0) and the last key is a C (C8). Therefore, middle C, i.e., C4, is the fourth octave. Give it a try! https://recursivearts.com/virtual-piano/

Conclusion:

```
(Note.XY, Note.Z)
```

AI in music

AI finds patterns in data and works best with lots and lots of data.

Data: any form of recorded information. In our case, this includes music notes, parts of the song, and dance moves.

What we are doing is creating data-lots and lots of data for AI applications to find patterns in and do something with it.

Take the Shazam music app on your phone as an example:
https://www.shazam.com/apps

Do you think it is AI-powered?
Not quite yet. It's not an AI-powered mobile app, as it uses science and maths. The fingerprinting algorithm is where the analogue sound wave is converted into a digital signal. The resulting and recorded frequency pattern can then be used to create a fingerprint of each song, which is compared to a song database to uniquely identify the song.

Past technology and present times, in the age of AI, share the commonality of patterns in data.

Taking it up a notch: AI comes into play when pieces of music not only need to be recognised, but also categorised. For example, categorising music into genres. How do we use artificial intelligence to assign a song to a genre?

Statistical methods such as Gaussian Mixture Models (GMM), Nearest Neighbour Classification, Linear Discriminant Analysis (LDA) or Support Vector Machines (SVP) are used.

Wouldn't it be cool if you could create an AI application that could predict the types of instruments being played when we play a tune? Especially with reggae music, it should recognise and categorise that drums are being used! At the University of Waikato, you could choose a paper, either Work Integrated or Machine Learning, where you could collaborate with a professor in designing such an AI application.

Note Frequencies

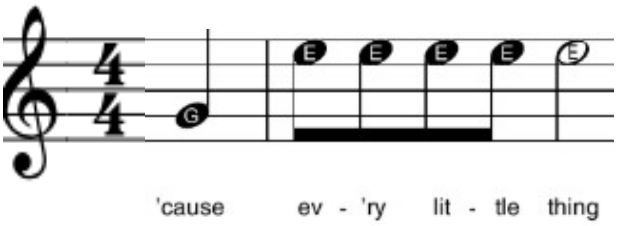
Referring to *Python NoteFrequencies.pdf*, you now have the ability to convert the coded music notes and the duration of each beat into more precise data that the program can understand. There are two steps to this:

- **Convert the music notes into Hz** – depending on the octave, but generally, the higher in scale the piano key is placed, the higher the number of frequencies in Hz.
- **Convert the duration of each beat to time in seconds** – you will need your sense of music to complete this exercise.

A tip: It is fun to get into a group and check each other's work to see if some of the notes played may sound out of tune.

Here, we are creating music with science based on the song "`Three Little Birds`"

Example:



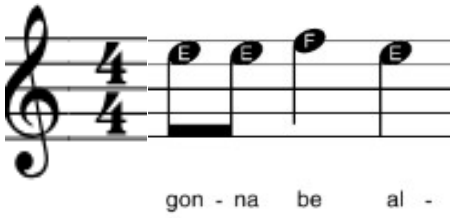
```
await robot.play_note(391,0.40)
await robot.play_note(659,0.15)
await robot.play_note(659,0.15)
await robot.play_note(659,0.15)
await robot.play_note(659,0.15)

await robot.play_note(659,1.30)
```

Figure 5: Note frequencies conversion

Pause the video and see if you can convert the notes for the next set of lyrics:

`gon -na be al -`

I think you are getting the hang of it, nice work! Here's what you should have so far:



```
await robot.play_note(659,0.15)
await robot.play_note(659,0.15)
await robot.play_note(698,0.4)
await robot.play_note(659,0.4)
```

Figure 6: 2nd set of note frequencies conversion

Music and coding

There are many preprogrammed actions provided by iRobot that are used straight out of the box. Refer to *Web Playground Command Reference.pdf*, Bot Marley can be programmed with iRobot Coding App (Level 1 to 3 available depending on student capabilities; see Figure 7 and 8), Python Web Playground, or ROS 2 to make it sing.

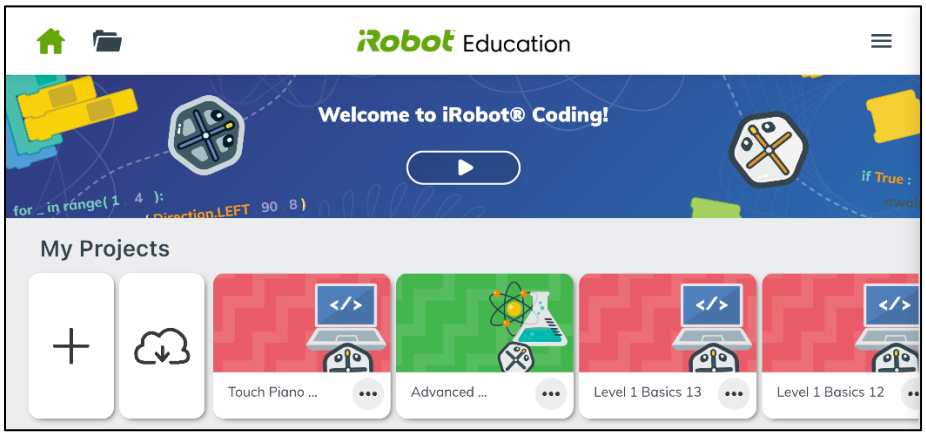


Figure 7: iRobot Mobile App

Figure 8: iRobot Coding Blocks Interface

For advanced users, to program Bot Marley, we are going to use Python via **Python Web Playground v1.2.3, https://python.irobot.com/**. Once opened, you should be presented with the screen shown in Figure 9.

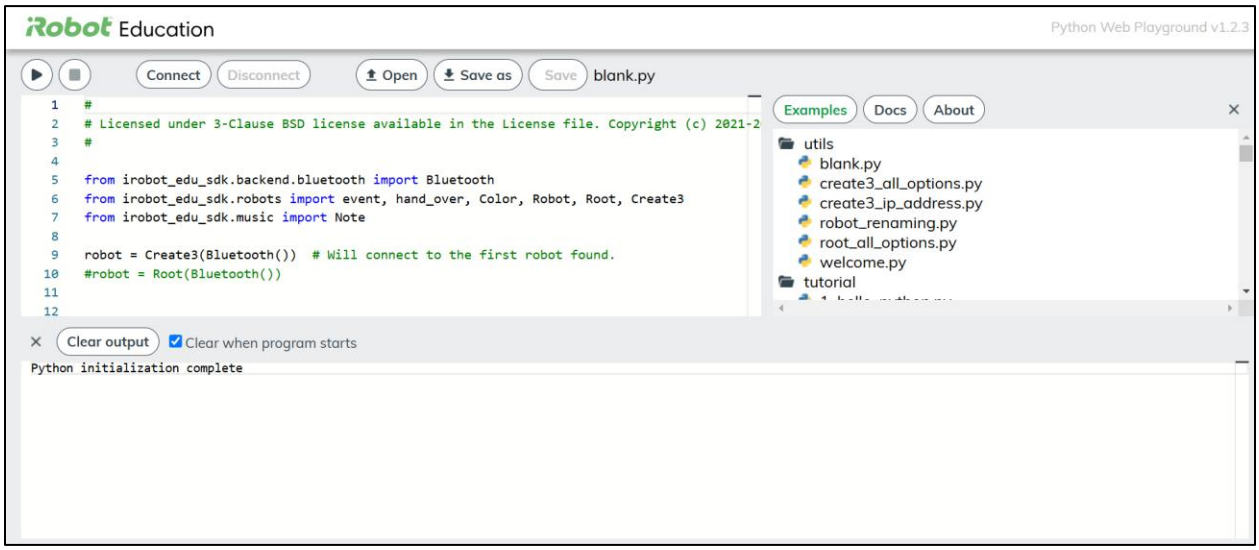


Figure 9: Python Web Playground v1.2.3

Create a new file called `botmarleythreelittlebirds.py` and save it your folder (see Figure 10). This is where we will write our code for today's session. Feel free to ask staff for assistance if you get stuck.

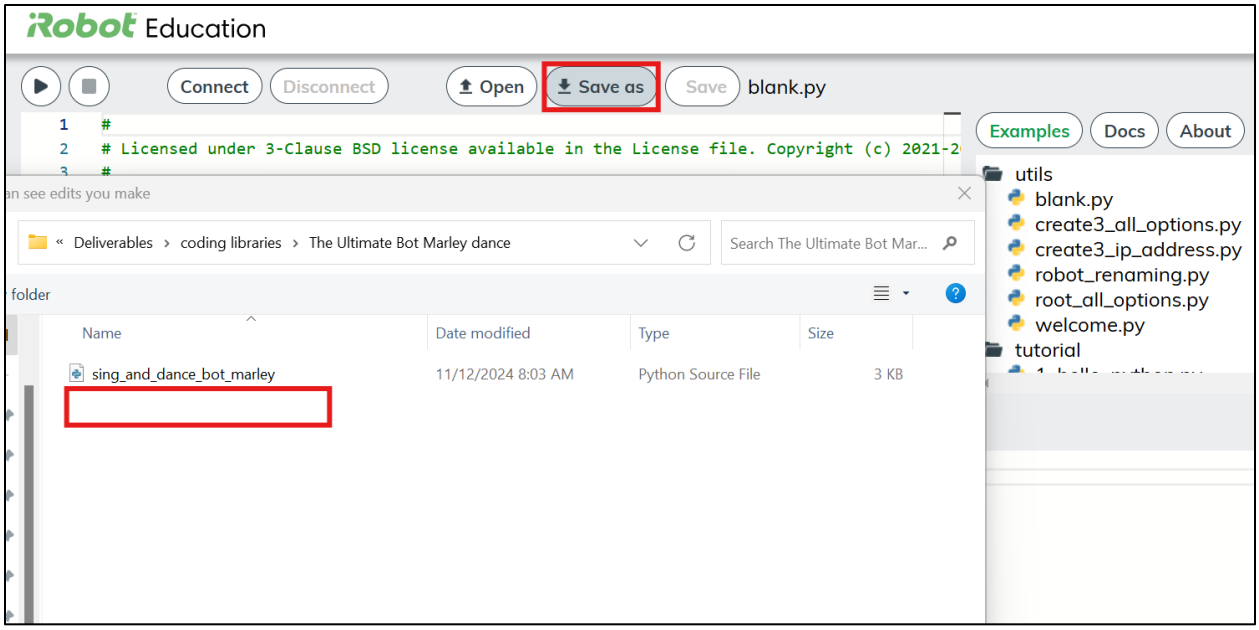


Figure 10: Creating a Python File in Web Playground

Now we can start writing code to make Bot Marley sing "`Three Little Birds`" song.

Firstly, we must include this code:

```
01    from irobot_edu_sdk.backend.bluetooth import Bluetooth
02
03    from irobot_edu_sdk.robots import event, Robot, Create3
04
05    robot = Create3(Bluetooth())
```

Line 1 and 3 tells the program to use the **irobot_edu_sdk** library; without it, none of the commands specific to the robot would be recognised (i.e., Bot Marley cannot be programmed without importing this library). Once the specific library is recognised, Line 1 requests the **Bluetooth** module to be imported from the **irobot_edu_sdk.backend.bluetooth** package. This allows the program to establish a Bluetooth connection with the robot.

Line 3 instructs the program to import the following items from the **irobot_edu_sdk.robots** module:

- **event:** used for event-driven programming to respond to specific actions.
- **Robot:** a general base class interacting with robots.
- **Create3:** a specific class for interacting with the **iRobot Create 3** robot.

Line 5 initialises an instance of the **Create 3** class and assigns it to the variable *robot*. The **Bluetooth()** call creates a Bluetooth connection object, which is passed as an argument to **Create 3** to enable communication between the Python program and the iRobot Create 3 robot.

In summary, what the three lines of code do:

- Imports the necessary modules to enable Bluetooth communication and control the **iRobot Create 3 robot**.
- Creates a connection to the robot using Bluetooth and stores this connection in the variable *robot*.
- This sets up the foundational connection to the robot, allowing you to send commands, receive feedback, and interact with it programmatically.

Secondly, we must know trigger an event:

```
07    @event(robot.when_play)
08    async def sing(robot):
09          print('Lets sing!')
10
11    robot.play()
```

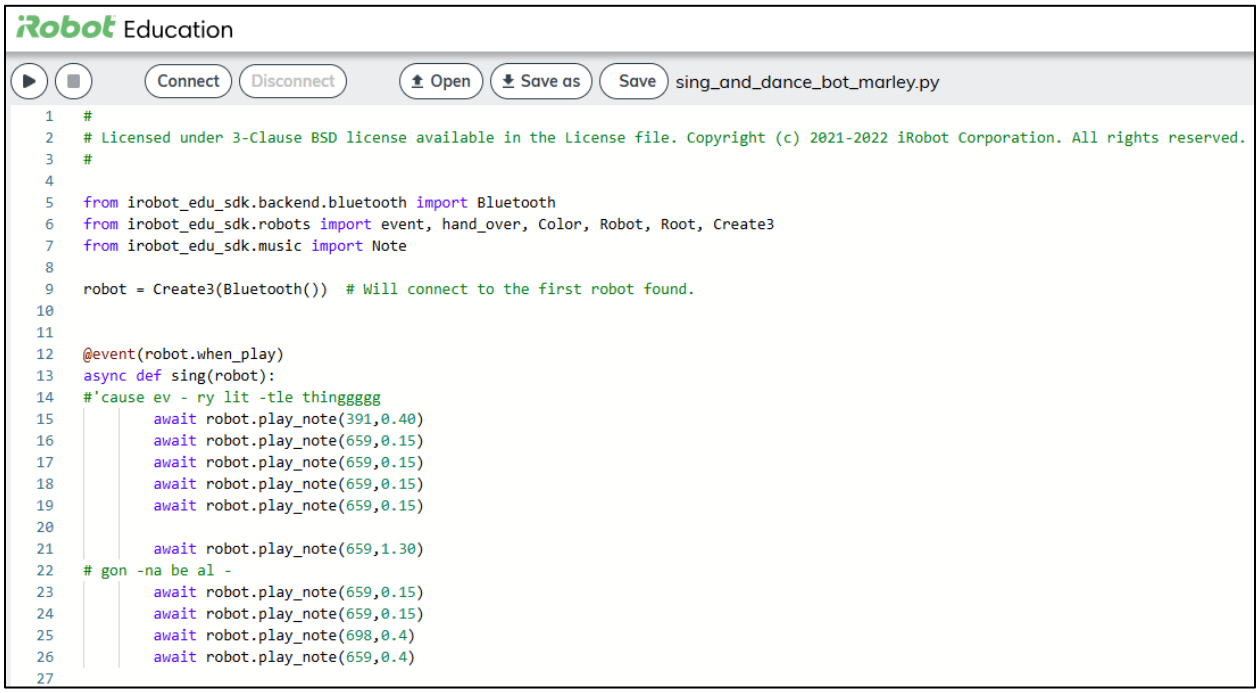
Line 7 and 8 tells the program to run commands (inserted in line 10) according to the event triggered, e.g., when play button is pressed. Line 8 uses **async** which is a shorthand for asynchronous functions. Asynchronous functions enable non-blocking execution, a programming model where a program can execute instructions independently, allowing multiple events or instructions to run without waiting for one to finish before moving on to the next. This time we changed play → sing on Line 8.

Line 9 prints the message **"Lets sing!"** to the console.

Line 10 is where we will write our blocks of commands.

Line 11 manually triggers the **when_play** event for the robot. **When_play** event is triggered, the play() functions runs.

Let's begin with instructing Bot Marley to sing **"**`Cause ev -ry lit -tle thinggggg`**"** **"**`gon -na be al –`**"** (see the code in Figure 11).




```
#
# Licensed under 3-Clause BSD license available in the License file. Copyright (c) 2021-2022 iRobot Corporation. All rights reserved.
#

from irobot_edu_sdk.backend.bluetooth import Bluetooth
from irobot_edu_sdk.robots import event, hand_over, Color, Robot, Root, Create3
from irobot_edu_sdk.music import Note

robot = Create3(Bluetooth())  # Will connect to the first robot found.


@event(robot.when_play)
async def sing(robot):
#'cause ev - ry lit -tle thinggggg
        await robot.play_note(391,0.40)
        await robot.play_note(659,0.15)
        await robot.play_note(659,0.15)
        await robot.play_note(659,0.15)
        await robot.play_note(659,0.15)

        await robot.play_note(659,1.30)
# gon -na be al -
        await robot.play_note(659,0.15)
        await robot.play_note(659,0.15)
        await robot.play_note(698,0.4)
        await robot.play_note(659,0.4)

```

Figure 11: Python Code for singing Three Little Birds song

Click on the ‘Run’ button to run the program. If the program quits without Bot Marley singing, the reason might be due to an incorrect skill string (e.g. ‘robot.play_note(+,+)’) or Bluetooth lost connection. Feel free to ask staff for assistance if you get stuck.

Can you complete the last part of chorus “`righttttt`” ?

The answer is:

```
# righttttt
        await robot.play_note(590,1.0)
```

Now that you have coded part of the song of "`The Hokey Pokey`" and "`Three Little Birds`", get into your groups and start your preparation for the Big Performance Night.

Hopefully understanding the breakdown of a song into music notes and converting them to Python Note frequencies was easy (i.e. `(Note.C4, Note.HALF)` and `await robot.play_note(391,0.40)`) but converting song into music notes may not be because you need another block of code and to store the music notes in a variable `notes = []`. Do insert this piece of code after:

```
while True:
    for note in notes:
        await robot.play_note(note[0], note[1])
```

Now that you know two different ways to make Bot Marley sing some of his reggae songs and musical classics, program your Bot Marley to sing your chosen song for Session 3.

AI memory

AI memory helps machines 'remember' and improve, similar to how humans' study and recall. Remember how Andrei wondered how Bot Marley could remember his birthday for the next year?

AI memory refers to how an AI system stores and recalls information to improve its decisions over time. Unlike human memory, AI memory is structured and relies on algorithms.

Types of Memory in AI

Short-Term Memory: Temporary storage of information. Used for immediate tasks (e.g., remembering recent moves in a chess game).
Long-Term Memory: Persistent storage of knowledge, like facts or patterns learned over time (e.g., how to play chess well).
Replay Memory: In RL, this is used to store past experiences to learn from them again later.

Which type of memory in AI do you think Bot Marley uses to store Andrei's information, such as his birthday?

Answer: Long term memory

Relatable Example:
Think of AI memory like a student studying for a test.

- Short-term memory is used for cramming answers quickly.
- Long-term memory stores concepts they've understood deeply, which they can use in future tests.
- Replay memory is like reviewing past test questions to learn from mistakes.

Short quiz
Grab a pen and paper and draw a robot's 'brain', illustrating short-term memory examples (e.g., quick notes) and long-term memory examples (e.g., books of knowledge). Then, brainstorm with your peers what other examples could be stored in robot's short-term memory and long-term memory PC slots.

Discussion Questions:

- How does memory help humans and machines make better decisions?

Both concepts are foundational for modern AI, including robotics, self-driving cars, and game AI.

Today's Exercise

Continuing with Andrei's amazing companion bot, Marley, it's time to focus on performance enhancements and completing some of the sing-offs! As Marley's programmer, your task is to create a program that:

1. Completes the rest of the chorus for the song "Three Little Birds."
2. Completes the rest of the "Hokey Pokey" song and showcases how beautifully Bot Marley can sing so far.
3. Once satisfied, choose one of the songs based on your preference and program Bot Marley to perform the sing-off with some dance moves.

Remember to save your code in a safe place, as you'll need it for the Big Performance Night in Session 3. Start identifying each other's strengths, whether it's a sense of music or a knack for choreography. Collaborate on combining your strengths to prepare for Session 3. See you all at the Big Performance Night!

Once you feel ready, ask a staff member to watch Bot Marley sing and dance while they tick off your worksheet.

Advanced Exercises:

1. **Complex Dance Routine**
   Now that you've mastered basic moves, create a more **intricate dance routine**. Try to:
   - Use at least **five different movement commands**.
   - Sync the movements with a song's rhythm.
   - Include a **360-degree spin** or a **zigzag pattern**.

   Bonus: Can you make Bot Marley **"freestyle"** by selecting random movements.
2. **Convert Sheet Music to Playable Frequencies**
   - Find **sheet music** for a new song (e.g., "Don't Worry, Be Happy" or another reggae tune).
   - Identify the **notes and durations**.
   - Convert them into **Python note frequencies**.
   - Program Bot Marley to **play the tune correctly**.

   Tip: Use online tools like Recursive Arts Virtual Piano to check your notes.

Reflection:

1. Reflect on what went well and what didn't.
2. Take notes on what your group needs to improve for Session 3.

Summary

In this session, we introduced you to Bot Marley's ability to sing. We instructed him to sing using musical notes, breaking this down further into musical hertz and the duration of beats. Using a Bluetooth connection and Python programming, we communicated with Bot Marley what specific actions he was to perform. If you are using Root, a web-based IDE (Integrated Development Environment) for the sing-off, you had the opportunity to explore more block programming, ranging from Level 1 to 3, to achieve a similar sing-off experience.
Next week, with your group, you'll have the chance to bring music and science to life through a staged performance. Get creative!

Useful Resources

- Create your own music: https://www.noteflight.com/scores/view/ac08337a5288ebdfe38abb4732f6eeab8671ec4a
- Virtual piano: https://recursivearts.com/virtual-piano/

- AI in the music industry: https://musicbusinessresearch.wordpress.com/2024/02/26/ai-in-the-music-industry-part-4-ai-in-music-recognition-and-recommendation/
- W3 Schools Python tutorials: https://www.w3schools.com/python/default.asp
- Python Web Playground v1.2.3: https://python.irobot.com/
- Introduction to Support Vector Machine statistical method: https://www.geeksforgeeks.org/support-vector-machine-algorithm/
- Reinforcement learning – a comprehensive guide: https://aiixx.ai/blog/reinforcement-learning-a-comprehensive-guide-with-interactive-examples
- Short term and Long term memory – your introduction to an *autonomous agents:* https://lilianweng.github.io/posts/2023-06-23-agent/
- PythonNoteFrequencies: https://python.irobot.com/assets/doc/sdk_commands.pdf