Python Code By Mission

Mission 9 - Game Spinner	
Using a logical operator:	`if buttons.is_pressed(BTN_A) or buttons.is_pressed(BTN_B):`

Define a function	`def show_random_arrow():`
	`    num = random.randrange(8)`
	`    display.show(pics.ALL_ARROWS[num])`

Call a function	`while True:`
	`    if buttons.is_pressed(BTN_A) or buttons.is_pressed(BTN_B):`
	`        show_random_arrow()`

Finite loop with condition
 	
(increment the control variable)
	`while index < 8:`
	`    my_arrow = pics.ALL_ARROWS[index]`
	`    display.show(my_arrow)`
	`    sleep(0.1)`
	`    index = index + 1`

Finite loop with condition and list wrapping	`while loops < count:`
	`    my_arrow = pics.ALL_ARROWS[index]`
	`    display.show(my_arrow)`
	`    sleep(delay)`
	`    delay = delay + 0.005`
	`    loops = loops + 1`
	`    index = index + 1`
	`    if index == 8:`
	`        index = 0`

Mission 10 - Reaction Tester	
Turn off all pixels using a list	`pixels.set([BLACK, BLACK, BLACK, BLACK])`
Turn all pixels a color using a list	`pixels.set([GREEN, GREEN, GREEN, GREEN])`
Clear the display	`display.clear()`

Get current clock time	`start_time = time.ticks_ms()`
Find the difference between two clock times	`reaction_time = time.ticks_diff(end_time, start_time)`
Reset the button state	`buttons.was_pressed(BTN_A)`
Mission 11 - Spirit Level	
Math module	`import math` used for math operations, like math.pi, math.asin, etc.
Get values from the accelerometer	`val = accel.read()`
Get a single value from the accelerometer	`val = accel.read()` `tilt_x = val[0]`
Change display color	`display.fill(WHITE)`
Draw a line	`display.draw_line(x1, y1, x2, y2, color)` `display.draw_line(CENTER, 0, CENTER, 105, BLACK)`
Draw a circle	`display.draw_circle(x, y, radius, color)` `display.draw_circle(x, CENTER, 15, ORANGE)`
Mission 11 Remix -- these commands are optional but can be used in the remix projects	
Filled in circle	`display.fill_circle(CENTER, CENTER, 15, RED)`
Display text with a specific location	`display.draw_text(str(score), x=20, y=20, scale=3, color=BLACK)`
Mission 12 - Night Light	
Read from the light sensor	`value = light.read()`
Set all pixels the same color	`pixels.fill(WHITE)` -- on `pixels.fill(BLACK)` -- off
Adjust brightness of pixels	`pixels.fill(WHITE, brightness=20)`

	`pixels.fill(WHITE, brightness = level)`