

8 Ball Fortune Teller

Teacher's Guide



Lesson Overview

In this lesson, students will construct a functional equivalent of the “Magic 8 Ball”, which is basically choosing a random number 1 to 8 and displaying one of 8 messages. In addition, the mp3 player module is used so that students can add their own voice saying the message.

Grade Level(s)

5-12

Duration

2-3 class periods

Objectives

1. Students will be introduced to the random number generator of MakeCode.
2. Students will reinforce their knowledge of how to use the “on-touch” event blocks.
3. Students will reinforce their knowledge of using arrays.
4. Students will learn to record their own voice as an mp3 file and to put those files on the microSD card.
5. These activities reinforce earlier lessons on the mp3 playback module and file/folder structure on devices.
6. Students will have the opportunity to use what they learned about controlling LEDs in earlier lessons, and apply it in this project for a visual effect while the 8 Ball Fortune Teller is “thinking”.
7. Students will reflect on probability and what that means in a practical way for the generation of random messages.



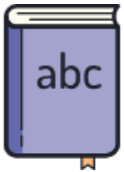
Supplies

Required

Re-use the Game of NIM model
[8 Ball overlay](#)

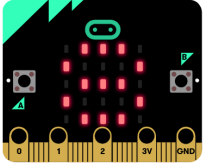
Optional

Masking, painter's or transparent tape



Vocabulary

Random numbers	Folders, files, filetypes
Array	Element of the array
Index	



MakeCode Blocks Introduced



(pick 1 of 11 random numbers, 0 to 10)



Instructional Steps

Warm Up

Time: 15 minutes

Ask Students: Have they seen a toy called “The Magic 8 Ball”? For someone that has seen it, ask them to describe it to the class. Here is a video showing the Magic 8 Ball:

<https://youtu.be/vZRrg6NI-1E>

In code, have them discuss, in general terms, how they think it would be coded, based on what they’ve learned so far in previous projects. Through the [Piano projects](#), they know about arrays of text messages, and playing digitized audio files, so the only new block for this project is “pick a random number”.

Now ask them to imagine a Magic 8 Ball toy that not only shows a message, but speaks the answer. Can they describe how they would use what they learned in the Piano (mp3) to play a voice recording, rather than a digitized piano note?

If they can play back an audio message in one language, could they use what they learned about switching instruments in the Piano (mp3) to play back digitized recordings of the message in different spoken languages?



Main Activity

Time: 3-4 class periods

Project Instructions:

Part 1: [Assembly](#)

Part 2: [Recording](#)

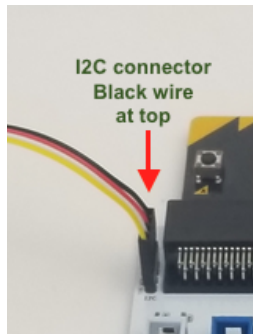
Part 3: [Coding](#)

[Challenges](#)

Trouble-shooting Tips

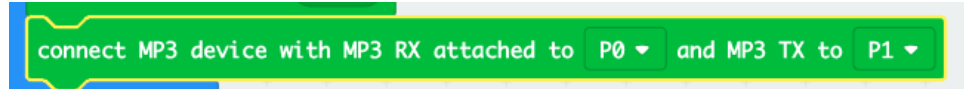
Have a totally functional model available. A fast way to diagnose a problem is to put a micro:bit that is pre-programmed with the 8 Ball program into the MakerBit of a student model. If the model then works, it is a coding error in the student model. If the physical model does not work (touch sensors don't respond, LCD doesn't display text, mp3 playback doesn't happen.) then there is most likely a wiring error in the student model.

1. **Touch sensor:** Confirm that it is a brown wire on the outside edge of the ribbon cable (T16) that is connected to the touch sensor that is being touched to initiate a new message.
2. **LCD:** Confirm that the wire order is **black-red-white-yellow** on **both** the MakerBit and the LCD. Black should be at the "top" on both connections.



3. **Mp3 playback:**
 - a. Confirm that the microSD card is inserted into the mp3 playback module. If the module is connected properly (totally inserted, component side "up"), then there will be an illuminated LED on the board.

- b. Confirm that the “connect mp3” block is in “on start” in the program code.



- c. Confirm that there is a folder on the microSD card with the proper numbering of the folder and mp3 file.



Challenges

Time: 1-2 class periods

1. Can you add a pause block and then cycle LEDs on and off, with a delay of 100ms so that you can see the 8 ball “thinking”?
Hint: pause block, for index = 5 to 16 loop, set digital pin (index) high/low
https://makecode.microbit.org/#pub:_AkuehPMY1Kkd
2. In the Piano (mp3) project, you learned how to switch folders to play different instrument sounds. Can you record the messages in a 2nd language, and use a touchpoint to switch between the spoken language used for the messages?
Hint: If you don’t speak a 2nd language, see if you can find a classmate or family member to help you. You can also use [Google translate](https://www.google.com/translate) and the “speak” option to get a spoken output. Use another computer or phone to record the Google translated voice.
A: https://makecode.microbit.org/#pub:_ev6J3dDtkEje
3. If the text to display on one line (16 characters) is too long, can you adapt your code for 2 lines for each message on the LCD?
Hint: use 2 arrays (lists). One for the top line, and the other for the second.
A: https://makecode.microbit.org/#pub:_RgRFpUFcEgE6

Closing

Time: 10 minutes



Reflection

In using the 8 Ball Fortune Teller, did you ever see the same response happen twice in a row? Three times in a row? If events are random, does that mean that even in a small number of tries, each possible outcome only happens once until the entire list repeats?

Is this 8 Ball Fortune Teller “fair”? Look at the list of 8 possible responses, and categorize each as a positive, negative or neutral answer. Is it equally likely to get a positive answer as a negative or neutral answer? Why or why not?

Can you express the chances (probability) of getting each of the categories (positive, negative, or neutral) as a numeric fraction or decimal?

Using the technology that you have learned about in this and previous devices, what kinds of different other devices or models can you think of that could be made using a new paper overlay, and that which you now know how to use?

The next project will be building your own board game, where you will simulate rolling dice for a number of moves, and a digital version of drawing cards with different messages on them. What can you point out from the code that you used in this project that could also be used in the board game?



Resources

Magic 8 Ball Video

<https://youtu.be/vZRrg6NI-1E>

8 Ball Overlay

<https://bit.ly/3bAb7ta>

8 Ball mp3 files (15 answers in English & Spanish)

<https://makerbit.com/resources/Magic%208%20Ball/8%20Ball%20Fortune%20Teller.zip>

Finished Coded Programs

8 Ball Code

https://makecode.microbit.org/#pub:_hgHf085To5gF

Challenge 1

"Thinking" code

https://makecode.microbit.org/#pub:_1pLJskHFR9qx

Challenge 2

Alternate Language (15 answers)

https://makecode.microbit.org/#pub:_ev6J3dDtEje

Challenge 3

Better LCD display for long text and non-repeat of last answer.

https://makecode.microbit.org/#pub:_RgRFpUFcEgE6



Standards

California Computer Science Standards		Indiana K-12 Computer Science Standards		New York Computer Science and Digital Fluency Learning Standards	
3-5.CS.1	6-8.AP.13	6-8.PA.2	CSI-4.4	3-5.CT.4	3-5.CT.9
3-5.CS.2	6-8.AP.14	6-8.PA.3	CSI-4.6	6-8.CT.4	6-8.CT.9
3-5.CS.3	6-8.AP.15	ICS-2.1	CSI-4.8	9-12.CT.4	9-12.CT.9
3-5.AP.10	6-8.AP.17	ICS-2.2	CSI-4.11	3-5.CT.5	3-5.CT.10
3-5.AP.11	6-8.AP.19	ICS-2.3	CSI-4.12	6-8.CT.5	6-8.CT.10
3-5.AP.12	6-8.IC.21	ICS-2.5	CSI-5.2	9-12.CT.5	9-12.CT.10
3-5.AP.13	9-12.CS.1	ICS-2.7	CSII-3.1	6-8.CT.6	3-5.CT.11
3-5.AP.14	9-12.CS.2	ICS-5.5	CSII-4.6	3-5.CT.7	6-8.CT.11
3-5.AP.17	9-12.AP.12	CSI-3.6	CSII-5.2	6-8.CT.7	3-5.NSD.1
3-5.IC.20	9-12.AP.14	CSI-4.2		9-12.CT.7	6-8.NSD.1
6-8.CS.3	9-12.AP.16	CSI-4.3		9-12.CT.8	3-5.NSD.2
6-8.AP.10	9-12.AP.22				3-5.NSD.3
6-8.AP.11	9-12S.AP.22				6-8.NSD.3
6-8.AP.12					9-12.NSD.3