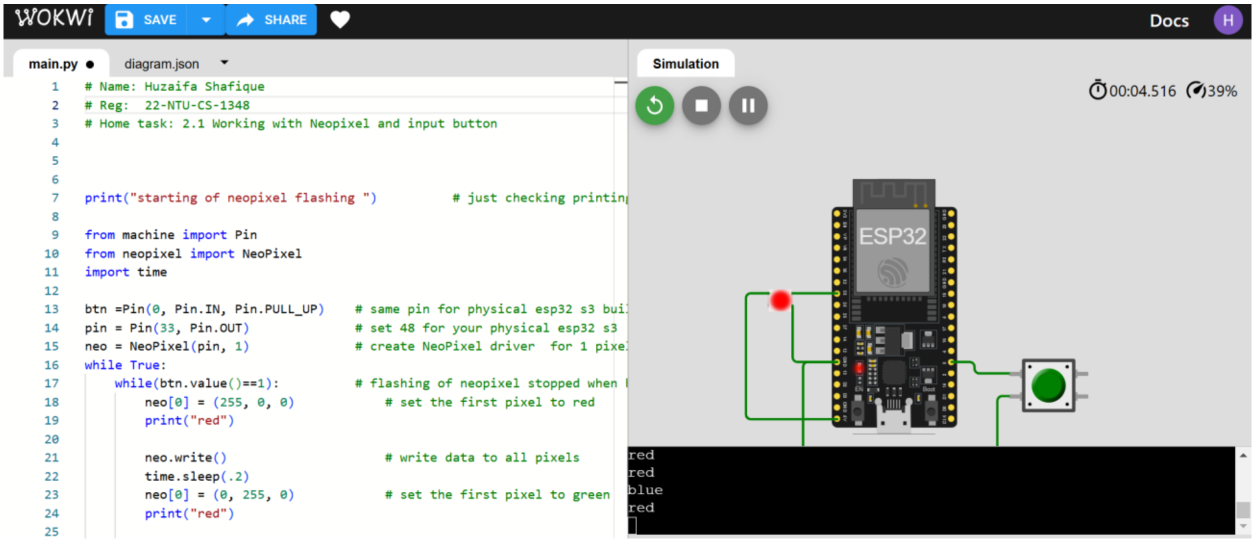
Task 1:

```
print("starting of neopixel flashing ")        # just checking printing output


from machine import Pin

from neopixel import NeoPixel

import time


btn =Pin(0, Pin.IN, Pin.PULL_UP)    # same pin for physical esp32 s3 built in Boot buton

pin = Pin(33, Pin.OUT)           # set 48 for your physical esp32 s3

neo = NeoPixel(pin, 1)           # create NeoPixel driver  for 1 pixel

while True:

  while(btn.value()==1):        # flashing of neopixel stopped when button is in pressed status

    neo[0] = (255, 0, 0)         # set the first pixel to red

    print("red")


    neo.write()                 # write data to all pixels

    time.sleep(.2)

    neo[0] = (0, 255, 0)         # set the first pixel to green

    print("red")


    neo.write()                 # write data to all pixels

    time.sleep(.2)

    neo[0] = (0, 0, 255)         # set the first pixel to blue

    print("blue")


    neo.write()                 # write data to all pixels

    time.sleep(.2)
```



```
# Name: Huzaifa Shafique
# Reg:  22-NTU-CS-1348
# Home task: 2.1 Working with Neopixel and input button



print("starting of neopixel flashing ")          # just checking printin

from machine import Pin
from neopixel import NeoPixel
import time

btn =Pin(0, Pin.IN, Pin.PULL_UP)    # same pin for physical esp32 s3 bui
pin = Pin(33, Pin.OUT)              # set 48 for your physical esp32 s3
neo = NeoPixel(pin, 1)              # create NeoPixel driver  for 1 pixe
while True:
    while(btn.value()==1):          # flashing of neopixel stopped when
        neo[0] = (255, 0, 0)            # set the first pixel to red
        print("red")

        neo.write()                     # write data to all pixels
        time.sleep(.2)
        neo[0] = (0, 255, 0)            # set the first pixel to green
        print("red")

```

```
red
red
blue
red
```

Task 2:

- Why does the Neopixel always turn blue when the button is pressed?

Answer:

The neopixel always turn blue when the button is pressed because when we press the button it give the loop value 0 . The loop continues and when it check the condition the loop terminate itself and the last color which it display is blue. That's why it always turn blue

- How can it be made to stop on different colors in real-time (e.g., sometimes red, sometimes green, sometimes blue)

Answer:

We can stop it on different colors by making a random call of list.

import time

from machine import Pin

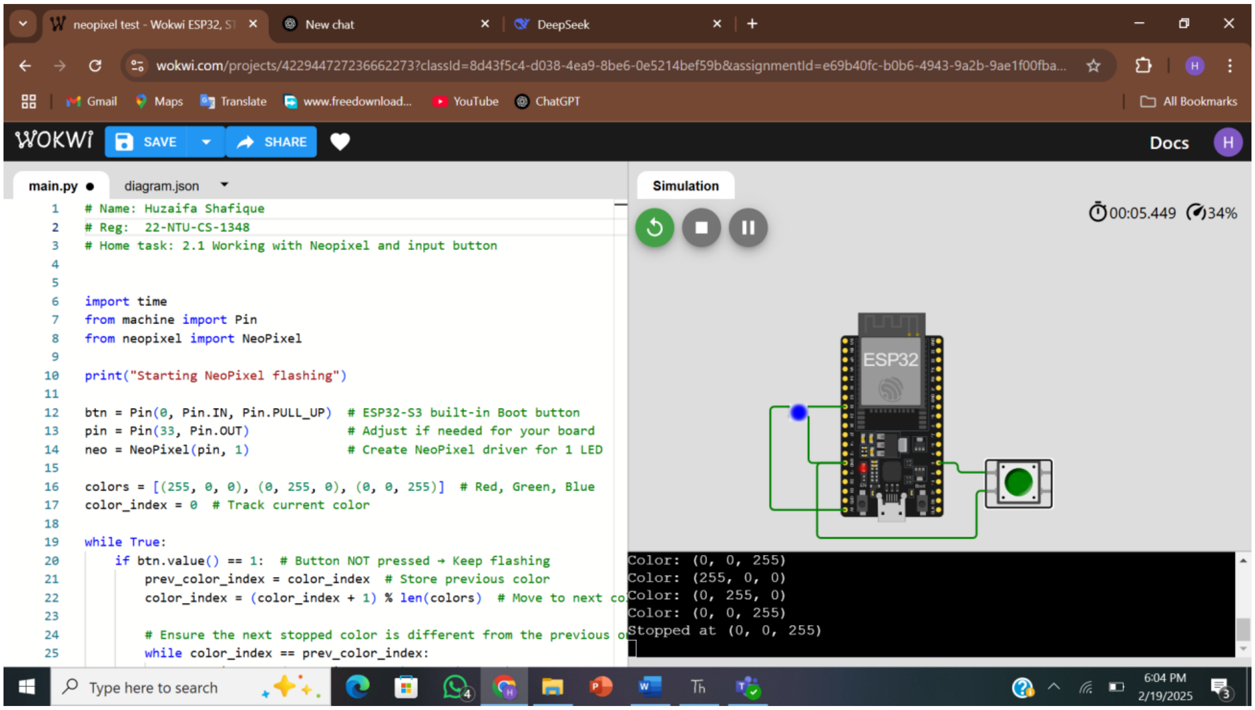
from neopixel import NeoPixel

print("Starting NeoPixel flashing")

btn = Pin(0, Pin.IN, Pin.PULL_UP) # ESP32-S3 built-in Boot button

pin = Pin(33, Pin.OUT)   # Adjust if needed for your board

```
neo = NeoPixel(pin, 1)          # Create NeoPixel driver for 1 LED


colors = [(255, 0, 0), (0, 255, 0), (0, 0, 255)]  # Red, Green, Blue
color_index = 0  # Track current color


while True:
   if btn.value() == 1:  # Button NOT pressed → Keep flashing
      prev_color_index = color_index  # Store previous color
      color_index = (color_index + 1) % len(colors)  # Move to next color


      # Ensure the next stopped color is different from the previous one
      while color_index == prev_color_index:
         color_index = (color_index + 1) % len(colors)


      neo[0] = colors[color_index]
      neo.write()
      print(f"Color: {colors[color_index]}")
      time.sleep(0.2)  # Flash delay


   else:  # Button PRESSED → Stop on current color
      print(f"Stopped at {colors[color_index]}")
      while btn.value() == 0:  # Wait until button is released
         time.sleep(0.1)  # Avoid excessive CPU usage
```



```
# Name: Huzaifa Shafique
# Reg:  22-NTU-CS-1348
# Home task: 2.1 Working with Neopixel and input button


import time
from machine import Pin
from neopixel import NeoPixel

print("Starting NeoPixel flashing")

btn = Pin(0, Pin.IN, Pin.PULL_UP)  # ESP32-S3 built-in Boot button
pin = Pin(33, Pin.OUT)             # Adjust if needed for your board
neo = NeoPixel(pin, 1)             # Create NeoPixel driver for 1 LED

colors = [(255, 0, 0), (0, 255, 0), (0, 0, 255)]  # Red, Green, Blue
color_index = 0  # Track current color

while True:
    if btn.value() == 1:  # Button NOT pressed → Keep flashing
        prev_color_index = color_index  # Store previous color
        color_index = (color_index + 1) % len(colors)  # Move to next co

        # Ensure the next stopped color is different from the previous o
        while color_index == prev_color_index:
```

```
Color: (0, 0, 255)
Color: (255, 0, 0)
Color: (0, 255, 0)
Color: (0, 0, 255)
Stopped at (0, 0, 255)
```

Task 3

- Change the color after every 5 button presses.

print("Starting NeoPixel flashing") # Just checking printing output

from machine import Pin

from neopixel import NeoPixel

import time

btn = Pin(0, Pin.IN, Pin.PULL_UP) # Same pin for physical ESP32-S3 built-in Boot button

pin = Pin(33, Pin.OUT)   # Set 48 for your physical ESP32-S3 if needed

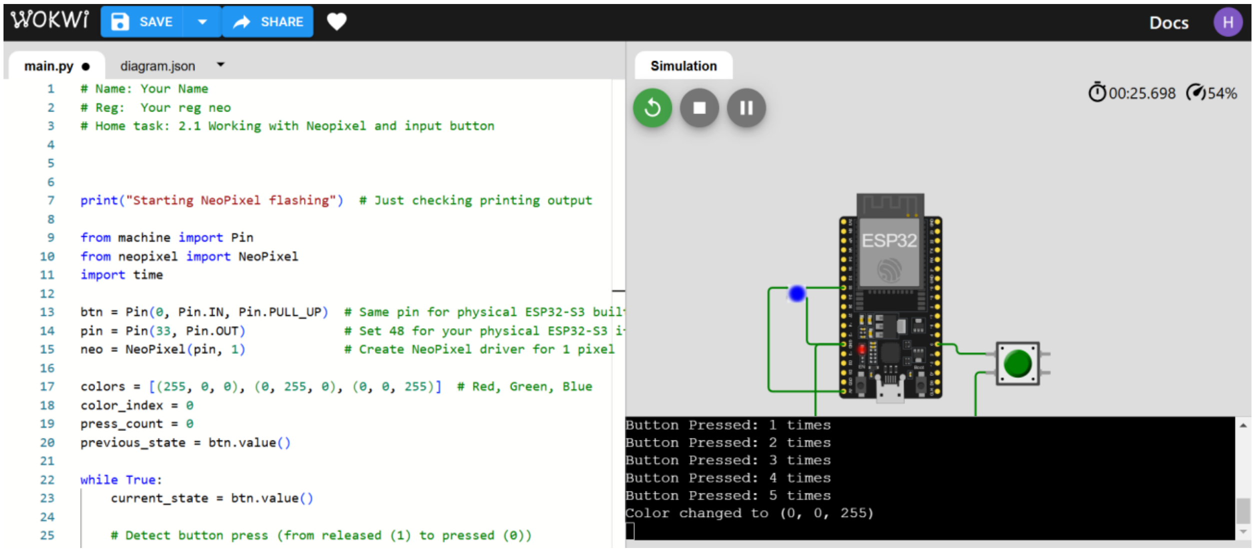
neo = NeoPixel(pin, 1)   # Create NeoPixel driver for 1 pixel

colors = [(255, 0, 0), (0, 255, 0), (0, 0, 255)] # Red, Green, Blue

color_index = 0

press_count = 0

```
previous_state = btn.value()

while True:
  current_state = btn.value()

  # Detect button press (from released (1) to pressed (0))
  if previous_state == 1 and current_state == 0:
    press_count += 1
    print(f"Button Pressed: {press_count} times")

    if press_count >= 5:  # Change color every 5 presses
      color_index = (color_index + 1) % len(colors)  # Cycle through colors
      neo[0] = colors[color_index]
      neo.write()
      print(f"Color changed to {colors[color_index]}")
      press_count = 0  # Reset counter

  previous_state = current_state  # Update previous state
  time.sleep(0.05)  # Small delay to avoid bouncing issues
```



```
# Name: Your Name
# Reg:  Your reg neo
# Home task: 2.1 Working with Neopixel and input button



print("Starting NeoPixel flashing")  # Just checking printing output

from machine import Pin
from neopixel import NeoPixel
import time

btn = Pin(0, Pin.IN, Pin.PULL_UP)  # Same pin for physical ESP32-S3 buil
pin = Pin(33, Pin.OUT)             # Set 48 for your physical ESP32-S3 i
neo = NeoPixel(pin, 1)             # Create NeoPixel driver for 1 pixel

colors = [(255, 0, 0), (0, 255, 0), (0, 0, 255)]  # Red, Green, Blue
color_index = 0
press_count = 0
previous_state = btn.value()

while True:
    current_state = btn.value()

    # Detect button press (from released (1) to pressed (0))
```

```
Button Pressed: 1 times
Button Pressed: 2 times
Button Pressed: 3 times
Button Pressed: 4 times
Button Pressed: 5 times
Color changed to (0, 0, 255)
```

- Examine the result: Does the color change exactly after 5 presses, or is there abnormal behavior?

No abnormal behavior

Task 4:

Implement your own changes to the code

Answer:

```
from machine import Pin
from neopixel import NeoPixel
import time

btn =Pin(0, Pin.IN, Pin.PULL_UP)
pin = Pin(48, Pin.OUT)
neo = NeoPixel(pin, 1)
while True:
    while(btn.value()==1):
        neo[0] = (255, 0, 0)
        print("red")

        neo.write()
        time.sleep(.2)
        neo[0] = (0, 255, 0)
        print("red")

        neo.write()
        time.sleep(.2)

        neo[0] = (0, 0, 255)
        print("blue")
```

```
neo.write()
time.sleep(.2)

neo[0] = (255, 255, 255)
print("white")

neo.write()
time.sleep(.2)
neo[0] = (25, 205, 155)
print("white")

neo.write()
time.sleep(.2)
```