

Mission 9 Remix

Create your own project from Mission 9

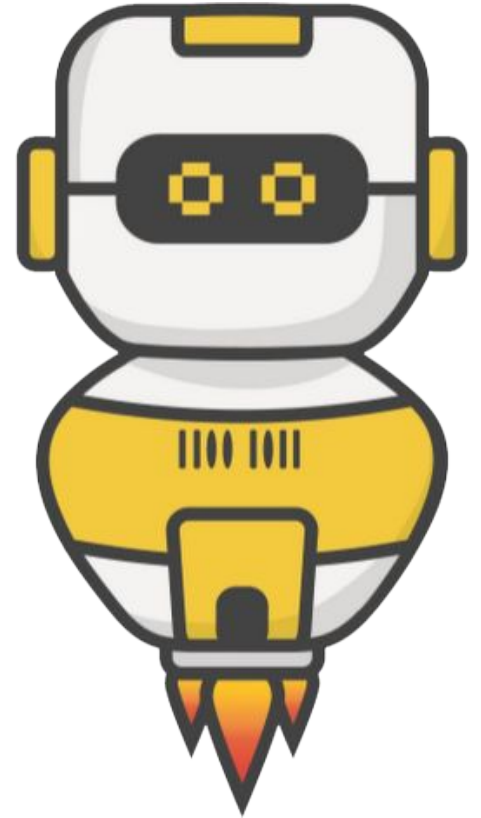


Warm-up

Mission 9 introduced functions, parameters and arguments.

In your Mission 9 Remix Log, answer the pre-mission preparation questions:

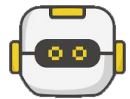
- What is a function?
- What is the difference between a parameter and an argument?



Time for a project remix!

A remix can be:

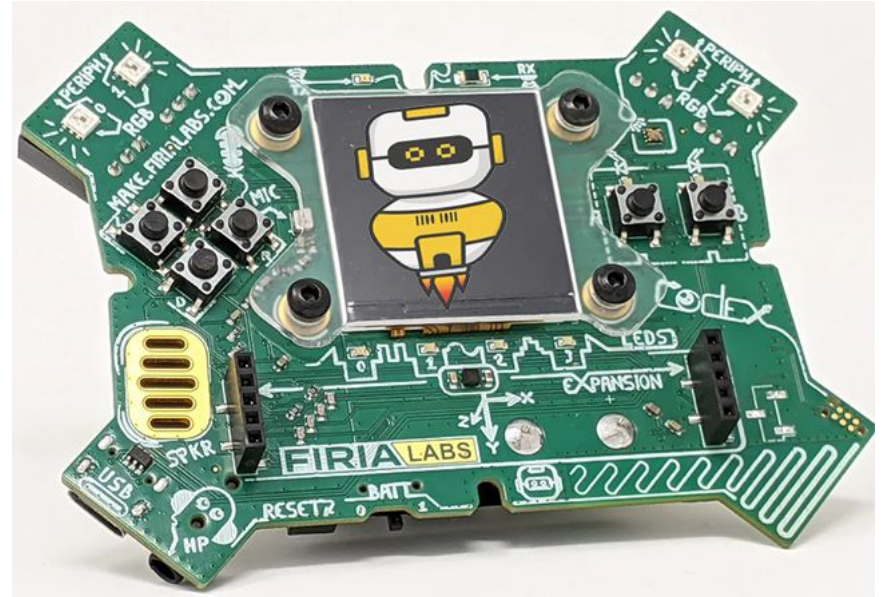
- A new program created by adding new code to a program you already created
- You can combine parts of two or more programs in a remix
- Use a similar idea in a different way



Project Remix

Creating a remix will let you:

- Improve your skills and practice the concepts from the mission
- Be creative
- Remember code from earlier programs and missions
- Work with other students
- Design an original program and write the code all on your own



Step #1

Review the mission

- Open your project from Mission 9
 - What does the program do?
 - What skills were used or concepts learned?

DO THIS:

- Fill out the information in the Mission 9 Remix Log for **Step 1**

```
from codex import *
import random
from time import sleep

def show_random_arrow():
    arrow = random.randrange(8)
    display.show(pics.ALL_ARROWS[arrow])

def spin_animation(count):
    index = 0
    loops = 0
    delay = 0.0
    while loops < count:
        loops = loops + 1
        display.show(pics.ALL_ARROWS[index])
        sleep(delay)
        delay = delay + 0.005
        index = index + 1
        if index == 8:
            index = 0

while True:
    if buttons.is_pressed(BTN_A) or buttons.is_pressed(BTN_B):
        spin_animation(20)
        show_random_arrow()

    if buttons.is_pressed(BTN_U):
        break
```



Step #2

Brainstorm ideas

- Read through remix suggestions from your teacher

Ten suggestions are on the next 3 slides.
You can use any of these ideas or come up with your own.

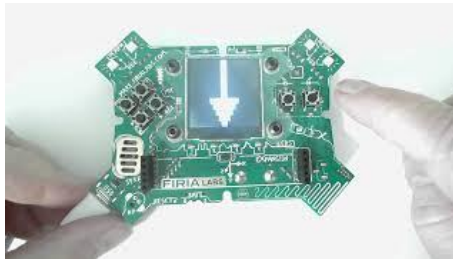


Step #2 Remix Ideas



Mild-1A

Create a function that will give an introduction and wait to start the game spinner. Then add a button to break out of the infinite loop and end the program with a message.



Mild-1B

Add a beep for each arrow spin.



Mild-1C

Make the animation spin counterclockwise instead of clockwise

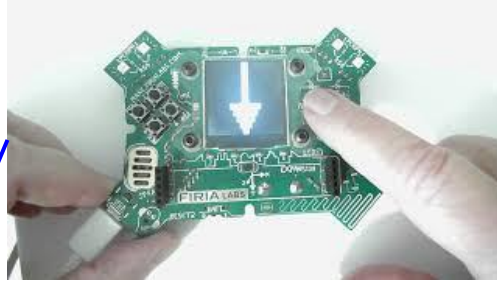


Step #2 Remix Ideas



Medium-2A

Use BTN_A for spinning clockwise and BTN_B for spinning counterclockwise.



Medium-2B

Many games use colors, numbers or specific images on the wheel. Show a random color (or large number) on the screen. Instead of ending with a random arrow, get a random number for count and whatever is showing at the end is the selection. (Only one function needed)

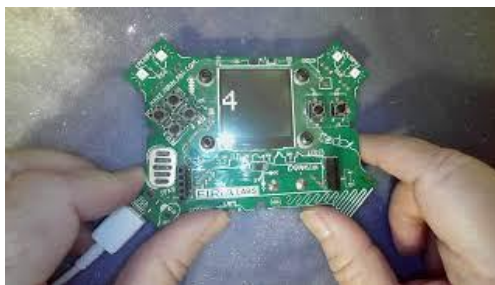


Step #2



Spicy-3A

Use both Button A and Button B for two different things. For example, use button A for the arrow spin and button B for the dice roll. Or button A for arrow spin and button B for colors.



Spicy-3B

Create another list and use a different beep for each arrow.



Spicy-3C

Create a 2-player game. Each player presses their own button. Use the dice roll or assign points to the arrows or colors. The first player to a point value wins.



Step #2



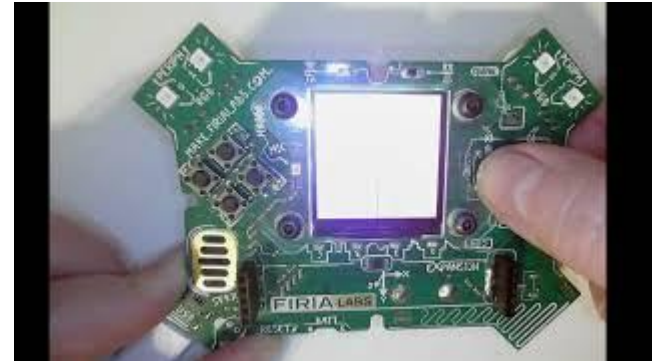
Extra Spicy-4A

Many games use two dice. Create a program that rolls two dice at the same time.



Extra Spicy-4B

Create a color-match game. Display two rectangles, each with a random color. How long does it take to get matching colors?



Step #2

Brainstorm ideas

- Read through remix suggestions from your teacher (previous slides)
- Use the suggestions as presented, or combine some of the options for your own mild, medium, spicy or extra spicy project
- Use your creativity to come up with your own idea for a project
- Decide with your partner what project you will do

DO THIS:

- Fill out the information in the Mission 9 Remix Log for **Step #2**



Step #3

Make a plan

- What variables will you need? What will you use them for?
- What functions will you create for the program?
- What lists will you use, and what information will they store?
- What buttons will you program, and what will each button do?


DO THIS:

- Fill out the information in the Mission 9 Remix Log for **Step #3**



Step #4

Code your project

- **IMPORTANT:** In CodeSpace, go to the sandbox 
- Start with a new file and give it a descriptive name (**Remix9**)
- You can leave any program open, including **Game Spinner**, and use it as a guide
- Import your modules
- Create your lists and define your variables
- Write your code, testing frequently



Step #4

Stop and test frequently!

- Don't try to write all the code at one time
- Think about the steps:
 - Just get one thing to work, then move on
 - Step by step!
- Mistakes happen, so find them early
- Type just a few lines of code and then run the program
- If there is an error, fix it before continuing
- Use the debugger and your other programs for help



Step #5

Documentation!

- Make sure your code is readable by adding blank lines
- Add comments to sections of your code that explain what they do



Step #5

Get feedback

- Show your code to other students.
- What do they think? Have them fill out the feedback form on your Mission 9 Remix Log.
- The Mission Log has space for two people to give feedback. The feedback can come from two peers or one peer and yourself.

Modify your code to make your project even better



And now you have your own remix!

Congratulations!

By completing this remix you have:

- learned more about programming
- practiced the skills and concepts from the missions
- been thinking! And problem solving and much more!



Mission Reflection

- Wow! Great job!
 - Share your project with your friends!
 - Run projects from other students
 - Complete your Mission 9 Remix Log
-
- Don't forget to clear your CodeX by running your **Clear** program

